Fundamental Numerical Methods and Data Analysis

by

George W. Collins, II

Download latest edition of this book here:
http://astrwww.cwru.edu/personal/collins/

Table of Contents

List of Figures

List of Tables

Preface

• • •

The origins of this book can be found years ago when I was a doctoral candidate working on my thesis and finding that I needed numerical tools that I should have been taught years before. In the intervening decades, little has changed except for the worse. All fields of science have undergone an information explosion while the computer revolution has steadily and irrevocability been changing our lives. Although the crystal ball of the future is at best "seen through a glass darkly", most would declare that the advent of the digital electronic computer will change civilization to an extent not seen since the coming of the steam engine. Computers with the power that could be offered only by large institutions a decade ago now sit on the desks of individuals. Methods of analysis that were only dreamed of three decades ago are now used by students to do homework exercises. Entirely new methods of analysis have appeared that take advantage of computers to perform logical and arithmetic operations at great speed. Perhaps students of the future may regard the multiplication of two two-digit numbers without the aid of a calculator in the same vein that we regard the formal extraction of a square root. The whole approach to scientific analysis may change with the advent of machines that communicate orally. However, I hope the day never arrives when the investigator no longer understands the nature of the analysis done by the machine.

Unfortunately instruction in the uses and applicability of new methods of analysis rarely appears in the curriculum. This is no surprise as such courses in any discipline always are the last to be developed. In rapidly changing disciplines this means that active students must fend for themselves. With numerical analysis this has meant that many simply take the tools developed by others and apply them to problems with little knowledge as to the applicability or accuracy of the methods. Numerical algorithms appear as neatly packaged computer programs that are regarded by the user as "black boxes" into which they feed their data and from which come the publishable results. The complexity of many of the problems dealt with in this manner makes determining the validity of the results nearly impossible. This book is an attempt to correct some of these problems.

Some may regard this effort as a survey and to that I would plead guilty. But I do not regard the word survey as pejorative for to survey, condense, and collate, the knowledge of man is one of the responsibilities of the scholar. There is an implication inherent in this responsibility that the information be made more comprehensible so that it may more readily be assimilated. The extent to which I have succeeded in this goal I will leave to the reader. The discussion of so many topics may be regarded by some to be an impossible task. However, the subjects I have selected have all been required of me during my professional career and I suspect most research scientists would make a similar claim.

Unfortunately few of these subjects were ever covered in even the introductory level of treatment given here during my formal education and certainly they were never placed within a coherent context of numerical analysis.

The basic format of the first chapter is a very wide ranging view of some concepts of mathematics based loosely on axiomatic set theory and linear algebra. The intent here is not so much to provide the specific mathematical foundation for what follows, which is done as needed throughout the text, but rather to establish, what I call for lack of a better term, "mathematical sophistication". There is a general acquaintance with mathematics that a student should have before embarking on the study of numerical methods. The student should realize that there is a subject called mathematics which is artificially broken into sub-disciplines such a linear algebra, arithmetic, calculus, topology, set theory, etc. All of these disciplines are related and the sooner the student realizes that and becomes aware of the relations, the sooner mathematics will become a convenient and useful language of scientific expression. The ability to use mathematics in such a fashion is largely what I mean by "mathematical sophistication". However, this book is primarily intended for scientists and engineers so while there is a certain familiarity with mathematics that is assumed, the rigor that one expects with a formal mathematical presentation is lacking. Very little is proved in the traditional mathematical sense of the word. Indeed, derivations are resorted to mainly to emphasize the assumptions that underlie the results. However, when derivations are called for, I will often write several forms of the same expression on the same line. This is done simply to guide the reader in the direction of a mathematical development. I will often give "rules of thumb" for which there is no formal proof. However, experience has shown that these "rules of thumb" almost always apply. This is done in the spirit of providing the researcher with practical ways to evaluate the validity of his or her results.

The basic premise of this book is that it can serve as the basis for a wide range of courses that discuss numerical methods used in science. It is meant to support a series of lectures, not replace them. To reflect this, the subject matter is wide ranging and perhaps too broad for a single course. It is expected that the instructor will neglect some sections and expand on others. For example, the social scientist may choose to emphasize the chapters on interpolation, curve-fitting and statistics, while the physical scientist would stress those chapters dealing with numerical quadrature and the solution of differential and integral equations. Others might choose to spend a large amount of time on the principle of least squares and its ramifications. All these approaches are valid and I hope all will be served by this book. While it is customary to direct a book of this sort at a specific pedagogic audience, I find that task somewhat difficult. Certainly advanced undergraduate science and engineering students will have no difficulty dealing with the concepts and level of this book. However, it is not at all obvious that second year students couldn't cope with the material. Some might suggest that they have not yet had a formal course in differential equations at that point in their career and are therefore not adequately prepared. However, it is far from obvious to me that a student’s first encounter with differential equations should be in a formal mathematics course. Indeed, since most equations they are liable to encounter will require a numerical solution, I feel the case can be made that it is more practical for them to be introduced to the subject from a graphical and numerical point of view. Thus, if the instructor exercises some care in the presentation of material, I see no real barrier to using this text at the second year level in some areas. In any case I hope that the student will at least be exposed to the wide range of the material in the book lest he feel that numerical analysis is limited only to those topics of immediate interest to his particular specialty.

Nowhere is this philosophy better illustrated that in the first chapter where I deal with a wide range of mathematical subjects. The primary objective of this chapter is to show that mathematics is "all of a piece". Here the instructor may choose to ignore much of the material and jump directly to the solution of linear equations and the second chapter. However, I hope that some consideration would be given to discussing the material on matrices presented in the first chapter before embarking on their numerical manipulation. Many will feel the material on tensors is irrelevant and will skip it. Certainly it is not necessary to understand covariance and contravariance or the notion of tensor and vector densities in order to numerically interpolate in a table of numbers. But those in the physical sciences will generally recognize that they encountered tensors for the first time too late in their educational experience and that they form the fundamental basis for understanding vector algebra and calculus. While the notions of set and group theory are not directly required for the understanding of cubic splines, they do form a unifying basis for much of mathematics. Thus, while I expect most instructors will heavily select the material from the first chapter, I hope they will encourage the students to at least read through the material so as to reduce their surprise when the see it again.

The next four chapters deal with fundamental subjects in basic numerical analysis. Here, and throughout the book, I have avoided giving specific programs that carry out the algorithms that are discussed. There are many useful and broadly based programs available from diverse sources. To pick specific packages or even specific computer languages would be to unduly limit the student's range and selection. Excellent packages are contain in the IMSL library and one should not overlook the excellent collection provided along with the book by Press et al. (see reference 4 at the end of Chapter 2). In general collections compiled by users should be preferred for they have at least been screened initially for efficacy.

Chapter 6 is a lengthy treatment of the principle of least squares and associated topics. I have found that algorithms based on least squares are among the most widely used and poorest understood of all algorithms in the literature. Virtually all students have encountered the concept, but very few see and understand its relationship to the rest of numerical analysis and statistics. Least squares also provides a logical bridge to the last chapters of the book. Here the huge field of statistics is surveyed with the hope of providing a basic understanding of the nature of statistical inference and how to begin to use statistical analysis correctly and with confidence. The foundation laid in Chapter 7 and the tests presented in Chapter 8 are not meant to be a substitute for a proper course of study in the subject. However, it is hoped that the student unable to fit such a course in an already crowded curriculum will at least be able to avoid the pitfalls that trap so many who use statistical analysis without the appropriate care.

Throughout the book I have tried to provide examples integrated into the text of the more difficult algorithms. In testing an earlier version of the book, I found myself spending most of my time with students giving examples of the various techniques and algorithms. Hopefully this initial shortcoming has been overcome. It is almost always appropriate to carry out a short numerical example of a new method so as to test the logic being used for the more general case. The problems at the end of each chapter are meant to be generic in nature so that the student is not left with the impression that this algorithm or that is only used in astronomy or biology. It is a fairly simple matter for an instructor to find examples in diverse disciplines that utilize the techniques discussed in each chapter. Indeed, the student should be encouraged to undertake problems in disciplines other than his/her own if for no other reason than to find out about the types of problems that concern those disciplines.

Here and there throughout the book, I have endeavored to convey something of the philosophy of numerical analysis along with a little of the philosophy of science. While this is certainly not the central theme of the book, I feel that some acquaintance with the concepts is essential to anyone aspiring to a career in science. Thus I hope those ideas will not be ignored by the student on his/her way to find some tool to solve an immediate problem. The philosophy of any subject is the basis of that subject and to ignore it while utilizing the products of that subject is to invite disaster.

There are many people who knowingly and unknowingly had a hand in generating this book. Those at the Numerical Analysis Department of the University of Wisconsin who took a young astronomy student and showed him the beauty of this subject while remaining patient with his bumbling understanding have my perpetual gratitude. My colleagues at The Ohio State University who years ago also saw the need for the presentation of this material and provided the environment for the development of a formal course in the subject. Special thanks are due Professor Philip C. Keenan who encouraged me to include the sections on statistical methods in spite of my shortcomings in this area. Peter Stoychoeff has earned my gratitude by turning my crude sketches into clear and instructive drawings. Certainly the students who suffered through this book as an experimental text have my admiration and well as my thanks.

George W. Collins, II
September 11, 1990

A Note Added for the Internet Edition

A significant amount of time has passed since I first put this effort together. Much has changed in Numerical Analysis. Researchers now seem often content to rely on packages prepared by others even more than they did a decade ago. Perhaps this is the price to be paid by tackling increasingly ambitious problems. Also the advent of very fast and cheap computers has enabled investigators to use inefficient methods and still obtain answers in a timely fashion. However, with the avalanche of data about to descend on more and more fields, it does not seem unreasonable to suppose that numerical tasks will overtake computing power and there will again be a need for efficient and accurate algorithms to solve problems. I suspect that many of the techniques described herein will be rediscovered before the new century concludes. Perhaps efforts such as this will still find favor with those who wish to know if numerical results can be believed.

George W. Collins, II
January 30, 2001

A Further Note for the Internet Edition

Since I put up a version of this book two years ago, I have found numerous errors which largely resulted from the generations of word processors through which the text evolved. During the last effort, not all the fonts used by the text were available in the word processor and PDF translator. This led to errors that were more wide spread that I realized. Thus, the main force of this effort is to bring some uniformity to the various software codes required to generate the version that will be available on the internet. Having spent some time converting *Fundamentals of Stellar Astrophysics* and *The Virial Theorem in Stellar Astrophysics* to Internet compatibility, I have learned to better understand the problems of taking old manuscripts and setting then in the contemporary format. Thus I hope this version of my Numerical Analysis book will be more error free and therefore useable. Will I have found all the errors? That is most unlikely, but I can assure the reader that the number of those errors is significantly reduced from the earlier version. In addition, I have attempted to improve the presentation of the equations and other aspects of the book so as to make it more attractive to the reader. All of the software coding for the index was lost during the travels through various word processors. Therefore, the current version was prepared by means of a page comparison between an earlier correct version and the current presentation. Such a table has an intrinsic error of at least ± 1 page and the index should be used with that in mind. However, it should be good enough to guide the reader to general area of the desired subject.

Having re-read the earlier preface and note I wrote, I find I still share the sentiments expressed therein. Indeed, I find the flight of the student to "black-box" computer programs to obtain solutions to problems has proceeded even faster than I thought it would. Many of these programs such as MATHCAD are excellent and provide quick and generally accurate 'first looks' at problems. However, the researcher would be well advised to understand the methods used by the "black-boxes" to solve their problems. This effort still provides the basis for many of the operations contained in those commercial packages and it is hoped will provide the researcher with the knowledge of their applicability to his/her particular problem. However, it has occurred to me that there is an additional view provided by this book. Perhaps, in the future, a historian may wonder what sort of numerical skills were expected of a researcher in the mid twentieth century. In my opinion, the contents of this book represent what I feel scientists and engineers of the mid twentieth century should have known and many did. I am confident that the knowledge-base of the mid twenty first century scientist will be quite different. One can hope that the difference will represent an improvement.

Finally, I would like to thank John Martin and Charles Knox who helped me adapt this version for the Internet and the Astronomy Department at the Case Western Reserve University for making the server-space available for the PDF files. As is the case with other books I have put on the Internet, I encourage anyone who is interested to down load the PDF files as they may be of use to them. I would only request that they observe the courtesy of proper attribution should they find my efforts to be of use.

George W. Collins, II
April, 2003
Case Western Reserve University

1

Introduction and Fundamental Concepts

• • •

The numerical expression of a scientific statement has traditionally been the manner by which scientists have verified a theoretical description of the physical world. During this century there has been a revolution in both the nature and extent to which this numerical comparison can be made. Indeed, it seems likely that when the history of this century is definitively written, it will be the development of the computer, which will be regarded as its greatest technological achievement - not nuclear power. While it is true that the origins of the digital computer can be traced through the work of Isaac Babbitt, Hermann Hollerith, and others in the nineteenth century, the real advance came after the Second World War when machines were developed that were able to carry out an extended sequence of instructions at a rate that was very much greater than a human could manage. We call such machines programmable.

The electronic digital computer of the sort developed by John von Neumann and others in the 1950s really ushered in the present computer revolution. While it is still to soon to delineate the form and consequences of this revolution, it is already clear that it has forever changed the way in which science and engineering will be done. The entire approach to numerical analysis has changed in the past two decades and that change will most certainly continue rapidly into the future. Prior to the advent of the electronic digital computer, the emphasis in computing was on short cuts and methods of verification which insured that computational errors could be caught before they propagated through the solution. Little attention was paid to "round off error" since the "human computer" could easily control such problems when they were encountered. Now the reliability of electronic machines has nearly eliminated concerns of random error, but round off error can be a persistent problem.

The extreme speed of contemporary machines has tremendously expanded the scope of numerical problems that may be considered as well as the manner in which such computational problems may even be approached. However, this expansion of the degree and type of problem that may be numerically solved has removed the scientist from the details of the computation. For this, most would shout "Hooray"! But this removal of the investigator from the details of computation may permit the propagation of errors of various types to intrude and remain undetected. Modern computers will almost always produce numbers, but whether they represent the solution to the problem or the result of error propagation may not be obvious. This situation is made worse by the presence of programs designed for the solution of broad classes of problems. Almost every class of problems has its pathological example for which the standard techniques will fail. Generally little attention is paid to the recognition of these pathological cases which have an uncomfortable habit of turning up when they are least expected.

Thus the contemporary scientist or engineer should be skeptical of the answers presented by the modern computer unless he or she is completely familiar with the numerical methods employed in obtaining that solution. In addition, the solution should always be subjected to various tests for "reasonableness". There is often a tendency to regard the computer and the programs which they run as "black boxes" from which come infallible answers. Such an attitude can lead to catastrophic results and belies the attitude of "healthy skepticism" that should pervade all science. It is necessary to understand, at least at some level, what the "Black Boxes" do. That understanding is one of the primary aims of this book.

It is not my intention to teach the techniques of programming a computer. There are many excellent texts on the multitudinous languages that exist for communicating with a computer. I will assume that the reader has sufficient capability in this area to at least conceptualize the manner by which certain processes could be communicated to the computer or at least recognize a computer program that does so. However, the programming of a computer does represent a concept that is not found in most scientific or mathematical presentations. We will call that concept an algorithm. An algorithm is simply a sequence of mathematical operations which, when preformed in sequence, lead to the numerical answer to some specified problem. Much time and effort is devoted to ascertaining the conditions under which a particular algorithm will work. In general, we will omit the proof and give only the results when they are known. The use of algorithms and the ability of computers to carry out vastly more operations in a short interval of time than the human programmer could do in several lifetimes leads to some unsettling differences between numerical analysis and other branches of mathematics and science.

Much as the scientist may be unwilling to admit it, some aspects of art creep into numerical analysis. Knowing when a particular algorithm will produce correct answers to a given problem often involves a non-trivial amount of experience as well as a broad based knowledge of machines and computational procedures. The student will achieve some feeling for this aspect of numerical analysis by considering problems for which a given algorithm should work, but doesn't. In addition, we shall give some "rules of thumb" which indicate when a particular numerical method is failing. Such "rules of thumb" are not guarantees of either success or failure of a specific procedure, but represent instances when a greater height of skepticism on the part of the investigator may be warranted.

As already indicated, a broad base of experience is useful when trying to ascertain the validity of the results of any computer program. In addition, when trying to understand the utility of any algorithm for calculation, it is useful to have as broad a range of mathematical knowledge as possible. Mathematics is

indeed the language of science and the more proficient one is in the language the better. So a student should realize as soon as possible that there is essentially one subject called mathematics, which for reasons of convenience we break down into specific areas such as arithmetic, algebra, calculus, tensors, group theory, etc. The more areas that the scientist is familiar with, the more he/she may see the relations between them. The more the relations are apparent, the more useful mathematics will be. Indeed, it is all too common for the modern scientist to flee to a computer for an answer. I cannot emphasize too strongly the need to analyze a problem thoroughly before any numerical solution is attempted. Very often a better numerical approach will suggest itself during the analyses and occasionally one may find that the answer has a closed form analytic solution and a numerical solution is unnecessary.

However, it is too easy to say "I don't have the background for this subject" and thereby never attempt to learn it. The complete study of mathematics is too vast for anyone to acquire in his or her lifetime. Scientists simply develop a base and then continue to add to it for the rest of their professional lives. To be a successful scientist one cannot know too much mathematics. In that spirit, we shall "review" some mathematical concepts that are useful to understanding numerical methods and analysis. The word review should be taken to mean a superficial summary of the area mainly done to indicate the relation to other areas. Virtually every area mentioned has itself been a subject for many books and has occupied the study of some investigators for a lifetime. This short treatment should not be construed in any sense as being complete. Some of this material will indeed be viewed as elementary and if thoroughly understood may be skimmed. However many will find some of these concepts as being far from elementary. Nevertheless they will sooner or later be useful in understanding numerical methods and providing a basis for the knowledge that mathematics is "all of a piece".

1.1 Basic Properties of Sets and Groups

Most students are introduced to the notion of a set very early in their educational experience. However, the concept is often presented in a vacuum without showing its relation to any other area of mathematics and thus it is promptly forgotten. Basically *a set is a collection of elements*. The notion of an element is left deliberately vague so that it may represent anything from cows to the real numbers. The number of elements in the set is also left unspecified and may or may not be finite. Just over a century ago Georg Cantor basically founded set theory and in doing so clarified our notion of infinity by showing that there are different types of infinite sets. He did this by generalizing what we mean when we say that two sets have the same number of elements. Certainly if we can identify each element in one set with a unique element in the second set and there are none left over when the identification is completed, then we would be entitled in saying that the two sets had the same number of elements. Cantor did this formally with the infinite set composed of the positive integers and the infinite set of the real numbers. He showed that it is not possible to identify each real number with a integer so that there are more real numbers than integers and thus different degrees of infinity which he called cardinality. He used the first letter of the Hebrew alphabet to denote the cardinality of an infinite set so that the integers had cardinality $\aleph_0$ and the set of real numbers had cardinality of $\aleph_1$. Some of the brightest minds of the twentieth century have been concerned with the properties of infinite sets.

Our main interest will center on those sets which have constraints placed on their elements for it will be possible to make some very general statements about these restricted sets. For example, consider a set

wherein the elements are related by some "law". Let us denote the "law" by the symbol ‡. If two elements are combined under the "law" so as to yield another element in the set, the set is said to be closed with respect to that law. Thus if a, b, and c are elements of the set and

$$a‡b = c \ , \tag{1.1.1}$$

then the set is said to be closed with respect to ‡. We generally consider ‡ to be some operation like + or ×, but we shouldn't feel that the concept is limited to such arithmetic operations alone. Indeed, one might consider operations such as b 'follows' a to be an example of a law operating on a and b.

If we place some additional conditions of the elements of the set, we can create a somewhat more restricted collection of elements called a group. Let us suppose that one of the elements of the set is what we call a unit element. Such an element is one which, when combined with any other element of the set under the law, produces that same element. Thus

$$a‡i = a \ . \tag{1.1.2}$$

This suggests another useful constraint, namely that there are elements in the set that can be designated "*inverses*". An inverse of an element is one that when combined with its element under the law produces the unit element or

$$a^{-1}‡a = i \ . \tag{1.1.3}$$

Now with one further restriction on the law itself, we will have all the conditions required to produce a group. The restriction is known as *associativity*. A law is said to be associative if the order in which it is applied to three elements does not determine the outcome of the application. Thus

$$(a‡b)‡c = a‡(b‡c) \ . \tag{1.1.4}$$

If a set possess a unit element and inverse elements and is closed under an associative law, that set is called a group under the law. Therefore the normal integers form a group under addition. The unit is zero and the inverse operation is clearly subtraction and certainly the addition of any two integers produces another integer. The law of addition is also associative. However, it is worth noting that the integers do not form a group under multiplication as the inverse operation (reciprocal) does not produce a member of the group (an integer). One might think that these very simple constraints would not be sufficient to tell us much that is new about the set, but the notion of a group is so powerful that an entire area of mathematics known as group theory has developed. It is said that Eugene Wigner once described all of the essential aspects of the thermodynamics of heat transfer on one sheet of paper using the results of group theory.

While the restrictions that enable the elements of a set to form a group are useful, they are not the only restrictions that frequently apply. The notion of commutivity is certainly present for the laws of addition and scalar multiplication and, if present, may enable us to say even more about the properties of our set. A law is said to be *communitative* if

$$a‡b = b‡a \ . \tag{1.1.5}$$

A further restriction that may be applied involves two laws say ‡ and ∧. These laws are said to be distributive with respect to one another if

$$a‡(b∧c) = (a‡b)∧(a‡c) \ . \tag{1.1.6}$$

Although the laws of addition and scalar multiplication satisfy all three restrictions, we will encounter common laws in the next section that do not. Subsets that form a group under addition and scalar

multiplication are called *fields*. The notion of a field is very useful in science as most theoretical descriptions of the physical world are made in terms of fields. One talks of gravitational, electric, and magnetic fields in physics. Here one is describing scalars and vectors whose elements are real numbers and for which there are laws of addition and multiplication which cause these quantities to form not just groups, but fields. Thus all the abstract mathematical knowledge of groups and fields is available to the scientist to aid in understanding physical fields.

1.2 Scalars, Vectors, and Matrices

In the last section we mentioned specific sets of elements called scalars and vectors without being too specific about what they are. In this section we will define the elements of these sets and the various laws that operate on them. In the sciences it is common to describe phenomena in terms of specific quantities which may take on numerical values from time to time. For example, we may describe the atmosphere of the planet at any point in terms of the temperature, pressure, humidity, ozone content or perhaps a pollution index. Each of these items has a single value at any instant and location and we would call them scalars. The common laws of arithmetic that operate on scalars are addition and multiplication. As long as one is a little careful not to allow division by zero (often known as the cancellation law) such scalars form not only groups, but also fields.

Although one can generally describe the condition of the atmosphere locally in terms of scalar fields, the location itself requires more than a single scalar for its specification. Now we need two (three if we include altitude) numbers, say the latitude and longitude, which locate that part of the atmosphere for further description by scalar fields. A quantity that requires more than one number for its specification may be called a vector. Indeed, some have defined a vector as an "*ordered n-tuple of numbers*". While many may not find this too helpful, it is essentially a correct statement, which emphasizes the multi-component side of the notion of a vector. The number of components that are required for the vector's specification is usually called the *dimensionality of the vector*. We most commonly think of vectors in terms of spatial vectors, that is, vectors that locate things in some coordinate system. However, as suggested in the previous section, vectors may represent such things as an electric or magnetic field where the quantity not only has a magnitude or scalar length associated with it at every point in space, but also has a direction. As long as such quantities obey laws of addition and some sort of multiplication, they may indeed be said to form vector fields. Indeed, there are various types of products that are associated with vectors. The most common of these and the one used to establish the field nature of most physical *vector fields* is called the "*scalar product*" or inner product, or sometimes simply the dot product from the manner in which it is usually written. Here the result is a scalar and we can operationally define what we mean by such a product by

$$\vec{A} \bullet \vec{B} = c = \sum_i A_i B_i \quad . \tag{1.2.1}$$

One might say that as the result of the operation is a scalar not a vector, but that would be to put to restrictive an interpretation on what we mean by a vector. Specifically, any scalar can be viewed as vector having only one component (i.e. a 1-dimensional vector). Thus scalars become a subgroup of vectors and since the vector scalar product degenerates to the ordinary scalar product for 1-dimensional vectors, they are actually a sub-field of the more general notion of a vector field.

It is possible to place additional constraints (laws) on a field without destroying the field nature of the elements. We most certainly do this with vectors. Thus we can define an additional type of product known as the "vector product" or simply cross product again from the way it is commonly written. Thus in Cartesian coordinates the cross product can be written as

$$\vec{A}\times\vec{B}=\begin{vmatrix}\hat{i} & \hat{j} & \hat{k}\\ A_i & A_j & A_k\\ B_i & B_j & B_k\end{vmatrix}=\hat{i}(A_jB_k-A_kB_j)-\hat{j}(A_iB_k-A_kB_i)+\hat{k}(A_iB_j-A_jB_i)\,. \qquad (1.2.2)$$

The result of this operation is a vector, but we shall see later that it will be useful to sharpen our definition of vectors so that this result is a special kind of vector.

Finally, there is the "tensor product" or vector outer product that is defined as

$$\left.\begin{aligned}&\vec{A}\vec{B}=\mathbf{C}\\ &C_{ij}=A_iB_j\end{aligned}\right\}\,. \qquad (1.2.3)$$

Here the result of applying the "law" is an ordered array of (n×m) numbers where n and m are the dimensions of the vectors $\vec{A}$ and $\vec{B}$ respectively. Again, here the result of applying the law is not a vector in any sense of the normal definition, but is a member of a larger class of objects we will call tensors. But before discussing tensors in general, let us consider a special class of them known as matrices.

The result of equation (1.2.3) while needing more than one component for its specification is clearly not simply a vector with dimension (n×m). The values of n and m are separately specified and to specify only the product would be to throw away information that was initially specified. Thus, in order to keep this information, we can represent the result as an array of numbers having n columns and m rows. Such an array can be called a matrix. For matrices, the products already defined have no simple interpretation. However, there is an additional product known as a *matrix product*, which will allow us to at least define a matrix group. Consider the product defined by

$$\left.\begin{aligned}&\mathbf{AB}=\mathbf{C}\\ &C_{ij}=\sum_k A_{ik}B_{kj}\end{aligned}\right\}\,. \qquad (1.2.4)$$

With this definition of a product, the unit matrix denoted by **1** will have elements δ_{ij} specified for n = m = 2 by

$$\delta_{ij}=\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}\,. \qquad (1.2.5)$$

The quantity δ_{ij} is called the *Kronecker delta* and may be generalized to n-dimensions.

Thus the inverse elements of the group will have to satisfy the relation

$$\mathbf{A}\mathbf{A}^{-1}=\mathbf{1}\,, \qquad (1.2.6)$$

and we shall spend some time in the next chapter discussing how these members of the group may be calculated. Since matrix addition can simply be defined as the scalar addition of the elements of the matrix,

and the 'unit' matrix under addition is simply a matrix with zero elements, it is tempting to think that the group of matrices also form a field. However, the matrix product as defined by equation (1.2.4), while being distributive with respect to addition, is not communitative. Thus we shall have to be content with matrices forming a group under both addition and matrix multiplication but not a field.

There is much more that can be said about matrices as was the case with other subjects of this chapter, but we will limit ourselves to a few properties of matrices which will be particularly useful later. For example, the transpose of a matrix with elements $\mathbf{A}_{ij}$ is defined as

$$\mathbf{A}^{T} = A_{ji} \quad . \tag{1.2.7}$$

We shall see that there is an important class of matrices (i.e. the orthonormal matrices) whose inverse is their transpose. This makes the calculation of the inverse trivial.

Another important scalar quantity is the *trace of a matrix* defined as

$$\mathrm{Tr}\mathbf{A} = \sum_{i} A_{ii} \quad . \tag{1.2.8}$$

A matrix is said to be *symmetric* if $A_{i\,j} = A_{ji}$. If, in addition, the elements are themselves complex numbers, then should the elements of the transpose be the complex conjugates of the original matrix, the matrix is said to be *Hermitian* or *self-adjoint*. The conjugate transpose of a matrix $\mathbf{A}$ is usually denoted by $\mathbf{A}^{\dagger}$. If the Hermitian conjugate of $\mathbf{A}$ is also $\mathbf{A}^{-1}$, then the matrix is said to be *unitary*. Should the matrix $\mathbf{A}$ commute with it Hermitian conjugate so that

$$\mathbf{A}\mathbf{A}^{\dagger} = \mathbf{A}^{\dagger}\mathbf{A} \quad , \tag{1.2.9}$$

then the matrix is said to be *normal*. For matrices with only real elements, Hermitian is the same as symmetric, unitary means the same as orthonormal and both classes would be considered to be normal.

Finally, a most important characteristic of a matrix is its *determinant*. It may be calculated by expansion of the matrix by "minors" so that

$$\det A = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{13} & a_{23} & a_{33} \end{vmatrix} = a_{11}(a_{22}a_{33} - a_{23}a_{32}) - a_{12}(a_{21}a_{33} - a_{23}a_{31}) + a_{13}(a_{21}a_{32} - a_{22}a_{13}) . \tag{1.2.10}$$

Fortunately there are more straightforward ways of calculating the determinant which we will consider in the next chapter. There are several theorems concerning determinants that are useful for the manipulation of determinants and which we will give without proof.

1. *If each element in a row or column of a matrix is zero, the determinant of the matrix is zero.*

2. *If each element in a row or column of a matrix is multiplied by a scalar q, the determinant is multiplied by q.*

3. *If each element of a row or column is a sum of two terms, the determinant equals the sum of the two corresponding determinants.*

4. *If two rows or two columns are proportional, the determinant is zero. This clearly follows from theorems 1, 2 and 3.*

5. *If two rows or two columns are interchanged, the determinant changes sign.*

6. *If rows and columns of a matrix are interchanged, the determinant of the matrix is unchanged.*

7. *The value of a determinant of a matrix is unchanged if a multiple of one row or column is added to another.*

8. *The determinant of the product of two matrices is the product of the determinants of the two matrices.*

One of the important aspects of the determinant is that it is a single parameter that can be used to characterize the matrix. Any such single parameter (i.e. the sum of the absolute value of the elements) can be so used and is often called a matrix norm. We shall see that various matrix norms are useful in determining which numerical procedures will be useful in operating on the matrix. Let us now consider a broader class of objects that include scalars, vectors, and to some extent matrices.

1.3 Coordinate Systems and Coordinate Transformations

There is an area of mathematics known as *topology,* which deals with the description of spaces. To most students the notion of a space is intuitively obvious and is restricted to the three dimensional Euclidian space of every day experience. A little reflection might persuade that student to include the flat plane as an allowed space. However, a little further generalization would suggest that any time one has several independent variables that they could be used to form a space for the description of some phenomena. In the area of topology the notion of a space is far more general than that and many of the more exotic spaces have no known counterpart in the physical world.

We shall restrict ourselves to spaces of independent variables, which generally have some physical interpretation. These variables can be said to constitute a coordinate frame, which describes the space and are fairly high up in the hierarchy of spaces catalogued by topology. To understand what is meant by a coordinate frame, imagine a set of rigid rods or vectors all connected at a point. We shall call such a collection of rods a reference frame. If every point in space can be projected onto the rods so that a unique set of rod-points represent the space point, the vectors are said to span the space.

If the vectors that define the space are locally perpendicular, they are said to form an orthogonal coordinate frame. If the vectors defining the reference frame are also unit vectors say $\hat{e}_i$ then the condition for orthogonality can be written as

$$\hat{e}_i \bullet \hat{e}_j = \delta_{ij} \quad , \tag{1.3.1}$$

where δ_{ij} is the Kronecker delta. Such a set of vectors will span a space of dimensionality equal to the

number of vectors $\hat{e}_j$. Such a space need not be Euclidian, but if it is then the coordinate frame is said to be a *Cartesian coordinate frame*. The conventional xyz-coordinate frame is Cartesian, but one could imagine such a coordinate system drawn on a rubber sheet, and then distorted so that locally the orthogonality conditions are still met, but the space would no longer be Euclidian or Cartesian.

Of the orthogonal coordinate systems, there are several that are particularly useful for the description of the physical world. Certainly the most common is the rectangular or Cartesian coordinate frame where coordinates are often denoted by x, y, z or x_1, x_2, x_3. Other common three dimensional frames include spherical polar coordinates (r,θ,φ) and cylindrical coordinates (ρ,ϑ,z). Often the most important part of solving a numerical problem is choosing the proper coordinate system to describe the problem. For example, there are a total of thirteen orthogonal coordinate frames in which Laplace's equation is separable (see Morse and Feshbach[1]).

In order for coordinate frames to be really useful it is necessary to know how to get from one to another. That is, if we have a problem described in one coordinate frame, how do we express that same problem in another coordinate frame? For quantities that describe the physical world, we wish their meaning to be independent of the coordinate frame that we happen to choose. Therefore we should expect the process to have little to do with the problem, but rather involve relationships between the coordinate frames themselves. These relationships are called *coordinate transformations*. While there are many such transformations in mathematics, for the purposes of this summary we shall concern ourselves with *linear transformations*. Such coordinate transformations relate the coordinates in one frame to those in a second frame by means of a system of linear algebraic equations. Thus if a vector $\vec{x}$ in one coordinate system has components x_j, in a primed-coordinate system a vector $\vec{x}'$ to the same point will have components x'_j

$$x_i = \sum_j A_{ij} x_j + B_i \,. \tag{1.3.2}$$

In vector notation we could write this as

$$\vec{x}' = \mathbf{A}\vec{x} + \vec{B} \quad . \tag{1.3.3}$$

This defines the general class of linear transformation where **A** is some matrix and $\vec{B}$ is a vector. This general linear form may be divided into two constituents, the matrix **A** and the vector. It is clear that the vector $\vec{B}$ may be interpreted as a shift in the origin of the coordinate system, while the elements A_{ij} are the cosines of the angles between the axes X_i and X'_i, and are called the directions cosines (see Figure 1.1). Indeed, the vector $\vec{B}$ is nothing more than a vector from the origin of the un-primed coordinate frame to the origin of the primed coordinate frame. Now if we consider two points that are fixed in space and a vector connecting them, then the length and orientation of that vector will be independent of the origin of the coordinate frame in which the measurements are made. That places an additional constraint on the types of linear transformations that we may consider. For instance, transformations that scaled each coordinate by a constant amount, while linear, would change the length of the vector as measured in the two coordinate systems. Since we are only using the coordinate system as a convenient way to describe the vector, the coordinate system can play no role in controlling the length of the vector. Thus we shall restrict our investigations of linear transformations to those that transform orthogonal coordinate systems while preserving the length of the vector.

Thus the matrix **A** must satisfy the following condition

$$\vec{x}' \bullet \vec{x}' = (\mathbf{A}\vec{x}) \bullet (\mathbf{A}\vec{x}) = \vec{x} \bullet \vec{x} \quad , \tag{1.3.4}$$

which in component form becomes

$$\sum_i \left(\sum_j A_{ij} x_j \right) \left(\sum_k A_{ik} x_k \right) = \sum_j \sum \left(\sum_i A_{ij} A_{ik} x_j x_k \right) = \sum_i x_i^2 \quad . \tag{1.3.5}$$

This must be true for all vectors in the coordinate system so that

$$\sum_i A_{ij} A_{ik} = \delta_{jk} = \sum_i A_{ji}^{-1} A_{ik} \quad . \tag{1.3.6}$$

Now remember that the Kronecker delta δ_{ij} is the unit matrix and any element of a group that multiplies another and produces that group's unit element is defined as the inverse of that element. Therefore

$$A_{ji} = [A_{ij}]^{-1} \quad . \tag{1.3.7}$$

Interchanging the rows with the columns of a matrix produces a new matrix which we have called the transpose of the matrix. Thus orthogonal transformations that preserve the length of vectors have inverses that are simply the transpose of the original matrix so that

$$\mathbf{A}^{-1} = \mathbf{A}^{T} \quad . \tag{1.3.8}$$

This means that given the transformation **A** in the linear system of equations (1.3.3), we may invert the transformation, or solve the linear equations, by multiplying those equations by the transpose of the original matrix or

$$\vec{x} = \mathbf{A}^{T} \vec{x}' - \mathbf{A}^{T} \vec{B} \quad . \tag{1.3.9}$$

Such transformations are called orthogonal unitary transformations, or orthonormal transformations, and the result given in equation (1.3.9) greatly simplifies the process of carrying out a transformation from one coordinate system to another and back again.

We can further divide orthonormal transformations into two categories. These are most easily described by visualizing the relative orientation between the two coordinate systems. Consider a transformation that carries one coordinate into the negative of its counterpart in the new coordinate system while leaving the others unchanged. If the changed coordinate is, say, the x-coordinate, the transformation matrix would be

$$\mathbf{A} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad , \tag{1.3.10}$$

which is equivalent to viewing the first coordinate system in a mirror. Such transformations are known as reflection transformations and will take a right handed coordinate system into a left handed coordinate system.

The length of any vectors will remain unchanged. The x-component of these vectors will simply be replaced by its negative in the new coordinate system. However, this will not be true of "vectors" that result from the vector cross product. The values of the components of such a vector will remain unchanged implying that a reflection transformation of such a vector will result in the orientation of that vector being changed. If you will, this is the origin of the "right hand rule" for vector cross products. A left hand rule results in a vector pointing in the opposite direction. Thus such vectors are not invariant to *reflection*

transformations because their orientation changes and this is the reason for putting them in a separate class, namely the axial (pseudo) vectors. It is worth noting that an orthonormal reflection transformation will have a determinant of -1. The unitary magnitude of the determinant is a result of the magnitude of the vector being unchanged by the transformation, while the sign shows that some combination of coordinates has undergone a reflection.

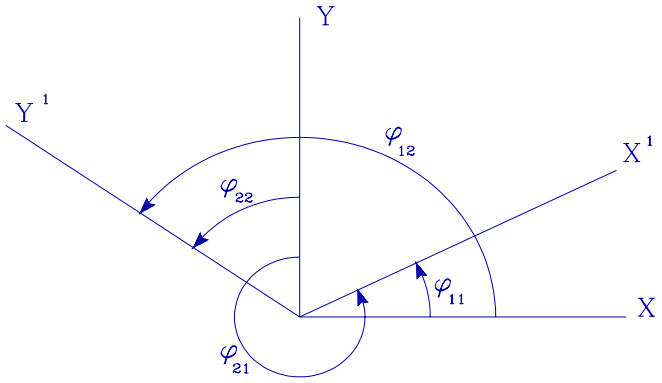


Figure 1.1 shows two coordinate frames related by the transformation angles φ_{ij}. *Four coordinates are necessary if the frames are not orthogonal*

As one might expect, the elements of the second class of orthonormal transformations have determinants of +1. These represent transformations that can be viewed as a rotation of the coordinate system about some axis. Consider a transformation between the two coordinate systems displayed in Figure 1.1. The components of any vector $\vec{C}$ in the primed coordinate system will be given by

$$\begin{pmatrix} C_{x'} \\ C_{y'} \\ C_{z'} \end{pmatrix} = \begin{pmatrix} \cos\varphi_{11} & \cos\varphi_{12} & 0 \\ \cos\varphi_{21} & \cos\varphi_{22} & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} C_x \\ C_y \\ C_z \end{pmatrix} . \tag{1.3.11}$$

If we require the transformation to be orthonormal, then the direction cosines of the transformation will not be linearly independent since the angles between the axes must be $\pi/2$ in both coordinate systems. Thus the angles must be related by

$$\left.\begin{aligned} \varphi_{11} &= \varphi_{22} = \varphi \\ \varphi_{12} &= \varphi_{11} + \pi/2 = \varphi + \pi/2 \\ (2\pi - \varphi_{21}) &= \pi/2 - \varphi_{11} = \pi/2 - \varphi \end{aligned}\right\} . \tag{1.3.12}$$

Using the addition identities for trigonometric functions, equation (1.3.11) can be given in terms of the single angle φ by

$$\begin{pmatrix} C_{x'} \\ C_{y'} \\ C_{z'} \end{pmatrix} = \begin{pmatrix} \cos\varphi & \sin\varphi & 0 \\ -\sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} C_x \\ C_y \\ C_z \end{pmatrix} \quad . \tag{1.3.13}$$

This transformation can be viewed as a simple rotation of the coordinate system about the Z-axis through an angle φ. Thus,

$$\mathrm{Det}\begin{vmatrix} \cos\varphi & \sin\varphi & 0 \\ -\sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{vmatrix} = \cos^2\varphi + \sin^2\varphi = +1 \quad . \tag{1.3.14}$$

In general, the rotation of any Cartesian coordinate system about one of its principal axes can be written in terms of a matrix whose elements can be expressed in terms of the rotation angle. Since these transformations are about one of the coordinate axes, the components along that axis remain unchanged. The rotation matrices for each of the three axes are

$$\left.\begin{aligned} P_x(\phi) &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{pmatrix} \\ P_y(\phi) &= \begin{pmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{pmatrix} \\ P_z(\phi) &= \begin{pmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{aligned}\right\} \quad . \tag{1.3.15}$$

It is relatively easy to remember the form of these matrices for the row and column of the matrix corresponding to the rotation axis always contains the elements of the unit matrix since that component is not affected by the transformation. The diagonal elements always contain the cosine of the rotation angle while the remaining off diagonal elements always contain the sine of the angle modulo a sign. For rotations about the x- or z-axes, the sign of the upper right off diagonal element is positive and the other negative. The situation is just reversed for rotations about the y-axis. So important are these rotation matrices that it is worth remembering their form so that they need not be re-derived every time they are needed.

One can show that it is possible to get from any given orthogonal coordinate system to another through a series of three successive coordinate rotations. Thus a general orthonormal transformation can always be written as the product of three coordinate rotations about the orthogonal axes of the coordinate systems. It is important to remember that the matrix product is not commutative so that the order of the rotations is important.

1.4 Tensors and Transformations

Many students find the notion of tensors to be intimidating and therefore avoid them as much as possible. After all Einstein was once quoted as saying that there were not more than ten people in the world that would understand what he had done when he published General Theory of Relativity. Since tensors are the foundation of general relativity that must mean that they are so esoteric that only ten people could manage them. Wrong! This is a beautiful example of misinterpretation of a quote taken out of context. What Einstein meant was that the notation he used to express the General Theory of Relativity was sufficiently obscure that there were unlikely to be more than ten people who were familiar with it and could therefore understand what he had done. So unfortunately, tensors have generally been represented as being far more complex than they really are. Thus, while readers of this book may not have encountered them before, it is high time they did. Perhaps they will be somewhat less intimidated the next time, for if they have any ambition of really understanding science, they will have to come to an understanding of them sooner or later.

In general a tensor has N^n components or elements. N is known as the dimensionality of the tensor by analogy with vectors, while n is called the rank of the tensor. Thus scalars are tensors of rank zero and vectors of any dimension are rank one. So scalars and vectors are subsets of tensors. We can define the law of addition in the usual way by the addition of the tensor elements. Thus the null tensor (i.e. one whose elements are all zero) forms the unit under addition and arithmetic subtraction is the inverse operation. Clearly tensors form a communitative group under addition. Furthermore, the scalar or dot product can be generalized for tensors so that the result is a tensor of rank $|m-n|$. In a similar manner the outer product can be defined so that the result is a tensor of rank $|m+n|$. It is clear that all of these operations are closed; that is, the results remain tensors. However, while these products are in general distributive, they are not communitative and thus tensors will not form a field unless some additional restrictions are made.

One obvious way of representing tensors of rank 2 is as N×N square matrices Thus, the scalar product of a tensor of rank 2 with a vector would be written as

$$\left.\begin{aligned} \mathbf{A}\bullet\vec{B}&=\vec{C}\\ C_i&=\sum_j A_{ij}B_j \end{aligned}\right\}, \tag{1.4.1}$$

while the tensor outer product of the same tensor and vector could be written as

$$\left.\begin{aligned} \mathbf{AB}&=\vec{\vec{C}}\\ C_{ijk}&=A_{ij}B_k \end{aligned}\right\}. \tag{1.4.2}$$

It is clear from the definition and specifically from equation (1.4.2) that tensors may frequently have

a rank of more than two. However, it becomes more difficult to display all the elements in a simple geometrical fashion so they are generally just listed or described. A particularly important tensor of rank three is known as the *Levi-Civita Tensor* (or correctly the Levi-Civita Tensor Density). It plays a role that is somewhat complimentary to that of the Kronecker delta in that when any two indices are equal the tensor element is zero. When the indices are all different the tensor element is +1 or -1 depending on whether the index sequence can be obtained as an even or odd permutation from the sequence 1, 2, 3 respectively. If we try to represent the tensor ε_{ijk} as a succession of 3×3 matrices we would get

$$\left.\begin{aligned}
\varepsilon_{1jk} &= \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & +1 \\ 0 & -1 & 0 \end{pmatrix} \\
\varepsilon_{2jk} &= \begin{pmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ +1 & 0 & 0 \end{pmatrix} \\
\varepsilon_{3jk} &= \begin{pmatrix} 0 & -1 & 0 \\ +1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}
\end{aligned}\right\} . \qquad (1.4.3)$$

This somewhat awkward looking third rank tensor allows us to write the equally awkward vector cross product in summation notation as

$$\vec{A} \times \vec{B} = \vec{\vec{\varepsilon}} : (\vec{A}\vec{B}) = \sum_j \sum_k \varepsilon_{ijk} A_j B_k = C_i \quad . \qquad (1.4.4)$$

Here the symbol **:** denotes the *double dot product* which is explicitly specified by the double sum of the right hand term. The quantity ε_{ijk} is sometimes called the permutation symbol as it changes sign with every permutation of its indices. This, and the identity

$$\sum_i \varepsilon_{ijk} \varepsilon_{ipq} = \delta_{jp}\delta_{kq} - \delta_{jq}\delta_{kp} \quad , \qquad (1.4.5)$$

makes the evaluation of some complicated vector identities much simpler (see exercise 13).

In section 1.3 we added a condition to what we meant by a vector, namely we required that the length of a vector be invariant to a coordinate transformation. Here we see the way in which additional constraints of what we mean by vectors can be specified by the way in which they transform. We further limited what we meant by a vector by noting that some vectors behave strangely under a reflection transformation and calling these pseudo-vectors. Since the Levi-Civita tensor generates the vector cross product from the elements of ordinary (polar) vectors, it must share this strange transformation property. Tensors that share this transformation property are, in general, known as tensor densities or pseudo-tensors. Therefore we should call ε_{ijk} defined in equation (1.4.3) the Levi-Civita tensor density. Indeed, it is the invariance of tensors, vectors, and scalars to orthonormal transformations that is most correctly used to define the elements of the group called tensors.

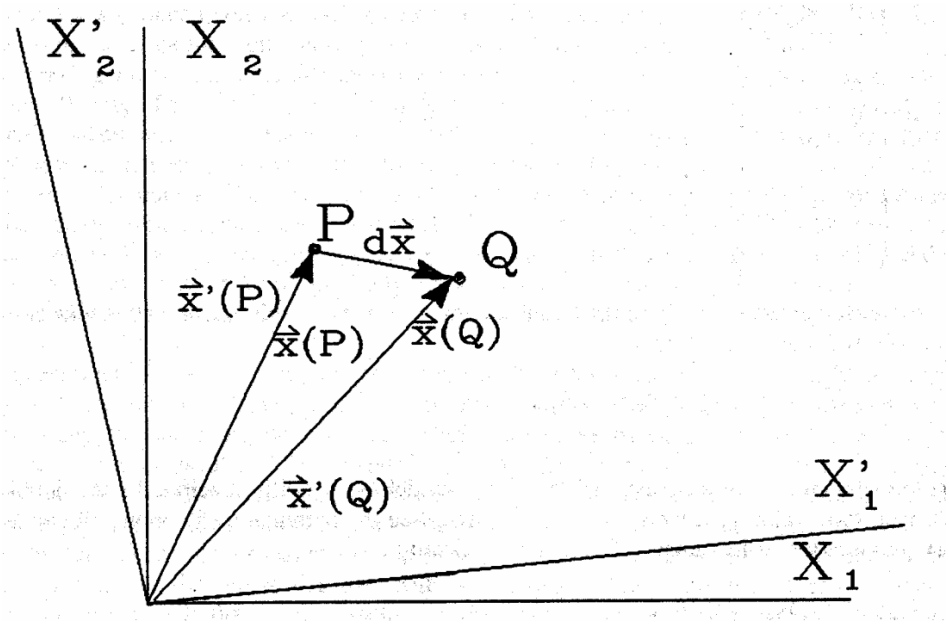


Figure 1.2 shows two neighboring points P *and* Q *in two adjacent coordinate systems* X *and* X'. *The differential distance between the two is* $d\vec{x}$. *The vectorial distance to the two points is* $\vec{X}$ *(P) or* $\vec{X}'$ *(P) and* $\vec{X}$(Q) *or* $\vec{X}'$(Q) *respectively.*

Since vectors are just a special case of the broader class of objects called tensors, we should expect these transformation constraints to extend to the general class. Indeed the only fully appropriate way to define tensors is to define the way in which they transform from one coordinate system to another. To further refine the notion of tensor transformation, we will look more closely at the way vectors transform. We have written a general linear transformation for vectors in equation (1.3.2). However, except for rotational and reflection transformations, we have said little about the nature of the transformation matrix **A**. So let us consider how we would express a coordinate transformation from some point P in a space to a nearby neighboring point Q. Each point can be represented in any coordinate system we choose. Therefore, let us consider two coordinate systems having a common origin where the coordinates are denoted by x_i and x'_i respectively.

Since P and Q are near each other, we can represent the coordinates of Q to those of P in either coordinate system by

$$\left.\begin{aligned} x_i(Q) &= x_i(P) + dx_i \\ x'_i(Q) &= x'_i(P) + dx'_i \end{aligned}\right\} . \tag{1.4.6}$$

Now the coordinates of the vector from P to Q will be dx_i and dx'_i, in the un-primed and primed coordinate systems respectively. By the chain rule the two coordinates will be related by

$$dx'_i = \sum_j \frac{\partial x'_i}{\partial x_j} dx_j \quad . \tag{1.4.7}$$

Note that equation (1.4.7) does not involve the specific location of point Q but rather is a general expression of the local relationship between the two coordinate frames. Since equation (1.4.7) basically describes how the coordinates of P or Q will change from one coordinate system to another, we can identify the elements A_{ij} from equation (1.3.2) with the partial derivatives in equation (1.4.6). Thus we could expect any vector $\vec{x}$ to transform according to

$$x'_i = \sum_j \frac{\partial x'_i}{\partial x_j} x_j \quad . \tag{1.4.8}$$

Vectors that transform in this fashion are called *contravariant vectors*. In order to distinguish them from covariant vectors, which we shall shortly discuss, we shall denote the components of the vector with superscripts instead of subscripts. Thus the correct form for the transformation of a contravariant vector is

$$x'^i = \sum_j \frac{\partial x'^i}{\partial x^j} x^j \quad . \tag{1.4.9}$$

We can generalize this transformation law to contravariant tensors of rank two by

$$T'^{ij} = \sum_k \sum_l T^{kl} \frac{\partial x'^i}{\partial x^l} \frac{\partial x'^j}{\partial x^k} \quad . \tag{1.4.10}$$

Higher rank contravariant tensors transform as one would expect with additional coordinate changes. One might think that the use of superscripts to represent contravariant indices would be confused with exponents, but such is generally not the case and the distinction between this sort of vector transformation and covariance is sufficiently important in physical science to make the accommodation. The sorts of objects that transform in a contravariant manner are those associated with, but not limited to, *geometrical objects*. For example, the infinitesimal displacements of coordinates that makes up the tangent vector to a curve show that it is a contravariant vector. While we have used vectors to develop the notion of contravariance, it is clear that the concept can be extended to tensors of any rank including rank zero. The transformation rule for such a tensor would simply be

$$T' = T \quad . \tag{1.4.11}$$

In other words scalars will be invariant to contravariant coordinate transformations.

Now instead of considering vector representations of geometrical objects imbedded in the space and their transformations, let us consider a scalar function of the coordinates themselves. Let such a function be $\Phi(x^i)$. Now consider components of the gradient of Φ in the x'^i-coordinate frame. Again by the chain rule

$$\frac{\partial \Phi}{\partial x'_i} = \sum_j \frac{\partial x^j}{\partial x_i} \frac{\partial \Phi}{\partial x'^j} \quad . \tag{1.4.12}$$

If we call $\partial\Phi / \partial x'_i$ a vector with components V_i , then the transformation law given by equation (1.4.12) appears very like equation (1.4.8), but with the partial derivatives inverted. Thus we would identify the elements A_i^j of the linear vector transformation represented by equation (1.3.2) as

$$A_i^{\ j} = \partial x^j / \partial x'_i \quad , \tag{1.4.13}$$

and the vector transformation would have the form

$$V_i = \sum_j A_i^{\ j} V_j \quad . \tag{1.4.14}$$

Vectors that transform in this manner are called *covariant vectors*. In order to distinguish them from contravariant vectors, the component indices are written as subscripts. Again, it is not difficult to see how the concept of covariance would be extended to tensors of higher rank and specifically for a second rank covariant tensor we would have

$$T'_{ij} = \sum_k \sum_l T_{lk} \frac{\partial x^l}{\partial x'_i} \frac{\partial x^k}{\partial x'_j} \quad . \tag{1.4.15}$$

The use of the scalar invariant Φ to define what is meant by a covariant vector is a clue as to the types of vectors that behave as covariant vectors. Specifically the gradient of physical scalar quantities such as temperature and pressure would behave as a covariant vector while coordinate vectors themselves are contravariant. Basically equations (1.4.15) and (1.4.10) define what is meant by a covariant or contravariant tensor of second rank. It is possible to have a mixed tensor where one index represents covariant transformation while the other is contravariant so that

$$T'_i{}^j = \sum_k \sum_l T_l{}^k \frac{\partial x^l}{\partial x'_i} \frac{\partial x^j}{\partial x_k} \quad . \tag{1.4.16}$$

Indeed the Kronecker delta can be regarded as a tensor as it is a two index symbol and in particular it is a mixed tensor of rank two and when covariance and contravariance are important should be written as δ^i_j.

Remember that both contravariant and covariant transformations are locally linear transformations of the form given by equation (1.3.2). That is, they both preserve the length of vectors and leave scalars unchanged. The introduction of the terms contravariance and covariance simply generate two subgroups of what we earlier called tensors and defined the members of those groups by means of their detailed transformation properties. One can generally tell the difference between the two types of transformations by noting how the components depend on the coordinates. If the components denote 'distances' or depend directly on the coordinates, then they will transform as contravariant tensor components. However, should the components represent quantities that change with the coordinates such as gradients, divergences, and curls, then dimensionally the components will depend inversely on the coordinates and the will transform covariantly. The use of subscripts and superscripts to keep these transformation properties straight is particularly useful in the development of tensor calculus as it allows for the development of rules for the manipulation of tensors in accord with their specific transformation characteristics. While coordinate systems have been used to define the tensor characteristics, those characteristics are properties of the tensors themselves and do not depend on any specific coordinate frame. This is of considerable importance when developing theories of the physical world as anything that is fundamental about the universe should be independent of man made coordinate frames. This is not to say that the choice of coordinate frames is unimportant when actually solving a problem. Quite the reverse is true. Indeed, as the properties of the physical world represented by tensors are independent of coordinates and their explicit representation and transformation properties from one coordinate system to another are well defined, they may be quite useful in reformulating numerical problems in different coordinate systems.

1.5 Operators

The notion of a mathematical operator is extremely important in mathematical physics and there are entire books written on the subject. Most students first encounter operators in calculus when the notation [d/dx] is introduced to denote the operations involved in finding the derivative of a function. In this instance the operator stands for taking the limit of the difference between adjacent values of some function of x divided by the difference between the adjacent values of x as that difference tends toward zero. This is a fairly complicated set of instructions represented by a relatively simple set of symbols.

The designation of some symbol to represent a collection of operations is said to represent the definition of an operator. Depending on the details of the definition, the operators can often be treated as if they were quantities and subject to algebraic manipulations. The extent to which this is possible is determined by how well the operators satisfy the conditions for the group on which the algebra or mathematical system in question is defined. The operator [d/dx] is a scalar operator. That is, it provides a single result after operating on some function defined in an appropriate coordinate space. It and the operator $\int$ represent the fundamental operators of the infinitesimal calculus. Since [d/dx] and $\int$ carry out inverse operations on functions, one can define an identity operator by $[d/dx]\int$ so that continuous differentiable functions will form a group under the action of these operators.

In numerical analysis there are analogous operators Δ and Σ that perform similar functions but without taking the limit to vanishingly small values of the independent variable. Thus we could define the forward finite difference operator Δ by its operation on some function f(x) so that

$$\Delta f(x) = f(x+h) - f(x) \ , \qquad (1.5.1)$$

where the problem is usually scaled so that h = 1. In a similar manner Σ can be defined as

$$\sum_{i=0}^{n} f(x_i) = f(x) + f(x+h) + f(x+2h) + f(x+ih) \cdots + f(x+nh) \ . \qquad (1.5.2)$$

Such operators are most useful in expressing formulae in numerical analysis. Indeed, it is possible to build up an entire calculus of finite differences. Here the base for such a calculus is 2 instead of e=2.7182818... as in the infinitesimal calculus . Other operators that are useful in the finite difference calculus are the shift operator E[f(x)] and the Identity operator I[f(x)] which are defined as

$$\left.\begin{aligned} E[f(x)] &\equiv f(x+h) \\ I[f(x)] &\equiv f(x) \end{aligned}\right\} \ . \qquad (1.5.3)$$

These operators are not linearly independent as we can write the forward difference operator as

$$\Delta = E - I \ . \qquad (1.5.4)$$

The finite difference and summation calculus are extremely powerful when summing series or evaluating convergence tests for series. Before attempting to evaluate an infinite series, it is useful to know if the series converges. If possible, the student should spend some time studying the calculus of finite differences.

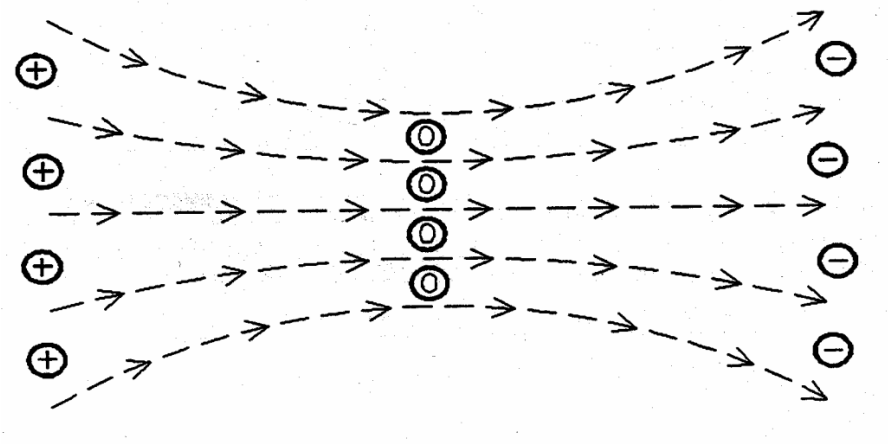
In addition to scalar operators, it is possible to define vector and tensor operators. One of the most common vector operators is the "del" operator or "nabla". It is usually denoted by the symbol ∇ and is defined in Cartesian coordinates as

$$\nabla = \hat{i}\frac{\partial}{\partial x} + \hat{j}\frac{\partial}{\partial y} + \hat{k}\frac{\partial}{\partial z} \quad . \tag{1.5.5}$$

This single operator, when combined with the some of the products defined above, constitutes the foundation of vector calculus. Thus the divergence, gradient, and curl are defined as

$$\left.\begin{aligned} \nabla \bullet \vec{A} &= b \\ \nabla a &= \vec{B} \\ \nabla \times \vec{A} &= \vec{C} \end{aligned}\right\} , \tag{1.5.6}$$

respectively. If we consider $\vec{A}$ to be a continuous vector function of the independent variables that make up the space in which it is defined, then we may give a physical interpretation for both the divergence and curl. The divergence of a vector field is a measure of the amount that the field spreads or contracts at some given point in the space (see Figure 1.3).



.Figure 1.3 schematically shows the divergence of a vector field. In the region where the arrows of the vector field converge, the divergence is positive, implying an increase in the source of the vector field. The opposite is true for the region where the field vectors diverge.

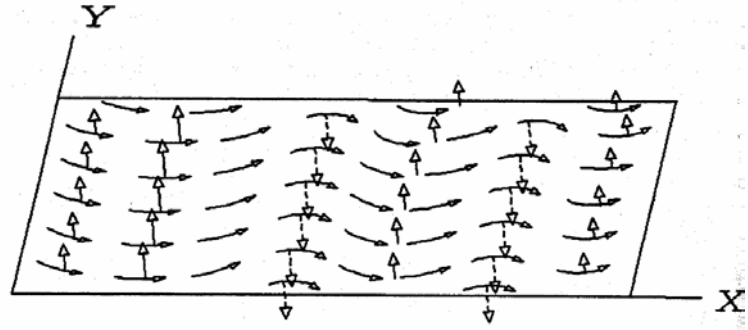


Figure 1.4 schematically shows the curl of a vector field. The direction of the curl is determined by the "right hand rule" while the magnitude depends on the rate of change of the x- *and* y- *components of the vector field with respect to* y *and* x..

The curl is somewhat harder to visualize. In some sense it represents the amount that the field rotates about a given point. Some have called it a measure of the "swirliness" of the field. If in the vicinity of some point in the field, the vectors tend to veer to the left rather than to the right, then the curl will be a vector pointing up normal to the net rotation with a magnitude that measures the degree of rotation (see Figure 1.4). Finally, the gradient of a scalar field is simply a measure of the direction and magnitude of the maximum rate of change of that scalar field (see Figure 1.5).

Figure 1.5 schematically shows the gradient of the scalar dot-density in the form of a number of vectors at randomly chosen points in the scalar field. The direction of the gradient points in the direction of maximum increase of the dot-density while the magnitude of the vector indicates the rate of change of that density.

With these simple pictures in mind and what we developed in section 1.4 it is possible to generalize the notion of the Del-operator to other quantities. Consider the gradient of a vector field. This represents the outer product of the Del-operator with a vector. While one doesn't see such a thing often in freshman physics, it does occur in more advanced descriptions of fluid mechanics (and many other places). We now know enough to understand that the result of this operation will be a tensor of rank two which we can represent as a matrix. What do the components mean? Generalize from the scalar case. The nine elements of the vector gradient can be viewed as three vectors denoting the direction of the maximum rate of change of each of the components of the original vector. The nine elements represent a perfectly well defined quantity and it has a useful purpose in describing many physical situations. One can also consider the divergence of a second rank tensor, which is clearly a vector.

In hydrodynamics, the divergence of the pressure tensor may reduce to the gradient of the scalar gas pressure if the macroscopic flow of the material is small compared to the internal speed of the particles that make up the material. With some care in the definition of a collection of operators, their action on the elements of a field or group will preserve the field or group nature of the original elements. These are the operators that are of the greatest use in mathematical physics.

By combining the various products defined in this chapter with the familiar notions of vector calculus, we can formulate a much richer description of the physical world. This review of scalar and vector mathematics along with the all-too-brief introduction to tensors and matrices will be useful in setting up problems for their eventual numerical solution. Indeed, it is clear from the transformations described in the last sections that a prime aspect in numerically solving problems will be dealing with linear equations and matrices and that will be the subject of the next chapter

Chapter 1 Exercises

1. Show that the rational numbers (not including zero) form a group under addition and multiplication. Do they also form a scalar field?

2. Show that it is not possible to put the rational numbers into a one to one correspondence with the real numbers.

3. Show that the vector cross product is not communitative.

4. Show that the matrix product is not communitative.

5. Is the scalar product of two second rank tensors communitative? If so show how you know. If not, give a counter example.

6. Give the necessary and sufficient conditions for a tensor field.

7. Show that the Kronecker delta δ^i_j is indeed a mixed tensor.

8. Determine the nature (i.e. contravariant, covariant, or mixed) of the Levi-Civita tensor density.

9. Show that the vector cross product does indeed give rise to a pseudo-vector.

10. Use the forward finite difference operator to define a second order finite difference operator and use it to evaluate $\Delta^2[f(x)]$, where $f(x) = x^2 + 5x + 12$.

11. If $g_n(x) = x^{(n)} \equiv x(x-1)(x-2)(x-3)\cdots(x-n+1)$, show that $\Delta[g_n(x)] = n g_{n-1}(x)$. $g_n(x)$ is known as the factorial function.

12. Show that if f(x) is a polynomial of degree n, then it can be expressed as a sum of factorial functions (see problem 11).

13. Show that

$$\sum_i \varepsilon_{i\,j\,k}\varepsilon_{\,i\,p\,q} = \delta_{j\,p}\delta_{k\,q} - \delta_{j\,q}\delta_{k\,p} \quad ,$$

and use the result to prove

$$\nabla \times (\nabla \times \vec{F}) = \nabla(\nabla \bullet \vec{F}) - \nabla^2 \vec{F} \quad .$$

Chapter 1 References and Additional Reading

One of the great books in theoretical physics, and the only one I know that gives a complete list of the coordinate frames for which Laplace's equation is separable is

1. Morse, P.M., and Feshbach, H., "Methods of Theoretical Physics" (1953) McGraw-Hill Book Co., Inc. New York, Toronto, London, pp. 665-666.

It is a rather formidable book on theoretical physics, but any who aspire to a career in the area should be familiar with its contents.

While many books give excellent introductions to modern set and group theory, I have found

2. Andree, R.V., "Selections from Modern Abstract Algebra" (1958) Henry Holt & Co. New York,to be clear and concise. A fairly complete and concise discussion of determinants can be found in

3. Sokolnikoff, I.S., and Redheffer, R.M., "Mathematics of Physics and Modern Engineering" (1958) McGraw-Hill Book Co., Inc. New York, Toronto, London, pp. 741-753.

A particularly clear and approachable book on Tensor Calculus which has been reprinted by Dover is

4. Synge, J.L., and Schild, A., "Tensor Calculus" (1949) University of Toronto Press, Toronto.

I would strongly advise any student of mathematical physics to become familiar with this book before attempting books on relativity theory that rely heavily on tensor notation. While there are many books on operator calculus, a venerable book on the calculus of finite differences is

5. Milne-Thomson, L.M., "The Calculus of Finite Differences" (1933) Macmillan and Co., LTD, London.

A more elementary book on the use of finite difference equations in the social sciences is

6. Goldberg, S., "Introduction to Difference Equations", (1958) John Wiley & Sons, Inc., London.

There are many fine books on numerical analysis and I will refer to many of them in later chapters. However, there are certain books that are virtually unique in the area and foremost is

7. Abramowitz, M. and Stegun, I.A., "Handbook of Mathematical Functions" National Bureau of Standards Applied Mathematics Series 55 (1964) U.S. Government Printing Office, Washington D.C.

While this book has also been reprinted, it is still available from the Government Printing Office and represents an exceptional buy. Approximation schemes and many numerical results have been collected and are clearly presented in this book. One of the more obscure series of books are collectively known as the Bateman manuscripts, or

8. Bateman, H., "The Bateman Manuscript Project" (1954) Ed. A. Erdélyi, 5 Volumns, McGraw-Hill Book Co., Inc. New York, Toronto, London.

Harry Bateman was a mathematician of considerable skill who enjoyed collecting obscure functional relationships. When he died, this collection was organized, catalogued, and published as the Bateman Manuscripts. It is a truly amazing collection of relations. When all else fails in an analysis of a problem, before fleeing to the computer for a solution, one should consult the Bateman Manuscripts to see if the problem could not be transformed to a different more tractable problem by means of one of the remarkable relations collected by Harry Bateman. A book of similar utility but easier to obtain and use is

9. Lebedev, N.N., "Special Functions and Their Applications" (1972), Trans. R.A.Silverman. Dover Publications, Inc. New York.

2

The Numerical Methods for Linear Equations and Matrices

• • •

We saw in the previous chapter that linear equations play an important role in transformation theory and that these equations could be simply expressed in terms of matrices. However, this is only a small segment of the importance of linear equations and matrix theory to the mathematical description of the physical world. Thus we should begin our study of numerical methods with a description of methods for manipulating matrices and solving systems of linear equations. However, before we begin any discussion of numerical methods, we must say something about the accuracy to which those calculations can be made.

2.1 Errors and Their Propagation

One of the most reliable aspects of numerical analysis programs for the electronic digital computer is that they almost always produce numbers. As a result of the considerable reliability of the machines, it is common to regard the results of their calculations with a certain air of infallibility. However, the results can be no better than the method of analysis and implementation program utilized by the computer and these are the works of highly fallible man. This is the origin of the aphorism "*garbage in – garbage out*". Because of the large number of calculations carried out by these machines, small errors at any given stage can rapidly propagate into large ones that destroy the validity of the result.

We can divide computational errors into two general categories: the first of these we will call round off error, and the second truncation error. Round off error is perhaps the more insidious of the two and is always present at some level. Indeed, its omnipresence indicates the first problem facing us. How accurate an answer do we require? Digital computers utilize a certain number of digits in their calculations and this base number of digits is known as the precision of the machine. Often it is possible to double or triple the number of digits and hence the phrase "double" or "triple" precision is commonly used to describe a calculation carried out using this expanded number of digits. It is a common practice to use more digits than are justified by the problem simply to be sure that one has "got it right". For the scientist, there is a subtle danger in this in that the temptation to publish all the digits presented by the computer is usually overwhelming. Thus published articles often contain numerical results consisting of many more decimal places than are justified by the calculation or the physics that went into the problem. This can lead to some reader unknowingly using the results at an unjustified level of precession thereby obtaining meaningless conclusions. Certainly the full machine precession is never justified, as after the first arithmetical calculation, there will usually be some uncertainty in the value of the last digit. This is the result of the first kind of error we called *round off error*. As an extreme example, consider a machine that keeps only one significant figure and the exponent of the calculation so that 6+3 will yield 9×10^0. However, 6+4, 6+5, and 6+8 will all yield the same answer namely 1×10^1. Since the machine only carries one digit, all the other information will be lost. It is not immediately obvious what the result of 6+9, or 7+9 will be. If the result is 2×10^1, then the machine is said to round off the calculation to the nearest significant digit. However, if the result remains 1×10^1, then the machine is said to truncate the addition to the nearest significant digit. Which is actually done by the computer will depend on both the physical architecture (hardware) of the machine and the programs (software) which instruct it to carry out the operation. Should a human operator be carrying out the calculation, it would usually be possible to see when this is happening and allow for it by keeping additional significant figures, but this is generally not the case with machines. Therefore, we must be careful about the propagation of round off error into the final computational result. It is tempting to say that the above example is only for a 1-digit machine and therefore unrealistic. However, consider the common 6-digit machine. It will be unable to distinguish between 1 million dollars and 1 million and nine dollars. Subtraction of those two numbers would yield zero. This would be significant to any accountant at a bank. Repeated operations of this sort can lead to a completely meaningless result in the first digit.

This emphasizes the question of 'how accurate an answer do we need?'. For the accountant, we clearly need enough digits to account for all the money at a level decided by the bank. For example, the Internal Revenue Service allows taxpayers to round all calculations to the nearest dollar. This sets a lower

bound for the number of significant digits. One's income usually sets the upper bound. In the physical world very few constants of nature are known to more than four digits (the speed of light is a notable exception). Thus the results of physical modeling are rarely important beyond four figures. Again there are exceptions such as in null experiments, but in general, scientists should not deceive themselves into believing their answers are better answers than they are.

How do we detect the effects of round off error? Entire studies have been devoted to this subject by considering that round off errors occurs in basically a random fashion. Although computers are basically deterministic (i.e. given the same initial state, the computer will always arrive at the same answer), a large collection of arithmetic operations can be considered as producing a random collection of round-ups and round-downs. However, the number of digits that are affected will also be variable, and this makes the problem far more difficult to study in general. Thus in practice, when the effects of round off error are of great concern, the problem can be run in double precession. Should both calculations yield the same result at the acceptable level of precession, then round off error is probably not a problem. An additional "rule of thumb" for detecting the presence of round off error is the appearance of a large number of zeros at the right-hand side of the answers. Should the number of zeros depend on parameters of the problem that determine the size or numerical extent of the problem, then one should be concerned about round off error. Certainly one can think of exceptions to this rule, but in general, they are just that - exceptions.

The second form of error we called *truncation error* and it should not be confused with errors introduced by the "truncation" process that happens half the time in the case of round off errors. This type of error results from the inability of the approximation method to properly represent the solution to the problem. The magnitude of this kind of error depends on both the nature of the problem and the type of approximation technique. For example, consider a numerical approximation technique that will give exact answers should the solution to the problem of interest be a polynomial (we shall show in chapter 3 that the majority of methods of numerical analysis are indeed of this form). Since the solution is exact for polynomials, the extent that the correct solution differs from a polynomial will yield an error. However, there are many different kinds of polynomials and it may be that a polynomial of higher degree approximates the solution more accurately than one of lower degree.

This provides a hint for the practical evaluation of truncation errors. If the calculation is repeated at different levels of approximation (i.e. for approximation methods that are correct for different degree polynomials) and the answers change by an unacceptable amount, then it is likely that the truncation error is larger than the acceptable amount. There are formal ways of estimating the truncation error and some 'black-box' programs do this. Indeed, there are general programs for finding the solutions to differential equations that use estimates of the truncation error to adjust parameters of the solution process to optimize efficiency. However, one should remember that these estimates are just that - estimates subject to all the errors of calculation we have been discussing. It many cases the correct calculation of the truncation error is a more formidable problem than the one of interest. In general, it is useful for the analyst to have some prior knowledge of the behavior of the solution to the problem of interest before attempting a detailed numerical solution. Such knowledge will generally provide a 'feeling' for the form of the truncation error and the extent to which a particular numerical technique will manage it.

We must keep in mind that both round-off and truncation errors will be present at some level in any calculation and be wary lest they destroy the accuracy of the solution. The acceptable level of accuracy is

determined by the analyst and he must be careful not to aim too high and carry out grossly inefficient calculations, or too low and obtain meaningless results.

We now turn to the solution of linear algebraic equations and problems involving matrices associated with those solutions. In general we can divide the approaches to the solution of linear algebraic equations into two broad areas. The first of these involve algorithms that lead directly to a solution of the problem after a finite number of steps while the second class involves an initial "guess" which then is improved by a succession of finite steps, each set of which we will call an iteration. If the process is applicable and properly formulated, a finite number of iterations will lead to a solution.

2.2 Direct Methods for the Solution of Linear Algebraic Equations

In general, we may write a system of linear algebraic equations in the form

$$\left.\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n &= c_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2n}x_n &= c_2 \\ a_{31}x_1 + a_{312}x_2 + a_{33}x_3 + \cdots + a_{3n}x_n &= c_3 \\ \cdot \quad \cdot \quad \cdot \quad \cdot \quad \cdot & \\ \cdot \quad \cdot \quad \cdot \quad \cdot \quad \cdot & \\ \cdot \quad \cdot \quad \cdot \quad \cdot \quad \cdot & \\ a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \cdots + a_{nn}x_n &= c_n \end{aligned}\right\}, \tag{2.2.1}$$

which in vector notation is

$$\mathbf{A}\vec{x} = \vec{c} \ . \tag{2.2.2}$$

Here x is an n-dimensional vector the elements of which represent the solution of the equations. $\vec{c}$ is the constant vector of the system of equations and **A** is the matrix of the system's coefficients.

We can write the solution to these equations as

$$\vec{x} = \mathbf{A}^{-1}\vec{c} \ , \tag{2.2.3}$$

thereby reducing the solution of any algebraic system of linear equations to finding the inverse of the coefficient matrix. We shall spend some time describing a number of methods for doing just that. However, there are a number of methods that enable one to find the solution without finding the inverse of the matrix. Probably the best known of these is *Cramer's Rule*

a. Solution by Cramer's Rule

It is unfortunate that usually the only method for the solution of linear equations that students remember from secondary education is Cramer's rule or expansion by minors. As we shall see, this method is rather inefficient and relatively difficult to program for a computer. However, as it forms sort of a standard by which other methods can by judged, we will review it here. In Chapter 1 [equation (1.2.10)] we gave the form for the determinant of a 3×3 matrix. The more general definition is inductive so that the determinant of the matrix **A** would be given by

$$\text{Det } \mathbf{A} = \sum_{i=1}^{n} (-1)^{i+j} a_{ij} M_{ij} \text{ , } \forall j \quad . \tag{2.2.4}$$

Here the summation may be taken over either i or j, or indeed, any monotonically increasing sequence of both. The quantity M_{ij} is the determinant of the matrix **A** with the ith row and jth column removed and, with the sign carried by $(-1)^{(i+j)}$ is called the *cofactor* of the *minor* element a_{ij}. With all this terminology, we can simply write the determinant as

$$\text{Det } \mathbf{A} = \sum_{i-1}^{n} C_{ij} a_{ij} \quad , \quad \forall j \ , = \sum_{j=1}^{n} a_{ij} C_{ij} \ , \forall i \quad . \tag{2.25}$$

By making use of theorems 2 and 7 in section 1.2, we can write the solution in terms of the determinant of **A** as

$$x_1 \begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ a_{n1} & a_{n2} & & a_{nn} \end{vmatrix} = \begin{vmatrix} a_{11}x_1 & a_{12} & \cdots & a_{1n} \\ a_{21}x_1 & a_{22} & \cdots & a_{2n} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ a_{n1}x_1 & a_{n2} & & a_{nn} \end{vmatrix} = \begin{vmatrix} (a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n) & a_{12} & \cdots & a_{1n} \\ (a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n) & a_{22} & \cdots & a_{2n} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ (a_{1n}x_1 + a_{1n}x_2 + \cdots + a_{1n}x_n) & a_{n2} & & a_{nn} \end{vmatrix}$$

$$= \begin{vmatrix} c_1 & a_{12} & \cdots & a_{1n} \\ c_2 & a_{22} & \cdots & a_{2n} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ c_n & a_{n2} & & a_{nn} \end{vmatrix} \quad , \tag{2.2.6}$$

which means that the general solution of equation (2.2.1) is given by

$$x_j = \begin{vmatrix} a_{11} \cdots a_{1\,j-1} c_1 & a_{1\,j+1} \cdots a_{1n} \\ a_{21} \cdots a_{2\,j-1} c_2 & a_{2\,j+1} \cdots a_{2n} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ a_{n1} \cdots a_{n\,j-1} c_n & a_{n\,j+1} \cdots a_{nn} \end{vmatrix} \times \ [\text{Det } \mathbf{A}]^{-1} \quad . \tag{2.2.7}$$

This requires evaluating the determinant of the matrix **A** as well as an augmented matrix where the jth column has been replaced by the elements of the constant vector c_i. Evaluation of the determinant of an n×n matrix requires about $3n^2$ operations and this must be repeated for each unknown, thus solution by Cramer's rule will require at least $3n^3$ operations. In addition, to maintain accuracy, an optimum path through the matrix (finding the least numerically sensitive cofactors) will require a significant amount of logic. Thus, solution by Cramer's rule is not a particularly desirable approach to the numerical solution of linear equations either for a computer or a hand calculation. Let us consider a simpler algorithm, which forms the basis for one of the most reliable and stable direct methods for the solution of linear equations. It also provides a method for the inversion of matrices. Let begin by describing the method and then trying to understand why it works.

b. Solution by Gaussian Elimination

Consider representing the set of linear equations given in equation (2.2.1) by

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ a_{n1} & a_{n2} & & a_{nn} \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ \cdot \\ \cdot \\ c_n \end{pmatrix} . \tag{2.2.8}$$

Here we have suppressed the presence of the elements of the solution vector x_j. Now we will perform a series of operations on the rows and columns of the coefficient matrix **A** and we shall carry through the row operations to include the elements of the constant vector c_i. In other words, we shall treat the rows as if they were indeed the equations so that anything done to one element is done to all. One begins by dividing each row including the constant element by the lead element in the row. The first row is then subtracted from all the lower rows. Thus all rows but the first will have zero in the first column. Now repeat these operations for all but the first equation starting with the second element of the second equation producing ones in the second column of the remaining equations. Subtracting the resulting second line from all below will yield zeros in the first two columns of equation three and below. This process can be repeated until one has arrived at the last line representing the last equation. When the diagonal coefficient there is unity, the last term of the constant vector contains the value of x_n. This can be used in the (n-1)th equation represented by the second to the last line to obtain x_{n-1} and so on right up to the first line which will yield the value of x_1. The name of this method simply derives from the elimination of each unknown from the equations below it producing a triangular system of equations represented by

$$\begin{pmatrix} 1 & a'_{12} & \cdots & a'_{1n} \\ 0 & 1 & \cdots & a'_{2n} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ 0 & 0 \cdot & \cdot & \cdot\ 1 \end{pmatrix} \begin{pmatrix} c'_1 \\ c'_2 \\ \cdot \\ \cdot \\ c'_n \end{pmatrix} , \tag{2.2.9}$$

which can then be easily solved by back substitution where

$$\left.\begin{aligned} x_n &= c'_n \\ x_i &= c'_i - \sum_{j=i+1}^{n} a'_{ij}\, x_j \end{aligned}\right\} . \tag{2.2.10}$$

One of the disadvantages of this approach is that errors (principally round off errors) from the successive subtractions build up through the process and accumulate in the last equation for x_n. The errors thus incurred are further magnified by the process of back substitution forcing the maximum effects of the round-off error into x_1. A simple modification to this process allows us to more evenly distribute the effects of round off error yielding a solution of more uniform accuracy. In addition, it will provide us with an efficient mechanism for calculation of the inverse of the matrix **A**.

c. Solution by Gauss Jordan Elimination

Let us begin by writing the system of linear equations as we did in equation (2.2.8), but now include a unit matrix with elements δ_{ij} on the right hand side of the expression. Thus,

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ a_{n1} & a_{n2} & & a_{nn} \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ \cdot \\ \cdot \\ c_n \end{pmatrix} \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ 0 & 0 & \cdots & 1 \end{pmatrix} . \tag{2.2.11}$$

We will treat the elements of this matrix as we do the elements of the constant vector c_i. Now proceed as we did with the Gauss elimination method producing zeros in the columns below and to the left of the diagonal element. However, in addition to subtracting the line whose diagonal element has been made unity from all those below it, also subtract from the equations above it as well. This will require that these equations be normalized so that the corresponding elements are made equal to one and the diagonal element will no longer be unity. In addition to operating on the rows of the matrix **A** and the elements of $\vec{C}$, we will operate on the elements of the additional matrix which is initially a unit matrix. Carrying out these operations row by row until the last row is completed will leave us with a system of equations that resemble

$$\begin{pmatrix} a'_{11} & 0 & \cdots & 0 \\ 0 & a'_{22} & \cdots & 0 \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ 0 & 0 & \cdots & a'_{nn} \end{pmatrix} \begin{pmatrix} c'_1 \\ c'_2 \\ \cdot \\ \cdot \\ c'_n \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ b_{n1} & b_{n2} & \cdots & b_{nn} \end{pmatrix} . \tag{2.2.12}$$

If one examines the operations we have performed in light of theorems 2 and 7 from section 1.2, it is clear that so far we have done nothing to change the determinant of the original matrix **A** so that expansion by minors of the modified matrix represent by the elements a'_{ij} is simply accomplished by multiplying the diagonal elements a_{ii} together. A final step of dividing each row by a'_{ii} will yield the unit matrix on the left hand side and elements of the solution vector x_i will be found where the C'_i s were. The final elements of **B** will be the elements of the inverse matrix of **A**. Thus we have both solved the system of equations and found the inverse of the original matrix by performing the same steps on the constant vector as well as an additional unit matrix. Perhaps the simplest way to see why this works is to consider the system of linear equations and what the operations mean to them. Since all the operations are performed on entire rows including the constant vector, it is clear that they constitute legal algebraic operations that won't change the nature of the solution in any way. Indeed these are nothing more than the operations that one would preform by hand if he/she were solving the system by eliminating the appropriate variables. We have simply formalized that procedure so that it may be carried out in a systematic fashion. Such a procedure lends itself to computation by machine and may be relatively easily programmed. The reason for the algorithm yielding the matrix inverse is somewhat less easy to see. However, the product of **A** and **B** will be the unit matrix **1**, and the operations that go into that matrix-multiply are the inverse of those used to generate **B**.

To see specifically how the Gauss-Jordan routine works, consider the following system of equations:

$$\left.\begin{aligned} x_1 + 2x_2 + 3x_3 &= 12 \\ 3x_1 + 2x_2 + x_3 &= 24 \\ 2x_1 + x_2 + 3x_3 &= 36 \end{aligned}\right\} . \tag{2.2.13}$$

If we put this in the form required by expression (2.2.11) we have

$$\begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \\ 2 & 1 & 3 \end{pmatrix}\begin{pmatrix} 12 \\ 24 \\ 36 \end{pmatrix}\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} . \tag{2.2.14}$$

Now normalize the all rows by factoring out the lead elements of the first column so that

$$(1)(3)(2)\begin{pmatrix} 1 & 2 & 3 \\ 1 & \frac{2}{3} & \frac{1}{3} \\ 1 & \frac{1}{2} & \frac{3}{2} \end{pmatrix}\begin{pmatrix} 12 \\ 8 \\ 18 \end{pmatrix}\begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{3} & 0 \\ 0 & 0 & \frac{1}{2} \end{pmatrix} . \tag{2.2.15}$$

The first row can then be subtracted from the remaining rows (i.e. rows 2 and 3) to yield

$$(6)\begin{pmatrix} 1 & 2 & 3 \\ 0 & -\frac{4}{3} & -\frac{8}{3} \\ 0 & -\frac{3}{2} & -\frac{3}{2} \end{pmatrix}\begin{pmatrix} 12 \\ -4 \\ +6 \end{pmatrix}\begin{pmatrix} 1 & 0 & 0 \\ -1 & \frac{1}{3} & 0 \\ -1 & 0 & \frac{1}{2} \end{pmatrix} . \tag{2.2.16}$$

Now repeat the cycle normalizing by factoring out the elements of the second column getting

$$(6)\left(\frac{-4}{3}\right)\left(\frac{-3}{2}\right)(2)\begin{pmatrix} \frac{1}{2} & 1 & \frac{3}{2} \\ 0 & 1 & 2 \\ 0 & 1 & 1 \end{pmatrix}\begin{pmatrix} +6 \\ +3 \\ -4 \end{pmatrix}\begin{pmatrix} \frac{1}{2} & 0 & 0 \\ \frac{3}{4} & \frac{1}{4} & 0 \\ \frac{2}{3} & 0 & -\frac{1}{3} \end{pmatrix} . \tag{2.2.17}$$

Subtracting the second row from the remaining rows (i.e. rows 1 and 3) gives

$$(24)\begin{pmatrix} \frac{1}{2} & 0 & -\frac{1}{2} \\ 0 & 1 & 2 \\ 0 & 1 & -1 \end{pmatrix}\begin{pmatrix} +3 \\ +3 \\ -7 \end{pmatrix}\begin{pmatrix} -\frac{1}{4} & \frac{1}{4} & 0 \\ \frac{3}{4} & -\frac{1}{4} & 0 \\ -\frac{1}{12} & \frac{1}{4} & -\frac{1}{3} \end{pmatrix} . \tag{2.2.18}$$

Again repeat the cycle normalizing by the elements of the third column so

$$(24)(-1/2)(2)(-1)\begin{pmatrix} -1 & 0 & 1 \\ 0 & \frac{1}{2} & 1 \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} -6 \\ \frac{3}{2} \\ +7 \end{pmatrix}\begin{pmatrix} \frac{1}{2} & -\frac{1}{2} & 0 \\ \frac{3}{8} & -\frac{1}{8} & 0 \\ \frac{1}{12} & -\frac{1}{4} & \frac{1}{3} \end{pmatrix} , \tag{2.2.19}$$

and subtract from the remaining rows to yield

$$(24)\begin{pmatrix} -1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} -13 \\ -11\frac{1}{2} \\ +7 \end{pmatrix}\begin{pmatrix} \frac{5}{12} & -\frac{1}{4} & -\frac{1}{3} \\ \frac{7}{24} & \frac{1}{8} & -\frac{1}{3} \\ \frac{1}{12} & -\frac{1}{4} & \frac{1}{3} \end{pmatrix} . \tag{2.2.20}$$

Finally normalize by the remaining elements so as to produce the unit matrix on the left hand side so that

$$(24)(-1)(1/2)(+1)\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} +13 \\ -11 \\ +7 \end{pmatrix}\begin{pmatrix} -5/12 & 1/4 & 1/3 \\ 7/24 & 1/8 & -1/3 \\ 1/12 & -1/4 & 1/3 \end{pmatrix}. \tag{2.2.21}$$

The solution to the equations is now contained in the center vector while the right hand matrix contains the inverse of the original matrix that was on the left hand side of expression (2.2.14). The scalar quantity accumulating at the front of the matrix is the determinant as it represents factors of individual rows of the original matrix. Here we have repeatedly use theorem 2 and 7 given in section (1.2) in chapter 1. Theorem 2 allows us to build up the determinant by factoring out elements of the rows, while theorem 7 guarantees that the row subtraction shown in expressions (2.2.16), (2.2.18), and (2.2.20) will not change the value of the determinant. Since the determinant of the unit matrix on left side of expression (2.2.21) is one, the determinant of the original matrix is just the product of the factored elements. Thus our complete solution is

$$\left.\begin{aligned} \vec{x} &= [13, -11, +7] \\ \text{Det } \mathbf{A} &= -12 \\ \mathbf{A}^{-1} &= \begin{pmatrix} -5/12 & 1/4 & 1/3 \\ 7/12 & 1/4 & -2/3 \\ 1/12 & -1/4 & 1/3 \end{pmatrix} \end{aligned}\right\}. \tag{2.2.22}$$

In carrying out this procedure, we have been careful to maintain full accuracy by keeping the fractions that explicitly appear as a result of the division. In general, this will not be practical and the perceptive student will have notice that there is the potential for great difficulty as a result of the division. Should any of the elements of the matrix **A** be zero when they are to play the role of divisor, then a numerical singularity will result. Indeed, should the diagonal elements be small, division would produce such large row elements that subtraction of them from the remaining rows would generate significant roundoff error. However, interchanging two rows or two columns of a system of equations doesn't alter the solution of these equations and, by theorem 5 of chapter 1 (sec 1.2), only the sign of the determinant is changed. Since the equations at each step represent a system of equations, which have the same solution as the original set, we may interchange rows and columns at any step in the procedure without altering the solution. Thus, most Gauss-Jordan programs include a search of the matrix to place the largest element on the diagonal prior to division by that element so as to minimize the effects of round off error. Should it be impossible to remove a zero from the division part of this algorithm, the one column of the matrix can be made to be completely zero. Such a matrix has a determinant, which is zero and the matrix is said to be *singular*. Systems of equations that are characterized by singular matrices have no unique solution.

It is clear that one could approach the singular state without actually reaching it. The result of this would be to produce a solution of only marginal accuracy. In such circumstances the initial matrix might have coefficients with six significant figures and the solution have one or less. While there is no *a priori* way of knowing how nearly singular the matrix may be, there are several "rules of thumb" which while not guaranteed to resolve the situation, generally work. First consider some characteristic of the matrix that measures the typical size of its elements. Most any reasonable criterion will do such as the absolute value of

the largest element, the sum of the absolute values of the elements, or possibly the trace. Divide this characteristic by the absolute value of the determinant and if the result exceeds the machine precision, the result of the solution should be regarded with suspicion. Thus if we denote this characteristic of the matrix by M, then

$$N \gtrsim \log_{10} \left| M/d \right| \quad , \tag{2.2.23}$$

where d is the determinant of the original matrix. This should be regarded as a necessary, but not sufficient, condition for the solution to be accurate. Indeed a rough guess as to the number of significant figures in the resultant solution is

$$N_s \sim N - \log_{10} \left| M/d \right| \quad . \tag{2.2.24}$$

Since most Gauss-Jordan routines return the determinant as a byproduct of the solution, it is irresponsible to fail to check to see if the solution passes this test.

An additional test would be the substitution of the solution back into the original equations to see how accurately the elements of the constant vector are reproduced. For the inverse matrix, one can always multiply the original matrix by the inverse and see to what extent the unit matrix results. This raises an interesting question. What do we mean when we say that a solution to a system of equations is accurate. One could mean that each element of the solution vector contains a certain number of significant figures, or one might mean that the solution vector satisfies the equations at some acceptable level of accuracy (i.e. all elements of the constant vector are reproduced to some predetermined number of significant figures). It is worth noting that these two conditions are not necessarily the same. Consider the situation of a poorly conditioned system of equations where the constant vector is only weakly specified by one of the unknowns. Large changes in its value will make little change in the elements of the constant vector so that tight tolerances on the constant vector will not yield values of the that particular unknown with commensurate accuracy. This system would not pass the test given by equation (2.2.23). In general, there should always be an *a priori* specification of the required accuracy of the solution and an effort must be made to ascertain if that level of accuracy has been reached.

d. Solution by Matrix Factorization: The Crout Method

Consider two triangular matrices **U** and **V** with the following properties

$$\left.\begin{aligned} \mathbf{U} &= \begin{pmatrix} u_{ij} & i \le j \\ 0 & i > j \end{pmatrix} \\ \mathbf{V} &= \begin{pmatrix} 0 & i < j \\ v_{ij} & i \ge j \end{pmatrix} \end{aligned}\right\} \quad . \tag{2.2.25}$$

Further assume that **A** can be written in terms of these triangular matrices so that

$$\mathbf{A} = \mathbf{VU} \quad . \tag{2.2.26}$$

Then our linear system of equations [equation (2.2.2)] could be written as

$$\mathbf{A}\vec{x} = \vec{c} = \mathbf{V}\left(\mathbf{U}\vec{x}\right) \quad . \tag{2.2.27}$$

Multiplying by $\mathbf{V}^{-1}$ we have that the solution will be given by a different set of equations

$$\mathbf{U}\vec{x} = \mathbf{V}^{-1}\vec{c} = \vec{c}' \quad , \tag{2.2.28}$$

where

$$\vec{c} = \mathbf{V}\vec{c}' \quad . \tag{2.2.29}$$

If the vector $\vec{c}'$ can be determined, then equation (2.2.28) has the form of the result of the Gauss elimination and would resemble expression (2.2.9) and have a solution similar to equation (2.2.10). In addition, equation (2.2.29) is triangular and has a similarly simple solution for the vector $\vec{c}'$. Thus, we have replaced the general system of linear equations by two triangular systems. Now the constraints on **U** and **V** only depend on the matrix **A** and the triangular constraints. In no way do they depend on the constant vector $\vec{c}$. Thus, if one has a large number of equations differing only in the constant vector, the matrices **U** and **V** need only be found once. \

The matrices **U** and **V** can be found from the matrix **A** in a fairly simple way by

$$\left.\begin{aligned} u_{ij} &= a_{ij} - \sum_{k=1}^{i-1} v_{ik} u_{kj} \\ v_{ij} &= \left(a_{ij} - \sum_{k=1}^{j-1} v_{ik} u_{kj} \right) \Big/ u_{ii} \end{aligned}\right\} , \tag{2.2.30}$$

which is justified by Hildebrandt[1]. The solution of the resulting triangular equations is then just

$$\left.\begin{aligned} c'_i &= \left(c_i - \sum_{k=1}^{i-1} v_{ik} c'_k \right) \Big/ v_{ii} \\ x_i &= \left(c'_i - \sum_{k=i+1}^{n} u_{ik} x_k \right) \Big/ u_{ii} \end{aligned}\right\} \quad . \tag{2.2.31}$$

Both equations (2.2.30) and (2.2.31) are recursive in nature in that the unknown relies on previously determined values of the same set of unknowns. Thus round-off error will propagate systematically throughout the solution. So it is useful if one attempts to arrange the initial equations in a manner which minimizes the error propagation. However, the method involves a minimum of readily identifiable divisions and so tends to be exceptionally stable. The stability will clearly be improved as long as the system of equations contains large diagonal elements. Therefore the Crout method provides a method of similar or greater stability to Gauss-Jordan method and considerable efficiency in dealing with systems differing only in the constant vector. In instances where the matrix **A** is symmetric the equations for u_{ij} simplify to

$$u_{ij} = v_{ji}/u_{ii} \quad . \tag{2.2.32}$$

As we shall see the normal equations for the least squares formalism always have this form so that the Crout method provides a good basis for their solution.

While equations (2.2.30) and (2.2.31) specifically delineate the elements of the factored matrices **U** and **V**, it is useful to see the manner in which they are obtained. Therefore let us consider the same equations that served as an example for the Gauss-Jordan method [i.e. equations (2.2.13)]. In order to implement the Crout method we wish to be able to express the coefficient matrix as

$$\mathbf{A} = \mathbf{VU} = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \\ 2 & 1 & 3 \end{pmatrix} = \begin{pmatrix} v_{11} & 0 & 0 \\ v_{12} & v_{22} & 0 \\ v_{13} & v_{23} & v_{33} \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{pmatrix} . \tag{2.2.33}$$

The constant vector $\vec{c}$ that appears in equation (2.2.31) is

$$\vec{c} = (\,12, 24, 36\,) \ . \tag{2.2.34}$$

To factor the matrix **A** into the matrices **U** and **V** in accordance with equation (2.2.30), we proceed column by column through the matrix so that the necessary elements of **U** and **V** required by equation (2.2.30) are available when they are needed. Carrying out the factoring process specified by equations (2.2.30) sequentially column by column yields

$$\left.\begin{aligned} u_{11} &= a_{11} - 0 = 1 \\ v_{11} &= (a_{11} - 0) \big/ u_{11} = 1 \\ v_{12} &= (a_{12} - 0) \big/ u_{11} = 3 \\ v_{13} &= (a_{13} - 0) \big/ u_{11} = 2 \end{aligned}\right\} \quad j = 1$$

$$\left.\begin{aligned} u_{12} &= a_{12} - 0 = 2 \\ u_{22} &= [a_{22} - (v_{21} u_{21})] = 2 - (3 \times 2) = 4 \\ v_{22} &= [a_{22} - (v_{21} u_{12})] \big/ u_{22} = [2 - (3 \times 2)] \big/ -4 = 1 \\ v_{32} &= [a_{32} - (v_{31} u_{13})] \big/ u_{22} = [1 - (2 \times 2)] \big/ -4 = \frac{3}{4} \end{aligned}\right\} \quad j = 2$$

$$\left.\begin{aligned} u_{13} &= a_{13} - 0 = 3 \\ u_{23} &= a_{23} - (v_{21} u_{13}) = 1 - (3 \times 3) = -8 \\ u_{33} &= a_{33} - (v_{31} u_{13} + v_{32} u_{23}) = 3 - [(2 \times 3) + (\tfrac{3}{4} \times -8)] = 3 \\ v_{33} &= [a_{33} - (v_{31} u_{13} + v_{32} u_{23})] \big/ u_{33} = [3 - (2 \times 3) - (-8 \times \tfrac{3}{4})] / 3 = 1 \end{aligned}\right\} \quad j = 3 \tag{2.2.35}$$

Therefore we can write the original matrix **A** in accordance with equation (2.2.33) as

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 \\ 3 & 1 & 0 \\ 2 & \frac{3}{4} & 1 \end{pmatrix}\begin{pmatrix} 1 & 2 & 3 \\ 0 & -4 & -8 \\ 0 & 0 & 3 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 \\ 3 & (6-4) & (9-8) \\ 2 & (4-3) & (6-6+3) \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \\ 2 & 1 & 3 \end{pmatrix} . \qquad (2.2.36)$$

Here the explicit multiplication of the two factored matrices **U** and **V** demonstrates that the factoring has been done correctly.

Now we need to obtain the augmented constant vector $\vec{c}'$ specified by equations (2.2.31). These equations must be solved recursively so that the results appear in the order in which they are needed. Thus

$$\left.\begin{aligned} c'_1 &= (c_1 - 0)/v_{11} = 12/1 = 12 \\ c'_2 &= [c_2 - (v_{21}c'_1)]/v_{22} = [24 - (3\times 12)]/1 = -12 \\ c'_3 &= [c_3 - (v_{31}c'_1 + v_{32}c'_2)]/v_{33} = [36 - (2\times 2) + (12\times \tfrac{3}{4})]/1 = 1 \end{aligned}\right\} . \qquad (2.2.37)$$

Finally the complete solution can be obtained by back-solving the second set of equations (2.2.31) so that

$$\left.\begin{aligned} x_3 &= c'_3/u_{33} = 21/3 = 7 \\ x_2 &= (c'_2 - u_{23}x_3)/u_{22} = [-12 + (8\times 7)]/(-4) = -11 \\ x_1 &= (c'_1 - u_{12}x_2 - u_{13}x_3)/u_{11} = [12 - (2\times -11) - (3\times 7)]/1 = 13 \end{aligned}\right\} . \qquad (2.2.38)$$

As anticipated, we have obtained the same solution as in equation (2.2.22). The strength of the Crout method resides in the minimal number of operations required to solve a second set of equations differing only in the constant vector. The factoring of the matrix remains the same and only the steps specified by equations (2.2.37) and (2.2.38) need be repeated. In addition, the method is particularly stable.

e. The Solution of Tri-diagonal Systems of Linear Equations

All the methods described so far generally require about n^3 operations to obtain the solution. However, there is one frequently occurring system of equations for which extremely efficient solution algorithms exist. This system of equations is called tri-diagonal because there are never more than three unknowns in any equation and they can be arranged so that the coefficient matrix is composed of non-zero elements on the main diagonal and the diagonal immediately adjacent to either side. Thus such a system would have the form

$$\left.\begin{array}{l}
a_{11}x_1 + a_{12}x_2 + 0 + 0 + \cdots + 0 = c_1 \\
a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + 0 + \cdots + 0 = c_2 \\
0 + a_{32}x_2 + a_{33}x_3 + a_{34}x_4 + \cdots + 0 = c_3 \\
\vdots \\
\vdots \\
\vdots \\
0 + 0 + 0 + 0 + \cdots + a_{n-1\,n-2}x_{n-2} + a_{n-1\,n-1}x_{n-1} + a_{n-1\,n}x_n = c_{n-1} \\
0 + 0 + 0 + 0 + \cdots + 0 + a_{n-1\,n}x_{n-1} + a_{n\,n}x_n = c_n
\end{array}\right\}. \quad (2.2.39)$$

Equations of this type often occur as a result of using a finite difference operator to replace a differential operator for the solution of differential equations (see chapter 5). A routine that performed straight Gauss elimination would only be involved in one subtraction below the diagonal normalization element and so would reach its 'triangular' form after n steps. Since the resulting equations would only contain two terms, the back substitution would also only require two steps meaning that the entire process would require something of the order of 3n steps for the entire solution. This is so very much more efficient than the general solution and equations of this form occur sufficiently frequently that the student should be aware of this specialized solution.

2.3 Solution of Linear Equations by Iterative Methods

So far we have dealt with methods that will provide a solution to a set of linear equations after a finite number of steps (generally of the order of n^3). The accuracy of the solution at the end of this sequence of steps is fixed by the nature of the equations and to a lesser extent by the specific algorithm that is used. We will now consider a series of algorithms that provide answers to a linear system of equations in considerably fewer steps, but at a level of accuracy that will depend on the number of times the algorithm is applied. Such methods are generally referred to as iterative methods and they usually require of the order of n^2 steps for each iteration. Clearly for very large systems of equations, these methods may prove very much faster than direct methods providing they converge quickly to an accurate solution.

a. Solution by the Gauss and Gauss-Seidel Iteration Methods

All iterative schemes begin by assuming that an approximate answer is known and then the scheme proceeds to improve that answer. Thus we will have a solution vector that is constantly changing from iteration to iteration. In general, we will denote this by a superscript in parentheses so that $x^{(i)}$ will denote the value of x at the ith iteration. Therefore in order to begin, we will need an initial value of the solution vector $\vec{x}^{(0)}$. The concept of the Gauss iteration scheme is extremely simple. Take the system of linear equations as expressed in equations (2.2.1) and solve each one for the diagonal value of x so that

$$x_i = \frac{\left[c_i - \sum_{i \neq j}^{n} a_{ij} x_j \right]}{a_{ii}} \quad . \tag{2.3.1}$$

Now use the components of the initial value of on the right hand side of equation (2.3.1) to obtain an improved value for the elements. This procedure can be repeated until a solution of the desired accuracy is obtained. Thus the general iteration formula would have the form

$$x_i^{(k)} = \frac{\left[c_i - \sum_{i \neq j}^{n} a_{ij} x_j^{(k-1)} \right]}{a_{ii}} \quad . \tag{2.3.2}$$

It is clear, that should any of the diagonal elements be zero, there will be a problem with the stability of the method. Thus the order in which the equations are arranged will make a difference to in the manner in which this scheme proceeds. One might suppose that the value of the initial guess might influence whether or not the method would find the correct answer, but as we shall see in section 2.4 that is not the case. However, the choice of the initial guess will determine the number of iterations required to arrive at an acceptable answer.

The Gauss-Seidel scheme is an improvement on the basic method of Gauss. Let us rewrite equations (2.3.1) as follows:

$$x_i^{(k)} = \frac{\left[c_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k-1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)} \right]}{a_{ii}} \quad . \tag{2.3.3}$$

When using this as a basis for an iteration scheme, we can note that all the values of x_j in the first

summation for the kth iteration will have been determined before the value of $x_i^{(k)}$ so that we could write the iteration scheme as

$$x_i^{(k)} = \frac{\left[c_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)} \right]}{a_{ii}} \quad . \tag{2.3.4}$$

Here the improved values of x_i are utilized as soon as they are obtained. As one might expect, this can lead to a faster rate of convergence, but there can be a price for the improved speed. The Gauss-Seidel scheme may not be as stable as the simple Gauss method. In general, there seems to be a trade off between speed of convergence and the stability of iteration schemes.

Indeed, if we were to apply either if the Gauss iterative methods to equations (2.2.13) that served as an example for the direct method, we would find that the iterative solutions would not converge. We shall see later (sections 2.3d and 2.4) that those equations fail to satisfy the simple sufficient convergence criteria given in section 2.3d and the necessary and sufficient condition of section 2.4. With that in mind, let us consider another 3×3 system of equations which does satisfy those conditions. These equations are much more strongly diagonal than those of equation (2.2.13) so

$$\left.\begin{aligned} 3x_1 + x_2 + x_3 &= 8 \\ x_1 + 4x_2 + 2x_3 &= 15 \\ 2x_1 + x_2 + 5x_3 &= 19 \end{aligned}\right\} \quad . \tag{2.3.5}$$

For these equations, the solution under the Gauss-iteration scheme represented by equations (2.3.2) takes the form

$$\left.\begin{aligned} x_1^{(k+1)} &= \left[8 - x_2^{(k)} - x_3^{(k)}\right] / 3 \\ x_2^{(k+1)} &= \left[15 - x_1^{(k)} - 2x_3^{(k)}\right] / 4 \\ x_3^{(k+1)} &= \left[19 - 2x_1^{(k)} - x_2^{(k)}\right] / 5 \end{aligned}\right\} \quad . \tag{2.3.6}$$

However, if we were to solve equations (2.3.5) by means of the Gauss-Seidel method the iterative equations for the solution would be

$$\left.\begin{aligned} x_1^{(k+1)} &= \left[8 - x_2^{(k)} - x_3^{(k)}\right]/3 \\ x_2^{(k+1)} &= \left[15 - x_1^{(k+1)} - 2x_3^{(k)}\right]/4 \\ x_3^{(k+1)} &= \left[19 - 2x_1^{(k+1)} - x_2^{(k+1)}\right]/5 \end{aligned}\right\} \quad . \tag{2.3.7}$$

If we take the initial guess to be

$$x_1^{(0)} = x_2^{(0)} = x_3^{(0)} = 1 \ , \tag{2.3.8}$$

then repetitive use of equations (2.3.6) and (2.3.7) yield the results given in Table 2.1.

Table 2.1

Convergence of Gauss and Gauss-Seidel Iteration Schemes

k	0		1		2		3		4		5		10	
	G	GS	G	GS	G	GS	G	GS	G	GS	G	GS	G	GS
x_1	1.00	1.00	2.00	2.00	0.60	0.93	1.92	0.91	0.71	0.98	1.28	1.00	0.93	1.00
x_2	1.00	1.00	3.00	2.75	1.65	2.29	2.64	2.03	1.66	1.99	2.32	2.00	1.92	2.00
x_3	1.00	1.00	3.20	2.45	1.92	2.97	3.23	3.03	2.51	3.01	3.18	3.00	2.95	3.00

As is clear from the results labeled "G" in table 2.1, the Gauss-iteration scheme converges very slowly. The correct solution which would eventually be obtained is

$$\vec{x}^{(\infty)} = [\,1,2,3\,] \ . \tag{2.3.9}$$

There is a tendency for the solution to oscillate about the correct solution with the amplitude slowly damping out toward convergence. However, the Gauss-Seidel iteration method damps this oscillation very rapidly by employing the improved values of the solution as soon as they are obtained. As a result, the Gauss-Seidel scheme has converged on this problem in about 5 iterations while the straight Gauss scheme still shows significant error after 10 iterations.

b. The Method of Hotelling and Bodewig

Assume that the correct solution to equation (2.2.3) can be written as

$$\vec{x}_c = \mathbf{A}^{-1}\vec{c} \ , \tag{2.3.10}$$

but that the actual solution that is obtained by matrix inversion is really

$$\vec{x}^{(k)} = (\mathbf{A}^{-1})^{(k)}\vec{c} \ . \tag{2.3.11}$$

Substitution of this solution into the original equations would yield a slightly different constant vector, namely

$$\vec{c}^{(k)} = \mathbf{A}\vec{x}^{(k)}. \tag{2.3.12}$$

Let us define a residual vector in terms of the constant vector we started with and the one that results from the substitution of the correct solution into the original equations so that

$$\vec{R}^{(k)} = \vec{c}^{(k)} - \vec{c} = \mathbf{A}\vec{x}^{(k)} - \mathbf{A}\vec{x}_c = \mathbf{A}(\vec{x}^{(k)} - \vec{x}_c). \tag{2.3.13}$$

Solving this for the true solution $\vec{x}_c$ we get

$$\vec{x}_c = \vec{x}^{(k)} - [\mathbf{A}^{-1}]^{(k)}\vec{R}^{(k)} = \vec{x}^{(k)} - [\mathbf{A}^{-1}]^{(k)}\vec{c}^{(k)} + [\mathbf{A}^{-1}]^{(k)}\vec{c} = [\mathbf{A}^{-1}]^{(k)}[2\vec{c} - \vec{c}^{(k)}] \quad . \tag{2.3.14}$$

The solution of equation (2.3.13) will involve basically the same steps as required to solve equation (2.3.11). Thus the quantity $(\vec{x}^{(k)} - \vec{x}_c)$ will be found with the same accuracy as $\vec{x}^{(k)}$ providing $\vec{R}^{(k)}$ is not too large.

Now we can write $\vec{c}^{(k)}$ in terms $\vec{c}$ of by using equations (2.3.11, 12) and get

$$\vec{c}^{(k)} = \mathbf{A}\vec{x}^{(k)} = \mathbf{A}[\mathbf{A}^{-1}]^{(k)}\vec{c} \ . \tag{2.3.15}$$

Using this result to eliminate $\vec{c}^{(k)}$ from equation (2.3.14) we can write the "correct" solution $\vec{x}_c$ in terms of the approximate matrix inverse $[\mathbf{A}^{-1}]^{(k)}$ as

$$\vec{x}_c = [\mathbf{A}^{-1}]^{(k)}\{2\times\mathbf{1} - \mathbf{A}[\mathbf{A}^{-1}]^{(k)}\}\vec{c}. \tag{2.3.16}$$

Here **1** denotes the unit matrix with elements equal to the Kronecker delta δ_{ij}. Round-off error and other problems that gave rise to the initially inaccurate answer will in reality keep $\vec{x}_c$ from being the correct answer, but it may be regarded as an improvement over the original solution. It is tempting to use equation (2.3.16) as the basis for a continuous iteration scheme, but in practice very little improvement can be made over a single application as the errors that prevent equation (2.3.16) from producing the correct answer will prevent any further improvement over a single iteration.

If we compare equations (2.3.10) and (2.3.16), we see that this method provides us with a mechanism for improving the inverse of a matrix since

$$\mathbf{A}^{-1} = [\mathbf{A}^{-1}]^{(k)}\{2\times\mathbf{1} - \mathbf{A}[\mathbf{A}^{-1}]^{(k)}\} \ . \tag{2.3.17}$$

All of the problems of using equation (2.3.16) as an iteration formula are present in equation (2.3.17). However, the matrix inverse as obtained from equation (2.3.17) should be an improvement over $[\mathbf{A}^{-1}]^{(k)}$.

To see how this method works, consider the equations used to demonstrate the Gauss-Jordan and Crout methods. The exact matrix inverse is given in equations (2.2.22) so we will be able to compare the iterated matrix with the correct value to judge the improvement. For demonstration purposes, assume that the inverse in equation (2.2.22) is known only to two significant figures so that

$$(\mathbf{A}^{-1})^{(k)} = \begin{pmatrix} -0.42 & 0.25 & 0.33 \\ 0.58 & 0.25 & -0.67 \\ 0.08 & -0.25 & 0.33 \end{pmatrix} . \tag{2.3.18}$$

Taking the constant vector to be the same as equation (2.2.13), the solution obtained from the imperfect matrix inverse would be

$$\vec{x}^{(k)} = (\mathbf{A}^{-1})^{(k)}\vec{c} = \begin{pmatrix} -0.42 & 0.25 & 0.33 \\ 0.58 & 0.25 & -0.67 \\ 0.08 & -0.25 & 0.33 \end{pmatrix}\begin{pmatrix} 12 \\ 24 \\ 36 \end{pmatrix} = \begin{pmatrix} 12.84 \\ -11.16 \\ 6.84 \end{pmatrix} . \tag{2.3.19}$$

and substitution of this solution into the original equations [i.e. equation (2.2.13)] will yield the constant vector $\vec{c}_k$ with the elements

$$\mathbf{A}\vec{x}^{(k)} = \vec{c}^{(k)} = \begin{pmatrix} 1.0 & 2.0 & 3.0 \\ 3.0 & 2.0 & 1.0 \\ 2.0 & 1.0 & 3.0 \end{pmatrix}\begin{pmatrix} 12.84 \\ -11.16 \\ 6.84 \end{pmatrix} = \begin{pmatrix} 11.04 \\ 23.04 \\ 35.04 \end{pmatrix} , \tag{2.3.20}$$

that are used to obtain the residual vector in equation (2.3.13).

The method of Hotelling and Bodewig operates by basically finding an improved value for the matrix inverse and then using that with the original constant vector to obtain an improved solution. Therefore, using equation (2.3.17) to improve the matrix inverse we get

$$\mathbf{A}^{-1} = [\mathbf{A}^{-1}]^{(k)}\{2\times\mathbf{1}-\mathbf{A}[\mathbf{A}^{-1}]^{(k)}\},$$

or for example

$$\begin{aligned} A^{-1} &= [A^{-1}]^{(k)}\left[\begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix} - \begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \\ 2 & 1 & 3 \end{pmatrix}\begin{pmatrix} -0.42 & 0.25 & 0.33 \\ 0.58 & 0.25 & -0.67 \\ 0.08 & 0.25 & 0.33 \end{pmatrix}\right] \\ &= [A^{-1}]^{(k)}\left[\begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix} - \begin{pmatrix} 0.98 & 0.00 & -0.02 \\ -0.02 & 1.00 & -0.02 \\ 0.02 & 0.00 & 0.98 \end{pmatrix}\right] \\ &= [A^{-1}]^{(k)}\begin{pmatrix} 1.02 & 0.00 & 0.02 \\ 0.02 & 1.00 & 0.02 \\ 0.02 & 0.00 & 1.02 \end{pmatrix} \end{aligned} . \tag{2.3.21}$$

and performing the final matrix multiplication we have

$$\mathbf{A}^{-1} = \begin{pmatrix} -0.42 & 0.25 & 0.33 \\ 0.58 & 0.25 & -0.67 \\ 0.08 & -0.25 & 0.33 \end{pmatrix} \begin{pmatrix} 1.02 & 0.00 & 0.02 \\ 0.02 & 1.00 & 0.02 \\ 0.02 & 0.00 & 1.02 \end{pmatrix} = \begin{pmatrix} -0.4168 & 0.2500 & 0.3332 \\ 0.5832 & 0.2500 & -0.6668 \\ 0.0832 & -0.2500 & 0.3332 \end{pmatrix}. \quad (2.3.22)$$

This can be compared with the six figure version of the exact inverse from equation (2.2.22) which is

$$\mathbf{A}^{-1} = \begin{pmatrix} -0.416667 & 0.250000 & 0.333333 \\ 0.583333 & 0.250000 & -0.666667 \\ 0.083333 & -0.250000 & 0.333333 \end{pmatrix}. \quad (2.3.23)$$

Every element experienced a significant improvement over the two figure value [equation(2.3.18)]. It is interesting that the elements of the original inverse for which two figures yield an exact result (i.e $a_{12}^{-1}, a_{22}^{-1}, a_{32}^{-1}$) remain unchanged. This result can be traced back to the augmentation matrix [i.e. the right hand matrix in equation (2.3.21) third line]. The second column is identical to the unit matrix so that the second column of the initial inverse will be left unchanged.

We may now use this improved inverse to re-calculate the solution from the initial constant vector and get

$$\vec{x}_c = \mathbf{A}^{-1}\vec{c} = \begin{pmatrix} -0.4168 & 0.2500 & 0.3332 \\ 0.5832 & 0.2500 & -0.6668 \\ 0.0832 & -0.2500 & 0.3332 \end{pmatrix} \begin{pmatrix} 12 \\ 24 \\ 36 \end{pmatrix} = \begin{pmatrix} 12.99 \\ -11.00 \\ 6.994 \end{pmatrix}. \quad (2.3.24)$$

As one would expect from the improved matrix inverse, the solution represents a significant improvement over the initial values given by equation (2.2.19). Indeed the difference between this solution and the exact solution given by equation (2.2.22) is in the fifth significant which is smaller than the calculation accuracy used to obtain the improved inverse. Thus we see that the method of Hotelling and Bodewig is a powerful algorithm for improving a matrix inverse and hence a solution to a system of linear algebraic equations.

c. Relaxation Methods for the Solution of Linear Equations

The Method of Hotelling and Bodewig is basically a specialized relaxation technique and such techniques can be used with virtually any iteration scheme. In general, relaxation methods tend to play off speed of convergence for stability. Rather than deal with the general theory of relaxation techniques, we will illustrate them by their application to linear equations.

As in equation (2.3.8) we can define a residual vector $\vec{R}^{(k)}$ as

$$\vec{R}^{(k)} = \mathbf{A}\vec{x}^{(k)} - \vec{c}. \quad (2.3.25)$$

Let us assume that each element of the solution vector $\vec{x}^{(k)}$ is subject to an improvement $\delta\vec{x}^{(k)}$ so that

$$x_j^{(k+1)} = x_j^{(k)} + \delta x_j \ . \tag{2.3.26}$$

Since each element of the solution vector may appear in each equation, a single correction to an element can change the entire residual vector. The elements of the new residual vector will differ from the initial residual vector by an amount

$$\delta R_{im} = - a_{im}\delta x_m \ . \tag{2.3.27}$$

Now search the elements of the matrix δR_{im} over the index m for the largest value of δR_{im} and reduce the corresponding residual by that maximum value so that

$$\rho_i^{(k)} \equiv -R_i^{(k)} / \tfrac{\text{Max}}{m}(-\delta R_{im}) . \tag{2.3.28}$$

The parameter ρ_i is known as the *relaxation parameter* for the ith equation and may change from iteration to iteration. The iteration formula then takes the form

$$x_j^{(k+1)} = x_j^{(k)} + \rho_j^{(k)}\delta x_j \ . \tag{2.3.29}$$

Clearly the smaller ρ_i is, the smaller the correction to x_i will be and the longer the iteration will take to converge. The advantage of this technique is that it treats each unknown in an individual manner and thus tends to be extremely stable.

Providing a specific example of a relaxation process runs the risk of appearing to limit the concept. Unlike the other iterative procedures we have described, relaxation schemes leave the choice of the correction to the elements of the solution vector completely arbitrary. However, having picked the corrections, the scheme describes how much of them to apply by calculating a relaxation parameter for each element of the solution vector. While convergence of these methods is generally slow, their stability is often quite good. We shall demonstrate that by applying the method to the same system of equations used to demonstrate the other iterative processes [i.e. equations (2.3.5)].

We begin by choosing the same initial solution that we used for the initial guess of the iterative schemes [i.e. $\vec{x}$=(1, 1, 1)]. Inserting that initial guess into equation (2.3.5), we obtain the approximate constant vector $\vec{c}^{(k)}$, which yields a residual vector

$$\vec{R}_0 = \begin{pmatrix} 3 & 1 & 1 \\ 1 & 4 & 2 \\ 2 & 1 & 5 \end{pmatrix}\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} - \begin{pmatrix} 8 \\ 15 \\ 19 \end{pmatrix} = \begin{pmatrix} 5 \\ 7 \\ 8 \end{pmatrix} - \begin{pmatrix} 8 \\ 15 \\ 19 \end{pmatrix} = \begin{pmatrix} -3 \\ -8 \\ -11 \end{pmatrix} . \tag{2.3.30}$$

It should be emphasized that the initial guesses are somewhat arbitrary as they only define a place from which to start the iteration scheme. However, we will be able to compare the results given in Table 2.2 with the other iterative methods .

We will further arbitrarily choose to vary all the unknowns by the same amount so that

$$\delta x_m = 0.3 \ . \tag{2.3.31}$$

Now calculate the variational residual matrix specified by equation (2.3.27) and get

$$\delta R_{im} = -\begin{pmatrix} 3 & 1 & 1 \\ 1 & 4 & 2 \\ 2 & 1 & 5 \end{pmatrix} \times 0.3 = -\begin{pmatrix} 0.9 & 0.3 & 0.3 \\ 0.3 & 1.2 & 0.6 \\ 0.6 & 0.3 & 1.5 \end{pmatrix} \quad . \tag{2.3.32}$$

The element of the matrix with the largest magnitude is $\delta R_{33} = 1.5$. We may now calculate the elements of the relaxation vector in accordance with equation (2.3.28) and modify the solution vector as in equation (2.3.29). Repeating this process we get the results in Table 2.2

Table 2.2

Sample Iterative Solution for the Relaxation Method

k	0	1	4	7	10	∞
	i\ x_i ρ_i	x_i ρ_i	x_i ρ_i	x_i ρ_i	x_i ρ_i	x_i ρ_i
1	1.00 2.00	1.60 -1.07	1.103 –0.02	1.036-.107	0.998 .006	1.00 0.00
2	1.00 4.44	2.33 -1.55	2.072 +0.41	2.07 -.224	2.00 .002	2.00 0.00
3	1.00 7.33	3.20 -7.02	3.011 -.119	3.01 -.119	2.99 .012	3.00 0.00

We see that the solution does indeed converge at a rate that is intermediate between that obtain for the Gauss method and that of the Gauss-Seidel method. This application of relaxation techniques allows the relaxation vector to change approaching zero as the solution converges. Another approach is to use the relaxation parameter to change the correction given by another type of iteration scheme such as Gauss-Seidel. Under these conditions, it is the relaxation parameter that is chosen and usually held constant while the corrections approach zero.

There are many ways to arrive at suitable values for the relaxation parameter but the result will usually be in the range ½ρ½. For values of ρ<½, the rate of convergence is so slow that one is not sure when the solution has been obtained. On rare occasions one may choose a relaxation parameter greater than unity. Such a procedure is said to be *over relaxed* and is likely to become unstable. If $\rho \geq 2$, then instability is almost guaranteed. We have said a great deal about convergence, but little that is quantitative so let us turn to a brief discussion of convergence within the confines of fixed-point iteration theory.

d. Convergence and Fixed-point Iteration Theory

The problems of deciding when a correct numerical solution to a system of equations has been reached are somewhat more complicated for the iterative methods than with the direct methods. Not only does the practical problem of what constitutes a sufficiently accurate solution have to be dealt with, but the problem of whether or not the iteration method is approaching that solution has to be solved. The iteration method will most certainly produce a new solution set, but whether that set is any closer to the

correct set is not immediately obvious. However, we may look to *fixed-point iteration theory* for some help with this problem.

Just as there is a large body of knowledge connected with relaxation theory, there is an equally large body of knowledge relating to fix-point iteration theory[2]. Before looking at iteration methods of many variables such as the Gauss iteration scheme, let us consider a much simpler iteration scheme of only one variable. We could write such a scheme as

$$x^{(k)} = \Phi[x^{(k-1)}] \quad . \tag{2.3.33}$$

Here $\Phi[x^{(k-1)}]$ is any function or algorithm that produces an new value of x based on a previous value. Such a function is said to posses a *fixed-point* x_0 if

$$x_0 = \Phi(x_0) \quad . \tag{2.3.34}$$

If $\Phi(x)$ provides a steady succession of values of x that approach the fixed-point x_0, then it can be said to be a convergent iterative function. There is a little-known theorem which states that a necessary and sufficient condition for $\Phi(x)$ to be a convergent iterative function is

$$\left|\frac{d\Phi(x)}{dx}\right| < 1 \quad \forall x\ \varepsilon \left|x^{(k)}\right| \le |x| \le |x_0| \quad . \tag{2.3.35}$$

For multidimensional iterative functions of the form

$$x_i^{(k+1)} = \Phi_i(x_j^{(k)}), \tag{2.3.36}$$

the theorem becomes

$$\sum_{j=1}^{n} \left|\frac{d\Phi_i(x_i)}{dx_j}\right| < 1, \quad \forall x_i\ \varepsilon \left|x_i^{(k)}\right| \le |x_i| \le |x_{i0}|. \tag{2.3.37}$$

However, it no longer provides necessary conditions, only sufficient ones. If we apply this to the Gauss iteration scheme as described by equation (2.3.1) we have

$$\sum_{j\neq i}^{n} \left|\frac{a_{ij}}{a_{ii}}\right| < 1, \quad \forall\ i \quad . \tag{2.3.38}$$

It is clear that the convergence process is strongly influenced by the size of the diagonal elements present in the system of equations. Thus the equations should be initially arranged so that the largest possible elements are present on the main diagonal of the coefficient matrix. Since the equations are linear, the sufficient condition given in equation (2.2.23) means that the convergence of a system of equations under the Gauss iteration scheme is independent of the solution and hence the initial guess. If equation (2.2.23) is satisfied then the Gauss iteration method is guaranteed to converge. However, the number of iterations required to achieve that convergence will still depend on the accuracy of the initial guess.

If we apply these conditions to equations (2.2.13) which we used to demonstrate the direct methods

of solution, we find that

$$\sum_{j\neq i}^{n}\left|\frac{a_{ij}}{a_{ii}}\right| = \begin{pmatrix} 5 \\ 2 \\ 1 \end{pmatrix} \quad . \tag{2.3.39}$$

Each equation fails to satisfy the sufficient convergence criteria given in equation (2.3.38). Thus it is unlikely that these equations can be solved by most iterative techniques. The fact that the method of Hotelling and Bodewig gave a significantly improved solution is a testament to the stability of that method. However, it must be remembered that the method of Hotelling and Bodewig is not meant to be used in an iterative fashion so comparison of iterative techniques with it is not completely justified.

The sufficient convergence criteria give by equation (2.3.38) essentially says that if the sum of the absolute values of the off-diagonal elements of every row is less than the absolute value of the diagonal element, then the iteration sequence will converge. The necessary and sufficient condition for convergence of this and the Gauss Seidel Scheme is that the eigenvalues of the matrix all be positive and less than one. Thus it is appropriate that we spend a little time to define what eigenvalues are, their importance to science, and how they may be obtained.

2.4 The Similarity Transformations and the Eigenvalues and Vectors of a Matrix

In Chapter 1 (section 1.3) we saw that it is often possible to represent one vector in terms of another by means of a system of linear algebraic equations which we called a coordinate transformation. If this transformation preserved the length of the vector, it was called an orthonormal transformation and the matrix of the transformation coefficients had some special properties. Many problems in science can be represented in terms of linear equations of the form

$$\vec{y} = \mathbf{A}\vec{x} \, . \tag{2.4.1}$$

In general, these problems could be made much simpler by finding a coordinate frame so that each element of the transformed vector is proportional to the corresponding element of the original vector. In other words, does there exist a space wherein the basis vectors are arranged so that the transformation is a diagonal matrix of the form

$$\vec{y}' = \mathbf{S}\vec{x}' \, , \tag{2.4.2}$$

where $\vec{x}'$ and $\vec{y}'$ represent the vectors $\vec{x}$ and $\vec{y}$ in this new space where the transformation matrix becomes diagonal. Such a transformation is called a *similarity transformation* as each element of $\vec{y}'$ would be similar (proportional) to the corresponding element of $\vec{x}'$. Now the space in which we express and is defined by a set of basis vectors $\hat{e}_i$ and the space in which $\vec{x}'$ and $\vec{y}'$ are expressed is spanned by $\hat{e}'_i$. If we let the transformation that relates the unprimed and primed coordinate frames be **D**, then the basis vectors are related by

$$\left.\begin{aligned}\hat{e}_i' &= \sum_j d_{ij}\hat{e}_j \\ \vec{e}' &= \mathbf{D}\vec{e}\end{aligned}\right\} \quad . \tag{2.4.3}$$

Any linear transformation that relates the basis vectors of two coordinate frames will transform any vector from one frame to the other. Therefore

$$\left.\begin{aligned}\vec{x} &= \mathbf{D}^{-1}\vec{x}' \\ \vec{e}' &= \mathbf{D}\vec{e}\end{aligned}\right\} \quad . \tag{2.4.4}$$

If we use the results of equation (2.4.4) to eliminate $\vec{x}$ and $\vec{y}$ from equation (2.4.1) in favor of $\vec{x}'$ and $\vec{y}'$ we get

$$\vec{y}' = [\mathbf{DAD}^{-1}]\vec{x}' = \mathbf{S}\vec{x}' \quad . \tag{2.4.5}$$

Comparing this result with equation (2.4.2) we see that the conditions for **S** to be diagonal are

$$\mathbf{DAD}^{-1} = \mathbf{S} \quad , \tag{2.4.6}$$

which we can rewrite as

$$\mathbf{AD}^{T} = \mathbf{D}^{T}\mathbf{S} \, . \tag{2.4.7}$$

Here we have made use of an implicit assumption that the transformations are orthonormal and so preserve the length of vectors. Thus the conditions that lead to equation (1.3.8) are met and $\mathbf{D}^{-1} = \mathbf{D}^{T}$. We can write these equations in component form as

$$\sum_{k=1}^{n} a_{ik}d_{jk} = d_{ji}s_{jj} = \sum_{k=1}^{n} d_{jk}\delta_{ki}s_{jj} \quad , \quad i = 1\cdots n,\ j = 1\cdots n \tag{2.4.8}$$

These are n systems of linear homogeneous equations of the form

$$\sum_{k=1}^{n} (a_{ik} - \delta_{ki}s_{jj})d_{jk} = 0, \quad i = 1\cdots n,\ j = 1\cdots n \quad , \tag{2.4.9}$$

which have a solution if and only if

$$\mathrm{Det}\left|a_{ik} - \delta_{ki}s_{jj}\right| = 0 \, , \quad \forall j \quad . \tag{2.4.10}$$

Now the nature of **D** and **S** depend only on the matrix **A** and in no way on the values of $\vec{x}$ or $\vec{y}$. Thus they may be regarded as properties of the matrix **A**. The elements s_{jj} are known as the *eigenvalues* (also as the proper values or characteristic values) of **A**, while the columns that make up **D** are called the *eigenvectors* (or proper vectors or characteristic vectors) of **A**. In addition, equation (2.4.10) is known as the *eigen* (or characteristic) *equation* of the matrix **A**. It is not obvious that a similarity transformation exists for all matrices and indeed, in general they do not. However, should the matrix be symmetric, then such a transformation is guaranteed to exist. Equation (2.4.10) suggests the manner by which we can find the eigenvalues of a matrix. The expansion of equation (2.4.10) by minors as in equation (1.2.10), or more generally in equation (2.2.5), makes it clear that the resulting expression will be a polynomial of degree n in s_{jj} which will have n roots which are the eigenvalues. Thus one approach to finding the eigenvalues of a matrix is equivalent to finding the roots of the eigen-equation (2.4.9). We shall say more about finding the roots of a polynomial in the next chapter so for the moment we will restrict ourselves to some special techniques for finding the eigenvalues and eigenvectors of a matrix.

We saw in section (2.2c) that diagonalization of a matrix will not change the value of its determinant. Since the application of the transformation matrix **D** and its inverse effectively accomplishes a diagonalization of **A** to the matrix **S** we should expect the determinant to remain unchanged. Since the determinant of **S** will just be the product of the diagonal elements we can write

$$\text{Det}\,|\,\mathbf{A}\,|_{i} = \Pi s_{ii} \quad . \tag{2.4.11}$$

The trace of a matrix is also invariant to a similarity transformation so

$$\text{Tr}\,|\,\mathbf{A}\,| = \Sigma_{i} s_{ii} \, . \tag{2.4.12}$$

These two constraints are always enough to enable one to find the eigenvalues of a 2□2 matrix and may be used to reduce the eigen-equation by two in its degree. However, for the more interesting case where n is large, we shall have to find a more general method. Since any such method will be equivalent to finding the roots of a polynomial, we may expect such methods to be fairly complicated as finding the roots of polynomials is one of the trickiest problems in numerical analysis. So it is with finding the eigenvalues of a matrix.

While we noted that the transformation that gives rise to **S** is a similarity transformation [equation (2.4.6)], not all similarity transformations need diagonalize a matrix, but simply have the form

$$\mathbf{B}^{-1}\mathbf{A}\mathbf{B} = \mathbf{Q} \quad . \tag{2.4.13}$$

The invariance of the eigenvalues to similarity transformations provide the basis for the general strategy employed by most "canned" eigenvalue programs. The basic idea is to force the matrix **A** toward diagonal form by employing a series of similarity transformations. The details of such procedures are well beyond the scope of this book but can be found in the references suggested at the end of this chapter[3, 4]. However, whatever approach is selected, the prudent investigator will see how well the constraints given by equations (2.4.11, 12) are met before being satisfied that the "canned" package has actually found the correct eigenvalues of the matrix.

Having found the eigenvalues, the corresponding eigenvectors can be found by appealing to equation (2.4.9). However, these equations are still homogeneous, implying that the elements of the eigenvectors are not uniquely determined. Indeed, it is the magnitude of the eigenvector that is usually considered to be unspecified so that all that is missing is a scale factor to be applied to each eigenvector. A common approach is to simply define one of the elements of the eigenvector to be unity thereby making the system of equations (2.4.9) nonhomogeneous and of the form

$$\sum_{k=2}^{n} (a_{ik} - \delta_{ik} s_{jj}) d_{jk} / d_{j1} = -a_{i1} \quad . \tag{2.4.14}$$

In this form the elements of the eigenvector will be found relative to the element d_{1j}.

Let us conclude our discussion of eigenvalues and eigen-vectors by again considering the matrix of the equations (2.2.13) used to illustrate the direct solution schemes. We have already seen from equation (2.3.39) that these equations failed the sufficient conditions for the existence of Gauss-Seidel iterative solution. By evaluating the eigenvalues for the matrix we can evaluate the *necessary and sufficient* conditions for convergence, namely that the eigenvalues all be positive and less than unity.

The matrix for equations (2.2.13) is

$$\mathbf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \\ 2 & 1 & 3 \end{pmatrix} \quad , \tag{2.4.14}$$

so that the eigen-equation delineated by equation (2.4.10) becomes

$$\text{Det } \mathbf{A} = \mathbf{Det}\begin{vmatrix} (1\text{-}s) & 2 & 3 \\ 3 & (2-s) & 1 \\ 2 & 1 & (3-s) \end{vmatrix} = -s^3 + 6s^2 + 2s - 12 = 0 \ . \tag{2.4.15}$$

The cubic polynomial that results has three roots which are the eigenvalues of the matrix. However before solving for the eigenvalues we can evaluate the constraints given by equations (2.4.11) and (2.4.12) and get

$$\left.\begin{aligned} \text{Det}\,|\mathbf{A}| &= \prod_i s_{ii} = -12 \\ \text{Tr}|\mathbf{A}| &= \sum_i s_{ii} = +6 \end{aligned}\right\} . \tag{2.4.16}$$

The determinant tells us that the eigenvalues cannot all be positive so that the *necessary and sufficient* conditions for the convergence of Gauss-Seidel are not fulfilled confirming the result of sufficient condition given by equation (2.3.39). The constraints given by equation (2.4.26) can also aid us in finding roots for the eigen-equation (2.4.15). The fact that the product of the roots is the negative of twice their sum suggests that two of the roots occur as a pair with opposite sign. This conjecture is supported by Descarte's "rule of signs" discussed in the next chapter (section 3.1a). With that knowledge coupled with the values for the trace and determinant we find that the roots are

$$s_i = \begin{pmatrix} 6 \\ +\sqrt{2} \\ -\sqrt{2} \end{pmatrix} \ . \tag{2.4.17}$$

Thus, not only does one of the eigenvalues violate the necessary and sufficient convergence criteria by being negative, they all do as they all have a magnitude greater than unity.

We may complete the study of this matrix by finding the eigen-vectors with the aid of equation (2.4.9) so that

$$\begin{pmatrix} (1\text{-}s) & 2 & 3 \\ 3 & (2-s) & 1 \\ 2 & 1 & (3-s) \end{pmatrix}\begin{pmatrix} d_{i1} \\ d_{i2} \\ d_{i3} \end{pmatrix} = 0 \ . \tag{2.4.18}$$

As we noted earlier, these equations are homogeneous so that they have no unique solution. This means that the length of the eigen-vectors is indeterminant. Many authors normalize them so that they are of unit length thereby constituting a set of unit basis vectors for further analysis. However, we shall simply take one component d_{11} to be unity thereby reducing the 3□3 system of homogeneous equations (2.4.18) to a 2□2 system of inhomogeneous equations,

$$\left.\begin{aligned}(2-s_i)d_{i2}+d_{i3}&=-3\\ d_{i2}+(3-s_i)d_{13}&=-2\end{aligned}\right\},\tag{2.4.19}$$

which have a unique solution for the remaining elements of the eigen-vectors. For our example the solution is

$$\left.\begin{aligned}s_1=+6:&\quad \vec{D}_1=[\,1.0,1.0,1.0\,]\\ s_2=+\sqrt{2}:&\quad \vec{D}_2=[\,1.0,-(7+3\sqrt{2})/(7-5\sqrt{2}),+(2\sqrt{2}-1)/(7-5\sqrt{2})\\ s_3=-\sqrt{2}:&\quad \vec{D}_3=[\,1.0,-(7+3\sqrt{2})/(7+5\sqrt{2}),-(2\sqrt{2}+1)/(7+5\sqrt{2})\end{aligned}\right\}\ .\tag{2.4.20}$$

Should one wish to re-normalize these vectors to be unit vectors, one need only divide each element by the magnitude of the vectors. Each eigenvalue has its own associated eigen-vector so that equation (2.4.20) completes the analysis of the matrix **A**.

We introduced the notion of an eigenvalue initially to provide a necessary and sufficient condition for the convergence of the Gauss-Seidel iteration method for a system of linear equations. Clearly, this is an excellent example of the case where the error or convergence criteria pose a more difficult problem than the original problem. There is far more to the detailed determination of the eigenvalues of a matrix than merely the inversion of a matrix. All the different classes of matrices described in section 1.2 pose special problems even in the case where distinct eigenvalues exist. The solution of the eigen-equation (2.4.10) involves finding the roots of polynomials. We shall see in the next chapter that this is a tricky problem indeed.

Chapter 2 Exercises

1. Find the inverse, eigenvalues, and eigenvectors for

   $a_{ij} = (i+j-1)^{-1}$ for i5, j5 .

   Describe the accuracy of your answer and how you know.

2. Solve the following set of equations both by means of a direct method and iterative method. Describe the methods used and why you chose them.

$$
\begin{aligned}
X_2 + 5X_3 - 7X_4 + 23X_5 - X_6 + 7X_7 + 8X_8 + X_9 - 5X_{10} &= 10\\
17X_1 - 24X_3 - 75X_4 + 100X_5 - 18X_6 + 10X_7 - 8X_8 + 9X_9 - 50X_{10} &= -40\\
3X_1 - 2X_2 + 15X_3 - 78X_5 - 90X_6 - 70X_7 + 18X_8 - 75X_9 + X_{10} &= -17\\
5X_1 + 5X_2 - 10X_3 - 72X_5 - X_6 + 80X_7 - 3X_8 + 10X_9 - 18X_{10} &= 43\\
100X_1 - 4X_2 - 75X_3 - 8X_4 + 83X_6 - 10X_7 - 75X_8 + 3X_9 - 8X_{10} &= -53\\
70X_1 + 85X_2 - 4X_3 - 9X_4 + 2X_5 + 3X_7 - 17X_8 - X_9 - 21X_{10} &= 12\\
X_1 + 15X_2 + 100X_3 - 4X_4 - 23X_5 + 13X_6 + 7X_8 - 3X_9 + 17X_{10} &= -60\\
16X_1 + 2X_2 - 7X_3 + 89X_4 - 17X_5 + 11X_6 - 73X_7 - 8X_9 - 23X_{10} &= 100\\
51X_1 + 47X_2 - 3X_3 + 5X_4 - 10X_5 + 18X_6 - 99X_7 - 18X_8 + 12X_{10} &= 0\\
X_1 + X_2 + X_3 + X_4 + X_5 + X_6 + X_7 + X_8 + X_9 &= 100
\end{aligned}
$$

3. Solve the equations $\mathbf{A}\vec{x} = \vec{c}$ where $a_{ij} = (i+j-1)^{-1}$, and $c_i = i$ for i5, and j5. Use both Gauss-Jordan and Gauss-Seidel methods and comment on which gives the better answer.

4. Solve the following system of equations by Gauss-Jordan and Gauss-Seidel iteration starting with an initial guess of X=Y=Z=1.

$$
\begin{aligned}
8X + 3Y + 2Z &= 20.00\\
16X + 6Y + 4.001Z &= 40.02\\
4X + 1.501Y + Z &= 10.01 \ .
\end{aligned}
$$

   Comment on the accuracy of your solution and the relative efficiency of the two methods.

5. Show that if **A** is an orthonormal matrix, the $\mathbf{A}^{-1} = \mathbf{A}^{T}$.

6. If $\vec{x} = \mathbf{A}\vec{x}'$ where

$$
\mathbf{A} = \begin{pmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 1 & 1 \end{pmatrix},
$$

   find the components of $\vec{x}$ ' in terms of the components of $\vec{x}$ for $\varphi = \pi/6$.

Chapter 2 References and Supplemental Reading

A reasonable complete description of the Crout factorization method is given by

1. Hildebrand, F.B., "Introduction to Numerical Analysis" (1956) McGraw-Hill Book Co., Inc., New York, Toronto, London.

A very nice introduction to fixed-point iteration theory is given by

2. Moursund, D.G., and Duris, C.S., "Elementary Theory and Applications of Numerical Analysis" (1988) Dover Publications, Inc. New York.

The next two references provide an excellent introduction to the determination of eigenvalues and eigenvectors. Householder's discussion is highly theoretical, but provides the underpinnings for contemporary methods. The work titled "Numerical Recipes" is just that with some description on how the recipes work. It represents probably the most complete and useful compilation of contemporary numerical algorithms currently available.

3. Householder, A.S., "Principles of Numerical Analysis" (1953) McGraw-Hill Book Co., Inc., New York, Toronto, London, pp.143-184.

4. Press, W.H., Flannery, B.P., Teukolsky, S.A., Vetterling, W.T., "Numerical Recipes The Art of Scientific Computing" (1986), Cambridge University Press, Cambridge, New York, Melbourne, pp. 335-380.

Richard Hamming's most recent numerical analysis provides a good introduction to the methods for handling error analysis, while reference 6 is an excellent example of the type of effort one may find in the Russian literature on numerical methods. Their approach tends to be fundamentally different than the typical western approach and is often superior as they rely on analysis to a far greater degree than is common in the west.

5. Hamming, R.W., "Introduction to Applied Numerical Analysis" (1971) McGraw-Hill Book Co., Inc., New York, San Francisco, Toronto, London.

6. Faddeeva, V.N., "Computational Methods of Linear Algebra",(1959), Trans. C.D. Benster, Dover Publications, Inc. New York.

3

Polynomial Approximation, Interpolation, and Orthogonal Polynomials

. . .

In the last chapter we saw that the eigen-equation for a matrix was a polynomial whose roots were the eigenvalues of the matrix. However, polynomials play a much larger role in numerical analysis than providing just eigenvalues. Indeed, the foundation of most numerical analysis methods rests on the understanding of polynomials. As we shall see, numerical methods are usually tailored to produce exact answers for polynomials. Thus, if the solution to a problem is a polynomial, it is often possible to find a method of analysis, which has zero formal truncation error. So the extent to which a problem's solution resembles a polynomial will generally determine the accuracy of the solution. Therefore we shall spend some time understanding polynomials themselves so that we may better understand the methods that rely on them.

3.1 Polynomials and Their Roots

When the term polynomial is mentioned, one generally thinks of a function made up of a sum of terms of the form $a_i x^i$. However, it is possible to have a much broader definition where instead of the simple function x^i we may use any general function $\varphi_i(x)$ so that a general definition of a polynomial would have the form

$$P(x) = \sum_{i=0}^{n} a_i \phi_i(x) \quad . \tag{3.1.1}$$

Here the quantity n is known as the degree of the polynomial and is usually one less than the number of terms in the polynomial. While most of what we develop in this chapter will be correct for general polynomials such as those in equation (3.1.1), we will use the more common representation of the polynomial so that

$$\phi_i(x) = x^i \quad . \tag{3.1.2}$$

Thus the common form for a polynomial would be

$$P(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_n x^n \quad . \tag{3.1.3}$$

Familiar as this form may be, it is not the most convenient form for evaluating the polynomial. Consider the last term in equation (3.1.3). It will take n+1 multiplications to evaluate that term alone and n multiplications for the next lowest order term. If one sums the series, it is clear that it will take (n+1)n/2 multiplications and n additions to evaluate P(x). However, if we write equation (3.1.3) as

$$P(x) = a_0 + (a_1 + \cdots (a_{n-1} + a_n x)x)x \cdots)x \,, \tag{3.1.4}$$

then, while there are still n additions required for the evaluation of P(x), the number of multiplications has been reduced to n. Since the time required for a computer to carry out a multiplication is usually an order of magnitude greater than that required for addition, equation (3.1.4) is a considerably more efficient way to evaluate P(x) than the standard form given by equation (3.1.3). Equation (3.1.4) is sometimes called the "*factored form*" of the polynomial and can be immediately written down for any polynomial. However, there is another way of representing the polynomial in terms of factors, namely

$$P(x) = a_n (x - x_1)(x - x_2)(x - x_3) \cdots (x - x_n) \,. \tag{3.1.5}$$

Here the last n coefficients of the polynomial have been replaced by n quantities known as the roots of the polynomial. It is important to note that, in general, there are (n+1) parameters specifying a polynomial of degree n. These parameters can be either the (n+1) coefficients or the n roots and a multiplicative scale factor a_n. In order to fully specify a polynomial this many parameters must be specified. We shall see that this requirement sets constraints for interpolation.

The n quantities known as the roots are not related to the coefficients in a simple way. Indeed, it is not obvious that the polynomial should be able to be written in the form of equation (3.1.5). The fact that a

polynomial of degree n has exactly n such roots is known as the *fundamental theorem of algebra* and its proof is not simple. As we shall see, simply finding the roots is not simple and constitutes one of the more difficult problems in numerical analysis. Since the roots may be either real or complex, the most general approach will have to utilize complex arithmetic. Some polynomials may have multiple roots (i.e. more than one root with the same numerical value). This causes trouble for some root finding methods. In general, it is useful to remove a root (or a pair if they are complex) once it is found thereby reducing the polynomial to a lower degree. Once it has been reduced to a quadratic or even a cubic, the analytic formulae for these roots maybe used. There is an analytic form for the general solution of a quartic (i.e. polynomial of 4th degree), but it is so cumbersome that it is rarely used. Since it has been shown that there is no general form for the roots of polynomials of degree 5 or higher, one will usually have to resort to numerical methods in order to find the roots of such polynomials. The absence of a general scheme for finding the roots in terms of the coefficients means that we shall have to learn as much about the polynomial as possible before looking for the roots.

a. Some Constraints on the Roots of Polynomials

This subject has been studied by some of the greatest mathematical minds of the last several centuries and there are numerous theorems that can be helpful in describing the roots. For example, if we re-multiply equation (3.1.5) the coefficient of x^{n-1} is just a_n times the negative summation of the roots so that

$$a_{n-1} = a_n \sum_{i=1}^{n} x_i \quad . \tag{3.1.6}$$

In a similar manner we find that

$$a_{n-2} = a_n \sum_{i \neq j} \sum_{j} x_i x_j \, . \tag{3.1.7}$$

We will see that it is possible to use these relations to obtain estimates of the magnitude of the roots. In addition, the magnitude of the roots is bounded by

$$\left(|a_{max}| + 1 \right)^{-1} \leq |x_j| \leq \left(|a_{max}| + 1 \right). \tag{3.1.8}$$

Finally there is Descarte's *rule of signs* which we all learned at one time but usually forgot. If we reverse the order of equation (3.1.3) so that the terms appear in descending powers of x as

$$P(x) = a_n x^n + a_{n-1} x^{n-1} + a_{n-2} x^{n-2} + \cdots + a_0 \quad , \tag{3.1.9}$$

then any change of sign between two successive terms is called a *variation* in sign. Coefficients that are zero are ignored. With that definition of a sign variation we can state Descarte's rule of signs as

> *The number of positive roots of P(x)=0 cannot exceed the number of variations of sign in P(x) and, in any case, differs from the number of variations by an even integer.*

A useful and easily proved corollary to this is

> *The number of negative roots of P(x)=0 cannot exceed the number of variations in sign in P(-x) and, in any case, differs from the number of variations by an even integer.*

The phrasing concerning the "even integer" results from the possibility of the existence of complex roots which occur in pairs (providing the coefficients are real) where one is the complex conjugate of the other. With these tools, it is often possible to say a good deal about the properties of the roots of the polynomial in question. Since most of the methods for finding roots are sequential and require the removal of the roots leading to a new polynomial of lower degree, we should say something about how this is accomplished.

b. Synthetic Division

If we wish to remove a factor from a polynomial we may proceed as if we were doing long division with the added proviso that we keep track of the appropriate powers of x. Thus if (x-r) is to be factored out of P(x) we could proceed in exactly the same fashion as long division. Consider the specific case where r = 2 and

$$P(x) = x^4 + 3x^3 - 17x^2 + 6x - 18 \quad . \tag{3.1.10}$$

The long division would then look like

$$\begin{array}{r} x^3 + 5x^2 - 7x - 11 \\ (x-2)\overline{)\,x^4 + 3x^3 - 17x^2 + 3x - 18} \\ \underline{x^4 - 2x^3} \\ 5x^3 - 17x^2 \\ \underline{5x^3 - 10x^2} \\ -7x^2 + 3x \\ \underline{-7x^2 + 14x} \\ -11x - 18 \\ \underline{-11x + 22} \\ -40 \end{array} \quad . \tag{3.1.11}$$

Thus we can write P(x) as

$$P(x) = (x-2)(x^3+5x^2-7x-11) - 40/(x-2) \quad , \tag{3.1.12}$$

or in general as

$$P(x) = (x-r)Q(x) + R \; . \tag{3.1.13}$$

So if we evaluate the polynomial for x = r we get

$$P(r) = R \quad . \tag{3.1.14}$$

Now if R(r) is zero, then r is a root by definition. Indeed, one method for improving roots is to carry out repeated division, varying r until the remainder R is acceptably close to zero. A cursory inspection of the long division expression (3.1.11) shows that much more is being written down than is necessary. In order for the division to proceed in an orderly fashion, there is no freedom in what is to be done with the lead coefficients of the largest powers of x. Indeed, the coefficients of the resultant polynomial Q(x) are repeated below. Also, when searching for a root, the lead coefficient of x in the divisor is always one and therefore need not be written down. Thus if we write down only the coefficients and r-value for the division process, we can compress the notation so that

$$\left.\begin{array}{r} r = 2\overline{)+1+3-17\ +\ 3-18} = P(x) \\ \underline{+2+10-14-22} \\ Q(x) = +1+5-\ 7\ -11\,\big|\,-40 = R \end{array}\right\}. \tag{3.1.15}$$

This shorthand form of keeping track of the division is known as *synthetic division*. Even this notation can be formulated in terms of a recursive procedure. If we let the coefficients of the quotient polynomial Q(x) be b_i so that

$$Q(x) = b_0 + b_1x + b_2x^2 + \ldots + b_{n-1}x^{n-1}\ , \tag{3.1.16}$$

then the process of finding the b_i's in terms of the coefficients a_i of the original polynomial P(x) can be written as

$$\left.\begin{array}{l} b_{n-1} = a_n \\ b_{i-1} = rb_i + a_i \quad i = 0\cdots n-1 \\ R = b_{-1} \end{array}\right\}. \tag{3.1.17}$$

Here the remainder R is given by b_{-1} and should it be zero, then r is a root. Therefore, once a root has been found, it can be removed by synthetic division leading to an new polynomial Q(x). One can then begin again to find the roots of Q(x) until the original polynomial has been reduced to a cubic or less. Because of the complexity of the general cubic, one usually uses the quadratic formula. However, even here Press et al[1] suggest caution and recommend the use of both forms of the formula, namely

$$\left.\begin{array}{l} x = \dfrac{-b \pm \sqrt{b^2-4ac}}{2a} \\ x = \dfrac{2c}{-b \pm \sqrt{b^2-4ac}} \end{array}\right\}. \tag{3.1.18}$$

Should a or c be small the discriminate will be nearly the same as b and the resultant solution will suffer from round-off error. They suggest the following simple solution to this problem. Define

$$q = -[b + \mathrm{sgn}(b)\sqrt{b^2-4ac}\,]/2\,. \tag{3.1.19}$$

Then the two roots will be given by

$$\left.\begin{aligned} x &= q/a \\ x &= c/q \end{aligned}\right\} \quad . \tag{3.1.20}$$

Let us see how one might analyze our specific polynomial in equation (3.1.10). Descartes' rule of signs for P(x) tells us that we will have no more than three real positive roots while for P(-x) it states that we will have no more than one real negative root. The degree of the polynomial itself indicates that there will be four roots in all. When the coefficients of a polynomial are integer, it is tempting to look for integer roots. A little exploring with synthetic division shows that we can find two roots so that

$$P(x) = (x-3)(x+6)(x^2+1) \quad , \tag{3.1.21}$$

and clearly the last two roots are complex. For polynomials with real coefficients, one can even use synthetic division to remove complex roots. Since the roots will appear in conjugate pairs, simply form the quadratic polynomial

$$(x-r)(x-r^*) = x^2 - (r+r^*)x + r\,r^* \quad , \tag{3.1.22}$$

which will have real coefficients as the imaginary part of r cancels out of $(r+r^*)$ and rr^* is real by definition. One then uses synthetic division to divide out the quadratic form of equation (3.1.22). A general recurrence relation similar to equation (3.1.17) can be developed for the purposes of machine computation.

Normally the coefficients of interesting polynomials are not integers and the roots are not simple numbers. Therefore the synthetic division will have a certain round off error so that R(r) will not be zero. This points out one of the greatest difficulties to be encountered in finding the roots of a polynomial. The round off error in R(r) accumulates from root to root and will generally depend on the order in which the roots are found. Thus the final quadratic polynomial that yields the last two roots may be significantly different than the correct polynomial that would result in the absence of round off error. One may get a feeling for the extent of this problem by redoing the calculation but finding the roots in a different order. If the values are independent of the order in which they are found, then they are probably accurate. If not, then they are not.

c. The Graffe Root-Squaring Process

We discuss this process not so much for its practical utility as to show the efficacy of the constraints given in equations (3.1.6,7). Consider evaluating a polynomial for values of $x = x_i$ where x_i are the roots so that

$$P(x_i) = \sum_j a_i x_i^j = \sum_k a_{2k} x_i^{2k} + a_{2k+1} x_i^{2k+1} \quad . \tag{3.1.23}$$

We may separate the terms of the polynomial into even and odd powers of x and since $P(x_i)=0$, we may arrange the odd powers so that they are all on one side of the equation as

$$\left[\sum_k a_{2k} x_i^{2k}\right]^2 = \left[\sum_k a_{2k+1} x_i^{2k+1}\right]^2 . \tag{3.1.24}$$

Squaring both sides produces exponents with even powers and a polynomial with new coefficients $a_i^{(p)}$ and having the form

$$S(x) = a_n^{(p)} x^{2pn} + a_n^{(p)} x^{2pn-2} + \cdots + a_0^{(p)} \quad . \tag{3.1.25}$$

These new coefficients can be generated by the recurrence relation from

$$\left.\begin{aligned} a_i^{(p+1)} &= 2a_n^{(p)} a_{2\ell}^{(p)} - 2\sum_{k=\ell-1}^{n-1} a_k^{(p)} a_{2\ell-k}^{(p)} + (-1)^{\ell} (a_1^{(p)})^2 \\ a_i^{(p)} &= 0, \quad i > n \end{aligned}\right\} . \tag{3.1.26}$$

If we continue to repeat this process it is clear that the largest root will dominate the sum in equation (3.1.6) so that

$$x_{max}^{2p} = \lim_{p\to\infty} \sum_{i=1}^{n} x_i^{2p} = \lim_{p\to\infty} \left[a_{n-1}^{(p)} \Big/ a_n^{(p)} \right] . \tag{3.1.27}$$

Since the product of the largest two roots will dominate the sums of equation (3.1.7), we may generalize the result of eq (3.1.27) so that each root will be given by

$$x_i^{2p} \cong \lim_{p\to\infty} \left[a_{i-1}^{(p)} \Big/ a_n^{(p)} \right] \quad . \tag{3.1.28}$$

While this method will in principle yield all the roots of the polynomial, the coefficients grow so fast that roundoff error quickly begins to dominate the polynomial. However, in some instance it may yield approximate roots that will suffice for initial guesses required by more sophisticated methods. Impressive as this method is theoretically, it is rarely used. While the algorithm is reasonably simple, the large number of digits required by even a few steps makes the programming of the method exceedingly difficult.

d. Iterative Methods

Most of the standard algorithms used to find the roots of polynomials scan the polynomial in an orderly fashion searching for the root. Any such scheme requires an initial guess, a method for predicting a better guess, and a system for deciding when a root has been found. It is possible to cast any such method in the form of a fixed-point iterative function such as was discussed in section 2.3d. Methods having this form are legion so we will discuss only the simplest and most widely used. Putting aside the problem of establishing the initial guess, we will turn to the central problem of predicting an improved value for the root. Consider the simple case of a polynomial with real roots and having a value $P(x_k)$ for some value of the independent variable x_k (see Figure 3.1).

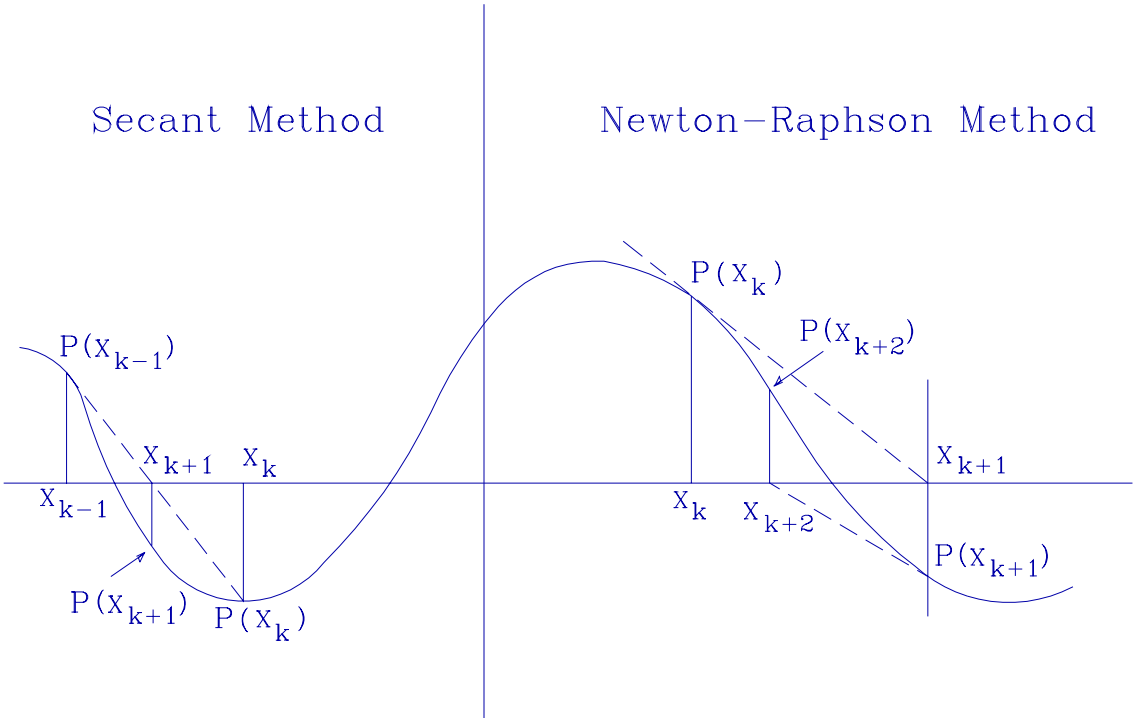


Figure 3.1 depicts a typical polynomial with real roots. Construct the tangent to the curve at the point x_k *and extend this tangent to the x-axis. The crossing point* x_{k+1} *represents an improved value for the root in the Newton-Raphson algorithm. The point* x_{k-1} *can be used to construct a secant providing a second method for finding an improved value of* x.

Many iterative techniques use a straight line extension of the function P(x) to the x-axis as a means of determining an improved value for x. In the case where the straight-line approximation to the function is obtained from the local tangent to the curve at the point x_k, we call the method the *Newton-Raphson* method. We can cast this in the form of a fixed-point iterative function since we are looking for the place where P(x) = 0. In order to find the iterative function that will accomplish this let us assume that an improved value of the root $x^{(k)}$ will be given by

$$x^{(k+1)} = x^{(k)} + [x^{(k+1)} - x^{(k)}] \equiv x^{(k)} + \Delta x^{(k)} \ . \tag{3.1.29}$$

Now since we are approximating the function locally by a straight line, we may write

$$\left.\begin{aligned} P[x^{(k)}] &= \alpha x^{(k)} + \beta \\ P[x^{(k+1)}] &= \alpha x^{(k+1)} + \beta \end{aligned}\right\} . \tag{3.1.30}$$

Subtracting these two equations we get

$$P[x^{(k)}] = \alpha[x^{(k)} - x^{(k+1)}] = -\alpha \Delta x^{(k)} \ . \tag{3.1.31}$$

However the slope of the tangent line α is given by the derivative so that

$$\alpha = dP[x^{(k)}]/dx \quad . \tag{3.1.32}$$

Thus the Newton-Raphson iteration scheme can be written as

$$x^{(k+1)} = x^{(k)} - P[x^{(k)}]/P'[x^{(k)}] \quad . \tag{3.1.33}$$

By comparing equation (3.1.33) to equation (2.3.18) it is clear that the fixed-point iterative function for Newton-Raphson iteration is

$$\Phi(x) = x - P(x)/P'(x) \quad . \tag{3.1.34}$$

We can also apply the convergence criterion given by equation (2.3.20) and find that the necessary and sufficient condition for the convergence of the Newton-Raphson iteration scheme is

$$\left| \frac{P(x)P''(x)}{[P'(x)]^2} \right| < 1 , \quad \forall x \; \varepsilon \; x^{(k)} \le x \le x_0 \quad . \tag{3.1.35}$$

Since this involves only one more derivative than is required for the implementation of the scheme, it provides a quite reasonable convergence criterion and it should be used in conjunction with the iteration scheme.

The Newton-Raphson iteration scheme is far more general than is implied by its use in polynomial root finding. Indeed, many non-linear equations can be dealt with by means of equations (3.1.34, 35). From equation (3.1.33), it is clear that the scheme will yield 'exact' answers for first degree polynomials or straight lines. Thus we can expect that the error at any step will depend on $[\Delta x^{(k)}]^2$. Such schemes are said to be second order schemes and converge quite rapidly. In general, if the error at any step can be written as

$$E(x) = \mathbf{K} \times (\Delta x)^n \quad , \tag{3.1.36}$$

where K is approximately constant throughout the range of approximation, the approximation scheme is said to be of (order) $O(\Delta x)^n$. It is also clear that problems can occur for this method in the event that the root of interest is a multiple root. Any multiple root of P(x) will also be a root of P'(x). Geometrically this implies that the root will occur at a point where the polynomial becomes tangent to the x-axis. Since the denominator of equation (3.1.35) will approach zero at least quadratically while the numerator may approach zero linearly in the vicinity of the root(s), it is unlikely that the convergence criterion will be met. In practice, the shallow slope of the tangent will cause a large correction to $x^{(k)}$ moving the iteration scheme far from the root.

A modest variation of this approach yields a rather more stable iteration scheme. If instead of using the local value of the derivative to obtain the slope of our approximating line, we use a prior point from the iteration sequence, we can construct a secant through the prior point and the present point instead of the local tangent. The straight line approximation through these two points will have the form

$$\left.\begin{aligned} P[x^{(k)}] &= \alpha x^{(k)} + \beta \\ P[x^{(k-1)}] &= \alpha x^{(k-1)} + \beta \end{aligned}\right\} , \tag{3.1.37}$$

which, in the same manner as was done with equation (3.1.30) yields a value for the slope of the line of

$$\alpha = \frac{P[x^{(k)}] - P[x^{(k-1)}]}{x^{(k)} - x^{(k-1)}} \quad . \tag{3.1.38}$$

So the iterative form of the *secant iteration scheme* is

$$x^{(k+1)} = x^{(k)} - \frac{P[x^{(k)}]\left[x^{(k)} - x^{(k-1)}\right]}{P[x^{(k)}] - P[x^{(k-1)}]} \quad . \tag{3.1.39}$$

Useful as these methods are for finding real roots, as presented, they will be ineffective in locating complex roots. There are numerous methods that are more sophisticated and amount to searching the complex plane for roots. For example *Bairstow's method* synthetically divides the polynomial of interest by an initial quadratic factor which yields a remainder of the form

$$R = \alpha x + \beta \ , \tag{3.1.40}$$

where α and β depend on the coefficients of the trial quadratic form. For that form to contain two roots of the polynomial both α and β must be zero. These two constraints allow for a two-dimensional search in the complex plane to be made usually using a scheme such as Newton-Raphson or versions of the secant method. Press et al strongly suggest the use of the *Jenkins-Taub method* or the *Lehmer-Schur method*. These rather sophisticated schemes are well beyond the scope of this book, but may be studied in Acton[2].

Before leaving this subject, we should say something about the determination of the initial guess. The limits set by equation (3.1.8) are useful in choosing an initial value of the root. They also allow for us to devise an orderly progression of finding the roots - say from large to small. While most general root finding programs will do this automatically, it is worth spending a little time to see if the procedure actually follows an orderly scheme. Following this line, it is worth repeating the cautions raised earlier concerning the difficulties of finding the roots of polynomials. The blind application of general programs is almost certain to lead to disaster. At the very least, one should check to see how well any given root satisfies the original polynomial. That is, to what extent is $P(x_i) = 0$. While even this doesn't guarantee the accuracy of the root, it is often sufficient to justify its use in some other problem.

3.2 Curve Fitting and Interpolation

The very processes of interpolation and curve fitting are basically attempts to get "something for nothing". In general, one has a function defined at a discrete set of points and desires information about the function at some other point. Well that information simply doesn't exist. One must make some assumptions about the behavior of the function. This is where some of the "art of computing" enters the picture. One needs some knowledge of what the discrete entries of the table represent. In picking an interpolation scheme to generate the missing information, one makes some assumptions concerning the functional nature of the tabular entries. That assumption is that they behave as polynomials. All interpolation theory is based on polynomial approximation. To be sure the polynomials need not be of the simple form of equation (3.1.3), but nevertheless they will be polynomials of some form such as equation (3.1.1).

Having identified that missing information will be generated on the basis that the tabular function is represented by a polynomial, the problem is reduced to determining the coefficients of that polynomial. Actually some thought should be given to the form of the functions $\varphi_i(x)$ which determines the basic form of the polynomial. Unfortunately, more often than not, the functions are taken to be x^i and any difficulties in representing the function are offset by increasing the order of the polynomial. As we shall see, this is a dangerous procedure at best and can lead to absurd results. It is far better to see if the basic data is - say exponential or periodic in form and use basis functions of the form e^{ix}, $\sin(i\,\pi\,x)$, or some other appropriate functional form. One will be able to use interpolative functions of lower order which are subject to fewer large and unexpected fluctuations between the tabular points thereby producing a more reasonable result.

Having picked the basis functions of the polynomial, one then proceeds to determine the coefficients. We have already observed that an nth degree polynomial has (n+1) coefficients which may be regarded as (n+1) degrees of freedom, or n+1 free parameters to adjust so as to provide the best fit to the tabular entry points. However, one still has the choice of how much of the table to fit at any given time. For interpolation or curve-fitting, one assumes that the tabular data are known with absolute precision. Thus we expect the approximating polynomial to reproduce the data points exactly, but the number of data points for which we will make this demand at any particular part of the table remains at the discretion of the investigator. We shall develop our interpolation formulae initially without regard to the degree of the polynomial that will be used. In addition, although there is a great deal of literature developed around interpolating equally spaced data, we will allow the spacing to be arbitrary. While we will forgo the elegance of the finite difference operator in our derivations, we will be more than compensated by the generality of the results. These more general formulae can always be used for equally spaced data. However, we shall limit our generality to the extent that, for examples, we shall confine ourselves to basis functions of the form x^i. The generalization to more exotic basis functions is usually straightforward. Finally, some authors make a distinction between interpolation and curve fitting with the latter being extended to a single functional relation, which fits an entire tabular range. However, the approaches are basically the same so we shall treat the two subjects as one. Let us then begin by developing *Lagrange Interpolation formulae*.

a. Lagrange Interpolation

Let us assume that we have a set of data points $Y(x_i)$ and that we wish to approximate the behavior of the function between the data points by a polynomial of the form

$$\Phi(x) = \sum_{j=0}^{n} a_j x^j \quad . \tag{3.2.1}$$

Now we require exact conformity between the interpolative function $\Phi(x_i)$ and the data points $Y(x_i)$ so that

$$Y(x_i) = \Phi(x_i) = \sum_{j=0}^{n} a_j x_i^j \,, \quad i = 0 \cdots n \quad . \tag{3.2.2}$$

Equation (3.2.2) represents n+1 inhomogeneous equations in the n+1 coefficients a_j which we could solve using the techniques in chapter 2. However, we would then have a single interpolation formula that would have to be changed every time we changed the values of the dependent variable $Y(x_i)$. Instead, let us combine equations (3.2.1) and (3.2.2) to form n+2 homogeneous equations of the form

$$\left.\begin{aligned}\sum_{j=0}^{n} a_j x_i^j - Y(x_i)\\ \sum_{j=0}^{n} a_j x^j - \Phi(x)\end{aligned}\right\} = 0 \quad . \tag{3.2.3}$$

These equations will have a solution if and only if

$$\text{Det}\begin{vmatrix} 1 & x_0 & x_0^2 & x_0^3 & \cdots & x_0^n & -Y_0 \\ 1 & x_1 & x_1^2 & x_1^3 & \cdots & x_1^n & -Y_1 \\ 1 & x_2 & x_2^2 & x_2^3 & \cdots & x_2^n & -Y_2 \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots \\ 1 & x & x^2 & x^3 & \cdots & x^n & -\Phi(x) \end{vmatrix} = 0 \quad . \tag{3.2.4}$$

Now let $x = x_i$ and subtract the last row of the determinant from the ith row so that expansion by minors along that row will yield

$$[\Phi(x_i) - Y_i] \left| x_k^j \right|_i = 0 \quad . \tag{3.2.5}$$

Since $\left| x_k^j \right|_i \neq 0$, the value of $\Phi(x_i)$ must be $Y(x_i)$ satisfying the requirements given by equation (3.2.2). Now expand equation (3.2.4) by minors about the last column so that

$$\Phi(x)\left|x_k^j\right| = \begin{vmatrix} 1 & x_0 & x_0^2 & x_0^3 & \cdots & x_0^n & -Y_0 \\ 1 & x_1 & x_1^2 & x_1^3 & \cdots & x_1^n & -Y_1 \\ 1 & x_2 & x_2^2 & x_2^3 & \cdots & x_2^n & -Y_2 \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots \\ 1 & x & x^2 & x^3 & \cdots & x^n & 0 \end{vmatrix} = \sum_{i=0}^{n} Y(x_i) A_i(x) \quad . \tag{3.2.3}$$

Here the $A_i(x)$ are the minors that arise from the expansion down the last column and they are independent of the Y_i's. They are simply linear combinations of the x^j' s and the coefficients of the linear combination depend only on the x_i's. Thus it is possible to calculate them once for any set of independent variables x_i and use the results for any set of Y_i's. The determinant $\left|x_k^j\right|$ depends only on the spacing of the tabular values of the independent variable and is called the *Vandermode determinant* and is given by

$$V_d = \left| x_k^j = \prod_{i>j=0}^{n} (x_i - x_j) \right| \quad . \tag{3.2.7}$$

Therefore dividing $A_i(x)$ in equation (3.2.6) by the Vandermode determinant we can write the interpolation formula given by equation (3.2.6) as

$$\Phi(x) = \sum_{i=0}^{n} Y(x_i) L_i(x) \quad , \tag{3.2.8}$$

where $L_i(x)$ is known as the *Lagrange Interpolative Polynomial* and is given by

$$L_i(x) = \prod_{\substack{j\neq i \\ j=0}}^{n} (x - x_j) \Big/ (x_i - x_j) \quad . \tag{3.2.9}$$

This is a polynomial of degree n with roots x_j for $j \neq i$ since one term is skipped (i.e. when i = j) in a product of n+1 terms. It has some interesting properties. For example

$$L_i(x_k) = \prod_{\substack{j\neq i \\ j=0}}^{n} (x_k - x_j) \Big/ (x_i - x_j) \quad = \quad \delta_{ki} \quad , \tag{3.2.10}$$

where δ_{ik} is Kronecker's delta. It is clear that for values of the independent variable equally separated by an amount h the Lagrange polynomials become

$$L_i(x) = \frac{(-1)^n}{(n-i)!i!h^n} \prod_{\substack{j\neq i \\ j=0}}^{n} (x - x_j) \quad . \tag{3.2.11}$$

The use of the Lagrangian interpolation polynomials as described by equations (3.2.8) and (3.2.9) suggest that entire range of tabular entries be used for the interpolation. This is not generally the case. One picks a subset of tabular points and uses them for the interpolation. The use of all available tabular data will generally result in a polynomial of a very high degree possessing rapid variations between the data points that are unlikely to represent the behavior of the tabular data.

Here we confront specifically one of the "artistic" aspects of numerical analysis. We know only the values of the tabular data. The scheme we choose for representing the tabular data at other values of the independent variable must only satisfy some aesthetic sense that we have concerning that behavior. That sense cannot be quantified for the objective information on which to evaluate it simply does not exist. To illustrate this and quantify the use of the Lagrangian polynomials, consider the functional values for x_i and Y_i given in Table 3.1. We wish to obtain a value for the dependent variable Y when the independent variable x = 4. As shown in figure 3.2, the variation of the tabular values Y_i is rapid, particularly in the vicinity of x = 4. We must pick some set of points to determine the interpolative polynomials.

Table 3.1

Sample Data and Results for Lagrangian Interpolation Formulae

i	x	${}_2^1L_i(4)$	${}_1^2L_i(4)$	${}_2^2L_i(4)$	${}_1^3L_i(4)$	Y_i	${}_1^1\Phi_i(4)$	${}_1^2\Phi_i(4)$	${}_2^2\Phi_i(4)$	${}_1^3\Phi_i(4)$
0	1					1				
1	2		-1/3		-2/9	3				
3	3	+1/2	+1	+2/5	+4/5	8				
	4						6	25/3	86/15	112/15
4	5	+1/2	+1/3	+2/3	+4/9	4				
5	8			-1/15	-1/45	2				
6	10					1				

The number of points will determine the order and we must decide which points will be used. The points are usually picked for their proximity to the desired value of the independent variable. Let us pick them consecutively beginning with tabular entry x_k. Then the nth degree Lagrangian polynomials will be

$$ {}^{n}_{k}L_i(x) = \prod_{\substack{j=k \\ j\neq i}}^{n+k} (x - x_j) \Big/ (x_i - x_j) \quad . \tag{3.2.12} $$

Should we choose to approximate the tabular entries by a straight line passing through points bracketing the desired value of $x = 4$, we would get

$$ \left.\begin{aligned} {}^{1}_{2}L_1(x) &= \frac{(x - x_3)}{(x_2 - x_3)} = \tfrac{1}{2} \quad \text{for } x = 4 \\ {}^{1}_{2}L_2(x) &= \frac{(x - x_2)}{(x_3 - x_2)} = \tfrac{1}{2} \quad \text{for } x = 4 \end{aligned}\right\} . \tag{3.2.13} $$

Thus the interpolative value ${}^{1}_{2}\Phi(4)$ given in table 3.1 is simply the average of the adjacent values of Y_i. As can be seen in figure 3.2, this instance of linear interpolation yields a reasonably pleasing result. However, should we wish to be somewhat more sophisticated and approximate the behavior of the tabular function with a parabola, we are faced with the problem of which three points to pick. If we bracket the desired point with two points on the left and one on the right we get Lagrangian polynomials of the form

$$ \left.\begin{aligned} {}^{2}_{1}L_1(x) &= \frac{(x - x_2)(x - x_3)}{(x_1 - x_2)(x_1 - x_3)} = -\frac{1}{3}, \quad x = 4 \\ {}^{2}_{1}L_2(x) &= \frac{(x - x_1)(x - x_3)}{(x_2 - x_1)(x_2 - x_3)} = -1, \quad x = 4 \\ {}^{2}_{1}L_3(x) &= \frac{(x - x_1)(x - x_2)}{(x_3 - x_1)(x_3 - x_2)} = +\frac{1}{3}, \quad x = 4 \end{aligned}\right\} . \tag{3.2.14} $$

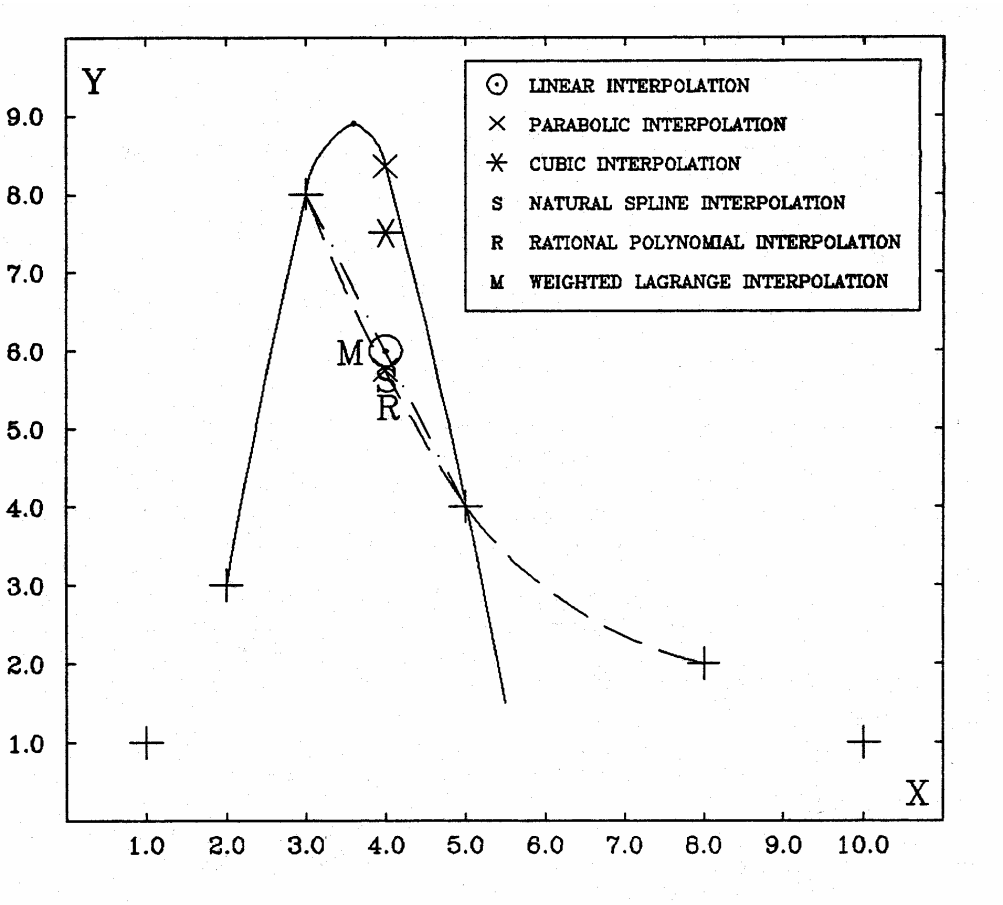


Figure 3.2 shows the behavior of the data from Table 3.1. The results of various forms of interpolation are shown. The approximating polynomials for the linear and parabolic Lagrangian interpolation are specifically displayed. The specific results for cubic Lagrangian interpolation, weighted Lagrangian interpolation and interpolation by rational first degree polynomials are also indicated.

Substituting these polynomials into equation (3.2.8) and using the values for Y_i from Table 3.1, we get an interpolative polynomial of the form

$$P_1(x) = 3\,{}^2_1L_1(x) + 8\,{}^2_1L_2(x) + 4\,{}^2_1L_3(x) = -(7x^2-50x+63)/3 \quad . \tag{3.2.15}$$

Had we chosen the bracketing points to include two on the left and only one on the right the polynomial would have the form

$$P_2(x) = 8\,{}^2_2L_1(x) + 4\,{}^2_2L_2(x) + 2\,{}^2_2L_3(x) = 2(2x^2-31x+135)/15 \quad . \tag{3.2.16}$$

However, it is not necessary to functionally evaluate these polynomials to obtain the interpolated value. Only the numerical value of the Lagrangian polynomials for the specific value of the independent variable given

on the right hand side of equations (3.2.14) need be substituted directly into equation (3.2.8) along with the appropriate values of Y_i. This leads to the values for ${}^{2}_{1}\Phi(4)$ and ${}^{2}_{2}\Phi(4)$ given in Table 3.1. The values are quite different, but bracket the result of the linear interpolation.

While equations (3.13) - (3.16) provide an acceptable method of carrying out the interpolation, there are more efficiently and readily programmed methods. One of the most direct of these is a recursive procedure where values of an interpolative polynomial of degree k are fit to successive sets of the data points. In this method the polynomial's behavior with x is not found, just its value for a specific choice of x. This value is given by

$$\left.\begin{aligned} P_{i,i+1,\cdots,i+k}(x) &= \frac{(x - x_{i+k})P_{i,i+1,\cdots,i+k-1} + (x_i - x)P_{i+1,i+2,\cdots,i+k}(x)}{(x_i - x_{i+k})} \\ P_{i,i}(x) &= Y_i, \quad \text{for } k = 0 \end{aligned}\right\} . \qquad (3.2.17)$$

For our test data given in table 3.1 the recursive formula given by equation (3.2.17) yields

$$\left.\begin{aligned} P_{1,2}(4) &= \frac{(4 - x_2)Y_1 + (x_1 - 4)Y_2}{(x_1 - x_2)} = \frac{(4-3)\times 3 + (2-4)\times 8}{(2-3)} = +13 \\ P_{2,3}(4) &= \frac{(4 - x_3)Y_2 + (x_2 - 4)Y_3}{(x_2 - x_3)} = \frac{(4-5)\times 8 + (3-4)\times 4}{(3-5)} = +6 \\ P_{3,4}(4) &= \frac{(4 - x_4)Y_3 + (x_3 - 4)Y_4}{(x_3 - x_4)} = \frac{(4-5)\times 4 + (5-4)\times 2}{(5-8)} = +\frac{14}{3} \end{aligned}\right\} . \qquad (3.2.18)$$

for k = 1. Here we see that $P_{2,3}(4)$ corresponds to the linear interpolative value obtained using points x_2 and x_3 given in table 3.1 as ${}^{1}_{2}\Phi(4)$. In general, the values of $P_{i,i+1}(x)$ correspond to the value of the straight line passing through points x_i and x_{i+1} evaluated at x. The next generation of recursive polynomial-values will correspond to parabolas passing through the points x_i, x_{i+1}, and x_{i+2} evaluated at x.

For this example they are

$$\left.\begin{aligned} P_{1,2,3}(4) &= \frac{(4 - x_2)P_{1,2}(4) + (x_1 - 4)P_{2,3}(4)}{(x_1 - x_3)} = \frac{(4-3)\times 3 + (2-4)\times 8}{(2-5)} = +\frac{25}{3} \\ P_{2,3,4}(4) &= \frac{(4 - x4)P_{2,3}(4) + (x_2 - 4)P_{3,4}(4)}{(x_2 - x_4)} = \frac{(4-8)\times 6 + (3-4)\times (14/3)}{(3-8)} = +\frac{86}{15} \end{aligned}\right\} , \qquad (3.2.20)$$

which correspond to the values for ${}^{2}_{1}\Phi(4)$ and ${}^{2}_{2}\Phi(4)$ in table 3.1 respectively. The cubic which passes through points x_1, x_2, x_3, and x_4 is the last generation of the polynomials calculated here by this recursive procedure and is

$$P_{1,2,3,4}(4) = \frac{(4 - x_3)P_{1,2,3}(4) + (x_1 - 4)P_{2,3,4}(4)}{(x_1 - x_4)} = \frac{(4-8)\times (25/3) + (2-4)\times (86/15)}{(2-8)} = +\frac{112}{15} . \qquad (3.2.21)$$

The procedure described by equation (3.2.17) is known as Neville's algorithm and can nicely be summarized by a Table 3.2.

The fact that these results exactly replicate those of table 3.1 is no surprise as the polynomial of a particular degree k that passes through a set of k+1 points is unique. Thus this algorithm describes a particularly efficient method for carrying out Lagrangian interpolation and, like most recursive proceedures, is easily implemented on a computer.

How are we to decide which of the parabolas is "better". In some real sense, both are equally likely. The large value of ${}^{2}_{1}\Phi(4)$ results because of the rapid variation of the tabular function through the three chosen points (see figure 3.1) and most would reject the result as being too high. However, we must remember that this is a purely subjective judgment. Perhaps one would be well advised to always have the same number of points on either side so as to insure the tabular variation on either side is equally weighted. This would lead to interpolation by polynomials of an odd degree. If we chose two points either side of the desired value of the independent variable, we fit a cubic through the local points and obtain ${}^{3}_{1}\Phi(4)$ which is rather close to ${}^{2}_{1}\Phi(4)$. It is clear that the rapid tabular variation of the points preceding x = 4 dominate the interpolative polynomials. So which one is correct? We must emphasis that there is no objectively "correct" answer to this question. Generally one prefers an interpolative function that varies no more rapidly that the tabular values themselves, but when those values are sparse this criterion is difficult to impose. We shall consider additional interpolative forms that tend to meet this subjective notion over a wide range of conditions. Let us now turn to methods of their construction.

Table 3.2

Parameters for the Polynomials Generated by Neville's Algorithm

i	x	Y_i	$P_{i,\,i}$	$P_{i,\,i+1}$	$P_{i,\,i+1,\,i+2}$	$P_{i,\,i+1,\,i+2,\,i+3}$
0	1	1	0			
				0		
1	2	3	3		0	
				+13		
2	3	8	8		+25/3	0
	4			+6		112/15
3	5	4	4		+86/15	0
				+14/3		
4	8	2	2		0	
				0		
5	10	1	0			

It is possible to place additional constraints on the interpolative function which will make the appropriate interpolative polynomials somewhat more difficult to obtain, but it will always be possible to obtain them through consideration of the determinantal equation similar to equation (3.2.6). For example, let

us consider the case where constraints are placed on the derivative of the function at a given number of values for the independent variable.

b. Hermite Interpolation

While we will use the Hermite interpolation formula to obtain some highly efficient quadrature formulae later, the primary reason for discussing this form of interpolation is to show a powerful approach to generating interpolation formulae from the properties of the Lagrange polynomials. In addition to the functional constraints of Lagrange interpolation given by equation (3.2.2), let us assume that the functional values of the derivative $Y'(x_i)$ are also specified at the points x_i. This represents an additional (n+1) constraints. However, since we have assumed that the interpolative function will be a polynomial, the relationship between a polynomial and its derivative means we shall have to be careful in order that these 2n+2 constraints remain linearly independent. While a polynomial of degree n has (n+1) coefficients, its derivative will have only n coefficients. Thus the specification of the derivative at the various values of the independent variable allow for a polynomial with 2n+2 coefficients to be used which is a polynomial of degree 2n+1.

Rather than obtain the determinantal equation for the 2n+2 constraints and the functional form of the interpolative function, let us derive the interpolative function from what we know of the properties of $L_i(x)$. For the interpolative function to be independent of the values of the dependent variable and its derivative, it must have a form similar to equation (3.2.8) so that

$$\Phi(x) = \sum_{j=0}^{n} Y(x_j)h_j(x) + Y'(x_j)H_j(x) \quad . \tag{3.2.21}$$

As before we shall require that the interpolative function yield the exact values of the function at the tabular values of the independent variable. Thus,

$$\Phi(x_i) = Y(x_i) = \sum_{j=0}^{n} Y(x_j)h_j(x_i) + Y'(x_j)H_j(x_i) \quad . \tag{3.2.22}$$

Now the beauty of an interpolation formula is that it is independent of the values of the dependent variable and, in this case, its derivative. Thus equation (3.2.22) must hold for any set of data points Y_i and their derivatives Y'_i. So lets consider a very specific set of data points given by

$$\left.\begin{aligned} &Y(x_i) = 1 \\ &Y(x_j) = 0, \quad j \neq i \\ &Y'(x_j) = 0, \quad \forall j \end{aligned}\right\} \quad . \tag{3.2.23}$$

This certainly implies that $h_i(x_i)$ must be one. A different set of data points that have the properties that

$$\left.\begin{aligned} &Y(x_i) = 0, \quad j \neq k \\ &Y(x_k) = 0, \\ &Y'(x_j) = 0, \quad \forall j \end{aligned}\right\} \quad , \tag{3.2.24}$$

will require that $h_k(x_j)$ be zero. However, the conditions on $h_i(x_j)$ must be independent of the values of the independent variable so that both conditions must hold. Therefore

$$h_j(x_i) = \delta_{ij} \quad . \tag{3.2.25}$$

where δ_{ij} is Kronecker's delta. Finally one can consider a data set where

$$\left.\begin{aligned} Y(x_j) &= 0 \\ Y'(x_j) &= 1, \end{aligned}\right\} \quad \forall\, j \quad . \tag{3.2.26}$$

Substitution of this set of data into equation (3.2.22) clearly requires that

$$H_j(x_i) = 0 \quad . \tag{3.2.27}$$

Now let us differentiate equation (3.2.21) with respect to x and evaluate at the tabular values of the independent variable x_i. This yields

$$\Phi'(x_i) = Y(x_i) = \sum_{j=0}^{n} Y(x_j) h'_j(x_i) + Y'(x_j) H'_j(x_i) \quad . \tag{3.2.28}$$

By choosing our data sets to have the same properties as in equations (3.2.23,24) and (3.2.26), but with the roles of the function and its derivative reversed, we can show that

$$\left.\begin{aligned} h'_j(x_i) &= 0 \\ H'_j(x_i) &= \delta_{ij} \end{aligned}\right\} \quad . \tag{3.2.29}$$

We have now place constraints on the interpolative functions $h_j(x)$, $H_j(x)$ and their derivatives at each of the n+1 values of the independent variable. Since we know that both $h_j(x)$ and $H_j(x)$ are polynomials, we need only express them in terms of polynomials of the correct degree which have the same properties at the points x_i to uniquely determine their form.

We have already shown that the interpolative polynomials will have a degree of (2n+1). Thus we need only find a polynomial that has the form specified by equations (3.2.25) and (3.2.29). From equation (3.2.10) we can construct such a polynomial to have the form

$$h_j(x) = v_j(x) L_j^2(x) \quad , \tag{3.2.30}$$

where $v_j(x)$ is a linear polynomial in x which will have only two arbitrary constants. We can use the constraint on the amplitude and derivative of $h_j(x_i)$ to determine those two constants. Making use of the constraints in equations (3.2.25) and (3.2.29) we can write that

$$\left.\begin{aligned} h_i(x_i) &= v_i(x_i) L_i^2(x_i) = 1 \\ h'_j(x_i) &= v'_j(x_i) L_j^2(x_i) + 2 v_j(x_i) L'_j(x_i) L_j(x_i) = 0 \end{aligned}\right\} \quad . \tag{3.2.31}$$

Since $v_i(x)$ is a linear polynomial, we can write

$$v_i(x) = a_i x + b_i \quad . \tag{3.2.32}$$

Specifically putting the linear form for $v_i(x)$ into equation (3.2.31) we get

$$\left.\begin{aligned} v_i(x_i) &= a_i x_i + b_i = 1 \\ v'_i(x_i) &= a_i = -2(a_i x_i + b_i)L'_i(x\ ) \end{aligned}\right\}, \tag{3.2.33}$$

which can be solved for a_i and b_i to get

$$\left.\begin{aligned} a_i &= -2L'_i(x_i) \\ b_i &= 1 + 2x_i L'_i(x_i) \end{aligned}\right\}. \tag{3.2.34}$$

Therefore the linear polynomial $v_i(x)$ will have the particular form

$$v_i(x) = 1 - 2(x-x_i)L_i'(x_i)\ . \tag{3.2.35}$$

We must follow the same procedure to specify $H_j(x)$. Like $h_j(x)$, it will also be a polynomial of degree 2n+1 so let us try the same form for it as we did for $h_j(x)$. So

$$H_j(x) = u_j(x)L_j^2(x)\ , \tag{3.2.36}$$

where $u_j(x)$ is also a linear polynomial whose coefficients must be determined from

$$\left.\begin{aligned} H_i(x_i) &= u_i(x_i)L_i^2(x_i) = 0 \\ H'_i(x_i) &= u'_i(x_i)L_i^2(x_i) + 2u_i(x_i)L'_i(x_i)L_i(x_i) = 1 \end{aligned}\right\}. \tag{3.2.37}$$

Since $L_i^2(x_i)$ is unity, these constraints clearly limit the values of $u_i(x)$ and its derivative at the tabular points to be

$$\left.\begin{aligned} u_i(x_i) &= 0 \\ u'_i(x_i) &= 1 \end{aligned}\right\}. \tag{3.2.38}$$

Since $u_i(x)$ is linear and must have the form

$$u_i(x) = \alpha_i x + \beta_i\ , \tag{3.2.39}$$

we can use equation (3.2.38) to fine the constants α_i and β_i as

$$\left.\begin{aligned} \alpha_i &= 1 \\ \beta_i &= -x_i \\ u_i(x) &= (x - x_i) \end{aligned}\right\}, \tag{3.2.40}$$

thereby completely specifying $u_i(x)$. Therefore, the two functions $h_j(x)$ and $H_j(x)$ will have the specific form

$$\left.\begin{aligned} h_j(x) &= [1 - 2(x - x_j)L'_j(x_j)]L_j^2(x) \\ H_j(x) &= (x - x_j)L_j^2(x) \end{aligned}\right\}. \tag{3.2.41}$$

All that remains is to find $L_j'(x_j)$. By differentiating equation (3.2.9) with respect to x and setting x to x_j, we get

$$L'_j(x_j) = \sum_{k \neq j} (x_j - x_k)^{-1}\ , \tag{3.2.42}$$

which means that $v_j(x)$ will simplify to

$$v_j(x) = 1 - 2\sum_{k\neq j}\frac{(x - x_j)}{(x_j - x_k)} \quad . \tag{3.2.43}$$

Therefore the Hermite interpolative function will take the form

$$\Phi(x) = \sum_i [Y_i v_i(x) + Y'_i\, u_i(x)]\left[\prod_{j\neq i}(x - x_j)/(x_i - x_j)\right]^2 \quad . \tag{3.2.44}$$

This function will match the original function Y_i and its derivative at each of the tabular points. This function is a polynomial of degree 2n-1 with 2n coefficients. These 2n coefficients are specified by the 2n constraints on the function and its derivative. Therefore this polynomial is unique and whether it is obtained in the above manner, or by expansion of the determinantal equation is irrelevant to the result. While such a specification is rarely needed, this procedure does indicate how the form of the Lagrange polynomials can be used to specify interpolative functions that meet more complicated constraints. We will now consider the imposition of a different set of constraints that lead to a class of interpolative functions that have found wide application.

c. Splines

Splines are interpolative polynomials that involve information concerning the derivative of the function at certain points. Unlike Hermite interpolation that explicitly invokes knowledge of the derivative, splines utilize that information implicitly so that specific knowledge of the derivative in not required. Unlike general interpolation formulae of the Lagrangian type, which maybe used in a small section of a table, splines are constructed to fit an entire run of tabular entries of the independent variable. While one can construct splines of any order, the most common ones are cubic splines as they generate tri-diagonal equations for the coefficients of the polynomials. As we saw in chapter 2, tri-diagonal equations lend themselves to rapid solution involving about N steps. In this case N would be the number of tabular entries of the independent variable. Thus for relatively few arithmetic operations, one can construct a set of cubic polynomials which will represent the function over its entire tabular range. If one were to make a distinction between interpolation and curve fitting, that would be it. That is, one may obtain a local value of a function by interpolation, but if one desires to describe the entire range of a tabular function, one would call that *curve fitting*. Because of the common occurrence of cubic splines, we shall use them as the basis for our discussion. Generalization to higher orders is not difficult, but will generate systems of equations for their coefficients that are larger than tri-diagonal. That removes much of the attractiveness of the splines for interpolation.

To understand how splines can be constructed, consider a function with n tabular points whose independent variable we will denote as x_i and dependent values as Y_i. We will approximate the functional values between any two adjacent points x_i and x_{i+1} by a cubic polynomial denoted by $\Psi_i(x)$. Also let the interval between x_{i+1} and x_i be called

$$\Delta x_i \equiv x_{i+1} - x_i \quad . \tag{3.2.45}$$

Since the cubic interpolative polynomials $\Psi_i(x)$ cover each of the n-1 intervals between the n tabular

points, there will be 4(n-1) constants to be determined to specify the interpolative functions. As with Lagrange interpolation theory we will require that the interpolative function reproduce the tabular entries so that

$$\left.\begin{aligned}\Psi_i(x_i) &= Y_i \\ \Psi_i(x_{i+1}) &= Y_{i+1}\end{aligned}\right\} \quad i = 1 \cdots n-1 \quad . \tag{3.2.46}$$

Requiring that a single polynomial match two successive points means that two adjacent polynomials will have the same value where they meet, or

$$\Psi_i(x_{i+1}) = \Psi_{i+1}(x_{i+1}) \quad i = 1 \cdots n-2 \quad . \tag{3.2.47}$$

The requirement to match n tabular points represents n linearly independent constraints on the 4n-4 coefficients of the polynomials. The remaining constraints come from conditions placed on the functional derivatives. Specifically we shall require that

$$\left.\begin{aligned}\Psi'_{i-1}(x_i) &= \Psi'_i(x_i) \\ \Psi''_{i-1}(x_i) &= \Psi''_i(x_i)\end{aligned}\right\} \quad i = 2 \cdots n-1 \quad . \tag{3.2.48}$$

Unlike Hermite interpolation, we have not specified the magnitude of the derivatives at the tabular points, but only that they are the same for two adjacent functions $\Psi_{i-1}(x_i)$ and $\Psi_i(x_i)$ at the points x_i all across the tabular range. Only at the end points have we made no restrictions. Requiring the first two derivatives of adjacent polynomials to be equal where they overlap will guarantee that the overall effect of the splines will be to generate a smoothly varying function over the entire tabular range. Since all the interpolative polynomials are cubics, their third derivatives are constants throughout the interval Δx_i so that

$$\Psi'''_i(x_i) = \Psi'''_i(x_{i+1}) = \text{const.}, \quad i = 1 \cdots n-1 \quad . \tag{3.2.49}$$

Thus the specification of the functional value and equality of the first two derivatives of adjacent functions essentially forces the value of the third derivative on each of the functions $\Psi_i(x)$. This represents n-1 constraints. However, the particular value of that constant for all polynomials is not specified so that this really represents only n-2 constraints. In a similar manner, the specification of the equality of the derivative of two adjacent polynomials for n-2 points represents another n-2 constraints. Since two derivatives are involved we have an additional 2n-4 constraints bringing the total to 4n-6. However, there were 4n-4 constants to be determined in order that all the cubic splines be specified. Thus the system as specified so far is under-determined. Since we have said nothing about the end points it seems clear that that is where the added constraints must be made. Indeed, we shall see that additional constraints must be placed either on the first or second derivative of the function at the end points in order that the problem have a unique solution. However, we shall leave the discussion of the specification of the final two constraints until we have explored the consequences of the 4n-6 constraints we have already developed.

Since the value of the third derivative of any cubic is a constant, the constraints on the equality of the second derivatives of adjacent splines require that the constant be the same for all splines. Thus the second derivative for all splines will have the form

$$\Psi''_i(x) = ax + b \, . \tag{3.2.50}$$

If we apply this form to two successive tabular points, we can write

$$\left.\begin{aligned}\Psi''_i(x_i) &= ax_i + b = Y''_i \\ \Psi''_{i+1}(x_{i+1}) &= ax_{i+1} + b = Y''_{i+1}\end{aligned}\right\} \quad . \tag{3.2.51}$$

Here we have introduced the notation that $\Psi''_i(x_i)=Y''_i$. The fact of the matter is that Y''_i doesn't exist. We

have no knowledge of the real values of the derivatives of the tabular function anywhere. All our constraints are applied to the interpolative polynomials $\Psi_i(x)$ otherwise known as the cubic splines. However, the notation is clear, and as long as we keep the philosophical distinction clear, there should be no confusion about what Y''_i means. In any event they are unknown and must eventually be found. Let us press on and solve equations (3.2.51) for a and b getting

$$\left.\begin{aligned} a &= (Y''_i + Y''_{i+1})/(x_i - x_{i+1}) = (Y''_i + Y''_{i+1})/\Delta x_i \\ b &= Y''_i - x_i(Y''_{i+1} - Y''_i)/\Delta x_i \end{aligned}\right\} . \qquad (3.2.52)$$

Substituting these values into equation (3.2.50) we obtain the form of the second derivative of the cubic spline as

$$\Psi''_i(x) = [Y''_{i+1}(x-x_i) - Y''_i(x-x_{i+1})]/\Delta x_i \ . \qquad (3.2.53)$$

Now we may integrate this expression twice making use of the requirement that the function and its first derivative are continuous across a tabular entry point, and evaluate the constants of integration to get

$$\Psi_i(x) = \{Y_i - Y''_i[(\Delta x_i)^2-(x_{i+1}-x)^2]/6\}[(x_{i+1}-x)/\Delta x_i] - \{Y_{i+1} - Y''_{i+1}[(\Delta x_i)^2-(x_i-x)^2]/6\}[(x_i-x)/\Delta x_i] \ . \qquad (3.2.54)$$

This fairly formidable expression for the cubic spline has no quadratic term and depends on those unknown constants Y''_i.

To get equation (3.2.54) we did not explicitly use the constraints on $\Psi'_i(x)$ so we can use them now to get a set of equations that the constants Y''_i must satisfy. If we differentiate equation (3.2.54) and make use of the condition on the first derivative that

$$\Psi'_{i-1}(x_i) = \Psi'_i(x_i) \ , \qquad (3.2.55)$$

we get after some algebra that

$$Y''_{i-1}\Delta x_{i-1} + 2Y''_i(\Delta x_{i-1}+\Delta x_i) + Y''_{i+1}\Delta x_i = 6[(Y_{i+1}-Y_i)/\Delta x_i + (Y_i-Y_{i-1})/\Delta x_{i-1}] \quad i=2 \cdots n-1 \ . \qquad (3.2.56)$$

Everything on the right hand side is known from the tabular entries while the left hand side contains three of the unknown constants Y''_i. Thus we see that the equations have the form of a tri-diagonal system of equations amenable to fast solution algorithms. Equation (3.2.56) represents n-2 equations in n unknowns clearly pointing out that the problem is still under determined by two constants. If we arbitrarily take $Y''_1=Y''_n=0$, then the splines that arise from the solution of equation (3.2.56) are called *natural splines*. Keeping the second derivative zero will reduce the variation in the function near the end points and this is usually considered desirable. While this arbitrary choice may introduce some error near the end points, the effect of that error will diminish as one moves toward the middle of the tabular range. If one is given nothing more than the tabular entries Y_i and x_i, then there is little more that one can do and the natural splines are as good as any other assumption. However, should anything be known about the first or second derivatives at the end points one can make a more rational choice for the last two constants of the problem? For example, if the values of the first derivatives are known at the end points then differentiating equation (3.2.56) and evaluating it at the end points yields two more equations of condition which depend on the end point first derivatives as

$$\left.\begin{aligned} Y''_1 + \tfrac{1}{2}Y''_2 &= 3[(Y_2 - Y_1)/\Delta x_1 - Y'_1]/\Delta x_1 \\ Y''_n - Y''_{n-1}/6 &= 2[(Y_{n-1} - Y_n)/\Delta x_{n-1} - Y'_n]/\Delta x_{n-1} \end{aligned}\right\} \ . \qquad (3.2.57)$$

These two added conditions complete the system of equations without destroying their tri-diagonal form and pose a significant alternative to natural splines should some knowledge of the endpoint derivatives exist. It is clear that any such information at any point in the tabular range could be used to further constrain the system so that a unique solution exists. In the absence of such information one has little choice but to use the aesthetically pleasing natural splines. One may be somewhat disenchanted that it is necessary to appeal to esthetics to justify a solution to a problem, but again remember that we are trying to get "something for nothing" in any interpolation or curve fitting problem. The "true" nature of the solution between the tabular points simply does not exist. Thus we have another example of where the "art of computing" is utilized in numerical analysis.

In order to see the efficacy of splines, consider the same tabular data given in Table 3.1 and investigate how splines would yield a value for the table at x = 4. Unlike Lagrangian interpolation, the constraints that determine the values for the splines will involve the entire table. Thus we shall have to set up the equations specified by equation (3.2.56). We shall assume that natural splines will be appropriate for the example so that

$$Y_0'' = Y_5'' = 0 \quad . \tag{3.2.58}$$

For i = 1, equation (3.2.56) and the tabular values from table 3.1 yield

$$4Y_1'' + Y_2'' = 6[(8-3)/1 + (3-1)/1] = 42 \ , \ i=1 \ , \tag{3.2.59}$$

and the entire system of linear equations for the Y_i'''s can be written as

$$\begin{pmatrix} 4 & 1 & 0 & 0 \\ 1 & 6 & 2 & 0 \\ 0 & 2 & 10 & 3 \\ 0 & 0 & 3 & 10 \end{pmatrix} \begin{pmatrix} Y_1'' \\ Y_2'' \\ Y_3'' \\ Y_4'' \end{pmatrix} = \begin{pmatrix} 42 \\ 18 \\ -16 \\ -7 \end{pmatrix} \ . \tag{3.2.60}$$

The solution for this tri-diagonal system can be found by any of the methods described in Chapter 2, but it is worth noting the increase in efficiency afforded by the tri-diagonal nature of the system. The solution is given in Table 3.3.

The first item to note is that the assumption of natural splines may not be the best, for the value of $Y_1'' \times 10$ is significantly different from the zero assumed for Y_0''. The value of Y" then proceeds to drop smoothly toward the other boundary implying that the assumption of $Y_5'' = 0$ is pretty good. Substituting the solution for Y_i'' into equation (3.2.54) we get that

$$\Psi_2(4) = \{8 - 1.9876[4-(5-4)^2]/6\}(4-3)/2 \ - \{4 - (-1.9643)[4-(3-4)^2]/6\}(3-4)/2 = 5.9942 \ . \tag{3.2.61}$$

As can be seen from Table 3.3, the results for the natural cubic splines are nearly identical to the linear interpolation, and are similar to that of the second parabolic Lagrangian interpolation. However, the most appropriate comparison would be with the cubic Lagrangian interpolation ${}^3{}_1\Phi(4)$ as both approximating functions are cubic polynomials. Here the results are quite different illustrating the importance of the constraints on the derivatives of the cubic splines. The Lagrangian cubic interpolation utilizes tabular information for 2≤x≤8 in order to specify the interpolating polynomial. The splines rely on the more local

information involving the function and its derivatives specified in the range 3≤x≤5. This minimizes the large tabular variations elsewhere in the table that affect the Lagrangian cubic polynomial and make for a smoother functional variation. The negative aspect of the spline approach is that it requires a solution throughout the table. If the number of tabular entries is large and the required number of interpolated values is small, the additional numerical effort maybe difficult to justify. In the next section we shall find esthetics and efficiency playing an even larger role in choosing the appropriate approximating method.

Table 3.3

A Comparison of Different Types of Interpolation Formulae

i	x	${}^{1}_{2}Y_i$	${}^{1}_{1}\Phi(4)$	${}^{2}_{1}\Phi(4)$	${}^{2}_{2}\Phi(4)$	${}^{3}_{1}\Phi(4)$	Δx_i	Y_i''	$\Psi_2(4)$	$R_{1,2,3,4}$	${}^{2}_{1,2}\Phi_w(4)$
0	1	1					1	0.0000			
1	2	3					1	10.003			
2	3	8					2	1.988			
	4		6.000	8.333	5.733	7.467			5.994	5.242	6.000
3	5	4					3	-1.965			
4	8	2					2	-0.111			
5	10	1					--	-0.000			

d. Extrapolation and Interpolation Criteria

So far we have obtained several different types of interpolation schemes, but said little about choosing the degree of the polynomial to be used, or the conditions under which one uses Lagrange interpolation or splines to obtain the information missing from the table. The reason for this was alluded to in the previous paragraph - there is no correct answer to the question. One can dodge the philosophical question of the "correctness" of the interpolated number by appealing to the foundations of polynomial approximation - namely that to the extent that the function represented by the tabular points can be represented by a polynomial, the answer is correct. But this is indeed a dodge. For if it were true that the tabular function was indeed a polynomial, one would simply use the interpolation scheme to determine the polynomial that fit the entire table and use it. In science, one generally does know something about the nature of a tabular function. For example, many such tables result from a complicated computation of such length that it is not practical to repeat the calculation to obtain additional tabular values. One can usually guarantee that the results of such calculations are at least continuous differentiable functions. Or if there are discontinuities, their location is known and can be avoided. This may not seem like much knowledge, but it guarantees that one can locally approximate the table by a polynomial. The next issue is what sort of polynomial should be used and over what part of the tabular range.

In section 3.1 we pointed out that a polynomial can have a very general form [see equation (3.1.1)].

While we have chosen our basis functions $\varphi_i(x)$ to be x^i for most of the discussion, this need not have been the case. Interpolation formulae of the type developed here for x^i can be developed for any set of basis functions $\varphi_i(x)$. For example, should the table exhibit exponential growth with the independent variable, it might be advisable to choose

$$\varphi_i(x) = e^{i\alpha x} = [e^{\alpha x}]^i \; z^i \quad . \tag{3.2.62}$$

The simple transformation of $z = e^{\alpha x}$ allows all previously generated formulae to be immediately carried over to the exponential polynomials. The choice of α will be made to suit the particular table. In general, it is far better to use basis functions $\varphi_i(x)$ that characterize the table than to use some set of functions such as the convenient x^i and a larger degree for interpolation. One must always make the choice between fitting the tabular form and using the *lowest* degree polynomial possible. The choice of basis functions that have the proper form will allow the use of a lower degree polynomial.

Why is it so desirable to choose the lowest degree polynomial for interpolation? There is the obvious reason that the lower the degree the faster the computation and there are some cases where this may be an overriding concern. However, plausibility of the result is usually the determining factor. When one fits a polynomial to a finite set of points, the value of the polynomial tends to oscillate between the points of constraint. The higher the degree of the polynomial, the larger is the amplitude and frequency of these oscillations. These considerations become even more important when one considers the use of the interpolative polynomial outside the range specified by the tabular entries. We call such use *extrapolation* and it should be done with only the greatest care and circumspection. It is a fairly general characteristic of polynomials to vary quite rapidly outside the tabular range to which they have been constrained. The variation is usually characterized by the largest exponent of the polynomial. Thus if one is using polynomials of the forth degree, he/she is likely to find the interpolative polynomial varying as x^4 immediately outside the tabular range. This is likely to be unacceptable. Indeed, there are some who regard any extrapolation beyond the tabular range that varies more than linearly to be unjustified. There are, of course, exceptions to such a hard and fast rule. Occasionally asymptotic properties of the function that yield the tabular entries are known, then extrapolative functions that mimic the asymptotic behavior maybe justifiable.

There is one form of extrapolation that reduces the instabilities associated with polynomials. It is a form of approximation that abandons the classical basis for polynomial approximation and that is approximation by *rational functions* or more specifically *quotient polynomials*. Let us fit such a function through the $(k - i + 1)$ points $i \rightarrow k$. Then we can define a quotient polynomial as

$$R_{i,i+1,\cdots,i+k}(x) = \frac{P(x)}{Q(x)} = \frac{\sum_{j=0} a_0 x^j}{\sum_{j=0} b_0 x^j} \quad . \tag{3.2.63}$$

This function would appear to have $(m+n+2)$ free parameters, but we can factor a_0 from the numerator and b_0 from the denominator so that only their ratio is a free parameter. Therefore there are only $(m+n+1)$ free parameters so we must have

$$k+1 = m+n+1 \; , \tag{3.2.64}$$

functional points to determine them. However, the values of n and m must also be specified separately. Normally the determination of the coefficients of such a function is rather difficult, but Stoer and Bulirsch[3]

have obtained a recurrence formula for the value of the function itself, which is

$$\left.\begin{aligned} R_{i,i+1,\cdots,i+k}(x) &= R_{i+1,\cdots,i+k}(x) + \frac{R_{i+1,\cdots,i+k}(x) - R_{i,\cdots,i+k-1}(x)}{\left[\dfrac{(x - x_i)}{(x - x_{i+k})}\right]\left[1 - \dfrac{R_{i+1,\cdots,i+k}(x) - R_{i,i+1,\cdots,i+k-1}(x)}{R_{i+1,\cdots,i+k}(x) - R_{i+1,\cdots,i+k-1}(x)}\right] - 1} \\ R_{i,i} &= f(x_i) \\ R_{i,k} &= 0, \quad k < 1 \end{aligned}\right\} . \qquad (3.2.65)$$

This recurrence relation produces a function where n = m if the number of points used is odd, but where m = n+1 should the number of points be even. However, its use eliminates the need for actually knowing the values of the coefficients as the relationship gives the value of the approximated function itself. That is

$$f(x) \cong R_{i,i+1,\cdots,i+k} \quad . \qquad (3.2.66)$$

Equation (3.2.65) conceals most of the difficulties of using rational functions or quotient polynomials. While the great utility of such approximating functions are their stability for extrapolation, we shall demonstrate their use for interpolation so as to compare the results with the other forms of interpolation we have investigated. Since the bulk of the other methods have four parameters available for the specification of the interpolating polynomial (i.e. they are cubic polynomials), we shall consider a quotient polynomial with four free parameters. This will require that we use four tabular entries which we shall choose to bracket the point x = 4 symmetrically. Such an approximating function would have the form

$$R_{1,2,3,4}(x) = (ax+b)/(\alpha x+\beta) \quad . \qquad (3.2.67)$$

However, the recursive form of equation (3.2.65) means that we will never determine the values of a, b, α, and β. The subscript notation used in equations (3.2.63) – (3.2.66) is designed to explicitly convey the recursive nature of the determination of the interpolative function. Each additional subscript denotes a successive "generation" in the development of the final result. One begins with the tabular data and the second of equations (3.2.65). Taking the data from table 3.3 so that $f(x_i) = Y_i$, we get for the second generation that represents the interpolative value at x = 4

$$\left.\begin{aligned} R_{1,2}(x) &= R_{2,2}(x) + \frac{R_{2,2}(x) - R_{1,1}(x)}{\left[\frac{(x-x_1)}{(x-x_2)}\right]\left[1 - \frac{R_{2,2}(x) - R_{1,1}(x)}{R_{2,2}(x)}\right] - 1} = 8 + \left[\frac{8-3}{\left[\frac{4-2}{4-3}\right]\left[1 - \frac{8-3}{8}\right] - 1}\right] = -12 \\ R_{2,3}(x) &= R_{3,3}(x) + \frac{R_{3,3}(x) - R_{2,2}(x)}{\left[\frac{(x-x_2)}{(x-x_3)}\right]\left[1 - \frac{R_{3,3}(x) - R_{2,2}(x)}{R_{3,3}(x)}\right] - 1} = 4 + \left[\frac{4-8}{\left[\frac{4-3}{4-5}\right]\left[1 - \frac{4-8}{4}\right] - 1}\right] = +\frac{16}{3} \\ R_{3,4}(x) &= R_{4,4}(x) + \frac{R_{4,4}(x) - R_{3,3}(x)}{\left[\frac{(x-x_3)}{(x-x_4)}\right]\left[1 - \frac{R_{4,4}(x) - R_{3,3}(x)}{R_{4,4}(x)}\right] - 1} = 2 + \left[\frac{2-4}{\left[\frac{4-2}{4-5}\right]\left[1 - \frac{2-4}{4}\right] - 1}\right] = +\frac{26}{5} \end{aligned}\right\}. \quad (3.2.68)$$

The third generation will contain only two terms so that

$$\left.\begin{aligned} R_{1,2,3}(x) &= R_{2,3}(x) + \frac{R_{2,2}(x) - R_{1,1}(x)}{\left[\frac{(x-x_1)}{(x-x_2)}\right]\left[1 - \frac{R_{2,2}(x) - R_{1,1}(x)}{R_{2,3}(x) - R_{2,2}(x)}\right] - 1} \\ R_{2,3,4}(x) &= R_{3,4}(x) + \frac{R_{3,4}(x) - R_{2,3}(x)}{\left[\frac{(x-x_2)}{(x-x_4)}\right]\left[1 - \frac{R_{3,4}(x) - R_{2,3}(x)}{R_{3,4}(x) - R_{3,3}(x)}\right] - 1} \end{aligned}\right\}. \quad (3.2.69)$$

Finally the last generation will have the single result.

$$\left.R_{1,2,3,4}(x) = R_{2,3,4}(x) + \frac{R_{2,3,4}(x) - R_{1,2,3}(x)}{\left[\frac{(x-x_1)}{(x-x_4)}\right]\left[1 - \frac{R_{2,3,4}(x) - R_{1,2,3}(x)}{R_{2,3,4}(x) - R_{2,3,3}(x)}\right] - 1}\right\}. \quad (3.2.70)$$

We can summarize this process neatly in the form of a "difference" Table (similar to Table 3.2 and Table 4.2) below.

Note how the recursion process drives the successive 'generations' of R toward the final result. This is a clear demonstration of the stability of this sort of scheme. It is this type of stability that makes the method desirable for extrapolation. In addition, such recursive procedures are very easy to program and quite fast in execution. The final result is given in equation (3.2.70), tabulated for comparison with other methods in Table 3.3, and displayed in Figure 3.2. This result is the smallest of the six results listed indicating that the

rapid tabular variation of the middle four points has been minimized. However, it still compares favorably with the second parabolic Lagrangian interpolation. While there is not a great deal of differentiation between these methods for interpolation, there is for extrapolation. The use of quotient polynomials for extrapolation is vastly superior to the use of polynomials, but one should always remember that one is basically after "free-lunch" and that more sophisticated is not necessarily better. Generally, it is risky to extrapolate any function far beyond one typical tabular spacing.

We have seen that the degree of the polynomial that is used for interpolation should be as low as possible to avoid unrealistic rapid variation of the interpolative function. This notion of providing a general "smoothness" to the function was also implicit in the choice of constraints for cubic splines. The constraints at the interior tabular points guarantee continuity up through the second derivative of the interpolative function throughout the full tabular range. The choice of $Y''_1 = Y''_n = 0$ that produces "natural" splines means that the interpolative function will vary no faster than linearly near the endpoints. In general, when one has to make an assumption concerning unspecified constants in an interpolation scheme, one chooses them so as to provide a slowly varying function. The extension of this concept to more complicated interpolation schemes is illustrated in the following highly successful interpolation algorithm.

Table 3.4

Parameters for Quotient Polynomial Interpolation

i	x	Yi	$R_{i,\ i}$	$R_{i,\ i+1,}$	$R_{i,\ i+1,\ i+2}$	$R_{i,\ i+1,\ i+2,\ i+3}$
0	1	1	0			
				0		
1	2	3	3		0	
				-12		
2	3	8	8		6.5714	0
	4			+16/3		5.2147
3	5	4	4		5.3043	0
				+26/5		
4	8	2	2		0	
				0		
5	10	1	0			

One of the most commonly chosen polynomials to be used for interpolation is the parabola. It tends not to vary rapidly and yet is slightly more sophisticated than linear interpolation. It will clearly require three tabular points to specify the three arbitrary constants of the parabola. One is then confronted with the problem of which of the two intervals between the tabular points should the point to be interpolated be placed. A scheme that removes this problem while far more importantly providing a gently varying function

proceeds as follows: Use four points symmetrically placed about the point to be interpolated. But instead of fitting a cubic to these four points, fit two parabolas, one utilizing the first three points and one utilizing the last three points. At this point one exercises an artistic judgment. One may choose to use the parabola with that exhibits the least curvature (i.e. the smallest value of the quadratic coefficient).

However, one may combine both polynomials to form a single quadratic polynomial where the contribution of each is weighted inversely by its curvature. Specifically, one could write this as

$$ {}^{2}_{k}\Phi_w(x) = \{a_{k+1}\,[{}^{2}_{k}\Phi(x)] + a_k\,[{}^{2}_{k+1}\Phi(x)]\}/(a_k+a_{k+1}) \quad , \tag{3.2.71} $$

where a_ks are the inverse of the coefficient of the x^2 term of the two polynomials and are given by

$$ a_k = \sum_{i=k}^{k+2} Y(x_i) \Big/ \prod_{j\neq i} (x_i - x_j) \quad , \tag{3.2.72} $$

and are just twice the inverse of the curvature of that polynomial. The ${}_k\Phi(x)$ are the Lagrange polynomials of second degree and are

$$ {}^{2}_{k}\Phi(x) = \sum_{i=k}^{k+2} Y(x_i)L_i(x) \quad . \tag{3.2.73} $$

Since each of the ${}_k\Phi(x)$s will produce the value of $Y(x_i)$ when $x = x_i$, it is clear that equation (3.2.71) will produce the values of $Y(x_2)$ and $Y(x_3)$ at the points x_2 and x_3 adjacent to the interpolative point. The functional behavior between these two points will reflect the curvature of both polynomials giving higher weight to the flatter, or more slowly varying polynomial. This scheme was developed in the 1960s by researchers at Harvard University who needed a fast and reliable interpolation scheme for the construction of model stellar atmospheres. While the justification of this algorithm is strictly aesthetic, it has been found to function well in a wide variety of situations. We may compare it to the other interpolation formulae by applying it to the same data from tables 3.1 and 3.3 that we have used throughout this section. In developing the parabolic Lagrangian formulae in section 3.1, we obtained the actual interpolative polynomials in equations (3.2.15) and (3.2.16). By differentiating these expressions twice, we obtain the a_ks required by equation (3.2.71) so that

$$ \left.\begin{aligned} a_1 &= 2\left|P_1''(4)\right|^{-1} = 3/7 \\ a_2 &= 2\left|P_2''(4)\right|^{-1} = 15/4 \end{aligned}\right\} \quad . \tag{3.2.74} $$

Substitution of these values into equation (3.2.71) yields a weighted Lagrangian interpolated value of

$$ {}^{2}_{1,2}\Phi_w = \{[3P_1(4)/7] + [15P_2(4)/4]\}/[(3/7)+(15/4)] = 6.000 \tag{3.2.75} $$

We have evaluated equation (3.2.75) by using the rational fraction values for $P_1(4)$ and $P_2(4)$ which are identical to the interpolative values given in table 3.1. The values for the relative weights given in equation (3.2.74) show that the first parabola will only contribute about 15% to the final answer do to its rapid variation. The more gently changing second parabola contributes the overwhelming majority of the final result reflecting our aesthetic judgment that slowly varying functions are more plausible for interpolating

functions. The fact that the result is identical to the result for linear interpolation is a numerical accident. Indeed, had round-off error not been a factor, it is likely that the result for the cubic splines would have also been exactly 6. However, this coincidence points up a common truth: "more sophisticated is not necessarily better".

Although slightly more complicated than quadratic Lagrangian interpolation, this scheme is rather more stable against rapid variation and is certainly more sophisticated than linear interpolation. In my opinion, its only real competition is the use of cubic splines and then only when the entire range of the table is to be used as in curve fitting. Even here there is no clear distinction as to which produces the more appropriate interpolative values, but an edge might be given to cubic splines on the basis of speed depending on the table size and number of required interpolative values.

It is worth taking a last look at the results in Table 3.3. We used the accuracy implied by the tables to provide a basis for the comparison of different interpolative methods. Indeed, some of the calculations were carried out as rational fractions to eliminate round-off error as the possible source of the difference between methods. The plausible values range from about 5.2 to 6.00. However, based on the tabular data, there is no real reason to prefer one value over another. The appropriate choice should revolve around the extent that one should expect an answer of a particular accuracy. None of the tabular data contain more than two significant figures. There would have to be some compelling reason to include more in the final result. Given the data spacing and the tabular variability, even two significant figures are difficult to justify. With that in mind, one could argue persuasively that linear interpolation is really all that is justified by this problem. This is an important lesson to be learned for it lies at the root of all numerical analysis. There is no need to use numerical methods that are vastly superior to the basic data of the problem.

3.3 Orthogonal Polynomials

Before leaving this chapter on polynomials, it is appropriate that we discuss a special, but very important class of polynomials known as the *orthogonal polynomials*. Orthogonal polynomials are defined in terms of their behavior with respect to each other and throughout some predetermined range of the independent variable. Therefore the orthogonality of a specific polynomial is not an important notion. Indeed, by itself that statement does not make any sense. The notion of orthogonality implies the existence of something to which the object in question is orthogonal. In the case of polynomials, that something happens to be other polynomials. In section 1.3 we discussed the notion of orthogonality for vectors and found that for a set of vectors to be orthogonal, no element of the set could be expressed in term of the other members of the set. This will also be true for orthogonal polynomials. In the case of vectors, if the set was complete it was said to span a vector space and any vector in that space could be expressed as a linear combination of the orthogonal basis vectors. Since the notion of orthogonality seems to hinge on two things being perpendicular to each other, it seems reasonable to say that two functions $f_1(x)$ and $f_2(x)$ are orthogonal if they are everywhere perpendicular to each other. If we imagine tangent vectors $\vec{t}_1(x)$ and $\vec{t}_2(x)$ defined at every point of each function, then if

$$\vec{t}_1(x) \bullet \vec{t}_2(x) = 0 \quad \forall x \quad , \tag{3.3.1}$$

one could conclude from equation (3.3.1) that $f_1(x)$ and $f_2(x)$ were mutually perpendicular at each value of x. If one considers the range of x to represent an infinite dimension vector space with each value of x representing a dimension so that the vectors $\vec{t}_1(x)$ represented basis vectors in that space, then orthogonality could be expressed as

$$\int_a^b t_1(x)t_2(x)dx = 0 \quad . \tag{3.3.2}$$

Thus, it is not unreasonable to generalize orthogonality of the functions themselves by

$$\int_a^b f_i(x)f_j(x)dx = 0 \quad , \quad i \neq j. \tag{3.3.3}$$

Again, by analogy to the case of vectors and linear transformations discussed in chapter 1 we can define two functions as being orthonormal if

$$\int_a^b w(x)\, f_i(x)f_j(x)\, dx = \delta_{ij} \quad . \tag{3.3.4}$$

Here we have included an additional function w(x) which is called a weight function. Thus the proper statement is that two functions are said to be orthonormal in the interval a≤ x≤ b, relative to a weight function w(x), if they satisfy equation (3.3.4). In this section we shall consider the subset of functions known as polynomials.

It is clear from equation (3.3.4) that orthonormal polynomials come in sets defined by the weight function and range of x. These parameters provide for an infinite number of such sets, but we will discuss only a few of the more important ones. While we will find it relatively easy to characterize the range of the independent variable by three distinct categories, the conditions for the weight function are far less stringent. Indeed the only constraint on w(x) is

$$w(x) > 0 \quad \forall x \in a \leq x \leq b \quad . \tag{3.3.5}$$

While one can find orthogonal functions for non-positive weight functions, it turns out that they are not unique and therefore not well defined. Simply limiting the weight function to positive definite functions in the interval a-b, still allows for an infinite number of such weight functions and hence an infinite number of sets of orthogonal polynomials.

Let us begin our search for orthogonal polynomials by using the orthogonality conditions to see how such polynomials can be generated. For simplicity, let us consider a finite interval from a to b. Now an orthogonal polynomial $\varphi_i(x)$ will be orthogonal to every member of the set of polynomials other than itself. In addition, we will assume (it can be proven) that the polynomials will form a complete set so that any polynomial can be generated from a linear combination of the orthogonal polynomials of the same degree or less. Thus, if $q_i(x)$ is an arbitrary polynomial of degree i, we can write

$$\int_a^b w(x)\phi_i(x)q_{i-1}(x)\, dx = 0 . \tag{3.3.6}$$

Now let

$$w(x)\phi_i(x) = \frac{d^i U_i(x)}{dx^i} \equiv U_i^{(i)}(x) \quad . \tag{3.3.7}$$

The function $U_i(x)$ is called the *generating function* of the polynomials $\varphi_i(x)$ and is itself a polynomial of degree 2i so that the ith derivative $U_i^{(i)}$ is an ith degree polynomial. Now integrate equation (3.3.7) by parts i-times to get

$$\int_a^b U_i^{(i)}(x) q_{i-1}(x)\,dx = 0 = \left[U_i^{(i-1)}(x)q_{i-1}(x) - U_i^{(i-2)}(x)q_{i-1}'(x) + \cdots + (-1)^{i-1} U_i(x) q_{i-1}^{(i-1)}(x)\right]\Big|_b^a \quad . \tag{3.3.8}$$

Since $q_i(x)$ is an arbitrary polynomial each term in equation (3.3.8) must hold separately so that

$$\left.\begin{aligned} U_i(a) = U_i'(a) = \cdots = U_i^{(i-1)}(a) = 0 \\ U_i(b) = U_i'(b) = \cdots = U_i^{(i-1)}(b) = 0 \end{aligned}\right\} \quad . \tag{3.3.9}$$

Since $\phi_i(x)$ is a polynomial of degree i we may differentiate it i+1 times to get

$$\frac{d^{i+1}}{dx^{i=1}}\left[\frac{1}{w(x)}\frac{d^i U_i(x)}{dx^i}\right] = 0 \quad . \tag{3.3.10}$$

This constitutes a differential equation of order 2i+1 subject to the 2i boundary conditions given by equation (3.3.9). The remaining condition required to uniquely specify the solution comes from the normalization constant required to make the integral of $\phi_i^2(x)$ unity. So at this point we can leave $U_i(x)$ uncertain by a scale factor. Let us now turn to the solution of equation (3.3.10) subject to the boundary conditions given by equation (3.3.9) for some specific weight functions w(x).

a. The Legendre Polynomials

Let us begin by restricting the range to $-1 \leq x \leq 1$ and taking the simplest possible weight function, namely

$$w(x) = 1 \quad , \tag{3.3.11}$$

so that equation (3.3.9) becomes

$$\frac{d^{2i+1}}{dx^{2i=1}}\left[U_i(x)\right] = 0 \quad . \tag{3.3.12}$$

Since $U_i(x)$ is a polynomial of degree 2i, an obvious solution which satisfies the boundary conditions is

$$U_i(x) = C_i(x^2\text{-}1)^i \quad . \tag{3.3.13}$$

Therefore the polynomials that satisfy the orthogonality conditions will be given by

$$\phi_i(x) = C_i \frac{d^i(x^2-1)^i}{dx^i} \quad . \tag{3.3.14}$$

If we apply the normalization criterion we get

$$\int_{-1}^{+1} \phi_i^2(x)\,dx = 1 = C_i \int_{-1}^{+1}\left[\frac{d^i(x^2-1)}{dx^i}\right] dx \quad , \tag{3.3.15}$$

so that

$$C_i = [2^i i!]^{-1} \quad . \tag{3.3.16}$$

We call the orthonormal polynomials with that normalization constant and satisfying equation (3.3.14) the *Legendre polynomials* and denote them by

$$P_i(x) = [2^i i!]^{-1} d^i(x^2-1)^i/dx^i \quad . \tag{3.3.17}$$

One can use equation (3.3.17) to verify that these polynomials will satisfy the recurrence relation

$$\left.\begin{aligned} P_{i+1}(x) &= \left[\frac{2i+1}{i+1}\right] x P_i(x) - \left[\frac{i}{i+1}\right] P_{i-1}(x) \\ P_0(x) &= 1 \\ P_1(x) &= x \end{aligned}\right\} \quad , \tag{3.3.18}$$

The set of orthogonal polynomials that covers the finite interval from -1 to +1 and whose members are orthogonal relative to the weight function w(x) = 1 are clearly the simplest of the orthogonal polynomials. One might be tempted to say that we have been unduly restrictive to limit ourselves to such a specific interval, but such is not the case. We may transform equation (3.3.15) to any finite interval by means of a linear transformation of the form

$$y(x) = x[(b-a)/2] + (a+b)/2 \quad , \tag{3.3.19}$$

so that we obtain an integral

$$\left[\frac{2}{b-a}\right] \int_a^b \phi_i(y)\phi_j(y)\, dy = \delta_{ij} \quad , \tag{3.3.20}$$

that resembles equation (3.3.4). Thus the Legendre polynomials form an orthonormal set that spans any finite interval relative to the unit weight function.

b. The Laguerre Polynomials

While we noted that the Legendre polynomials could be defined over any finite interval since the linear transformation required to reach such as interval didn't affect the polynomials, we had earlier mentioned that there are three distinct intervals that would have to be treated differently. Here we move to the second of these - the semi-infinite interval where $0 \leq x \leq \infty$. Clearly the limits of this interval cannot be reached from any finite interval by a linear transformation. A non-linear transformation that would accomplish that result would destroy the polynomic nature of any polynomials obtained in the finite interval. In addition, we shall have to consider a weight function that asymptotically approaches zero as x□□ as any polynomials in x will diverge making it impossible to satisfy the normalization condition. Perhaps the simplest weight function that will force a diverging polynomial to zero as $x \rightarrow \infty$ is $e^{-\alpha x}$**.** Therefore our orthogonal polynomials will take the form

$$\phi_I(x) = e^{\alpha x} \frac{d^i U_i(x)}{dx^i} \quad , \tag{3.3.21}$$

where the generating function will satisfy the differential equation

$$\frac{d^{i+1}}{dx^{i=1}}\left[e^{\alpha x}\,\frac{d^{i}U_{i}(x)}{dx^{i}}\right]=0 \quad , \tag{3.3.22}$$

and be subject to the boundary conditions

$$\left.\begin{aligned} U_{i}(0)=U_{i}^{'}(0)=\cdots=U_{i}^{(i-1)}(0)=0 \\ U_{i}(\infty)=U_{i}^{'}(\infty)=\cdots=U_{i}^{(i-1)}(\infty)=0 \end{aligned}\right\} \quad . \tag{3.3.23}$$

When subjected to those boundary conditions, the general solution to equation (3.3.22) will be

$$U_i(x) = C_i x^i e^{-\alpha x} \quad , \tag{3.3.24}$$

so that the polynomials can be obtained from

$$\phi_{i}(x)=\frac{e^{\alpha x}}{i!}\frac{d^{i}(x^{i}e^{-\alpha x})}{dx^{i}} \quad , \tag{3.3.25}$$

If we set $\alpha = 1$, then the resultant polynomials are called the *Laguerre polynomials* and when normalized have the form

$$\mathbf{L}_{i}=\frac{e^{x}}{i!}\frac{d^{i}(x^{i}e^{-x})}{dx^{i}} \quad , \tag{3,3,26}$$

and will satisfy the recurrence relation

$$\left.\begin{aligned} &\mathbf{L}_{i+1}(x)=\left[\frac{2i+1-x}{i+1}\right]\mathbf{L}_{i}(x)-\left[\frac{i}{i+1}\right]\mathbf{L}_{i-1}(x) \\ &\mathbf{L}_{0}(x)=1 \\ &\mathbf{L}_{1}(x)=1-x \end{aligned}\right\} \quad . \tag{3.3.27}$$

These polynomials form an orthonormal set in the semi-infinite interval relative to the weight function e^{-x}.

c. The Hermite Polynomials

Clearly the remaining interval that cannot be reached from either a finite interval or semi-infinite interval by means of a linear transformation is the full infinite interval $-\infty \leq x \leq +\infty$. Again we will need a weight function that will drive the polynomial to zero at both end points so that it must be symmetric in x. Thus the weight function for the semi-infinite interval will not do. Instead, we pick the simplest symmetric exponential $e^{\alpha^2 x^2}$, which leads to polynomials of the form

$$\phi_{i}(x)=e^{\alpha^{2}x^{2}}\,\frac{d^{i}U_{i}(x)}{dx^{i}} \quad , \tag{3.3.28}$$

that satisfy the differential equation

$$\frac{d^{i+1}}{dx^{i}}\left[e^{\alpha^{2}x^{2}}\,\frac{d^{i}U_{i}(x)}{dx^{i}}\right]=0 \quad , \tag{3.3.29}$$

subject to the boundary conditions

$$U_i(\pm\infty) = U_i'(\pm\infty) = \cdots = U_i^{(i-1)}(\pm\infty) = 0 \quad . \tag{3.3.30}$$

This has a general solution satisfying the boundary conditions that look like

$$U_i(x) = C_i e^{-\alpha^2 x^2} \quad , \tag{3.3.31}$$

which when normalized and with $\alpha = 1$, leads to the *Hermite polynomials* that satisfy

$$H_i(x) = (-1)^i e^{x^2} \frac{d^i e^{-x^2}}{dx^i} \quad . \tag{3.3.32}$$

Table 3.5

The First Five Members of the Common Orthogonal Polynomials

i	$P_i(x)$	$L_i(x)$	$H_i(x)$
0	1	1	1
1	x	1-x	2x
2	$(3x^2-1)/2$	$(2-4x+x^2)/2$	$2(2x^2-1)$
3	$x(5x^2-3)/2$	$(6-18x+9x^2-x^3)/6$	$4x(2x^2-3)$
4	$(35x^4-30x^2+3)/8$	$(24-96x+72x^2-6x^3+x^4)/24$	$4(4x^4-16x^2+3)$

Like the other polynomials, the Hermite polynomials can be obtained from a recurrence relation. For the Hermite polynomials that relation is

$$\left.\begin{aligned} H_{i+1}(x) &= 2xH_i(x) - 2iH_{i-1}(x) \\ H_0(x) &= 1 \\ H_1(x) &= 2x \end{aligned}\right\} \quad . \tag{3.3.31}$$

.We have now developed sets of orthonormal polynomials that span the three fundamental ranges of the real variable. Many other polynomials can be developed which are orthogonal relative to other weight functions, but these polynomials are the most important and they appear frequently in all aspects of science.

d. Additional Orthogonal Polynomials

There are as many additional orthogonal polynomials as there are positive definite weight functions. Below we list some of those that are considered to be classical orthogonal polynomials as they turn up frequently in mathematical physics. A little inspection of Table 3.6 shows that the Chebyschev polynomials are special cases of the more general Gegenbauer or Jacobi polynomials. However, they turn up sufficiently frequently that it is worth saying more about them. They can be derived from the generating function in the same manner that the other orthogonal polynomials were, so we will only quote the results. The Chebyschev polynomials of the first kind can be obtained from the reasonably simple trigonometric formula

$$T_i(x) = \cos[i \cos^{-1}(x)] \quad . \tag{3.3.34}$$

Table 3.6

Classical Orthogonal Polynomials of the Finite Interval

NAME	WEIGHT FUNCTION W(X)
Legendre	1
Gegenbauer or Ultraspherical	$(1-x^2)^{\lambda-\frac{1}{2}}$
Jacobi or Hypergeometric	$(1-x)^{\alpha}(1-x)^{\beta}$
Chebyschev of the first kind	$(1-x^2)^{-\frac{1}{2}}$
Chebyschev of the second kind	$(1-x^2)^{+\frac{1}{2}}$

However, in practice they are usually obtained from a recurrence formula similar to those for the other polynomials. Specifically

$$\left.\begin{aligned} T_{i+1}(x) &= 2xT_i(x) - T_{i-1}(x) \\ T_0(x) &= 1 \\ T_1(x) &= x \end{aligned}\right\}. \tag{3.3.35}$$

The Chebyschev polynomials of the second kind are represented by the somewhat more complicated trigonometric formula

$$V_i(x) = \sin[(i+1)\cos^{-1}(x)]/\sin[\cos^{-1}(x)]\ , \tag{3.3.36}$$

and obey the same recurrence formula as Chebyschev polynomials of the first kind so

$$\left.\begin{aligned} V_{i+1}(x) &= 2xV_i(x) - V_{i-1}(x) \\ V_0(x) &= 1 \\ V_1(x) &= 2x \end{aligned}\right\}. \tag{3.3.37}$$

Only the starting values are slightly different. Since they may be obtained from a more general class of polynomials, we should not be surprised if there are relations between them. There are, and they take the form

$$\left.\begin{aligned} T_i(x) &= V_i(x) - xV_{i-1}(x) \\ (1-x^2)V_{i-1}(x) &= xT_i(x) - T_{i+1}(x) \end{aligned}\right\}. \tag{3.3.38}$$

Since the orthogonal polynomials form a complete set enabling one to express an arbitrary polynomial in terms of a linear combination of the elements of the set, they make excellent basis functions for interpolation formulae. We shall see in later chapters that they provide a basis for curve fitting that provides great numerical stability and ease of solution. In the next chapter, they will enable us to generate formulae to evaluate integrals that yield great precision for a minimum of effort. The utility of these

functions is of central importance to numerical analysis. However, all of the polynomials that we have discussed so far form orthogonal sets over a continuous range of x. Before we leave the subject of orthogonality, let us consider a set of functions, which form a complete orthogonal set with respect to a *discrete* set of points in the finite interval.

e. The Orthogonality of the Trigonometric Functions

At the beginning of the chapter where we defined polynomials, we represented the most general polynomial in terms of basis functions $\varphi_i(x)$. Consider for a moment the case where

$$\varphi_i(x) = \sin(i\pi x) \quad . \tag{3.3.39}$$

Now integration by parts twice, recovering the initial integral but with a sign change, or perusal of any good table of integrals[4] will convince one that

$$\int_{-1}^{+1} \sin(k\pi x)\, \sin(j\pi x)\, dx = \int_{-1}^{+1} \cos(k\pi x)\, \cos(j\pi x)\, dx \quad = \quad \delta_{kj} \; . \tag{3.3.40}$$

Thus sines and cosines form orthogonal sets of functions of the real variable in the finite interval. This will come as no surprise to the student with some familiarity with Fourier transforms and we will make much of it in chapters to come. But what is less well known is that

$$\frac{1}{N} \sum_{x=0}^{2N-1} \sin(k\pi x / N)\, \sin(j\pi x / N) = \frac{1}{N} \sum_{x=0}^{2N-1} \cos(k\pi x / N)\, \cos(j\pi x / N) = \delta_{kj} \; , \quad 0 < (k + j) < 2N \; , \tag{3.3.41}$$

which implies that these functions also form an orthogonal set on the finite interval for a discrete set of points. The proof of this result can be obtained in much the same way as the integral, but it requires some knowledge of the finite difference calculus (see Hamming[5] page 44, 45). We shall see that it is this discrete orthogonality that allows for the development of Fourier series and the numerical methods for the calculation of Power Spectra and "Fast Fourier Transforms". Thus the concept of orthogonal functions and polynomials will play a role in much of what follows in this book.

Chapter 3 Exercises

1. Find the roots of the following polynomial

   $2x^5 - 225x^4 + 2613x^3 - 11516x^2 + 21744x - 14400 = P(x)$,

   a. by the Graffe Root-squaring method,
   b. any interative method,
   c. then compare the accuracy of the two methods.

2. Find the roots of the following polynomials:

   a. $P(x) = x^4 - 7x^3 + 13x^2 - 7x + 12$

   b. $P(x) = 2x^4 - 15x^3 + 34x^2 - 25x + 14$

   c. $P(x) = 4x^4 - 9x^3 - 12x^2 - 35x - 18$

   d. $P(x) = +0.0021(x^3+x) + 1.000000011x^2 + 0.000000011.$

   Comment of the accuracy of your solution.

3. Find Lagrangian interpolation formulae for the cases where the basis functions are

   a. $\varphi_i(x) = e^{ix}$

   b. $\varphi_i(x) = \sin(i\pi x/h)$,

   where h is a constant interval spacing between the points x_i.

4. Use the results from problem 3 to obtain values for f(x) at x=0.5, 0.9 and 10.3 in the following table:

x_i	$f(x_i)$
0.0	1.0
0.4	2.0
0.8	3.0
1.2	5.0
2.0	3.0
5.0	1.0
8.0	8.0 .

   Compare with ordinary Lagrangian interpolation for the same degree polynomials and cubic splines. Comment on the result.

5. Given the following table, approximate f(x) by

$$f(x) = \sum_{i=1}^{n} a_i \sin(ix).$$

Determine the "best" value of n for fitting the table. Discuss your reasoning for making the choice you made.

x_i	$f(x_i)$
1.0	+.4546
2.0	-.3784
3.0	-.1397
4.0	+.4947
5.0	-.2720
6.0	-.2683
7.0	+.4953
8.0	-.1439

6. Find the normalization constants for

   a. Hermite polynomials
   b. Laguerre polynomials
   c. Legendre polynomials that are defined in the interval $-1 \rightarrow +1$.

7. Use the rules for the manipulation of determinants given in chapter 1 (page 8) to show how the Vandermode determinant takes the form given by equation (3.3.7)

8. In a manner similiar to problem 7, show how the Lagrangian polynomials take the form given by equation (3.2.9).

9. Explicitly show how equation (3.2.29) is obtained from equations (3.2.23), (3.2.24), and (3.2.26).

10. Integrate equation (3.2.53) to obtain the tri-diagonal equations (3.2.54). Show explicitly how the constraints of the derivatives of Y_i enter into the problem.

11. By obtaining equation (3.3.18) from equation (3.3.17) show that one can obtain the recurrence relations for orthogonal polynomials from the defining differential equation.

12. Find the generating function for Gegenbauer polynomials and obtain the recurrence relation for them.

13. Show that equation (3.3.41) is indeed correct.

Chapter 3 References and Supplemental Reading

1. Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T., "Numerical Recipies the Art of Scientific Computing" (1986), Cambridge University Press Cambridge, New York, New Rochelle, Melbourne, Sydney.

2. Acton, Forman S., "Numerical Methods That Work", (1970) Harper and Row, New York.

3. Stoer, J. and Bulirsch, R., "Introduction to Numerical Analysis" (1980), Springer-Verlag, New York, §2.2.

4. Gradshteyn, I.S. and Ryzhik,I.M., "Table of Integrals, Series, and Products : corrected and enlarged edition" (1980), (ed. A. Jeffrey), Academic Press, New York, London, Toronto, Sydney, San Francisco, pp 139-140.

5. Hamming, R.W., "Numerical Methods for Scientists and Engineers" (1962) McGraw-Hill Book Co., Inc., New York, San Francisco, Toronto, London.

For an excellent general discussion of polynomials one should read

6. Moursund, D.G., and Duris, C.S., "Elementary Theory and Applications of Numerical Analysis" (1988) Dover Publications, Inc. New York, pp 108-140.

A very complete discussion of classical orthogonal polynomials can be found in

7. Bateman, H., The Bateman Manuscript Project, "Higher Transcendental Functions" (1954) Ed. A. Erdélyi, Vol. 3, McGraw-Hill Book Co., Inc. New York, Toronto, London, pp 153-228.

4

Numerical Evaluation of Derivatives and Integrals

• • •

The mathematics of the Greeks was insufficient to handle the concept of time. Perhaps the clearest demonstration of this is Zeno's Paradox regarding the flight of arrows. Zeno reasoned that since an arrow must cover half the distance between the bow and the target before traveling all the distance and half of that distance (i.e. a quarter of the whole) before that, etc., that the total number of steps the arrow must cover was infinite. Clearly the arrow could not accomplish that in a finite amount of time so that its flight to the target was impossible. This notion of a limiting process of an infinitesimal distance being crossed in an infinitesimal time producing a constant velocity seems obvious to us now, but it was a fundamental barrier to the development of Greek science. The calculus developed in the 17th century by Newton and Leibnitz has permitted, not only a proper handling of time and the limiting process, but the mathematical representation of the world of phenomena which science seeks to describe. While the analytic representation of the calculus is essential in this description, ultimately we must numerically evaluate the analytic expressions that we may develop in order to compare them with the real world.

Again we confront a series of subjects about which books have been written and entire courses of study developed. We cannot hope to provide an exhaustive survey of these areas of numerical analysis, but only develop the basis for the approach to each. The differential and integral operators reviewed in chapter 1 appear in nearly all aspects of the scientific literature. They represent mathematical processes or operations to be carried out on continuous functions and therefore can only be approximated by a series of discrete numerical operations. So, as with any numerical method, we must establish criteria for which the discrete operations will accurately represent the continuous operations of differentiation and integration. As in the case of interpolation, we shall find the criteria in the realm of polynomial approximation.

4.1 Numerical Differentiation

Compared with other subjects to be covered in the study of numerical methods, little is usually taught about numerical differentiation. Perhaps that is because the processes should be avoided whenever possible. The reason for this can be seen in the nature of polynomials. As was pointed out in the last chapter on interpolation, high degree polynomials tend to oscillate between the points of constraint. Since the derivative of a polynomial is itself a polynomial, it too will oscillate between the points of constraint, but perhaps not quite so wildly. To minimize this oscillation, one must use low degree polynomials which then tend to reduce the accuracy of the approximation. Another way to see the dangers of numerical differentiation is to consider the nature of the operator itself. Remember that

$$\frac{df(x)}{dx} = \lim_{\Delta x \to 0} \frac{f(x+\Delta x) - f(x)}{\Delta x} \quad . \tag{4.1.1}$$

Since there are always computational errors associated with the calculation of f(x), they will tend to be present as $\Delta x \to 0$, while similar errors will not be present in the calculation of Δx itself. Thus the ratio will end up being largely determined by the computational error in f(x). Therefore numerical differentiation should only be done if no other method for the solution of the problem can be found, and then only with considerable circumspection.

a. Classical Difference Formulae

With these caveats clearly in mind, let us develop the formalisms for numerically differentiating a function f(x). We have to approximate the continuous operator with a finite operator and the finite difference operators described in chapter 1 are the obvious choice. Specifically, let us take the finite difference operator to be defined as it was in equation (1.5.1). Then we may approximate the derivative of a function f(x) by

$$\frac{df(x)}{dx} = \frac{\Delta f(x)}{\Delta x} \quad . \tag{4.1.2}$$

The finite difference operators are linear so that repeated operations with the operator lead to

$$\Delta^n f(x) = \Delta[\Delta^{n-1} f(x)] \quad . \tag{4.1.3}$$

This leads to the *Fundamental Theorem of the Finite Difference Calculus* which is

> *The* nth *difference of a polynomial of degree* n *is a constant* (a_n n! h^n)*, and the* (n+1) *st difference is zero.*

Clearly the extent to which equation (4.1.3) is satisfied will depend partly on the value of h. Also the ability to repeat the finite difference operation will depend on the amount of information available. To find a non-trivial nth order finite difference will require that the function be approximated by an nth degree polynomial which has n+1 linearly independent coefficients. Thus one will have to have knowledge of the function for at least n+1 points. For example, if one were to calculate finite differences for the function x^2 at a finite set of points x_i, then one could construct a finite difference table of the form:

Table 4.1

A Typical Finite Difference Table for $f(x) = x^2$

X_I	$F(X_I)$	$\Delta F(X)$	$\Delta^2 F(X)$	$\Delta^3 F(X)$
2	f(2)=4			
		Δf(2)=5		
3	f(3)=9		$\Delta^2 f(2)=2$	
		Δf(3)=7		$\Delta^3 f(2)=0$
4	f(4)=16		$\Delta^2 f(3)=2$	
		Δf(4)=9		$\Delta^3 f(3)=0$
5	f(5)=25		$\Delta^2 f(4)=2$	
		Δf(5)=11		
6	f(6)=36			

This table nicely demonstrates the fundamental theorem of the finite difference calculus while pointing out an additional problem with repeated differences. While we have chosen f(x) to be a polynomial so that the differences are exact and the fundamental theorem of the finite difference calculus is satisfied exactly, one can imagine the situation that would prevail should f(x) only approximately be a polynomial. The truncation error that arises from the approximation would be quite significant for $\Delta f(x_i)$ and compounded for $\Delta^2 f(x_i)$. The propagation of the truncation error gets progressively worse as one proceeds to higher and higher differences. The table illustrates an additional problem with finite differences. Consider the values of $\Delta f(x_i)$. They are not equal to the values of the derivative at x_i implied by the definition of the forward difference operator at which they are meant to apply. For example Δf(3)=7 and with h=1 for this table would suggest that f '(3)=7, but simple differentiation of the polynomial will show that f '(3)=6. One might think that this could be corrected by averaging Δf (2) and Δf (3), or by re-defining the difference operator so that it didn't always refer backward. Such an operator is known as the central difference operator which is defined as

$$\delta f(x) \equiv f(x+\tfrac{1}{2}h) - f(x-\tfrac{1}{2}h) \ . \qquad (4.1.4)$$

However, this does not remove the problem that the value of the nth difference, being derived from information spanning a large range in the independent variable, may not refer to the nth derivative at the point specified by the difference operator.

In Chapter 1 we mentioned other finite difference operators, specifically the shift operator E and the identity operator I (see equation 1.5.3). We may use these operators and the relation between them given by equation (1.5.4), and the binomial theorem to see that

$$\Delta^k[f(x)] = [E - I]^k[f(x)] = \sum_{i=0}^{k}(-1)^k\left(\frac{k}{i}\right)E^i[f(x)] = \sum_{i=0}^{k}(-1)^{k-1}\left(\frac{k}{i}\right)f(x+i) \quad , \tag{4.1.5}$$

where $\binom{k}{i}$ is the binomial coefficient which can be written as

$$\left(\frac{k}{i}\right) = \frac{k!}{(k-i)!i!} \quad . \tag{4.1.6}$$

One can use equation (4.1.5) to find the kth difference for equally spaced data without constructing the entire difference table for the function. If a specific value of $f(x_j)$ is missing from the table, and one assumes that the function can be represented by a polynomial of degree k-1, then, since $\Delta^k f(x_i) = 0$, equation (4.1.5) can be solved for the missing value of $f(x_j)$.

While equation (4.1.5) can be used to find the differences of any equally spaced function $f(x_i)$ and hence is an estimate of the kth derivative, the procedure is equivalent to finding the value of a polynomial of degree n-k at a specific value of x_i. Therefore, we may use any interpolation formula to obtain an expression for the derivative at some specific point by differentiation of the appropriate formula. If we do this for Lagrangian interpolation, we obtain

$$\Phi'(x) = \sum_{i=1}^{n} f(x_i)L_i'(x) \quad , \tag{4.1.7}$$

where

$$L_i'(x) = \sum_{k=1}^{n}\prod_{\substack{j\neq i\\ j\neq k}}^{n}(x - x_j) \Big/ (x_i - x_j) \quad . \tag{4.1.8}$$

Higher order formulae can be derived by successive differentiation, but one must always use numerical differentiation with great care.

b. Richardson Extrapolation for Derivatives

We will now consider a "clever trick" that enables the improvement of nearly all formulae that we have discussed so far in this book and a number yet to come. It is known as Richardson extrapolation, but differs from what is usually meant by extrapolation. In chapter 3 we described extrapolation in terms of extending some approximation formula beyond the range of the data which constrained that formula. Here we use it to describe a process that attempts to approximate the results of any difference or difference based formula to limit where the spacing h approaches zero. Since h is usually a small number, the extension, or extrapolation, to zero doesn't seem so unreasonable. Indeed, it may not seem very important, but remember the limit of the accuracy on nearly all approximation formulae is set by the influence of round-off error in the case where an approximating interval becomes small. This will be

particularly true for problems of the numerical solution of differential equations discussed in the next chapter. However, we can develop and use it here to obtain expressions for derivatives that have greater accuracy and are obtained with greater efficiency than the classical difference formulae. Let us consider the special case where a function f(x) can be represented by a Taylor series so that if

$$x = x_0 + kh \quad , \tag{4.1.9}$$

then

$$f(x_0 + kh) = f(x_0) + khf'(x_0) + \frac{(kh)^2 f''(x_0)}{2!} + \frac{(kh)^3 f^{(3)}(x_0)}{3!} + \cdots + \frac{(kh)^n f^{(n)}(x_0)}{n!} \quad . \tag{4.1.10}$$

Now let us make use of the fact that h appears to an odd power in even terms of equation (4.1.10). Thus if we subtract the a Taylor series for -k from one for +k, the even terms will vanish leaving

$$f(x_0 + kh) - f(x_0 - kh) = 2khf'(x_0) + \frac{2(kh)^3 f^{(3)}(x_0)}{3!} + \cdots + \frac{(kh)^{2n+1} f^{(2n+1)}(x_0)}{(2n+1)!} \quad . \tag{4.1.11}$$

The functional relationship on the left hand side of equation (4.1.11) is considered to be some mathematical function whose value is precisely known, while the right hand side is the approximate relationship for that function. That relationship now only involves odd powers of h so that it converges much faster than the original Taylor series. Now evaluate equation (4.1.11) for k = 1 and 2 explicitly keeping just the first two terms on the right hand side so that

$$\left.\begin{aligned} f(x_0 + h) - f(x_0 - h) &= 2hf'(x_0) + 2h^3 f^{(3)}(x_0)/6 \quad + \cdots + R(h^5) \\ f(x_0 + 2h) - f(x_0 - 2h) &= 4hf'(x_0) + 16h^3 f^{(3)}(x_0)/6 \quad + \cdots + \tilde{R}(h^5) \end{aligned}\right\} \quad . \tag{4.1.12}$$

We now have two equations from which the term involving the third derivative may be eliminated yielding

$$f(x_0\text{-}2h)\text{-}8f(x_0\text{-}h)+8f(x_0+h)\text{-}f(x_0+2h) = -12hf'(x_0)+R(h^5)\text{-}\tilde{R}(h^5) \, , \tag{4.1.13}$$

and solving for $f'(x_0)$ we get.

$$f'(x_0) = [f(x_0\text{-}2h) - 8f(x_0\text{-}h) + 8f(x_0+h) - f(x_0+2h)]/(12h) + O(h^4). \tag{4.1.14}$$

It is not hard to show that the error term in equation (4.1.13) divided by h is $O(h^4)$. Thus we have an expression for the *derivative* of the function f(x) evaluated at some value of $x = x_0$ which requires four values of the function and is exact for cubic polynomials. This is not too surprising as we have four free parameters with which to fit a Taylor series or alternately a cubic polynomial and such polynomials will be unique. What is surprising is the rapid rate of convergence with decreasing interval h. But what is even more amazing is that this method can be generalized to any approximation formulae that can be written as

$$\left.\begin{aligned} f(x) &= \Phi(x, \alpha h) + Ch^n + O(h^m) \\ m &> n, \; \alpha > 0, \; \alpha \neq 1 \end{aligned}\right\} \quad . \tag{4.1.15}$$

so that

$$f(x) = \frac{\alpha^n \Phi(x,h) - \Phi(x,\alpha h)}{\alpha^n - 1} + O(h^m) \quad . \tag{4.1.16}$$

Indeed, it could be used to obtain an even higher order approximation for the derivative utilizing more tabular points. We shall revisit this method when we consider the solution to differential equations in Chapter 5.

4.2 Numerical Evaluation of Integrals: Quadrature

While the term *quadrature* is an old one, it is the correct term to use for describing the numerical evaluation of integrals. The term *numerical integration* should be reserved for describing the numerical solution of differential equations (see chapter 5). There is a genuine necessity for the distinction because the very nature of the two problems is quite different. Numerically evaluating an integral is a rather common and usually stable task. One is basically assembling a single number from a series of independent evaluations of a function. Unlike numerical differentiation, numerical quadrature tends to average out random computational errors.

Because of the inherent stability of numerical quadrature, students are generally taught only the simplest of techniques and thereby fail to learn the more sophisticated, highly efficient techniques that can be so important when the integrand of the integral is extremely complicated or occasionally the result of a separate lengthy study. Virtually all numerical quadrature schemes are based on the notion of polynomial approximation. Specifically, the quadrature scheme will give the exact value of the integral if the integrand is a polynomial of some degree n. The scheme is then said to have a *degree of precision* equal to n. In general, since a nth degree polynomial has n+1 linearly independent coefficients, a quadrature scheme will have to have n+1 adjustable parameters in order to accurately represent the polynomial and its integral. Occasionally, one comes across a quadrature scheme that has a degree of precision that is greater than the number of adjustable parameters. Such a scheme is said to be hyper-efficient and there are a number of such schemes known for multiple integrals. For single, or one dimensional, integrals, there is only one which we will discuss later.

a. The Trapezoid Rule

The notion of evaluating an integral is basically the notion of evaluating a sum. After all the integral sign $\int$ is a stylized S that stands for a continuous "sum". The symbol Σ as introduced in equation (1.5.2) stands for a discrete or finite sum, which, if the interval is taken small enough, will approximate the value for the integral. Such is the motivation for the Trapezoid rule which can be stated as

$$\int_a^b f(x)\,dx = \sum_{i=1}^{n-1} \frac{f(x_{i+1}) + f(x_i)}{2} \Delta x_i \quad . \tag{4.2.1}$$

The formula takes the form of the sum of a discrete set of average values of the function each of which is multiplied by some sort of weight W_i. Here the weights play the role of the adjustable parameters of the quadrature formula and in the case of the trapezoid rule the weights are simply the intervals between functional evaluations. A graphical representation of this can be seen below in Figure 4.1

The meaning of the rule expressed by equation (4.2.1) is that the integral is approximated by a series of trapezoids whose upper boundaries in the interval Δx_i are straight lines. In each interval this formula would have a degree of precision equal to 1 (i.e. equal to the number of free parameters in the interval minus one). The other "adjustable" parameter is the 2 used in obtaining the average of $f(x_i)$ in the interval. If we divide the interval $a \rightarrow b$ equally then the Δx_i's have the particularly simple form

$$\Delta x_i = (b-a)/(n-1)\,. \tag{4.2.2}$$

In Chapter 3, we showed that the polynomic form of the integrand of an integral was unaffected by a linear transformation [see equations (3.3.19) and (3.3.20)]. Therefore, we can rewrite equation (4.2.1) as

$$\int_a^b f(x)\,dx = \frac{(b-a)}{2}\int_{-1}^{+1} f(y)\,dy = \frac{(b-a)}{2}\sum_{i=1}^{n}\frac{f[x(y_{i+1})]+f[x(y_i)]}{2}W'_i \quad , \qquad (4.2.3)$$

where the weights for an equally spaced interval are

$$W'_i = 2/(n-1) \quad . \qquad (4.2.4)$$

If we absorb the factor of (b-a)/2 into the weights we see that for both representations of the integral [i.e. equation (4.2.1) and equation (4.2.3)] we get

$$\sum_{i=1}^{n} W_i = b-a \quad . \qquad (4.2.5)$$

Notice that the function f(x) plays absolutely no role in determining the weights so that once they are determined; they can be used for the quadrature of any function. Since any quadrature formula that is exact for polynomials of some degree greater than zero must be exact for $f(x) = x^0$, the sum of the weights of any quadrature scheme must be equal to the total interval for which the formula holds.

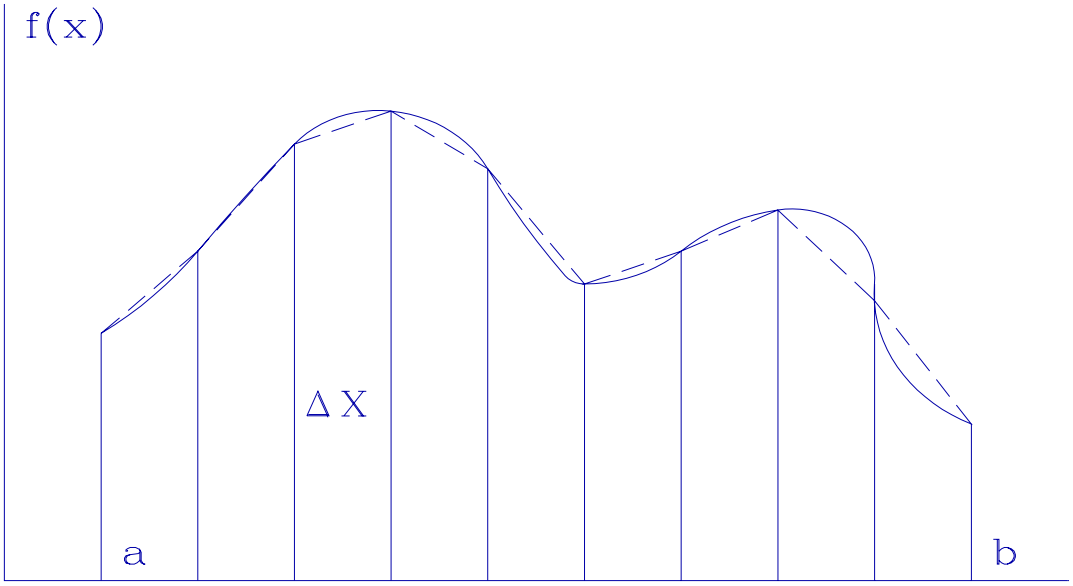


Figure 4.1 shows a function whose integral from a *to* b *is being evaluated by the trapezoid rule. In each interval a straight line approximates the function* Δx_i.

b. Simpson's Rule

The trapezoid rule has a degree of precision of 1 as it fits straight lines to the function in the interval. It would seem that we should be able to do better than this by fitting a higher order polynomial to the function. So instead of using the functional values at the endpoints of the interval to represent the function by a straight line, let us try three equally spaced points. That should allow us to fit a polynomial with three adjustable parameters (i.e. a parabola) and obtain a quadrature formula with a degree of precision

of 2. However, we shall see that this quadrature formula actually has a degree of precision of 3 making it a hyper-efficient quadrature formula and the only one known for integrals in one dimension.

In general, we can construct a quadrature formula from an interpolation formula by direct integration. In chapter 3 we developed interpolation formulae that were exact for polynomials of an arbitrary degree n. One of the more general forms of these interpolation formulae was the Lagrange interpolation formula given by equation (3.2.8). In that equation Φ(x) was a polynomial of degree n and was made up of a linear combination of the Lagrange polynomials $L_i(x)$. Since we are interested in using three equally spaced points, n will be 2. Also, we have seen that any finite interval is equivalent to any other for the purposes of fitting polynomials, so let us take the interval to be 2h so that our formula will take the form

$$\int_0^{2h} f(x)\,dx = \sum_{i=0}^{2} f(x_i) W_i = \sum_{i=0}^{2} f(x_i) \int_0^{2h} L_i(x)\,dx \quad . \tag{4.2.6}$$

Here we see that the quadrature weights W_i are given by

$$W_i = \int_0^{2h} L_i(x)\,dx = \int_0^{2h} \prod_{\substack{j\neq i \\ j=0}}^{2} \frac{(x - x_i)}{(x_i - x_j)}\,dx \quad . \tag{4.2.7}$$

Now the three equally spaced points in the interval 2h will have x = 0, h, and 2h. For equal intervals we can use equation (3.2.11) to evaluate the Lagrange polynomials to get

$$\left.\begin{aligned} L_0(x) &= \frac{(x-h)(x-2h)}{2h^2} = \frac{(x^2 - 3xh + 2h^2)}{2h^2} \\ L_1(x) &= \frac{(x-0)(x-2h)}{h^2} = \frac{(x^2 - 2xh)}{h^2} \\ L_2(x) &= \frac{(x-0)(x-h)}{2h^2} = \frac{(x^2 - xh)}{2h^2} \end{aligned}\right\} \quad . \tag{4.2.8}$$

Therefore the weights for Simpson's rule become

$$\left.\begin{aligned} W_0 &= \int_0^{2h} L_0(x)\,dx = \frac{(8h^3/3 - 12h^3/2 + 4h^3)}{2h^2} = \frac{h}{3} \\ W_1 &= \int_0^{2h} L_1(x)\,dx = \frac{(8h^3/3 - 8h^3/2)}{h^2} = \frac{4h}{3} \\ W_2 &= \int_0^{2h} L_2(x)\,dx = \frac{(8h^3/3 - 4h^3/2)}{h^2} = \frac{h}{3} \end{aligned}\right\} \quad . \tag{4.2.9}$$

Actually we need only to have calculated two of the weights since we know that the sum of the weights had to be 2h. Now since h is only half the interval we can write

$$h = \Delta x/2 \quad , \tag{4.2.10}$$

so that the approximation formula for Simpson's quadrature becomes

$$\int_0^{\Delta x} f(x)\,dx = \sum_{i=0}^{2} f(x_i)W_i = \frac{\Delta x}{6}\left[f(x_0) + 4f(x_1) + f(x_2)\right] . \qquad (4.2.11)$$

Now let us confirm the assertion that Simpson's rule is hyper-efficient. We know that the quadrature formula will yield exact answers for quadratic polynomials, so consider the evaluation of a quartic. We pick the extra power of x in anticipation of the result. Thus we can write

$$\int_0^{\Delta x} (\alpha x^3 + \beta x^4)\,dx = \frac{\alpha \Delta x^4}{4} + \frac{\beta \Delta x^5}{5} = \frac{\Delta x}{6}\left(4\alpha\left[\frac{\Delta x}{2}\right]^3 + \alpha(\Delta x)^3 + 4\beta\left[\frac{\Delta x}{2}\right]^4 + \beta(\Delta x)^4 \right) + R(\Delta x)$$

$$= \frac{\alpha(\Delta x)^4}{4} + \frac{5\beta(\Delta x)^5}{24} + R(\Delta x) \qquad . \qquad (4.2.12)$$

Here $R(\Delta x)$ is the error term for the quadrature formula. Completing the algebra in equation (4.2.12) we get

$$R(\Delta x) = \beta(\Delta x)^5/120 \quad . \qquad (4.2.13)$$

Clearly the error in the integral goes as the interval to the fifth power and not the fourth power. So the quadrature formula will have no error for cubic terms in the integrand and the formula is indeed hyper-efficient. Therefore Simpson's rule is a surprisingly good quadrature scheme having a degree of precision of 3 over the interval Δx. Should one wish to span a larger interval (or reduce the spacing for a given interval), one could write

$$\int_0^{h\Delta x} f(x)\,dx = \sum_{i=1}^{n} \int_{(i-1)\Delta x}^{i\Delta x} f(x_i)\,dx = \frac{\Delta x}{6}\left[f(x_1) + 4f(x_2) + 2f(x_3) + 4f(x_4) + \cdots + 4f(x_{n-1}) + f(x_n)\right] . \qquad (4.2.14)$$

By breaking the integral up into sub-intervals, the function need only be well approximated locally by a cubic. Indeed, the function need not even be continuous across the separate boundaries of the sub-intervals. This form of Simpson's rule is sometimes called a *running Simpson's rule* and is quite easy to implement on a computer. The hyper-efficiency of this quadrature scheme makes this a good "all purpose" equal interval quadrature algorithm.

c. Quadrature Schemes for Arbitrarily Spaced Functions

As we saw above, it is possible to obtain a quadrature formula from an interpolation formula and maintain the same degree of precision as the interpolation formula. This provides the basis for obtaining quadrature formula for functions that are specified at arbitrarily spaced values of the independent variable x_i. For example, simply evaluating equation (4.2.6) for an arbitrary interval yields

$$\int_a^b f(x)\,dx = \sum_{i=0}^{n} f(x_i) \int_a^b L_i(x)\,dx \quad , \qquad (4.2.15)$$

which means that the weights associated with the arbitrarily spaced points x_i are

$$W_i = \int_a^b L_i(x)\,dx \quad . \qquad (4.2.16)$$

However, the analytic integration of $L_i(x)$ can become tedious when n becomes large so we give an alternative strategy for obtaining the weights for such a quadrature scheme. Remember that the scheme is to

have a degree of precision of n so that it must give the exact answers for *any* polynomial of degree n. But there can only be one set of weights, so we specify the conditions that must be met for a set of polynomials for which we know the answer - namely x^i. Therefore we can write

$$\int_a^b x^i\,dx = \frac{b^{i+1} - a^{i+1}}{i+1} = \sum_{j=0}^{n} x_j^i W_j \quad , \quad i = 0 \cdots n \quad . \tag{4.2.17}$$

The integral on the left is easily evaluated to yield the center term which must be equal to the sum on the right if the formula is to have the required degree of precision n. Equations (4.2.17) represent n+1 linear equations in the n+1 weights W_i. Since we have already discussed the solution of linear equations in some detail in chapter 2, we can consider the problem of finding the weights to be solved.

While the spacing of the points given in equations (4.2.17) is completely arbitrary, we can use these equations to determine the weights for Simpson's rule as an example. Assume that we are to evaluate an integral in the interval $0 \rightarrow 2h$. Then the equations (4.2.17) for the weights would be

$$\int_0^{2h} x^i\,dx = \frac{(2h)^{i+1}}{i+1} = \sum_{j=0}^{n} x_j^i W_j \quad , \quad i = 0 \cdots n \quad . \tag{4.2.18}$$

For $x_j = [0,h,2h]$, the equations specifically take the form

$$\left.\begin{aligned} 2h &= W_1 + W_2 + W_3 \\ \frac{(2h)^2}{2} &= 2h^2 = h^2 W_2 + h^2 W_3 \\ \frac{(2h)^3}{3} &= \frac{8h^3}{3} = h^2 W_2 + 4h^2 W_3 \end{aligned}\right\} \quad . \tag{4.2.19}$$

which upon removal of the common powers of h are

$$\left.\begin{aligned} 2h &= W_1 + W_2 + W_3 \\ 2h &= W_2 + W_3 \\ \frac{8h}{3} &= W_2 + 4W_3 \end{aligned}\right\} \quad . \tag{4.2.20}$$

These have the solution

$$W_i = [1/3,\ 4/3,\ 1/3]h \quad . \tag{4.2.21}$$

The weights given in equation (4.2.21) are identical to those found for Simpson's rule in equation (4.2.9) which lead to the approximation formula given by equation (4.2.11). The details of finding the weights by this method are sufficiently simple that it is generally preferred over the method discussed in the previous section (section 4.2b).

There are still other alternatives for determining the weights. For example, the integral in equation (4.2.16) is itself the integral of a polynomial of degree n and as such can be evaluated exactly by any quadrature scheme with that degree of precision. It need not have the spacing of the desired scheme at all. Indeed, the integral could be evaluated at a sufficient level of accuracy by using a running Simpson's rule with a sufficient total number of points. Or the weights could be obtained using the highly efficient Gaussian type quadrature schemes described below. In any event, a quadrature scheme can be tailored to fit nearly any problem by writing down the equations of condition that the weights must satisfy in order to have the desired degree of precision. There are, of course, some potential pitfalls with this approach. If very high degrees of precision formulae are sought, the equations (4.2.17) may become nearly singular and be quite difficult to solve with the accuracy required for reliable quadrature schemes. If such high degrees of precision formulae are really required, then one should consider Gaussian quadrature schemes.

d. Gaussian Quadrature Schemes

We turn now to a class of quadrature schemes first suggested by that brilliant 19th century mathematician Karl Friedrich Gauss. Gauss noted that one could obtain a much higher degree of precision for a quadrature scheme designed for a function specified at a given number of points, if the location of those points were regarded as additional free parameters. So, if in addition to the N weights one also had N locations to specify, one could obtain a formula with a degree of precision of 2N-1 for a function specified at only N points. However, they would have to be the proper N points. That is, their location would no longer be arbitrary so that the function would have to be known at a particular set of values of the independent variable x_i. Such a formula would not be considered a hyper-efficient formula since the degree of precision does not exceed the number of adjustable parameters. One has simply enlarged the number of such parameters available in a given problem.

The question then becomes how to locate the proper places for the evaluation of the function given the fact that one wishes to obtain a quadrature formula with this high degree of precision. Once more we may appeal to the notion of obtaining a quadrature formula from an interpolation formula. In section (3.2b) we developed Hermite interpolation which had a degree of precision of 2N-1. (Note: in that discussion the actual numbering if the points began with zero so that N=n+1 where n is the limit of the sums in the discussion.) Since equation (3.2.12) has the required degree of precision, we know that its integral will provide a quadrature formula of the appropriate degree. Specifically

$$\int_a^b \Phi(x)\,dx = \sum_{j=0}^{n} f(x_j)\int_a^b h_j(x)\,dx + \sum_{j=0}^{n} f'(x_j)\int_a^b H_j(x)\,dx \quad . \tag{4.2.22}$$

Now equation (4.2.22) would resemble the desired quadrature formula if the second sum on the right hand side could be made to vanish. While the weight functions $H_j(x)$ themselves will not always be zero, we can ask under what conditions their integral will be zero so that

$$\int_a^b H_j(x)\,dx = 0 \quad . \tag{4.2.23}$$

Here the secret is to remember that those weight functions are polynomials [see equation (3.2.32)] of degree 2n+1 (i.e. 2N-1) and in particular $H_j(x)$ can be written as

$$H_i(x) = \frac{\prod(x) L_i(x)}{\prod_{j \neq i}^{n} (x_i - x_j)} \quad , \tag{4.2.24}$$

where

$$\prod(x) \equiv \prod_{j=0}^{n} (x - x_j) \quad . \tag{4.2.25}$$

Here the additional multiplicative linear polynomial $u_j(x)$ that appears in equation has been included in one of the Lagrange polynomials $L_j(x)$ to produce the n+1 degree polynomial $\Pi(x)$. Therefore the condition for the weights of $f'(x_i)$ to vanish [equation(4.2.23)] becomes

$$\frac{\int_a^b \prod(x) L_i(x)\, dx}{\prod_{j \neq i}^{n} (x_i - x_j)} = 0 \quad . \tag{4.2.26}$$

The product in the denominator is simply a constant which is not zero so it may be eliminated from the equation. The remaining integral looks remarkably like the integral for the definition of orthogonal polynomials [equation (3.3.6)]. Indeed, since $L_i(x)$ is a polynomial of degree n [or (N-1)] and $\Pi(x)$ is a polynomial of degree n+1 (also N), the conditions required for equation (4.2.26) to hold will be met if $\Pi(x)$ is a member of the set of polynomials which are orthogonal in the interval $a \rightarrow b$. But we have not completely specified $\Pi(x)$ for we have not chosen the values x_j where the function f(x) and hence $\Pi(x)$ are to be evaluated. Now it is clear from the definition of $\Pi(x)$ [equation (4.2.25)] that the values of x_j are the roots of a polynomial of degree n+1 (or N) that $\Pi(x)$ represents. Thus, we now know how to choose the x_j's so that the weights of f'(x) will vanish. Simply choose them to be the roots of the (n+1)th degree polynomial which is a member on an orthogonal set on the interval $a \rightarrow b$. This will insure that the second sum in equation (4.2.22) will always vanish and the condition becomes

$$\int_a^b \Phi(x)\, dx = \sum_{j=0}^{n} f(x_j) \int_a^b h_j(x)\, dx \quad . \tag{4.2.27}$$

This expression is exact as long as $\Phi(x)$ is a polynomial of degree 2n+1 (or 2N-1) or less. Thus, Gaussian quadrature schemes have the form

$$\int_a^b f(x)\, dx = \sum_{j=0}^{n} f(x_j) W_j \quad , \tag{4.2.28}$$

where the x_i's are the roots of the Nth degree orthogonal polynomial which is orthogonal in the interval $a \rightarrow b$, and the weights W_i can be written with the aid of equation (3.2.32) as

$$W_i = \int_a^b h_i(x)\, dx = \int_a^b [1 - 2(x - x_i) L'_i(x)] L_i^2(x)\, dx \, . \tag{4.2.29}$$

Now these weights can be evaluated analytically should one have the determination, or they can be evaluated from the equations of condition [equation (4.2.17)] which any quadrature weights must satisfy. Since the extent of the finite interval can always be transformed into the interval $-1 \rightarrow +1$ where the appropriate orthonormal polynomials are the Legendre polynomials, and the weights are independent of the function f(x), they will be specified by the value of N alone and may be tabulated once and for all. Probably the most complete tables of the roots and weights for Gaussian quadrature can be found in Abramowitz and Stegun[1] and unless a particularly unusual quadrature scheme is needed these tables will suffice.

Before continuing with our discussion of Gaussian quadrature, it is perhaps worth considering a specific example of such a formula. Since the Gaussian formulae make use of orthogonal polynomials, we should first express the integral in the interval over which the polynomials form an orthogonal set. To that end, let us examine an integral with a finite range so that

$$\int_a^b f(x)\,dx = \left(\frac{b-a}{2}\right)\int_{-1}^{+1} f\{[(b-a)y + (a+b)]/2\}\,dy \quad . \tag{4.2.30}$$

Here we have transformed the integral into the interval $-1 \rightarrow +1$. The appropriate transformation can be obtained by evaluating a linear function at the respective end points of the two integrals. This will specify the slope and intercept of the straight line in terms of the limits and yields

$$\left.\begin{aligned} y &= [2x - (a+b)]/(b-a) \\ dy &= [2/(b-a)]dx \end{aligned}\right\} \quad . \tag{4.2.31}$$

We have no complicating weight function in the integrand so that the appropriate polynomials are the Legendre polynomials. For simplicity, let us take n=2. We gave the first few Legendre polynomials in Table 3.4 and for n = 2 we have

$$P_2(y) = (3y^2\text{-}1)/2 \quad . \tag{4.2.32}$$

The points at which the integrand is to be evaluated are simply the roots of that polynomial which we can fine from the quadratic formula to be

$$\left.\begin{aligned} (3y^2 - 1)/2 &= 0 \\ y_i &= \pm\sqrt{3} \end{aligned}\right\} \quad . \tag{4.2.33}$$

Quadrature formulae of larger n will require the roots of much larger degree polynomials which have been tabulated by Abramowitz and Stegun[1]. The weights of the quadrature formula are yet to be determined, but having already specified where the function is to be evaluated, we may use equations (4.2.17) to find them. Alternatively, for this simple case we need only remember that the weights sum to the interval so that

$$W_1 + W_2 = 2 \quad . \tag{4.2.34}$$

Since the weights must be symmetric in the interval, they must both be unity. Substituting the values for y_i and W_i into equation (4.2.28), we get

$$\int_a^b f(x)\,dx \cong \tfrac{(b-a)}{2}\{f[(\tfrac{(b-a)}{2\sqrt{3}}) + \tfrac{1}{2}(a+b)] + f[(\tfrac{(a-b)}{2\sqrt{3}}) + \tfrac{1}{2}(a+b)]\} \quad . \tag{4.2.35}$$

While equation (4.2.35) contains only two terms, it has a degree of precision of three (2n-1) or the same as the three term hyper-efficient Simpson's rule. This nicely illustrates the efficiency of the Gaussian schemes. They rapidly pass the fixed abscissa formulae in their degree of precision as [(2n-1)/n].

So far we have restricted our discussion of Gaussian quadrature to the finite interval. However, there is nothing in the entire discussion that would affect general integrals of the form

$$I = \int_{\alpha}^{\beta} w(x) f(x)\, dx \quad . \tag{4.2.36}$$

Here w(x) is a weight function which may not be polynomic and should not be confused with the quadrature weights W_i. Such integrals can be evaluated exactly as long as f(x) is a polynomial of degree 2N-1. One simply uses a Gaussian scheme where the points are chosen so that the values of x_i are the roots of the Nth degree polynomial that is orthogonal in the interval $\alpha \rightarrow \beta$ *relative to the weight function* w(x). We have already studied such polynomials in section 3.3 so that we may use Gaussian schemes to evaluate integrals in the semi-infinite interval $[0 \rightarrow +\infty]$ and full infinite interval $[-\infty \rightarrow +\infty]$ as well as the finite interval $[-1 \rightarrow +1]$ as long as the appropriate weight function is used. Below is a table of the intervals and weight functions that can be used for some common types of Gaussian quadrature.

Table 4.2

Types of Polynomials for Gaussian Quadrature

Interval	Weight Function w(x)	Type of Polynomial
$-1 \rightarrow +1$	$(1-x^2)^{-½}$	Chebyschev: 1st kind
$-1 \rightarrow +1$	$(1-x^2)^{+½}$	Chebyschev: 2nd kind
$0 \rightarrow +\infty$	e^{-x}	Laguerre
$-\infty \rightarrow +\infty$	e^{-x^2}	Hermite

It is worth noting from the entries in Table 4.2 that there are considerable opportunities for creativity available for the evaluation of integrals by a clever choice of the weight function. Remember that it is only f(x) of the product w(x)f(x) making up the integrand that need be well approximated by a polynomial in order for the quadrature formula to yield accurate answers. Indeed the weight function for Gaussian-Chebyschev quadrature of the first kind has singularities at the end points of the interval. Thus if one's integral has similar singularities, it would be a good idea to use Gauss-Chebyschev quadrature instead of Gauss-Legendre quadrature for evaluating the integral. Proper choice of the weight function may simply be used to improve the polynomic behavior of the remaining part of the integrand. This will certainly improve the accuracy of the solution.

In any event, the quadrature formulae can always be written to have the form

$$\int_{\alpha}^{\beta} w(x) f(x)\, dx = \sum_{j=0}^{n} f(x_j) W_j \quad , \tag{4.2.37}$$

where the weights, which may include the weight function w(x) can be found from

$$w_i = \int_\alpha^\beta w(x) h_i(x)\, dx \quad . \tag{4.2.38}$$

Here $h_i(x)$ is the appropriate orthogonal polynomial for the weight function and interval.

e. Romberg Quadrature and Richardson Extrapolation

So far we have given explicit formulae for the numerical evaluation of a definite integral. In reality, we wish the result of the application of such formulae to specific problems. Romberg quadrature produces this result without obtaining the actual form for the quadrature formula. The basic approach is to use the general properties of the equal-interval formulae such as the Trapezoid rule and Simpson's rule to generate the results for formulae successively applied with smaller and smaller step size. The results can be further improved by means of Richardson's extrapolation to yield results for formulae of greater accuracy [i.e. higher order $O(h^m)$]. Since the Romberg algorithm generates these results recursively, the application is extremely efficient, readily programmable, and allows an on-going estimate of the error. Let us define a step size that will always yield equal intervals throughout the interval $a \rightarrow b$ as

$$h_j = (b\text{-}a)/2^j \quad . \tag{4.2.39}$$

The general Trapezoid rule for an integral over this range can written as

$$F(b-a) = \int_a^b f(x)\, dx = \frac{h_j}{2}\left[f(a) + f(b) + 2\sum_{i=1}^{j-1} f(a + i h_j) \right] \quad . \tag{4.2.40}$$

The Romberg recursive quadrature algorithm states that the results of applying this formula for successive values of j (i.e. smaller and smaller step sizes h_j) can be obtained from

$$\left.\begin{aligned} F_j^0 &= \tfrac{1}{2}(F_{j-1}^0 + Q_{j-1}) \\ Q_{j-1} &= h_{j-1} \sum_{i=1}^{2^{(j-1)}} f[b + (i - \tfrac{1}{2}) h_{j-1}] \\ F_0^0 &= (b-a)[f(a) + f(b)]/2 \end{aligned}\right\} \quad . \tag{4.2.41}$$

Each estimate of the integral will require $2^{(j-1)}$ evaluations of the function and should yield a value for the integral, but can have a degree of precession no greater than $2^{(j-1)}$. Since a sequence of j steps must be execute to reach this level, the efficiency of the method is poor compared to Gaussian quadrature. However the difference $(F_j^0 - F_{j-1}^0)$ does provide an continuous estimate of the error in the integral.

We can significantly improve the efficiency of the scheme by using Romberg extrapolation to improve the nature of the quadrature formulae that the iteration scheme is using. Remember that successive values of h differ by a factor of two. This is exactly the form that we used to develop the Richardson formula for the derivative of a function [equation (4.1.15)]. Thus we can use the generalization of the Richardson algorithm given by equation (4.1.15) and utilizing two successive values of F_j^0 to "extrapolate" to the result

for a higher order formula. Each value of integral corresponding to the higher order quadrature formula can, in turn, serve as the basis for an additional extrapolation. This procedure also can be cast as a recurrence formula where

$$F_j^k = \frac{2^{2k} F_{j+1}^{k-1} - F_j^{k-1}}{2^{2k} - 1} \quad . \tag{4.2.42}$$

There is a trade off between the results generated by equation (4.2.42) and equation (4.2.41). Larger values of j produce values for F^k_j which correspond to decreasing values of h (see table 4.3). However, increasing values of k yield values for F^k_j which correspond to quadrature formulae smaller error terms, but with larger values of h. Thus it is not obvious which sequence, equation (4.2.41) or equation (4.2.42) will yield the better value for the integral.

In order to see how this method works, consider applying it to the analytic integral

$$\int_0^{+1} e^{5x}\, dx = \frac{e^5 - 1}{5} = 29.48263182 \quad . \tag{4.2.43}$$

Table 4.3

Sample Results for Romberg Quadrature

i	F_j^0	F_j^1	F_j^2	F_j^3	F_j^4
0	74.7066	33.0238	29.6049	29.4837	29.4827
1	43.4445	29.8186	29.4856	29.4826	
2	33.2251	29.5064	29.4827		
3	30.4361	29.4824			
4	29.722113				

Here it is clear that improving the order of the quadrature formula rapidly leads to a converged solution. The convergence of the non-extrapolated quadrature is not impressive considering the number of evaluations required to reach, say, F^0_4. Table 4.4 gives the results of applying some of the other quadrature methods we have developed to the integral in equation (4.2.43).

We obtain the results for the Trapezoid rule by applying equation (4.2.1) to the integral given by equation (4.2.43). The results for Simpson's rule and the two-point Gaussian quadrature come from equations (4.2.11) and (4.2.35) respectively. In the last two columns of Table 4.4 we have given the percentage error of the method and the number of evaluations of the function required for the determination of the integral. While the Romberg extrapolated integral is five times more accurate that it nearest competitor, it takes twice the number of evaluations. This situation gets rapidly worse so that the Gaussian quadrature becomes the most efficient and accurate scheme when n exceeds about five. The trapezoid rule and Romberg F_0^0 yield identical results as they are the same approximation. Similarly Romberg F^1_0 yields the same results as Simpson's rule. This is to be expected as the Richardson extrapolation of the Romberg quadrature equivalent to the Trapezoid rule should lead to the next higher order quadrature formula which is Simpson's rule.

Table 4.4

Test Results for Various Quadrature Formulae

TYPE	F(X)	|ΔF(%)|	N[F(X)]
Analytic Result	29.48263182	0.0	1
Trapezoid Rule	74.70658	153.39	2
Simpson's Rule	33.02386	12.01	3
2-point Gauss Quad.	27.23454	7.63	2
Romberg Quadrature F_0^0	74.70658	153.39	2
Romberg Quadrature F_1^1	29.8186	1.14	4

f. Multiple Integrals

Most of the work on the numerical evaluation of multiple integrals has been done in the middle of this century at the University of Wisconsin by Preston C. Hammer and his students. A reasonably complete summary of much of this work can be found in the book by Stroud[2]. Unfortunately the work is not widely known since problems associated with multiple integrals occur frequently in the sciences particularly in the area of the modeling of physical systems. From what we have already developed for quadrature schemes one can see some of the problems. For example, should it take N points to accurately represent an integral in one dimension, then it will take N^m points to calculate an m-dimensional integral. Should the integrand be difficult to calculate, the computation involved in evaluating it at N^m points can be prohibitive. Thus we shall consider only those quadrature formulae that are the most efficient - the Gaussian formulae. The first problem in numerically evaluating multiple integrals is to decide what will constitute an approximation criterion. Like integrals of one dimension, we shall appeal to polynomial approximation. That is, in some sense, we shall look for schemes that are exact for polynomials of the multiple variables that describe the multiple dimensions. However, there are many distinct types of such polynomials so we shall choose a subset. Following Stroud[2] let us look for quadrature schemes that will be exact for polynomials that can be written as simple products of polynomials of a single variable. Thus the approximating polynomial will be a product polynomial in m-dimensions. Now we will not attempt to derive the general theory for multiple Gaussian quadrature, but rather pick a specific space. Let the space be m-dimensional and of the full infinite interval. This allows us, for the moment, to avoid the problem of boundaries. Thus we can represent our integral by

$$V = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\cdots\int_{-\infty}^{+\infty} e^{-(x_1^2+x_2^2+\cdots+x_m^2)} f(x_1, x_2, \cdots, x_m)\, dx_1 dx_2 \cdots dx_m \,. \tag{4.2.44}$$

Now we have seen that we lose no generality by assuming that our nth order polynomial is a monomial of the form x^α so let us continue with this assumption that $f(x_1, x_2, \cdots x_m)$ has the form

$$f(x) = \prod_{i=1}^{n} x_i^{\alpha_i} \quad . \tag{4.2.45}$$

We can then write equation (4.2.44) as

$$V = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\cdots\int_{-\infty}^{+\infty} e^{-\sum_{i=1}^{m} x_i^2} \prod_{j=1}^{m} x_j^{\alpha_j} dx_j = \prod_{j=1}^{m}\int_{-\infty}^{+\infty} e^{x_j^2} x_j^{\alpha_j} dx_j \quad . \tag{4.2.46}$$

The right hand side has the relatively simple form due to the linearity of the integral operator. Now make a coordinate transformation to general spherical coordinates by means of

$$\left.\begin{aligned} x_1 &= r\cos\theta_{m-1}\cos\theta_{m-2}\cdots\cos\theta_2\cos\theta_1 \\ x_2 &= r\cos\theta_{m-1}\cos\theta_{m-2}\cdots\cos\theta_2\sin\theta_1 \\ &\vdots \qquad\quad \vdots \qquad\quad \vdots \\ x_{m-1} &= r\cos\theta_{m-1}\sin\theta_{m-2} \\ x_m &= r\sin\theta_{m-1} \end{aligned}\right\}, \tag{4.2.47}$$

which has a Jacobian of the transformation equal to

$$J(x_i \mid r,\theta_i) = r^{m-1}\cos^{m-2}(\theta_{m-1})\cos^{m-3}(\theta_{m-2})\cdots\cos(\theta_2) \quad . \tag{4.2.48}$$

This allows the expression of the integral to take the form

$$V = \left(\int_{-\infty}^{+\infty} e^{-r^2}|r|^{m-1} r^{(\sum_{j=1}^{m}\alpha_i)} dr\right)\prod_{i=1}^{m-1}\int_{-\pi/2}^{+\pi/2}(\cos\theta_i)^{i-1}(\cos\theta_i)^{(-\sum_{j=1}^{m}\alpha_i)}(\sin\theta_i)^{\alpha_{i+1}} d\theta_i \quad . \tag{4.2.49}$$

Consider how we could represent a quadrature scheme for any single integral in the running product. For example

$$\int_{-\pi/2}^{+\pi/2}(\cos\theta_i)^{i-1}(\cos\theta_i)^{\alpha_i}(\sin\theta_i)^{\alpha_{i+1}} d\theta_i = \sum_{j=1}^{N} B_{ij}(\cos\theta_i)^{\alpha_i} \quad . \tag{4.2.50}$$

Here we have chosen the quadrature points for θ_i to be at θ_{ij} and we have let

$$\alpha = \Sigma\alpha_i \quad . \tag{4.2.51}$$

Now make one last transformation of the form $y_i = \cos\theta_i$, (4.2.52)

which leads to

$$\int_{-1}^{+1}(1-y_i^2)^{(i-2)/2} y_i^{\alpha}\, dy = \sum_{j=1}^{N} B_{ij} y_j = \int_{-1}^{+1} w(y_i) y_i^{\alpha} dy_i \quad , \quad i = 1\cdots(m-1) \quad . \tag{4.2.53}$$

The integral on the right hand side can be evaluated exactly if we take the y_i's to be the roots of a polynomial of degree $(\alpha+1)/2$ which is a member of an orthogonal set in the interval $-1 \rightarrow +1$, relative to the weight function $w(y_i)$ which is

$$w(y_i) = (1-y_i^2)^{(i-2)/4}(1+y_i^2)^{(i-2)/4}. \tag{4.2.54}$$

By considering Table 3.1 we see that the appropriate polynomials will be members of the Jacobi

polynomials for $\alpha = \beta = (i\text{-}2)/4$. The remaining integral over the radial coordinate has the form

$$\int_{-\infty}^{+\infty} e^{-r^2} r^{\alpha'} dr \ , \tag{4.2.55}$$

which can be evaluated using Gauss-Hermite quadrature. Thus we see that multiple dimensional quadratures can be carried out with a Gaussian degree of precision for product polynomials by considering each integral separately and using the appropriate Gaussian scheme for that dimension. For example, if one desires to integrate over the solid sphere, one would choose Gauss-Hermite quadrature for the radial quadrature, Gauss-Legendre quadrature for the polar angle θ, and Gauss-Chebyschev quadrature for the azimuthal angle φ. Such a scheme can be used for integrating over the surface of spheres or surfaces that can be distorted from a sphere by a polynomial in the angular variables with good accuracy. The use of Gaussian quadrature schemes can save on the order of $N^{m/2}$ evaluations of the functions which is usually significant.

For multi-dimensional integrals, there are a number of hyper-efficient quadrature formulae that are known. However, they depend on the boundaries of the integration and are generally of rather low order. Nevertheless such schemes should be considered when the boundaries are simple and the function well behaved. When the boundaries are not simple, one may have to resort to a modeling scheme such a Monte Carlo method.

It is clear that the number of points required to evaluate an integral in m-dimensions will increase as N^m. It does not take many dimensions for this to require an enormous number of points and hence, evaluations of the integrand. Thus for multiple integrals, efficiency may dictate another approach.

4.3 Monte Carlo Integration Schemes and Other Tricks

The Monte Carlo approach to quadrature is a philosophy as much as it is an algorithm. It is an application of a much more widely used method due to John von Neumann. The method was developed during the Second World War to facilitate the solution to some problems concerning the design of the atomic bomb. The basic philosophy is to describe the problem as a sequence of causally related physical phenomena. Then by determining the probability that each separate phenomenon can occur, the joint probability that all can occur is a simple product. The procedure can be fashioned sequentially so that even probabilities that depend on prior events can be handled. One can conceptualize the entire process by following a series of randomly chosen initial states each of which initiates a causal sequence of events leading to the desired final state. The probability distribution of the final state contains the answer to the problem. While the method derives it name from the casino at Monte Carlo in order to emphasize the probabilistic nature of the method, it is most easily understood by example. One of the simplest examples of Monte Carlo modeling techniques involves the numerical evaluation of integrals.

a. Monte Carlo Evaluation of Integrals

Let us consider a one dimensional integral defined over a finite interval. The graph of the integrand might look like that in Figure 4.2. Now the area under the curve is related to the integral of the function. Therefore we can replace the problem of finding the integral of the function to that of finding the area under the curve. However, we must place some units on the integral and we do that by finding the *relative area*

under the curve. For example, consider the integral

$$\int_a^b f_{max}\,dx = (b-a)f_{max} \quad . \tag{4.3.1}$$

The graphical representation of this integral is just the area of the rectangle bounded by y = 0, x = a, x = b, and $y = f_{max}$. Now if we were to *randomly* select values of x_i and y_i, one could ask if

$$y_i \leq f(x_i) \quad . \tag{4.3.2}$$

If we let ratio of the number of successful trials to the total number of trials be R, then

$$\int_a^b f(x)dx = R(b-a)f_{max} \quad . \tag{4 .3.3}$$

Clearly the accuracy of the integral will depend on the accuracy of R and this will improve with the number N of trials. In general, the value of R will approach its actual value as N. This emphasizes the major difference between Monte Carlo quadrature and the other types of quadrature. In the case of the quadrature formulae that depend on a direct calculation of the integral, the error of the result is determined by the extent to which the integrand can be approximated by a polynomial (neglecting round-off error). If one is sufficiently determined he/she can determine the magnitude of the error term and thereby place an absolute limit on the magnitude of the error. However, Monte Carlo schemes are not based on polynomial approximation so such an absolute error estimate cannot be made even in principle. The best we can hope for is that there is a certain probability that the value of the integral lies within ε of the correct answer. Very often this is sufficient, but it should always remembered that the certainty of the calculation rests on a statistical basis and that the approximation criterion is different from that used in most areas of numerical analysis.

If the calculation of f(x) is involved, the time required to evaluate the integral may be very great indeed. This is one of the major drawbacks to the use of Monte Carlo methods in general. Another lesser problem concerns the choice of the random variables x_i and y_i. This can become a problem when very large numbers of random numbers are required. Most random number generators are subject to periodicities and other non-random behavior after a certain number of selections have been made. Any non-random behavior will destroy the probabilistic nature of the Monte Carlo scheme and thereby limit the accuracy of the answer. Thus, one may be deceived into believing the answer is better than it is. One should use Monte Carlo methods with great care. It should usually be the method of last choice. However, there are problems that can be solved by Monte Carlo methods that defy solution by any other method. This modern method of modeling the integral is reminiscent of a method used before the advent of modern computers. One simply graphed the integrand on a piece of graph paper and then cut out the area that represented the integral. By comparing the carefully measured weight of the cutout with that of a known area of graph paper, one obtained a crude estimate of the integral.

While we have discussed Monte Carlo schemes for one-dimensional integrals only, the technique can easily be generalized to multiple dimensions. Here the accuracy is basically governed by the number of points required to sample the "volume" represented by the integrand and limits. This sampling can generally be done more efficiently than the N^m points required by the direct multiple dimension quadrature schemes. Thus, the Monte-Carlo scheme is likely to efficiently compete with those schemes as the number of dimensions increases. Indeed, should m > 2, this is likely to be the case.

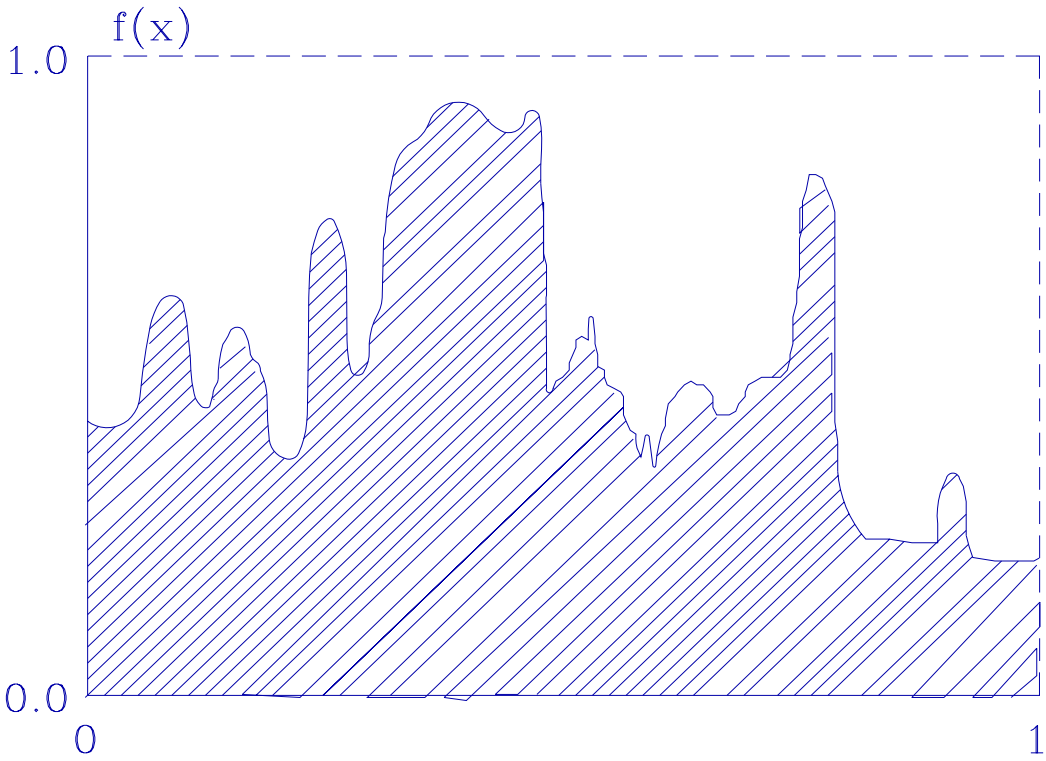


Figure 4.2 shows the variation of a particularly complicated integrand. Clearly it is not a polynomial and so could not be evaluated easily using standard quadrature formulae. However, we may use Monte Carlo methods to determine the ratio area under the curve compared to the area of the rectangle.

One should not be left with the impression that other quadrature formulae are without their problems. We cannot leave this subject without describing some methods that can be employed to improve the accuracy of the numerical evaluation of integrals.

b. The General Application of Quadrature Formulae to Integrals

Additional tricks that can be employed to produce more accurate answers involve the proper choice of the interval. Occasionally the integrand will display pathological behavior at some point in the interval. It is generally a good idea to break the interval at that point and represent the integral by two (or more) separate integrals each of which may separately be well represented by a polynomial. This is particularly useful in dealing with integrals on the semi-infinite interval, which have pathological integrands in the vicinity of zero. One can separate such an integral into two parts so that

$$\int_0^{+\infty} f(x)\,dx = \int_0^{a} f(x)\,dx + \int_a^{+\infty} f(x)\,dx \ . \qquad (4.3.4)$$

The first of these can be transformed into the interval -1$\rightarrow$ +1 and evaluated by means of any combination of the finite interval quadrature schemes shown in table 4.2. The second of these integrals can be transformed back into the semi-infinite interval by means of the linear transformation

$$y = x - a \ , \tag{4.3.5}$$

so that

$$\int_a^{+\infty} f(x)\,dx = \int_0^{+\infty} e^{-y}[e^{+y} f(y+a)\,dy \ . \tag{4.3.6}$$

Gauss-Laguerre quadrature can be used to determine the value of the second integral. By judiciously choosing places to break an integral that correspond to locations where the integrand is not well approximated by a polynomial, one can significantly increase the accuracy and ease with which integrals may be evaluated.

Having decided on the range over which to evaluate the integral, one has to pick the order of the quadrature formula to be used. Unlike the case for numerical differentiation, the higher the degree of precision of the quadrature formula, the better. However, there does come a point where the round-off error involved in the computation of the integrand exceeds the incremental improvement from the increased degree of precision. This point is usually difficult to determine. However, if one evaluates an integral with formulae of increasing degree of precision, the value of the integral will steadily change, reach a plateau, and then change slowly reflecting the influence of round-off error. As a rule of thumb 8 to 10 point Gauss-Legendre quadrature is sufficient to evaluate any integral over a finite range. If this is not the case, then the integral is somewhat pathological and other approaches should be considered. In some instances, one may use very high order quadrature (roots and weights for Legendre polynomials can be found up to N = 212), but these instances are rare. There are many other quadrature formulae that have utility in specific circumstances. However, should the quadrature present special problems, or require highly efficient evaluation, these formulae should be considered.

Chapter 4 Exercises

1. Numerically differentiate the function

   $f(x) = e^{-x}$,

   at the points x = 0, .5, 1, 5, 10. Describe the numerical method you used and why you chose it. Discuss the accuracy by comparing your results with the analytic closed form derivatives.

2. Numerically evaluate

   $f = \int_0^1 e^{-x}\ dx$.

   Carry out this evaluation using
   a. 5-point Gaussian quadrature
   b. a 5-point equal interval formula that you choose
   c. 5 point trapezoid rule
   d. analytically.

   Compare and discuss your results.

3. Repeat the analysis of problem 2 for the integral

   $\int_{-1}^{+1} |x|\ dx$.

   Comment on your results

4. What method would you use to evaluate

   $\int_1^{+\infty} (x^{-4} + 3x^{-2})\ \mathrm{Tanh}(x)\ dx$?

   Explain your choice.

5. Use the techniques described in section (4.2e) to find the volume of a sphere. Discuss all the choices you make regarding the type of quadrature use and the accuracy of the result.

Chapter 4 References and Supplemental Reading

1. Abramowitz, M. and Stegun, I.A., "Handbook of Mathematical Functions" National Bureau of Standards Applied Mathematics Series 55 (1964) U.S. Government Printing Office, Washington D.C.

2. Stroud, A.H., "Approximate Calculation of Multiple Integrals", (1971), Prentice-Hall Inc. Englewood Cliffs.

Because to the numerical instabilities encountered with most approaches to numerical differentiation, there is not a great deal of accessible literature beyond the introductory level that is available. For example

3. Abramowitz, M. and Stegun, I.A., "Handbook of Mathematical Functions" National Bureau of Standards Applied Mathematics Series 55 (1964) U.S. Government Printing Office, Washington D.C., p. 877, devote less than a page to the subject quoting a variety of difference formulae.

The situation with regard to quadrature is not much better. Most of the results are in technical papers in various journals related to computation. However, there are three books in English on the subject:

4. Davis, P.J., and Rabinowitz,P., "Numerical Integration", Blaisdell,

5. Krylov, V.I., "Approximate Calculation of Integrals" (1962) (trans. A.H.Stroud), The Macmillian Company

6. Stroud, A.H., and Secrest, D. "Gaussian Quadrature Formulas", (1966), Prentice-Hall Inc., Englewood Cliffs.

Unfortunately they are all out of print and are to be found only in the better libraries. A very good summary of various quadrature schemes can be found in

7. Abramowitz, M. and Stegun, I.A., "Handbook of Mathematical Functions" National Bureau of Standards Applied Mathematics Series 55 (1964) U.S. Government Printing Office, Washington D.C., pp. 885-899.

This is also probably the reference for the most complete set of Gaussian quadrature tables for the roots and weights with the possible exception of the reference by Stroud and Secrest (i.e. ref 4). They also give some hyper-efficient formulae for multiple integrals with regular boundaries. The book by Art Stroud on the evaluation of multiple integrals

6. Stroud, A.H., "Approximate Calculation of Multiple Integrals", (1971), Prentice-Hall Inc., Englewood Cliffs.

represents largely the present state of work on multiple integrals , but it is also difficult to find.

5

Numerical Solution of Differential and Integral Equations

• • •

The aspect of the calculus of Newton and Leibnitz that allowed the mathematical description of the physical world is the ability to incorporate derivatives and integrals into equations that relate various properties of the world to one another. Thus, much of the theory that describes the world in which we live is contained in what are known as differential and integral equations. Such equations appear not only in the physical sciences, but in biology, sociology, and all scientific disciplines that attempt to understand the world in which we live. Innumerable books and entire courses of study are devoted to the study of the solution of such equations and most college majors in science and engineering require at least one such course of their students. These courses generally cover the analytic closed form solution of such equations. But many of the equations that govern the physical world have no solution in closed form. Therefore, to find the answer to questions about the world in which we live, we must resort to solving these equations numerically. Again, the literature on this subject is voluminous, so we can only hope to provide a brief introduction to some of the basic methods widely employed in finding these solutions. Also, the subject is by no means closed so the student should be on the lookout for new techniques that prove increasingly efficient and accurate.

5.1 The Numerical Integration of Differential Equations

When we speak of a differential equation, we simply mean any equation where the dependent variable appears as well as one or more of its derivatives. The highest derivative that is present determines the *order* of the differential equation while the highest power of the dependent variable or its derivative appearing in the equation sets its *degree*. Theories which employ differential equations usually will not be limited to single equations, but may include sets of simultaneous equations representing the phenomena they describe. Thus, we must say something about the solutions of sets of such equations. Indeed, changing a high order differential equation into a system of first order differential equations is a standard approach to finding the solution to such equations. Basically, one simply replaces the higher order terms with new variables and includes the equations that define the new variables to form a set of first order simultaneous differential equations that replace the original equation. Thus a third order differential equation that had the form

$$f'''(x) + \alpha f''(x) + \beta f'(x) + \gamma f(x) = g(x) \quad , \tag{5.1.1}$$

could be replaced with a system of first order differential equations that looked like

$$\left.\begin{aligned} y'(x) + \alpha z'(x) + \beta f'(x) + \gamma f(x) &= g(x) \\ z'(x) &= y(x) \\ f'(x) &= z(x) \end{aligned}\right\} \quad . \tag{5.1.2}$$

This simplification means that we can limit our discussion to the solution of sets of first order differential equations with no loss of generality.

One remembers from beginning calculus that the derivative of a constant is zero. This means that it is always possible to add a constant to the general solution of a first order differential equation unless some additional constraint is imposed on the problem. These are generally called the *constants of integration*. These constants will be present even if the equations are inhomogeneous and in this respect differential equations differ significantly from functional algebraic equations. Thus, for a problem involving differential equations to be fully specified, the constants corresponding to the derivative present must be given in advance. The nature of the constants (i.e. the fact that their derivatives are zero) implies that there is some value of the independent variable for which the dependent variable has the value of the constant. Thus, constants of integration not only have a value, but they have a "place" where the solution has that value. If all the constants of integration are specified at the same place, they are called *initial values* and the problem of finding a solution is called an *initial value problem*. In addition, to find a numerical solution, the range of the independent variable for which the solution is desired must also be specified. This range must contain the initial value of the independent variable (i.e. that value of the independent variable corresponding to the location where the constants of integration are specified). On occasion, the constants of integration are specified at different locations. Such problems are known as boundary value problems and, as we shall see, these require a special approach. So let us begin our discussion of the numerical solution of ordinary differential equations by considering the solution of first order initial value differential equations.

The general approach to finding a solution to a differential equation (or a set of differential equations) is to begin the solution at the value of the independent variable for which the solution is equal to the initial values. One then proceeds in a step by step manner to change the independent variable and move across the required range. Most methods for doing this rely on the local polynomial approximation of the

solution and all the stability problems that were a concern for interpolation will be a concern for the numerical solution of differential equations. However, unlike interpolation, we are not limited in our choice of the values of the independent variable to where we can evaluate the dependent variable and its derivatives. Thus, the spacing between solution points will be a free parameter. We shall use this variable to control the process of finding the solution and estimating this error.

Since the solution is to be locally approximated by a polynomial, we will have constrained the solution and the values of the coefficients of the approximating polynomial. This would seem to imply that before we can take a new step in finding the solution, we must have prior information about the solution in order to provide those constraints. This "chicken or egg" aspect to solving differential equations would be removed if we could find a method that only depended on the solution at the previous step. Then we could start with the initial value(s) and generate the solution at as many additional values of the independent variable as we needed. Therefore let us begin by considering one-step methods.

a. One Step Methods of the Numerical Solution of Differential Equations

Probably the most conceptually simple method of numerically integrating differential equations is *Picard's method*. Consider the first order differential equation

$$y'(x) = g(x,y) \ . \tag{5.1.3}$$

Let us directly integrate this over the small but finite range h so that

$$\int_{y_0}^{y} dy = \int_{x_0}^{x_0+h} g(x,y)\,dx \ , \tag{5.1.4}$$

which becomes

$$y(x) = y_0 + \int_{x_0}^{x_0+h} g(x,y)\,dx \ , \tag{5.1.5}$$

Now to evaluate the integral and obtain the solution, one must know the answer to evaluate g(x,y). This can be done iteratively by turning eq (5.1.5) into a fixed-point iteration formula so that

$$\left.\begin{aligned} y^{(k)}(x_0+h) &= y_0 + \int_{x_0}^{x_0+h} g[x, y^{(k-1)}(x)]\,dx \\ y^{(k-1)}(x) &= y^{(k-1)}(x_0+h) \end{aligned}\right\} \ . \tag{5.1.6}$$

A more inspired choice of the iterative value for $y^{(k-1)}(x)$ might be

$$y^{(k-1)}(x) = \tfrac{1}{2}[y_0 + y^{(k-1)}(x_0+h)] \ . \tag{5.1.7}$$

However, an even better approach would be to admit that the best polynomial fit to the solution that can be achieved for two points is a straight line, which can be written as

$$y(x) = y_0 + a(x-x_0) = \{[y^{(k-1)}(x_0+h)](x-x_0) + [y_0(x0\ )](x_0+h-x)\}/h \ . \tag{5.1.8}$$

While the right hand side of equation (5.1.8) can be used as the basis for a fixed point iteration scheme, the iteration process can be completely avoided by taking advantage of the functional form of g(x,y). The linear

form of y can be substituted directly into g(x,y) to find the best value of a. The equation that constrains a is then simply

$$ah = \int_{x_0}^{x_0+h} g[x,(ax + y_0)]\,dx \quad . \tag{5.1.9}$$

This value of a may then be substituted directly into the center term of equation (5.1.8) which in turn is evaluated at $x = x_0+h$. Even should it be impossible to evaluate the right hand side of equation (5.1.9) in closed form any of the quadrature formulae of chapter 4 can be used to directly obtain a value for a. However, one should use a formula with a degree of precision consistent with the linear approximation of y.

To see how these various forms of Picard's method actually work, consider the differential equation

$$y'(x) = xy \quad , \tag{5.1.10}$$

subject to the initial conditions

$$y(0) = 1 \quad . \tag{5.1.11}$$

Direct integration yields the closed form solution

$$y = e^{x^2/2} \quad . \tag{5.1.12}$$

The rapidly varying nature of this solution will provide a formidable test of any integration scheme particularly if the step size is large. But this is exactly what we want if we are to test the relative accuracy of different methods.

In general, we can cast Picard's method as

$$y(x) = 1 + \int_0^z zy(z)\,dz \quad , \tag{5.1.13}$$

where equations (5.1.6) - (5.1.8) represent various methods of specifying the behavior of y(z) for purposes of evaluating the integrand. For purposes of demonstration, let us choose h = 1 which we know is unreasonably large. However, such a large choice will serve to demonstrate the relative accuracy of our various choices quite clearly. Further, let us obtain the solution at x = 1, and 2. The naive choice of equation (5.1.6) yields an iteration formula of the form

$$y(x_0 + h) = 1 + \int_{x_0}^{x_0+h} zy^{(k-1)}(x_0 + h)\,dz + 1 + [h(x_0 + h)/2]\,y^{(k-1)}(x_0 + h) \quad . \tag{5.1.14}$$

This may be iterated directly to yield the results in column (a) of table 5.1, but the fixed point can be found directly by simply solving equation (5.1.14) for $y^{(\infty)}(x_0+h)$ to get

$$y^{(\infty)}(x_0+h) = (1-hx_0-h^2/2)^{-1} \quad . \tag{5.1.15}$$

For the first step when $x_0 = 0$, the limiting value for the solution is 2. However, as the solution proceeds, the iteration scheme clearly becomes unstable.

Table 5.1

Results for Picard's Method

	(a)	(b)	(c)	(d)
i	**y(1)**	**y(1)**	**y(1)**	**y_c(1)**
0	1.0	1.0		
1	1.5	1.5		
2	1.75	1.625		
3	1.875	1.6563		
4	1.938	1.6641		
5	1.969	1.6660		
∞	2.000	5/3	7/4	1.6487
i	**y(2)**	**y(2)**	**y(2)**	**y_c(2)**
0	4.0	1.6666		
1	7.0	3.0000		
2	11.5	4.5000		
3	18.25	5.6250		
4	28.375	6.4688		
5	43.56	7.1015		
∞	∞	9.0000	17.5	7.3891

Estimating the appropriate value of y(x) by averaging the values at the limits of the integral as indicated by equation (5.1.7) tends to stabilize the procedure yielding the iteration formula

$$y^{(k)}(x_0+h) = 1+\tfrac{1}{2}\int_{x_0}^{x_0+h} z[y(x_0)+y^{(k-1)}(x_0+h)\,dz = 1+[h(x_0+h)/2][y(x_0)+y^{(k-1)}(x_0+h)]/2, \tag{5.1.16}$$

the application of which is contained in column (b) of Table 5.1. The limiting value of this iteration formula can also be found analytically to be

$$y^{(\infty)}(x_0+h) = \frac{1 + [h(x_0+h/2)y(x_0)]/2}{[1 - h(x_0+h/2)/2]}, \tag{5.1.17}$$

which clearly demonstrates the stabilizing influence of the averaging process for this rapidly increasing solution.

Finally, we can investigate the impact of a linear approximation for y(x) as given by equation (5.1.8). Let us assume that the solution behaves linearly as suggested by the center term of equation (5.1.8). This can be substituted directly into the explicit form for the solution given by equation (5.1.13) and the value for the slope, a, obtained as in equation (5.1.9). This process yields

$$a = y(x_0)(x_0+h/2)/[1-(x_0h/2)-(h^2/3)] \ , \qquad (5.1.18)$$

which with the linear form for the solution gives the solution without iteration. The results are listed in table 5.1 in column (c). It is tempting to think that a combination of the right hand side of equation (5.1.7) integrated in closed form in equation (5.1.13) would give a more exact answer than that obtained with the help of equation (5.1.18), but such is not the case. An iteration formula developed in such a manner can be iterated analytically as was done with equations (5.1.15) and (5.1.17) to yield exactly the results in column (c) of table 5.1. Thus the best one can hope for with a linear Picard's method is given by equation (5.1.8) with the slope, a, specified by equation (5.1.9).

However, there is another approach to finding one-step methods. The differential equation (5.1.3) has a full family of solutions depending on the initial value (i.e. the solution at the beginning of the step). That family of solutions is restricted by the nature of g(x,y). The behavior of that family in the neighborhood of $x=x_0+h$ can shed some light on the nature of the solution at $x = x_0+h$. This is the fundamental basis for one of the more successful and widely used one-step methods known as the *Runge-Kutta method*. The Runge-Kutta method is also one of the few methods in numerical analysis that does not rely directly on polynomial approximation for, while it is certainly correct for polynomials, the basic method assumes that the solution can be represented by a Taylor series.

So let us begin our discussion of Runge-Kutta formulae by assuming that the solution can be represented by a finite taylor series of the form

$$y_{n+1} = y_n + hy'_n + (h^2/2!)y''_n + \cdots + (h^k/k!)y_n^{(k)} \ . \qquad (5.1.19)$$

Now assume that the solution can also be represented by a function of the form

$$y_{n+1} = y_n + h\{\alpha_0 g(x_n,y_n)+\alpha_1 g[(x_n+\mu_1 h),(y_n+b_1 h)] +\alpha_2 g[(x_n+\mu_2 h),(y_n+b_2 h)]+ \cdots +\alpha_k g[(x_n+\mu_k h),(y_n+b_k h)]\} \ . \qquad (5.1.20)$$

This rather convoluted expression, while appearing to depend only on the value of y at the initial step (i.e. y_n) involves evaluating the function g(x,y) all about the solution point x_n, y_n (see Figure 5.1).

By setting equations (5.1.19) and (5.1.20) equal to each other, we see that we can write the solution in the from

$$y_{n+1} = y_n + \alpha_0 t_0 + \alpha_1 t_1 + \cdots + \alpha_k t_k \ , \qquad (5.1.21)$$

where the t_is can be expressed recursively by

$$\left.\begin{aligned} t_0 &= hg(x_n, y_n) \\ t_1 &= hg[(x_n + \mu_1 h),(y_n + \lambda_{1,0} t_0)] \\ t_2 &= hg[(x_n + \mu_2 h),(y_n + \lambda_{2,0} t_0 + \lambda_{2,1} t_1)] \\ &\vdots \qquad\qquad\qquad \vdots \\ t_k &= hg[(x_n + \mu_k h),(y_n + \lambda_{k,0} t_0 + \lambda_{k,1} t_1 + \cdots + \lambda_{k,k-1} t_{k-1})] \end{aligned}\right\} \ . \qquad (5.1.22)$$

Now we must determine k+1 values of α, k values of μ and k×(k+1)/2 values of $\lambda_{i,j}$. But we only have k+1 terms of the Taylor series to act as constraints. Thus, the problem is hopelessly under-determined. Thus indeterminency will give rise to entire families of Runge-Kutta formulae for any order k. In addition, the algebra to eliminate as many of the unknowns as possible is quite formidable and not unique due to the undetermined nature of the problem. Thus we will content ourselves with dealing only with low order formulae which demonstrate the basic approach and nature of the problem. Let us consider the lowest order that provides some insight into the general aspects of the Runge-Kutta method. That is k=1. With k=1 equations (5.1.21) and (5.1.22) become

$$\left.\begin{aligned} y_{n+1} &= y_n + \alpha_0 t_0 + \alpha_1 t_1 \\ t_0 &= hg(x_n y_n) \\ t_1 &= hg[(x_n + \mu h), (y_n + \lambda t_0)] \end{aligned}\right\} . \qquad (5.1.23)$$

Here we have dropped the subscript on λ as there will only be one of them. However, there are still four free parameters and we really only have three equations of constraint.

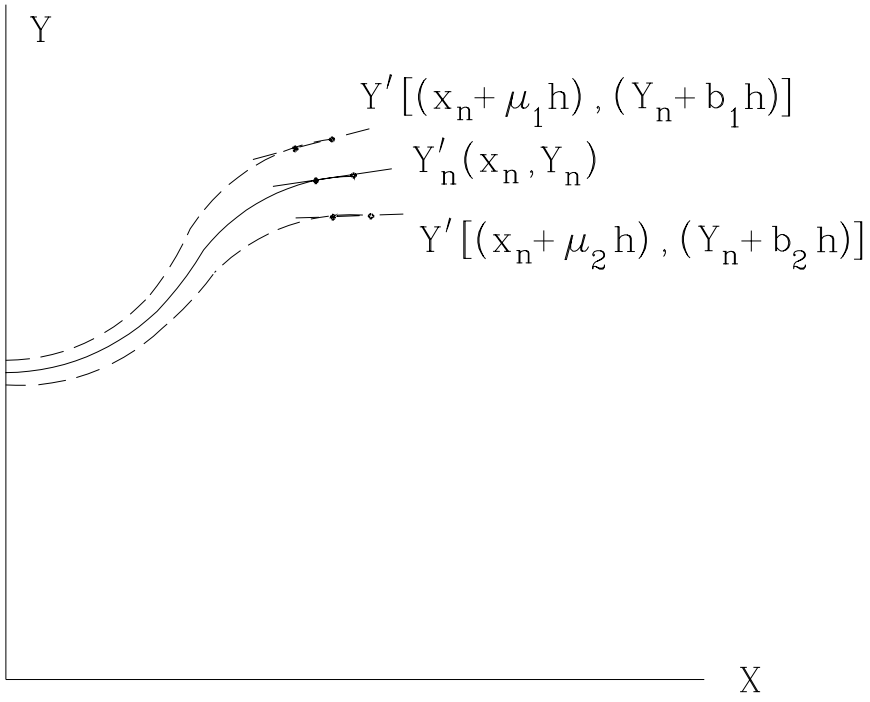


Figure 5.1 show the solution space for the differential equation y' = g(x,y). *Since the initial value is different for different solutions, the space surrounding the solution of choice can be viewed as being full of alternate solutions. The two dimensional Taylor expansion of the Runge-Kutta method explores this solution space to obtain a higher order value for the specific solution in just one step.*

If we expand g(x,y) about x_n, y_n, in a two dimensional taylor series, we can write

$$g[(x_n+\mu h),(y_n+\lambda t_0)] = g(x_n,y_n) + \mu h\frac{\partial g(x_n,y_n)}{\partial x} + \lambda t_0\frac{\partial g(x_n,y_n)}{\partial y} + \tfrac{1}{2}\mu^2 h^2\frac{\partial g(x_n,y_n)}{\partial x^2}$$
$$+ \tfrac{1}{2}\lambda^2 t_0^2\frac{\partial^2 g(x_n,y_n)}{\partial y^2} + \mu\lambda t_0\frac{\partial^2 g(x_n,y_n)}{\partial x\partial y} + \cdots + \quad . \qquad (5.1.24)$$

Making use of the third of equations (5.1.23), we can explicitly write t_1 as

$$t_1 = hg(x_n,y_n) + h^2\left[\mu\frac{\partial g(x_n,y_n)}{\partial x} + \lambda g(x_n,y_n)\frac{\partial g(x_n,y_n)}{\partial y}\right]$$
$$\tfrac{1}{2}h^3\left[\mu^2\frac{\partial^2 g(x_n,y_n)}{\partial x^2} + \lambda^2 g^2(x_n,y_n)\frac{\partial^2 g(x_n,y_n)}{\partial y^2} + 2\mu\lambda g(x_n,y_n)\frac{\partial^2 g(x_n,y_n)}{\partial x\partial y}\right] \quad . \qquad (5.1.25)$$

Direct substitution into the first of equations (5.1.23) gives

$$y_{n+1} = y_n + h(\alpha_0+\alpha_1)g(x_n,y_n) + h^2\left[\mu\frac{\partial g(x_n,y_n)}{\partial x} + \lambda g(x_n,y_n)\frac{\partial g(x_n,y_n)}{\partial y}\right]$$
$$\tfrac{1}{2}h^3\alpha_1\left[\mu^2\frac{\partial^2 g(x_n,y_n)}{\partial x^2} + \lambda^2 g^2(x_n,y_n)\frac{\partial^2 g(x_n,y_n)}{\partial y^2} + 2\mu\lambda g(x_n,y_n)\frac{\partial^2 g(x_n,y_n)}{\partial x\partial y}\right] \quad . \qquad (5.1.26)$$

We can also expand y' in a two dimensional taylor series making use of the original differential equation (5.1.3) to get

$$\left.\begin{aligned} y' &= g(x,y) \\ y'' &= \frac{\partial g(x,y)}{\partial x} + y'\frac{\partial g(x,y)}{\partial y} = \frac{\partial g(x,y)}{\partial x} + g(x,y)\frac{\partial g(x,y)}{\partial y} \\ y''' &= \frac{\partial y''}{\partial x} + y'\frac{\partial y''}{\partial y} = \frac{\partial^2 g(x,y)}{\partial x^2} + \frac{\partial g(x,y)}{\partial x}\bullet\frac{\partial g(x,y)}{\partial y} + g(x,y)\frac{\partial^2 g(x,y)}{\partial x\partial y} \\ &\quad + g(x,y)\frac{\partial^2 g(x,y)}{\partial y\partial x} + g(x,y)\left[\frac{\partial g(x,y)}{\partial y}\right]^2 + g(x,y)\frac{\partial^2 g(x,y)}{\partial y^2} \end{aligned}\right\} \quad . \qquad (5.1.27)$$

Substituting this into the standard form of the Taylor series as given by equation (5.1.19) yields

$$\left.\begin{aligned} y_{n+1} = y_n + hg(x,y) + h^2\left[\frac{\partial g(x,y)}{\partial x} + \lambda g(x,y)\frac{\partial g(x,y)}{\partial y}\right] + \frac{h^3}{6}\left(\frac{\partial^2 g(x,y)}{\partial x^2} + g^2(x,y)\frac{\partial^2 g(x,y)}{\partial y^2}\right. \\ \left. + 2g(x,y)\frac{\partial^2 g(x,y)}{\partial x\partial y} + \frac{\partial g(x,y)}{\partial y}\left[\frac{\partial g(x,y)}{\partial x} + g(x,y)\frac{\partial g(x,y)}{\partial y}\right]\right) \end{aligned}\right\} \quad . \qquad (5.1.28)$$

Now by comparing this term by term with the expansion shown in equation (5.1.26) we can conclude that the free parameters α_0, α_1, μ, and λ must be constrained by

$$\left.\begin{aligned}(\alpha_0+\alpha_1)&=1\\ \alpha_1\mu&=\tfrac{1}{2}\\ \alpha_1\lambda&=\tfrac{1}{2}\end{aligned}\right\}. \tag{5.1.29}$$

As we suggested earlier, the formula is under-determined by one constraint. However, we may use the constraint equations as represented by equation (5.1.29) to express the free parameters in terms of a single constant c. Thus the parameters are

$$\left.\begin{aligned}\alpha_0&=1-c\\ \alpha_1&=c\\ \mu=\lambda&=\tfrac{1}{2}c\end{aligned}\right\}. \tag{5.1.30}$$

and the approximation formula becomes

$$\left.\begin{aligned}y_{n+1}=y_n+hg(x,y)+h^2\left[\frac{\partial g(x,y)}{\partial x}+\lambda g(x,y)\frac{\partial g(x,y)}{\partial y}\right]+\frac{h^3}{8c}\left[\frac{\partial^2 g(x,y)}{\partial x^2}+g^2(x,y)\frac{\partial^2 g(x,y)}{\partial y^2}\right.\\ \left.+2g(x,y)\frac{\partial^2 g(x,y)}{\partial x\partial y}\right]\end{aligned}\right\}. \tag{5.1.31}$$

We can match the first two terms of the Taylor series with any choice of c. The error term will than be of order $O(h^3)$ and specifically has the form

$$R_{n+1}=-\frac{h^3}{24c}\left([3-4c]y_n'''-3\frac{\partial g(x_n,y_n)}{\partial y}y_n''\right). \tag{5.1.32}$$

Clearly the most effective choice of c will depend on the solution so that there is no general "best" choice. However, a number of authors recommend c = ½ as a general purpose value.

If we increase the number of terms in the series, the under-determination of the constants gets rapidly worse. More and more parameters must be chosen arbitrarily. When these formulae are given, the arbitrariness has often been removed by *fiat*. Thus one may find various Runge-Kutta formulae of the same order. For example, a common such fourth order formula is

$$\left.\begin{aligned}y_{n+1}&=y_n+(t_0+2t_1+2t_2+t_3)/6\\ t_0&=hg(x_n,y_n)\\ t_1&=hg[(x_n+\tfrac{1}{2}h),(y_n+\tfrac{1}{2}t_0)]\\ t_2&=hg[(x_n+\tfrac{1}{2}h),(y_n+\tfrac{1}{2}t_1)]\\ t_3&=hg[(x_n+h),(y_n+t_2)]\end{aligned}\right\}. \tag{5.1.33}$$

Here the "best" choice for the under-determined parameters has already been made largely on the basis of experience.

If we apply these formulae to our test differential equation (5.1.10), we need first specify which Runge-Kutta formula we plan to use. Let us try the second order (i.e. exact for quadratic polynomials) formula given by equation (5.1.23) with the choice of constants given by equation (5.1.29) when c = ½. The

formula then becomes

$$\left.\begin{aligned} y_{n+1} &= y_n + \tfrac{1}{2}t_0 + \tfrac{1}{2}t_1 \\ t_0 &= hg(x_n, y_n) \\ t_1 &= hg[(x_n + h), (y_n + t_0)] \end{aligned}\right\} . \qquad (5.1.34)$$

So that we may readily compare to the first order Picard formula, we will take h = 1 and y(0) = 1. Then taking g(x,y) from equation (5.1.10) we get for the first step that

$$\left.\begin{aligned} t_0 &= hx_0y_0 = (1)(0)(1) = 0 \\ t_1 &= h(x_0 + h)(y_0 + t_0) = (1)(0+1)(1+0) = 1 \\ y(x_0 + h) &= y_1 = (1) + (\tfrac{1}{2})(0) + (\tfrac{1}{2})(1) = \tfrac{3}{2} \end{aligned}\right\} . \qquad (5.1.35)$$

The second step yields

$$\left.\begin{aligned} t_0 &= hx_1y_1 = (1)(1)(\tfrac{3}{2}) = \tfrac{3}{2} \\ t_1 &= h(x_1 + h)(y_1 + t_0) = (1)(1+1)(1+\tfrac{3}{2}) = 5 \\ y(x_1 + h) &= y_2 = (\tfrac{3}{2}) + (\tfrac{1}{2})(\tfrac{3}{2}) + (\tfrac{1}{2})(5) = \tfrac{19}{4} \end{aligned}\right\} . \qquad (5.1.36)$$

Table 5.2
Sample Runge-Kutta Solutions

	Second Order Solution			**Fourth Order Solution**
		Step 1		
	h=1	h=1/2	y_c	h=1
i	t_i	t_i		t_i
0	0.0	[0 , 9/32]		0.00000
1	1.0	[1/4 , 45/64]		0.50000
2	----------	----------		0.62500
3	----------	----------		1.62500
y_1	1.5	1.6172	1.64587	1.65583
δy_1	0.1172			
h'_1	0.8532*			
		Step 2		
i	t_i	t_i		t_i
0	1.5	[0.8086 , 2.1984]		1.64583
1	5.0	[1.8193 , 5.1296]		3.70313
2	----------	----------		5.24609
3	----------	----------		13.78384
y_2	4.75	6.5951	7.38906	7.20051
δy_2	1.8451			
h'_2	0.0635			

* This value assumes that $\delta y_0 = 0.1$

The Runge-Kutta formula tends to under-estimate the solution in a systematic fashion. If we reduce the step size to h = ½ the agreement is much better as the error term in this formula is of $O(h^3)$. The results for h = ½ are given in table 5.2 along with the results for h = 1. In addition we have tabulated the results for the fourth order formula given by equation (5.1.33) For our example, the first step would require that equation (5.1.33) take the form

$$\left.\begin{aligned} t_0 &= hx_0y_0 = (1)(0)(1) = 0 \\ t_1 &= h(x_0 + \tfrac{1}{2}h)(y_0 + \tfrac{1}{2}t_0) = (1)(0 + ½)(1 + 0) = \tfrac{1}{2} \\ t_2 &= h(x_0 + \tfrac{1}{2}h)(y_0 + \tfrac{1}{2}t_1) = (1)(0 + ½)[1 + (½)(½)] = \tfrac{5}{8} \\ t_3 &= h(x_0 + \tfrac{1}{2}h)(y_0 + \tfrac{1}{2}t_2) = (1)(0 + 1)[1 + (½)(⅝)] = \tfrac{13}{8} \\ y(x_0 + h) &= y_1 = (1) + [(0) + 2(½) + 2(⅝) + (\tfrac{13}{8})]/6) = \tfrac{79}{48} \end{aligned}\right\} . \qquad (5.1.37)$$

The error term for this formula is of $O(h^5)$ so we would expect it to be superior to the second order formula for h = ½ and indeed it is. These results demonstrate that usually it is preferable to increase the accuracy of a solution by increasing the accuracy of the integration formula rather than decreasing the step size. The calculations leading to Table 5.2 were largely carried out using fractional arithmetic so as to eliminate the round-off error. The effects of round-off error are usually such that they are more serious for a diminished step size than for an integration formula yielding suitably increased accuracy to match the decreased step size. This simply accentuates the necessity to improve solution accuracy by improving the approximation accuracy of the integration formula.

The Runge-Kutta type schemes enjoy great popularity as their application is quite straight forward and they tend to be quite stable. Their greatest appeal comes from the fact that they are one-step methods. Only the information about the function at the previous step is necessary to predict the solution at the next step. Thus they are extremely useful in initiating a solution starting with the initial value at the boundary of the range. The greatest drawback of the methods is their relative efficiency. For example, the forth order scheme requires four evaluations of the function at each step. We shall see that there are other methods that require far fewer evaluations of the function at each step and yet have a higher order.

b. Error Estimate and Step Size Control

A numerical solution to a differential equation is of little use if there is no estimate of its accuracy. However, as is clear from equation (5.1.32), the formal estimate of the truncation error is often more difficult than finding the solution. Unfortunately, the truncation error for most problems involving differential equations tends to mimic the solution. That is, should the solution be monotonically increasing, then the absolute truncation error will also increase. Even monotonically decreasing solutions will tend to have truncation errors that keep the same sign and accumulate as the solution progresses. The common effect of truncation errors on oscillatory solutions is to introduce a "phase shift" in the solution. Since the effect of truncation error tends to be systematic, there must be some method for estimating its magnitude.

Although the formal expression of the truncation error [say equation (5.1.32)] is usually rather formidable, such expressions always depend on the step size. Thus we may use the step size h itself to estimate the magnitude of the error. We can then use this estimate and an *a priori* value of the largest acceptable error to adjust the step size. Virtually all general algorithms for the solution of differential equations contain a section for the estimate of the truncation error and the subsequent adjustment of the step size h so that predetermined tolerances can be met. Unfortunately, these methods of error estimate will rely on the variation of the step size at each step. This will generally triple the amount of time required to effect the solution. However, the increase in time spent making a single step may be offset by being able to use much larger steps resulting in an over all savings in time. The general accuracy cannot be arbitrarily increased by decreasing the step size. While this will reduce the truncation error, it will increase the effects of round-off error due to the increased amount of calculation required to cover the same range. Thus one does not want to set the *a priori* error tolerance to low or the round-off error may destroy the validity of the solution. Ideally, then, we would like our solution to proceed with rather large step sizes (i.e. values of h) when the solution is slowly varying and automatically decrease the step size when the solution begins to change rapidly. With this in mind, let us see how we may control the step size from tolerances set on the truncation error.

Given either the one step methods discussed above or the multi-step methods that follow, assume that we have determined the solution y_n at some point x_n. We are about to take the next step in the solution to x_{n+1} by an amount h and wish to estimate the truncation error in y_{n+1}. Calculate this value of the solution two ways. First, arriving at x_{n+1} by taking a single step h, then repeat the calculation taking two steps of (h/2). Let us call the first solution $y_{1,n+1}$ and the second $y_{2,n+1}$. Now the exact solution (neglecting earlier accumulated error) at x_{n+1} could be written in each case as

$$\left.\begin{aligned} y_e &= y_{1,n+1} + \alpha h^{k+1} + \cdots + \\ y_e &= y_{2,n+1} + 2\alpha(\tfrac{1}{2} h^{k+1}) + \cdots + \end{aligned}\right\} , \tag{5.1.38}$$

where k is the order of the approximation scheme. Now α can be regarded as a constant throughout the interval h since it is just the coefficient of the Taylor series fit for the (k+1)th term. Now let us define δ as a measure of the error so that

$$\delta(y_{n+1}) \tilde{} \; y_{2,n+1} - y_{1,n+1} = \alpha h^{k+1}/(1-2^k) . \tag{5.1.39}$$

Clearly,

$$\delta(y_{n+1}) \sim h^{k+1} , \tag{5.1.40}$$

so that the step size h can be adjusted at each step in order that the truncation error remains uniform by

$$h_{n+1} = h_n \left| \delta(y_n)/\delta(y_{n+1}) \right|^{k+1} . \tag{5.1.41}$$

Initially, one must set the tolerance at some pre-assigned level ε so that

$$\left| \delta y_0 \right| . \; \varepsilon . \tag{5.1.42}$$

If we use this procedure to investigate the step sizes used in our test of the Runge-Kutta method, we see that we certainly chose the step size to be too large. We can verify this with the second order solution for we carried out the calculation for step sizes of h=1 and h=½. Following the prescription of equation (5.1.39) and (5.1.41) we have, that for the results specified in Table 5.2,

$$\left.\begin{aligned} \delta y_1 &= y_{2,2} - y_{1,1} = 1.6172 - 1.500 = 0.1172 \\ h_1 &= h_0 \left| \delta y_0 \big/ \delta y_1 \right| = (1)(0.1/0.1172) = 0.8532 \end{aligned}\right\} . \tag{5.1.43}$$

Here we have tacitly assumed an initial tolerance of $\delta y_0 = 0.1$. While this is arbitrary and rather large for a tolerance on a solution, it is illustrative and consistent with the spirit of the solution. We see that to maintain the accuracy of the solution within $\left| 0.1 \right|$ we should decrease the step size slightly for the initial step. The error at the end of the first step is 0.16 for h = 1, while it is only about 0.04 for h = ½. By comparing the numerical answers with the analytic answer, y_c, we see that factor of two change in the step size reduces the error by about a factor of four. Our stated tolerance of 0.1 requires only a reduction in the error of about 33% which implies a reduction of about 16% in the step size or a new step size $h_1' = 0.84h_1$. This is amazingly close to the recommended change, which was determined without knowledge of the analytic solution.

The amount of the step size adjustment at the second step is made to maintain the accuracy that exists at the end of the first step. Thus,

$$\left.\begin{aligned} \delta y_2 &= y_{2,2} - y_{1,2} = 6.5951 - 4.7500 = 1.8451 \\ h_2 &= h_1 \left| \delta y_1 \big/ \delta y_2 \right| = (1)(0.1172/1.8451) = 0.0635 \end{aligned}\right\} . \tag{5.1.44}$$

Normally these adjustments would be made cumulatively in order to maintain the initial tolerance. However, the convenient values for the step sizes were useful for the earlier comparisons of integration methods. The rapid increase of the solution after x = 1 causes the Runge-Kutta method to have an increasingly difficult time maintaining accuracy. This is abundantly clear in the drastic reduction in the step size suggested at the end of the second step. At the end of the first step, the relative errors where 9% and 2% for the h=1 and h=½ step size solutions respectively. At the end of the second step those errors, resulting from comparison with the analytic solution, had jumped to 55% and 12% respectively (see table 5.2). While a factor of two-change in the step size still produces about a factor of four change in the solution, to arrive at a relative error of 9%, we will need more like a factor of 6 change in the solution. This would suggest a change in the step size of a about a factor of three, but the recommended change is more like a factor of 16. This difference can be understood by noticing that equation (5.1.42) attempts to maintain the absolute error less than δy_n. For our problem this is about 0.11 at the end of step one. To keep the error within those tolerances, the accuracy at step two would have to be within about 1.5% of the correct answer. To get there from 55% means a reduction in the error of a *factor* of 36, which corresponds to a reduction in the step size of a factor of about 18, is close to that given by the estimate.

Thus we see that the equation (5.1.42) is designed to maintain an absolute accuracy in the solution by adjusting the step size. Should one wish to adjust the step size so as to maintain a relative or percentage accuracy, then one could adjust the step size according to

$$h_{n+1} = h_n \left| \{[\delta(y_n)y_{n+1}]/[\delta(y_{n+1})y_n]\} \right|^{k+1} . \tag{5.1.45}$$

While these procedures vary the step size so as to maintain constant truncation error, a significant price in the amount of computing must be paid at each step. However, the amount of extra effort need not be used only to estimate the error and thereby control it. One can solve equations (5.1.38) (neglecting terms of order greater than k) to provide an improved estimate of y_{n+1}. Specifically

$$y_e \cong y_{2,n+1} + \delta(y_{n+1})/(2^k-1) \quad . \tag{5.1.46}$$

However, since one cannot simultaneously include this improvement directly in the error estimate, it is advisable that it be regarded as a "safety factor" and proceeds with the *error estimate* as if the improvement had not been made. While this may seem unduly conservative, in the numerical solution of differential equations conservatism is a virtue.

c. Multi-Step and Predictor-Corrector Methods

The high order one step methods achieve their accuracy by exploring the solution space in the neighborhood of the specific solution. In principle, we could use prior information about the solution to constrain our extrapolation to the next step. Since this information is the direct result of prior calculation, far greater levels of efficiency can be achieved than by methods such as Runge-Kutta that explore the solution space in the vicinity of the required solution. By using the solution at n points we could, in principle, fit an (n-1) degree polynomial to the solution at those points and use it to obtain the solution at the (n+1)st point. Such methods are called *multi-step methods*. However, one should remember the caveats at the end of chapter 3 where we pointed out that polynomial extrapolation is extremely unstable. Thus such a procedure by itself will generally not provide a suitable method for the solution of differential equations. But when combined with algorithms that compensate for the instability such schemes can provide very stable solution algorithms. Algorithms of this type are called *predictor-corrector* methods and there are numerous forms of them. So rather than attempt to cover them all, we shall say a few things about the general theory of such schemes and give some examples.

A predictor-corrector algorithm, as the name implies, consists of basically two parts. The predictor extrapolates the solution over some finite range h based on the information at prior points and is inherently unstable. The corrector allows for this local instability and makes a correction to the solution at the end of the interval also based on prior information as well as the extrapolated solution. Conceptually, the notion of a predictor is quite simple. In its simplest form, such a scheme is the one-step predictor where

$$y_{n+1} = y_n + hy'_n \quad . \tag{5.1.47}$$

By using the value of the derivative at x_n the scheme will systematically under estimate the proper value required for extrapolation of any monotonically increasing solution (see figure 5.2). The error will build up cumulatively and hence it is unstable. A better strategy would be to use the value of the derivative midway between the two solution points, or alternatively to use the information from the prior two points to predict y_{n+1}. Thus a two point predictor could take the form

$$y_{n+1} = y_{n-1} + 2hy'_n \quad . \tag{5.1.48}$$

Although this is a two-point scheme, the extrapolating polynomial is still a straight line. We could have used the value of y_n directly to fit a parabola through the two points, but we didn't due to the instabilities to be associated with a higher degree polynomial extrapolation. This deliberate rejection of the some of the informational constraints in favor of increased stability is what makes predictor-corrector schemes non-trivial and effective. In the general case, we have great freedom to use the information we have regarding y_i and y'_i If we were to include all the available information, a general predictor would have the form

$$y_{n+1} = \sum_{i=0} a_i y_i + h \sum_{i=0} b_i y'_i + R \ , \tag{5.1.49}$$

where the a_i s and b_i s are chosen by imposing the appropriate constraints at the points x_i and R is an error term.

When we have decided on the form of the predictor, we must implement some sort of corrector scheme to reduce the truncation error introduced by the predictor. As with the predictor, let us take a simple case of a corrector as an example. Having produced a solution at x_{n+1} we can calculate the value of the derivative y'_{n+1} at x_{n+1}. This represents new information and can be used to modify the results of the prediction. For example, we could write a corrector as

$$y^{(k)}_{n+1} = y_n + \tfrac{1}{2}h[y'^{(k-1)}_{n+1} + y'_n] \ . \tag{5.1.50}$$

Therefore, if we were to write a general expression for a corrector based on the available information we would get

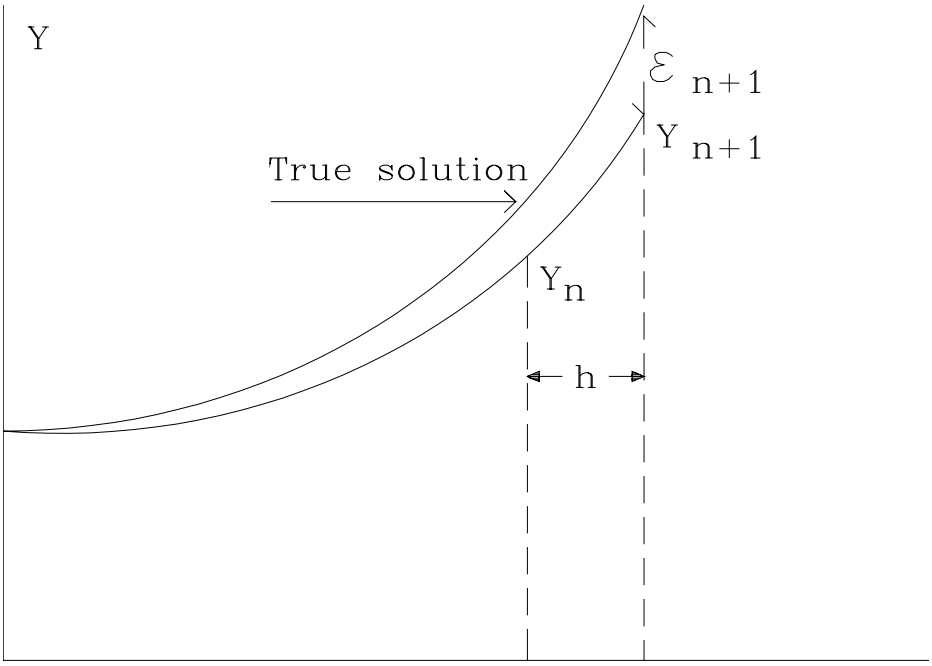


Figure 5.2 shows the instability of a simple predictor scheme that systematically underestimates the solution leading to a cumulative build up of truncation error.

$$y^{(k)}_{n+1} = \sum_{i=0}^{n} \alpha_i y_i + h \sum_{i=0}^{n} \beta_i y'_i + h\beta_{n+1} y'^{(k+1)}_{n+1} \ . \tag{5.1.51}$$

Equations (5.1.50) and (5.1.51) both are written in the form of iteration formulae, but it is not at all clear that

the fixed-point for these formulae is any better representation of the solution than single iteration. So in order to minimize the computational demands of the method, correctors are generally applied only once. Let us now consider certain specific types of predictor corrector schemes that have been found to be successful.

Hamming[1] gives a number of popular predictor-corrector schemes, the best known of which is the Adams-Bashforth-Moulton Predictor-Corrector. Predictor schemes of the Adams-Bashforth type emphasize the information contained in prior values of the derivative as opposed to the function itself. This is presumably because the derivative is usually a more slowly varying function than the solution and so can be more accurately extrapolated. This philosophy is carried over to the Adams-Moulton Corrector. A classical fourth-order formula of this type is

$$\left.\begin{aligned} y^{(1)}_{n+1} &= y_n + h(55y'_n - 59y'_{n-1} + 37y'_{n-2} - 9y'_{n-3})/24 \quad + O(h^5) \\ y_{n+1} &= y_n + h(9y'_{n+1} + 19y'_n - 5y'_{n-1})/24 \quad + O(h^5) \end{aligned}\right\} . \qquad (5.1.52)$$

Lengthy study of predictor-corrector schemes has evolved some special forms such as this one

$$\left.\begin{aligned} z_{n+1} &= (2y_{n-1} + y_{n-2})/3 \quad + h(191y'_n - 107y'_{n-1} + 109y'_{n-2} - 25y'_{n-3})/75 \\ u_{n+1} &= z_{n+1} - 707(z_n - c_n)/750 \\ c_{n+1} &= (2y_{n-1} + y_{n-2})/3 \quad + h(25u'_{n+1} + 91y'_n + 43y'_{n-1} + 9y'_{n-2})/72 \\ y_{n+1} &= c_{n+1} + 43(z_{n+1} - c_{n+1})/750 \quad + O(h^6) \end{aligned}\right\} . \qquad (5.1.53)$$

where the extrapolation formula has been expressed in terms of some recursive parameters u_i and c_i. The derivative of these intermediate parameters are obtained by using the original differential equation so that

$$u' = g(x,u) \ . \qquad (5.1.54)$$

By good chance, this formula [equation (5.1.53)] has an error term that varies as $O(h^6)$ and so is a fifth-order formula. Finally a classical predictor-corrector scheme which combines Adams-Bashforth and Milne predictors and is quite stable is parametrically (i.e. Hamming p206)

$$\left.\begin{aligned} z_{n+1} &= \tfrac{1}{2}(y_n + y_{n-1}) \quad + h(119y'_n - 99y'_{n-1} + 69y'_{n-2} - 17y'_{n-3})/48 \\ u_{n+1} &= z_{n+1} - 161(z_n - c_n)/170 \\ c_{n+1} &= \tfrac{1}{2}(y_n + y_{n-1}) \quad + h(17u'_{n+1} + 51y'_n + 3y'_{n-1} + y'_{n-2})/48 \\ y_{n+1} &= c_{n+1} + 9(z_{n+1} - c_{n+1})/170 \quad + O(h^6) \end{aligned}\right\} . \qquad (5.1.55)$$

Press et al[2] are of the opinion that predictor-corrector schemes have seen their day and are made obsolete by the Bulirsch-Stoer method which they discuss at some length[3]. They quite properly point out that the predictor-corrector schemes are somewhat inflexible when it comes to varying the step size. The step size can be reduced by interpolating the necessary missing information from earlier steps and it can be expanded in integral multiples by skipping earlier points and taking the required information from even earlier in the solution. However, the Bulirsch-Stoer method, as described by Press et. al. utilizes a predictor scheme with some special properties. It may be parameterized as

$$\left.\begin{aligned} z_0 &= y(x_0) \\ z_1 &= z_0 + hz'_0 \\ z_{k+1} &= z_{k-1} + hz'_k \quad k = 1,2,3,\cdots,n-1 \\ y_n^{(1)} &= \tfrac{1}{2}(z_n + z_{n-1} + hz'_n) \quad + O(h^5) \\ z' &= g(z,x) \end{aligned}\right\} . \tag{5.1.56}$$

It is an odd characteristic of the third of equations (5.1.56) that the error term only contains even powers of the step size. Thus, we may use the same trick that was used in equation (5.1.46) of utilizing the information generated in estimating the error term to improve the approximation order. But since only even powers of h appear in the error term, this single step will gain us two powers of h resulting in a predictor of order seven.

$$y_{nh} = \{4y_n^{(1)}(x+nh) - y_{n/2}^{(1)}\,[x+(n/2)(2h)]\}/3 + O(h^7) . \tag{5.1.57}$$

This yields a predictor that requires something on the order of 1½ evaluations of the function per step compared to four for a Runge-Kutta formula of inferior order.

Now we come to the aspect of the Bulirsch-Stoer method that begins to differentiate it from classical predictor-correctors. A predictor that operates over some *finite* interval can use a successively increasing number of steps in order to make its prediction. Presumably the prediction will get better and better as the step size decreases so that the number of steps to make the one prediction increases. Of course practical aspects of the problem such as roundoff error and finite computing resources prevent us from using arbitrarily small step sizes, but we can approximate what would happen in an ideal world without round-off error and utilizing unlimited computers. Simply consider the prediction at the end of the finite interval H where

$$H = \alpha h . \tag{5.1.58}$$

Thus $y_\alpha(x+H)$ can be taken to be a function of the step size h so that,

$$y_\alpha(x+H) = y(x+\alpha h) = f(h) . \tag{5.1.59}$$

Now we can phrase our problem to estimate the value of that function in the limit

$$\lim_{\substack{h\to 0 \\ \alpha\to\infty}} f(h) = Y_\infty(x+H) . \tag{5.1.60}$$

We can accomplish this by carrying out the calculation for successively smaller and smaller values of h and, on the basis of these values, extrapolating the result to h=0. In spite of the admonitions raised in chapter 3 regarding extrapolation, the range here is small. But to produce a truly powerful numerical integration algorithm, Bulirsch and Stoer carry out the extrapolation using rational functions in the manner described in section 3.2 [equation (3.2.65)]. The superiority of rational functions to polynomials in representing most analytic functions means that the step size can be quite large indeed and the conventional meaning of the 'order' of the approximation is irrelevant in describing the accuracy of the method.

In any case, remember that accuracy and order are not synonymous! Should the solution be described by a slowly varying function and the numerical integration scheme operate by fitting high order polynomials to prior information for the purposes of extrapolation, the high-order formula can give very inaccurate results. This simply says that the integration scheme can be unstable even for well behaved solutions.

Press et. al.[4] suggest that all one needs to solve ordinary differential equations is either a Runge-Kutta or Bulirsch-Stoer method and it would seem that for most problems that may well be the case. However, there are a large number of commercial differential equation solving algorithms and the majority of them utilize predictor-corrector schemes. These schemes are generally very fast and the more sophisticated ones carry out very involved error checking algorithms. They are generally quite stable and can involve a very high order when required. In any event, the user should know how they work and be wary of the results. It is far too easy to simply take the results of such programs at face value without ever questioning the accuracy of the results. Certainly one should always ask the question "Are these results reasonable?" at the end of a numerical integration. If one is genuinely skeptical, it is not a bad idea to take the final value of the calculation as an initial value and integrate back over the range. Should one recover the original initial value within the acceptable tolerances, one can be reasonably confident that the results are accurate. If not, the difference between the beginning initial value and what is calculated by the reverse integration over the range can be used to place limits on the accuracy of the initial integration.

d. Systems of Differential Equations and Boundary Value Problems

All the methods we have developed for the solution of single first order differential equations may be applied to the case where we have a coupled system of differential equations. We saw earlier that such systems arose whenever we dealt with ordinary differential equations of order greater than one. However, there are many scientific problems which are intrinsically described by coupled systems of differential equations and so we should say something about their solution. The simplest way to see the applicability of the single equation algorithms to a system of differential equations is to write a system like

$$\left.\begin{aligned} y'_1 &= g_1(x, y_1, y_2, \cdots y_n) \\ y'_2 &= g_2(x, y_1, y_2, \cdots y_n) \\ \vdots &\qquad\qquad \vdots \\ y'_n &= g_n(x, y_1, y_2, \cdots y_n) \end{aligned}\right\} , \tag{5.1.61}$$

as a vector where each element represents one of the dependent variables or unknowns of the system. Then the system becomes

$$\vec{y}' = \vec{g}(x, \vec{y}) \quad , \tag{5.1.62}$$

which looks just like equation (5.1.3) so that everything applicable to that equation will apply to the system of equations. Of course some care must be taken with the terminology. For example, equation (5.1.4) would have to be understood as standing for an entire system of equations involving far more complicated integrals, but in principle, the ideas carry over. Some care must also be extended to the error analysis in that the error term is also a vector $\vec{R}(x)$. In general, one should worry about the magnitude of the error vector, but in

practice, it is usually the largest element that is taken as characterizing the accuracy of the solution.

To generate a numerical integration method for a specific algorithm, one simply applies it to each of the equations that make up the system. By way of a specific example, let's consider a forth order Runge-Kutta algorithm as given by equation (5.1.33) and apply it to a system of two equations. We get

$$\left.\begin{aligned}
y_{1,n+1} &= y_{1,n} + (t_0 + 2t_1 + 2t_2 + t_3)/6 \\
y_{2,n+1} &= y_{2,n} + (u_0 + 2u_1 + 2u_2 + u_3)/6 \\
t_0 &= hg_1(x_n, y_{1,n}, y_{2,n}) \\
t_1 &= hg_1[(x_n + \tfrac{1}{2}h), (y_{1,n} + \tfrac{1}{2}t_0), (y_{2,n} + \tfrac{1}{2}u_0)] \\
t_2 &= hg_1[(x_n + \tfrac{1}{2}h), (y_{1,n} + \tfrac{1}{2}t_1), (y_{2,n} + \tfrac{1}{2}u_1)] \\
t_3 &= hg_1[(x_n + h), (y_{1,n} + t_t), (y_{2,n} + u_2)] \\
u_0 &= hg_2(x_n, y_{1,n}, y_{2,n}) \\
u_1 &= hg_2[(x_n + \tfrac{1}{2}h), (y_{1,n} + \tfrac{1}{2}t_0), (y_{2,n} + \tfrac{1}{2}u_0)] \\
u_2 &= hg_2[(x_n + \tfrac{1}{2}h), (y_{1,n} + \tfrac{1}{2}t_1), (y_{2,n} + \tfrac{1}{2}u_1)] \\
u_1 &= hg_2[(x_n + h), (y_{1,n} + t_t), (y_{2,n} + u_2)]
\end{aligned}\right\}. \qquad (5.1.63)$$

We can generalize equation (5.1.63) to an arbitrary system of equations by writing it in vector form as

$$\vec{y}_{n+1} = \vec{A}(\vec{y}_n). \qquad (5.1.64)$$

The vector $\vec{A}(\vec{y}_n)$ consists of elements which are functions of dependent variables $y_{i,n}$ and x_n, but which all have the same general form varying only with $g_i(x, \vec{y})$. Since an nth order differential equation can always be reduced to a system of n first order differential equations, an expression of the form of equation (5.1.63) could be used to solve a second order differential equation.

The existence of coupled systems of differential equations admits the interesting possibility that the constants of integration required to uniquely specify a solution are not all given at the same location. Thus we do not have a full compliment of $y_{i,0}$'s with which to begin the integration. Such problems are called *boundary value problems*. A comprehensive discussion of boundary value problems is well beyond the scope of this book, but we will examine the simpler problem of *linear two point boundary value problems*. This subclass of boundary value problems is quite common in science and extremely well studied. It consists of a system of linear differential equations (i.e. differential equations of the first degree only) where part of the integration constants are specified at one location x_0 and the remainder are specified at some other value of the independent variable x_n. These points are known as the boundaries of the problem and we seek a solution to the problem within these boundaries. Clearly the solution can be extended beyond the boundaries as the solution at the boundaries can serve as initial values for a standard numerical integration.

The general approach to such problems is to take advantage of the linearity of the equations, which guarantees that any solution to the system can be expressed as a linear combination of a set of basis

solutions. A set of basis solutions is simply a set of solutions, which are linearly independent. Let us consider a set of m linear first order differential equations where k values of the dependent variables are specified at x_0 and (m-k) values corresponding to the remaining dependent variables are specified at x_n. We could solve (m-k) initial value problems starting at x_0 and specifying (m-k) independent, sets of missing initial values so that the initial value problems are uniquely determined. Let us denote the missing set of initial values at x_0 by $\vec{y}^{(0)}(x_0)$ which we know can be determined from initial sets of linearly independent trial initial values ${}^j\vec{y}^{(t)}(x_0)$ by

$$\vec{y}^{(0)}(x_0) = \mathbf{A}\mathbf{y}^{(t)}(x_0), \tag{5.1.65}$$

The columns of $\mathbf{y}^{(t)}(x_0)$ are just the individual vectors ${}^j\vec{y}^{(t)}(x_0)$. Clearly the matrix **A** will have to be diagonal to always produce $\vec{y}^{(0)}(x_0)$. Since the trial initial values are arbitrary, we will choose the elements of the (m-k) sets to be

$${}^jy_i(x_0) = \delta_{ij}\ , \tag{5.1.66}$$

so that the missing initial values will be

$$\vec{y}^{(0)}(x_0)\,\mathbf{1} = \mathbf{1}\mathbf{A} = \mathbf{A}\ . \tag{5.1.67}$$

Integrating across the interval with these initial values will yield (m-k) solution ${}^j\vec{y}^{(t)}(x_n)$ at the other boundary. Since the equations are linear each trial solution will be related to the known boundary values ${}^j\vec{y}^{(t)}(x_n)$ by

$$\vec{y}^{(0)}(x_n) = \mathbf{A}[{}^j\vec{y}^{(t)}(x_n)]\ , \tag{5.1.68}$$

so that for the complete set of trial solutions we may write

$$\vec{y}^{(0)}(x_n)\,\mathbf{1} = \mathbf{A}\mathbf{y}^{(t)}(x_n)\ , \tag{5.1.69}$$

where by analogy to equation (5.1.65), the column vectors of $\mathbf{y}^{(t)}(x_n)$ are ${}^j\vec{y}^{(t)}(x_n)$. We may solve these equations for the unknown transformation matrix **A** so that the missing initial values are

$$\vec{y}^{(0)}(x_n)\,\mathbf{1} = \mathbf{A} = \mathbf{y}^{-1}\ \vec{y}^{(0)}(x_n)\,. \tag{5.1.70}$$

If one employs a one step method such as Runge-Kutta, it is possible to collapse this entire operation to the point where one can represent the complete boundary conditions at one boundary in terms of the values at the other boundary $\vec{y}_n$ a system of linear algebraic equations such as

$$\vec{y}(x_0) = \mathbf{B}\vec{y}(x_n)\,. \tag{5.1.71)}$$

The matrix **B** will depend only on the details of the integration scheme and the functional form of the equations themselves, not on the boundary values. Therefore it may be calculated for any set of boundary values and used repeatedly for problems differing only in the values at the boundary (see Day and Collins[5]).

To demonstrate methods of solution for systems of differential equations or boundary value

problems, we shall need more than the first order equation (5.1.10) that we used for earlier examples. However, that equation was quite illustrative as it had a rapidly increasing solution that emphasized the shortcomings of the various numerical methods. Thus we shall keep the solution, but change the equation. Simply differentiate equation (5.1.10) so that

$$Y'' = 2(1+2x^2)e^{x^2} = 2(1+x^2)y \quad . \tag{5.1.72}$$

Let us keep the same initial condition given by equation (5.1.11) and add a condition of the derivative at x = 1 so that

$$\left.\begin{aligned} y(0) &= 1 \\ y'(1) &= 2e = 5.43656 \end{aligned}\right\} \quad . \tag{5.1.73}$$

This insures that the closed form solution is the same as equation (5.1.12) so that we will be able to compare the results of solving this problem with earlier methods. We should not expect the solution to be as accurate for we have made the problem more difficult by increasing the order of the differential equation in addition to separating the location of the constants of integration. This is no longer an initial value problem since the solution value is given at x = 0, while the other constraint on the derivative is specified at x = 1. This is typical of the classical two-point boundary value problem.

We may also use this example to indicate the method for solving higher order differential equations given at the start of this chapter by equations (5.1.1) and (5.1.2). With those equations in mind, let us replace equation (5.1.72) by system of first order equations

$$\left.\begin{aligned} y'_1(x) &= y_2(x) \\ y'_2(x) &= 2(1+2x^2)y_1(x) \end{aligned}\right\} \quad , \tag{5.1.74}$$

which we can write in vector form as

$$\vec{y}' = \mathbf{A}(x)\vec{y} \, , \tag{5.1.75}$$

where

$$\mathbf{A}(x) = \begin{pmatrix} 0 & 1 \\ 2(1+x^2) & 0 \end{pmatrix} \quad . \tag{5.1.76}$$

The components of the solution vector $\vec{y}$ are just the solution we seek (i.e.) and its derivative. However, the form of equation (5.1.75) emphasizes its linear form and were it a scalar equation, we should know how to proceed.

For purposes of illustration, let us apply the fourth order Runge-Kutta scheme given by equation (5.1.63). Here we can take specific advantage of the linear nature of our problem and the fact that the dependence on the independent variable factors out of the right hand side. To illustrate the utility of this fact, let

$$g(x,y) = [f(x)]y \quad , \tag{5.1.77}$$

in equation (5.1.63).

Then we can write the fourth order Runge-Kutta parameters as

$$\left.\begin{aligned} t_0 &= hf_0 y_n \\ t_1 &= hf_1(y_n + \tfrac{1}{2}t_0) = hf_1(y_n + \tfrac{1}{2}hf_0 y_n = (hf_1 + \tfrac{1}{2}h^2 f_1 f_0)y_n \\ t_2 &= hf_1(y_n + \tfrac{1}{2}t_1) = (hf_1 + \tfrac{1}{2}h^2 f_1^2 + \tfrac{1}{4}h^3 f_1^2 f_0)y_n \\ t_3 &= hf_2(y_n + t_2) = (hf_2 + h^2 f_2 f_1 + \tfrac{1}{2}h^3 f_1^2 f_2 + \tfrac{1}{4}h^4 f_2 f_1^2 f_0)y_n \end{aligned}\right\} . \qquad (5.1.78)$$

where

$$\left.\begin{aligned} f_0 &= f(x_n) \\ f_1 &= f(x_n + \tfrac{1}{2}h) \\ f_2 &= f(x_n + h) \end{aligned}\right\} , \qquad (5.1.79)$$

so that the formula becomes

$$\begin{aligned} y_{n+1} &= y_n + (t_0 + 2t_1 2t_2 + t_3) \\ &= \left[1 + \frac{h}{6}(f_0 + 4f_1 + f_2) + \frac{h^2}{6}(f_1 f_0 + f_1^2 + f_2 f_1) + \frac{h^3}{12}(f_1^2 f_0 + f_2 f_1^2) + \frac{h^4}{24} f_2 f_1^2 f_0)\right] y_n \end{aligned} . \qquad (5.1.80)$$

Here we see that the linearity of the differential equation allows the solution at step n to be factored out of the formula so that the solution at step n appears explicitly in the formula. Indeed, equation (5.1.80) represents a power series in h for the solution at step (n+1) in terms of the solution at step n. Since we have been careful about the order in which the functions f_i multiplied each other, we may apply equation (5.1.80) directly to equation (5.1.75) and obtain a similar formula for systems of linear first order differential equations that has the form

$$\vec{y}_{n+1} = \left[1 + \frac{h}{6}(\mathbf{A}_0 + 4\mathbf{A}_1 + \mathbf{A}_2) + \frac{h^2}{6}(\mathbf{A}_0\mathbf{A}_1 + 4\mathbf{A}_1^2 + \mathbf{A}_2\mathbf{A}_1) + \frac{h^3}{12}(\mathbf{A}_1^2\mathbf{A}_0 + \mathbf{A}_2\mathbf{A}_1^2) + \frac{h^4}{24}\mathbf{A}_2\mathbf{A}_1^2\mathbf{A}_0\right]\vec{y}_n . \qquad (5.1.81)$$

Here the meaning of $\mathbf{A}_i$ is the same as f_i in that the subscript indicates the value of the independent variable x for which the matrix is to be evaluated. If we take h = 1, the matrices for our specific problem become

$$\left.\begin{aligned} \mathbf{A}_0 &= \begin{pmatrix} 0 & 1 \\ 2 & 0 \end{pmatrix} \\ \mathbf{A}_1 &= \begin{pmatrix} 0 & 1 \\ 3 & 0 \end{pmatrix} \\ \mathbf{A}_2 &= \begin{pmatrix} 0 & 1 \\ 4 & 0 \end{pmatrix} \end{aligned}\right\} . \qquad (5.1.82)$$

Keeping in mind that the order of matrix multiplication is important, the products appearing in the second order term are

$$\left.\begin{aligned} \mathbf{A}_1\mathbf{A}_0 &= \begin{pmatrix} 2 & 0 \\ 0 & 3 \end{pmatrix} \\ \mathbf{A}_1^2 &= \begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix} \\ \mathbf{A}_2\mathbf{A}_1 &= \begin{pmatrix} 3 & 0 \\ 0 & 6 \end{pmatrix} \end{aligned}\right\} . \tag{5.1.83}$$

The two products appearing in the third order term can be easily generated from equations (5.1.82) and (5.1.83) and are

$$\left.\begin{aligned} \mathbf{A}_1^2\mathbf{A}_0 &= \begin{pmatrix} 0 & 3 \\ 9 & 0 \end{pmatrix} \\ \mathbf{A}_2\mathbf{A}_1^2 &= \begin{pmatrix} 0 & 3 \\ 18 & 0 \end{pmatrix} \end{aligned}\right\} . \tag{5.1.84}$$

Finally the single matrix of the first order term can be obtain by successive multiplication using equations(5.1.82) and (5.1.84) yielding

$$\left.\mathbf{A}_2\mathbf{A}_1^2\mathbf{A}_0 = \begin{pmatrix} 9 & 0 \\ 0 & 18 \end{pmatrix}\right\} . \tag{5.1.85}$$

Like equation (5.1.80), we can regard equation (5.1.81) as a series solution in h that yields a system of linear equations for the solution at step n+1 in terms of the solution at step n. It is worth noting that the coefficients of the various terms of order h^k are similar to those developed for equal interval quadrature formulae in chapter 4. For example the lead term being the unit matrix generates the coefficients of the trapezoid rule while the h(+1, +4, +1)/6 coefficients of the second term are the familiar progression characteristic of Simpson's rule. The higher order terms in the formula are less recognizable since they depend on the parameters chosen in the under-determined Runge-Kutta formula.

If we define a matrix $\mathbf{P}(h^k)$ so that

$$\vec{y}_{n+1} = \mathbf{P}(h^k) \equiv {}^k\mathbf{P}\,\vec{y}_n , \tag{5.1.86}$$

the series nature of equation (5.1.81) can be explicitly represented in terms of the various values of ${}^k\mathbf{P}$.

For our problem they are:

$$\left.\begin{aligned}
{}^{0}\mathbf{P} &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \\
{}^{1}\mathbf{P} &= \begin{pmatrix} 1 & 1 \\ \frac{11}{6} & 0 \end{pmatrix} \\
{}^{2}\mathbf{P} &= \begin{pmatrix} \frac{7}{3} & 1 \\ \frac{11}{6} & 0 \end{pmatrix} \\
{}^{3}\mathbf{P} &= \begin{pmatrix} \frac{7}{3} & \frac{3}{2} \\ \frac{49}{12} & 3 \end{pmatrix} \\
{}^{4}\mathbf{P} &= \begin{pmatrix} \frac{65}{24} & \frac{3}{2} \\ \frac{49}{12} & \frac{15}{4} \end{pmatrix}
\end{aligned}\right\} . \qquad (5.1.87)$$

The boundary value problem now is reduced to solving the linear system of equations specified by equation (5.1.86) where the known values at the respective boundaries are specified. Using the values given in equation (5.1.73) the linear equations for the missing boundary values become

$$\left.\begin{aligned}
1 &= {}^{k}\mathbf{P}_{11}\, y_1(0) + {}^{k}\mathbf{P}_{12}(5.43656) \\
y_2(0) &= {}^{k}\mathbf{P}_{21}\, y_1(1) + {}^{k}\mathbf{P}_{22}(5.43656)
\end{aligned}\right\} . \qquad (5.1.88)$$

The first of these yields the missing solution value at x = 0 [i.e. $y_2(0)$]. With that value the remaining value can be obtained from the second equation. The results of these solutions including additional terms of order h^k are given in table 5.3. We have taken h to be unity, which is unreasonably large, but it serves to demonstrate the relative accuracy of including higher order terms and simplifies the arithmetic. The results for the missing values $y_2(0)$ and $y_1(1)$ (i.e. the center two rows) converge slowly, and not uniformly, toward their analytic values given in the column labeled $k = \infty$.

Had we chosen the step size h to be smaller so that a number of steps were required to cross the interval, then each step would have produced a matrix ${}^{k}_{i}\mathbf{P}$ and the solution at each step would have been related to the solution at the following step by equation (5.1.86). Repeated application of that equation would yield the solution at one boundary in terms of the solution at the other so that

$$\vec{y}_n = ({}^{k}_{n-1}\mathbf{P}\ {}^{k}_{n-2}\mathbf{P}\ {}^{k}_{n-3}\mathbf{P}\cdots{}^{0}_{0}\mathbf{P})\vec{y}_0 = {}^{k}\mathbf{Q}\,\vec{y}_0 \quad . \tag{5.1.89}$$

Table 5.3

Solutions of a Sample Boundary Value Problem for Various Orders of Approximation

\ k	0	1	2	3	4	∞
$y_1(0)$	1.0	1.0	1.0	1.0	1.0	1.0
$y_2(0)$	5.437	3.60	1.200	0.4510	0.3609	0.0
$y_1(1)$	1.0	4.60	3.53	3.01	3.25	2.71828
$y_2(1)$	5.437	5.437	5.437	5.437	5.437	2e

Thus one arrives at a similar set of linear equations to those implied by equation (5.1.86) and explicitly given in equation (5.1.88) relating the solution at one boundary in terms of the solution at the other boundary. These can be solved for the missing boundary values in the same manner as our example. Clearly the decrease in the step size will improve the accuracy as dramatically as increasing the order k of the approximation formula. Indeed the step size can be variable at each step allowing for the use of the error correcting procedures described in section 5.1b.

Table 5.4

Solutions of a Sample Boundary Value Problem

\ k	0	1	2	3	4	∞
$y_1(0)$	1.0	1.0	1.0	1.0	1.0	1.0
$y_2(0)$	0.0	0.0	0.0	0.0	0.0	0.0
$y_1(1)$	1.0	1.0	2.33	2.33	2.708	2.718
$y_2(1)$	0.0	1.83	1.83	4.08	4.08	5.437

Any set of boundary values could have been used with equations (5.1.81) to yield the solution elsewhere. Thus, we could treat our sample problem as an initial value problem for comparison. If we take the analytic values for $y_1(0)$ and $y_2(0)$ and solve the resulting linear equations, we get the results given in Table 5.4. Here the final solution is more accurate and exhibits a convergence sequence more like we would expect from Runge-Kutta. Namely, the solution systematically lies below the rapidly increasing analytic solution. For the boundary value problem, the reverse was true and the final result less accurate. This is not an uncommon result for two-point boundary value problems since the error of the approximation scheme is directly reflected in the determination of the missing boundary values. In an initial value problem, there is assumed to be no error in the initial values.

This simple example is not meant to provide a definitive discussion of even the restricted subset of linear two-point boundary value problems, but simply to indicate a way to proceed with their solution. Anyone wishing to pursue the subject of two-point boundary value problems further should begin with the venerable text by Fox[6].

e. Partial Differential Equations

The subject of partial differential equations has a literature at least as large as that for ordinary differential equations. It is beyond the scope of this book to provide a discussion of partial differential equations even at the level chosen for ordinary differential equations. Indeed, many introductory books on numerical analysis do not treat them at all. Thus we will only sketch a general approach to problems involving such equations.

Partial differential equations form the basis for so many problems in science, that to limit the choice of examples. Most of the fundamental laws of physical science are written in terms of partial differential equations. Thus one finds them present in computer modeling from the hydrodynamic calculations needed for airplane design, weather forecasting, and the flow of fluids in the human body to the dynamical interactions of the elements that make up a model economy.

A partial derivative simply refers to the rate of change of a function of many variables, with respect to just one of those variables. In terms of the familiar limiting process for defining differentials we would write

$$\frac{\partial F(x_1, x_2, \cdots, x_n)}{\partial x_j} = \underset{\Delta x_j \to 0}{\text{Lim}} \left[\frac{F(x_1, x_2, \cdots, x_j, \cdots x_n) - F(x_1, x_2, \cdots, x_j + \Delta x_j, \cdots x_n)}{\Delta x_j} \right] . \quad (5.1.90)$$

Partial differential equations usually relate derivatives of some function with respect to one variable to derivatives of the same function with respect to another. The notion of order and degree are the same as with ordinary differential equations.

Although a partial differential equation may be expressed in multiple dimensions, the smallest number for illustration is two, one of which may be time. Many of these equations, which describe so many aspects of the physical world, have the form

$$a(x,y)\frac{\partial^2 z(x,y)}{\partial x^2} + 2b(x,y)\frac{\partial^2 z(x,y)}{\partial x \partial y} + c(x,y)\frac{\partial^2 z(x,y)}{\partial y^2} = F\left[x, y, z\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y} \right] . \quad (5.1.91)$$

and as such can be classified into three distinct groups by the discriminate so that

$$\left.\begin{array}{ll} [b^2(x,y) - a(x,y)c(x,y)] < 0 & \text{Elliptic} \\ [b^2(x,y) - a(x,y)c(x,y)] = 0 & \text{Parabolic} \\ [b^2(x,y) - a(x,y)c(x,y)] > 0 & \text{Hyperbolic} \end{array}\right\} . \quad (5.1.92)$$

Should the equation of interest fall into one of these three categories, one should search for solution

algorithms designed to be effective for that class. Some methods that will be effective at solving equations of one class will fail miserably for another.

While there are many different techniques for dealing with partial differential equations, the most wide-spread method is to replace the differential operator by a finite difference operator thereby turning the differential equation into a finite difference equation in at least two variables. Just as a numerical integration scheme finds the solution to a differential equation at discrete points x_i along the real line, so a two dimensional integration scheme will specify the solution at a set of discrete points x_i, y_j. These points can be viewed as the intersections on a grid. Thus the solution in the x-y space is represented by the solution on a finite grid. The location of the grid points will be specified by the finite difference operators for the two coordinates. Unlike problems involving ordinary differential equations, the initial values for partial differential equations are not simply constants. Specifying the partial derivative of a function at some particular value of one of the independent variables still allows it to be a function of the remaining independent variables of the problem. Thus the functional behavior of the solution is often specified at some boundary and the solution proceeds from there. Usually the finite difference scheme will take advantage of any symmetry that may result for the choice of the boundary. For example, as was pointed out in section 1.3 there are thirteen orthogonal coordinate systems in which Laplace's equation is separable. Should the boundaries of a problem match one of those coordinate systems, then the finite difference scheme would be totally separable in the independent variables greatly simplifying the numerical solution. In general, one picks a coordinate system that will match the local boundaries and that will determine the geometry of the grid. The solution can then proceed from the initial values at a particular boundary and move across the grid until the entire space has been covered. Of course the solution should be independent of the path taken in filling the grid and that can be used to estimate the accuracy of the finite difference scheme that is being used. The details of setting up various types of schemes are beyond the scope of this book and could serve as the subject of a book by themselves. For a further introduction to the solution of partial differential equations the reader is referred to Sokolnikoff and Redheffer[7] and for the numerical implementation of some methods the student should consult Press et.al.[8]. Let us now turn to the numerical solution of integral equations.

5.2 The Numerical Solution of Integral Equations

For reasons that I have never fully understood, the mathematical literature is crowded with books, articles, and papers on the subject of differential equations. Most universities have several courses of study in the subject, but little attention is paid to the subject of integral equations. The differential operator is linear and so is the integral operator. Indeed, one is just the inverse of the other. Equations can be written where the dependent variable appears under an integral as well as alone. Such equations are the analogue of the differential equations and are called *integral equations*. It is often possible to turn a differential equation into an integral equation which may make the problem easier to numerically solve. Indeed many physical phenomena lend themselves to description by integral equations. So one would think that they might form as large an area for analysis are do the differential equations. Such is not the case. Indeed, we will not be able to devote as much time to the discussion of these interesting equations as we should, but we shall spend enough time so that the student is at least familiar with some of their basic properties. Of necessity, we will restrict our discussion to those integral equations where the unknown appears linearly. Such linear equations are more tractable and yet describe much that is of interest in science.

a. Types of Linear Integral Equations

We will follow the standard classification scheme for integral equations which, while not exhaustive, does include most of the common types. There are basically two main classes known as Fredholm and Volterra after the mathematicians who first studied them in detail. Fredholm equations involve definite integrals, while Volterra equations have the independent variable as one of the limits. Each of these categories can be further subdivided as to whether or not the dependent variable appears outside the integral sign as well as under it. Thus the two types of Fredholm equations for the unknown φ are

$$\left.\begin{aligned} F(x) &= \int_a^b K(x,t)\phi(t)\,dt \qquad \text{Fredholm Type I} \\ \phi(x) &= F(x) + \lambda\int_a^b K(x,t)\phi(t)\,dt \qquad \text{Fredholm Type II} \end{aligned}\right\}, \tag{5.2.1}$$

while the corresponding two types of Volterra equations for φ take the form

$$\left.\begin{aligned} F(x) &= \int_a^x K(x,t)\phi(t)\,dt \qquad \text{Volterra Type I} \\ \phi(x) &= F(x) + \lambda\int_a^x K(x,t)\phi(t)\,dt \qquad \text{Volterra Type II} \end{aligned}\right\}. \tag{5.2.2}$$

The parameter K(x,t) appearing in the integrand is known as the *kernel* of the integral equation. Its form is crucial in determining the nature of the solution. Certainly one can have homogeneous or inhomogeneous integral equations depending on whether or not F(x) is zero. Of the two classes, the Fredholm are generally easier to solve.

b. The Numerical Solution of Fredholm Equations

Integral equations are often easier to solve than a corresponding differential equation. One of the reasons is that the truncation errors of the solution tend to be averaged out by the process of quadrature while they tend to accumulate during the process of numerical integration employed in the solution of differential equations. The most straight-forward approach is to simply replace the integral with a quadrature sum. In the case of Fredholm equations of type one, this results in a functional equation for the unknown φ(x) at a discrete set of points t_j used by the quadrature scheme. Specifically

$$F(x) = \sum_{j=0}^{n} K(x,t_j)\varphi(t_j)W_j + R_n(x) \quad . \tag{5.2.3}$$

Since equation (5.2.3) must hold for all values of x, it must hold for values of x equal to those chosen for the quadrature points so that

$$x_j = t_j \qquad j = 0, 1, 2, \cdots, n \quad . \tag{5.2.4}$$

By picking those particular points we can generate a linear system of equations from the functional equation (5.2.3) and, neglecting the quadrature error term, they are

$$F(x_i) = \sum_{j=0}^{n} K(x_i,t_j)\varphi(t_j)W_j = \sum_{j=0}^{n} A_{ij}\varphi(x_j) \qquad i = 0, 1, 2, \cdots, n \ , \tag{5.2.5}$$

which can be solved by any of the methods discussed in Chapter 2 yielding

$$\phi(x_j) = \sum_{k=0}^{n} A_{jk}^{-1} F(x_k) \qquad j = 0,1,2,\cdots,n\,. \tag{5.2.6}$$

The solution will be obtained at the quadrature points x_j so that one might wish to be careful in the selection of a quadrature scheme and pick one that contained the points of interest. However, one can use the solution set $\varphi(x_j)$ to interpolate for missing points and maintain the same degree of precession that generated the solution set. For Fredholm equations of type 2, one can perform the same trick of replacing the integral with a quadrature scheme. Thus

$$\phi(x) = F(x) + \lambda \sum_{j=0}^{n} K(x,t_j)\phi(t_j)W_j \quad + \quad R_n(x)\,. \tag{5.2.7}$$

Here we must be a little careful as the unknown $\phi(x)$ appears outside the integral. Thus equation (5.2.7) is a functional equation for $\phi(x)$ itself. However, by evaluating this functional equation as we did for Fredholm equations of type 1 we get

$$\phi(x_i) = F(x_i) + \lambda \sum_{j=0}^{n} K(x_i,t_j)\phi(t_j)W_j \qquad , \tag{5.2.8}$$

which, after a little algebra, can be put in the standard form for linear equations

$$F(x_i) = \sum_{j=0}^{n} [\delta_{ij} - \lambda K(x_i,t_j)W_j]\phi(t_j) = \sum_{j=0}^{n} B_{ij}\phi(x_j) \qquad i = 0,1,2,\cdots,n \quad , \tag{5.2.9}$$

that have a solution

$$\phi(x_j) = \sum_{k=0}^{n} B_{jk}^{-1} F(x_k) \qquad j = 0,1,2,\cdots,n \quad . \tag{5.2.10}$$

Here the solution set $\phi(x_j)$ can be substituted into equation (5.2.7) to directly obtain an interpolation formula for $\phi(x)$ which will have the same degree of precision as the quadrature scheme and is valid for all values of x. Such equations can be solved efficiently by using the appropriate Gaussian quadrature scheme that is required by the limits. In addition, the form of the kernel K(x,t) may influence the choice of the quadrature scheme and it is useful to include as much of the behavior of the kernel in the quadrature weight functions as possible. We could also choose to break the interval $a \to b$ in several pieces depending on the nature of the kernel and what can be guessed about the solution itself. The subsequent quadrature schemes for the sub-intervals will not then depend on the continuity of polynomials from one sub-interval to another and may allow for more accurate approximation in the sub-interval.

For a specific example of the solution to Fredholm equations, let us consider a simple equation of the second type namely

$$y(x) = 1 + x\int_0^1 ty\, dt\,. \tag{5.2.11}$$

Comparing this to equation (5.2.7), we see that F(x) = 1, and that the kernel is separable which leads us immediately to an analytic solution. Since the integral is a definite integral, it may be regarded as some constant α and the solution will be linear of the form

$$y(x) = 1 + \alpha x\int_0^1 t(1+\alpha t)\, dt = 1 + x(\tfrac{1}{2} + \tfrac{\alpha}{3}) \quad . \tag{5.2.12}$$

This leads to a value for α of

$$\alpha = 3/4 \ . \tag{5.2.13}$$

However, had the equation required a numerical solution, then we would have proceeded by replacing the integral by a quadrature sum and evaluating the resulting functional equation at the points of the quadrature. Knowing that the solution is linear, let us choose the quadrature to be Simpson's rule which has a degree of precision high enough to provide an exact answer. The linear equations for the solution become

$$\left.\begin{aligned} y(0) &= 1 + (0)[(0)y(0) + 4(\tfrac{1}{2})y(\tfrac{1}{2}) + y(1)]/6 = 1 \\ y(\tfrac{1}{2}) &= 1 + (\tfrac{1}{2})[(0)y(0) + 4(\tfrac{1}{2})y(\tfrac{1}{2}) + y(1)]/6 = 1 + y(\tfrac{1}{2})/6 + y(1)/12 \\ y(1) &= 1 + (1)[(0)y(0) + 4(\tfrac{1}{2})y(\tfrac{1}{2}) + y(1)]/6 = 1 + y(\tfrac{1}{2})/3 + y(1)/6 \end{aligned}\right\} , \tag{5.2.14}$$

which have the immediate solution

$$\left.\begin{aligned} y(0) &= 1 \\ y(\tfrac{1}{2}) &= \frac{11}{8} \\ y(1) &= \frac{7}{4} \end{aligned}\right\} . \tag{5.2.15}$$

Clearly this solution is in exact agreement with the analytic form corresponding to α=3/4,

$$y(x) = 1 + 3x/4 \ . \tag{5.2.16}$$

While there are variations on a theme for the solution of these type of equations, the basic approach is nicely illustrated by this approach. Now let us turn to the generally more formidable Volterra equations.

c. The Numerical Solution of Volterra Equations

We may approach Volterra equations in much the same way as we did Fredholm equations, but there is the problem that the upper limit of the integral is the independent variable of the equation. Thus we must choose a quadrature scheme that utilizes the endpoints of the interval; otherwise we will not be able to evaluate the functional equation at the relevant quadrature points. One could adopt the view that Volterra equations are, in general, just special cases of Fredholm equations where the kernel is

$$K(x,t) = 0, \ \ t > x \ . \tag{5.2.17}$$

but this would usually require the kernel to be non-analytic However, if we choose such a quadrature formula then, for Volterra equations of type 1, we can write

$$\left.\begin{aligned} F(x_i) &= \sum_{j=0}^{i} K(x_i, x_j)\phi(x_j)W_j \qquad i = 0,1,2,\cdots, n \\ x_k &= a + kh \end{aligned}\right\} \ . \tag{5.2.18}$$

Not only must the quadrature scheme involve the endpoints, it must be an equal interval formula so that successive evaluations of the functional equation involve the points where the function has been previously

evaluated. However, by doing that we obtain a system of n linear equations in (n+1) unknowns. The value of $\phi(a)$ is not clearly specified by the equation and must be obtained from the functional behavior of F(x). One constraint that supplies the missing value of $\phi(x)$ is

$$\phi(a) - K(a,a) = \left.\frac{dF(x)}{dx}\right|_{x=a} . \tag{5.2.19}$$

The value of $\phi(a)$ reduces equations (5.2.18) to a triangular system that can be solved quickly by successive substitution (see section 2.2). The same method can be used for Volterra equations of type 2 yielding

$$\left.\begin{aligned} F(x_i) &= \phi(x_i) + \sum_{j=0}^{i} K(x_i, x_j)\phi(x_j)W_j \qquad i = 0,1,2,\cdots,n \\ x_k &= a + kh \end{aligned}\right\} . \tag{5.2.20}$$

Here the difficulty with $\phi(a)$ is removed since in the limit as $x \to a$

$$\phi(a) = F(a) . \tag{5.2.21}$$

Thus it would appear that type 2 equations are more well behaved that type 1 equations. To the extent that this is true, we may replace any type 1 equation with a type 2 equation of the form

$$F'(x) = K(x,x)\phi(x) + \int_a^x \frac{\partial K(x,t)}{\partial x}\phi(t)\,dt . \tag{5.2.22}$$

Unfortunately we must still obtain F'(x) which may have to be accomplished numerically.

Consider how these direct solution methods can be applied in practice. Let us choose equation (5.1.10), which served so well as a test case for differential equations. In setting that equation up for Picard's method, we turned it into a type 2 Volterra integral equation of the form

$$y(x) = 1 + x\int_0^x ty\,dt . \tag{5.2.23}$$

If we put this in the form suggested by equation (5.2.17) where the kernel vanishes for t > x, we could write

$$1 = y(x) - x\int_0^x ty\,dt = y(x_i) - x_i\sum_{j=0}^{n} t_j y(t_j)W_j , \quad W_j = 0,\ j > i . \tag{5.2.24}$$

Here we have insured that the kernel vanishes for t>x by choosing the quadrature weights to be zero when that condition is satisfied. The resulting linear equations for the solution become

$$\left.\begin{aligned} 1 &= y(0) - [(0)y(0) + 4(0)y(\tfrac{1}{2}) + (0)y(1)]/6 = y(0), \quad i = 1 \\ 1 &= y(\tfrac{1}{2}) - [(0)y(0) + 4(\tfrac{1}{2})y(\tfrac{1}{2}) + (0)y(1)]/6 = 2y(\tfrac{1}{2})/3, \quad i = 2 \\ 1 &= y(1) - [(0)y(0) + 4(\tfrac{1}{2})y(\tfrac{1}{2}) + y(1)]/6 = -y(\tfrac{1}{2})/3 + 5y(1)/6, \quad i = 3 \end{aligned}\right\} . \tag{5.2.25}$$

The method of using equal interval quadrature formulae of varying degrees of precision as x increases is expresses by equation (5.2.18), which for our example takes the form

$$1 = y(x) - x\int_0^x ty\,dt = y(x_i) - \sum_{j=0}^{i} t_j y(t_j)W_j . \tag{5.2.26}$$

This results in linear equations for the solution that are

$$\left.\begin{aligned} 1 &= y(0) - (0) \\ 1 &= y(½) - [(0)y(0) + (½)y(½)]/2 = 3y(½)/4, \\ 1 &= y(1) - [(0)y(0) + 4(½)y(½) + y(1)]/6 = -y(½)/3 + 5y(1)/6 \end{aligned}\right\} . \qquad (5.2.27)$$

The solutions to the two sets of linear equations (5.2.25) and (5.2.27) that represent these two different approaches are given in table 5.5

Table 5.5

Sample Solutions for a Type 2 Volterra Equation

	FREDHOLM SOLN.	TRIANGULAR SOLN.	ANALYTIC SOLN.
y(0)	1.0	1.0	1.0
% Error	0.0%	0.0%	---------
y(½)	1.5	1.333	1.284
% Error	16.8%	3.8%	---------
y(1)	1.8	1.733	2.718
% Error	-33.8%	-36.2%	---------

As with the other examples, we have taken a large step size so as to emphasize the relative accuracy. With the step size again being unity, we get a rather poor result for the rapidly increasing solution. While both method give answers that are slightly larger than the correct answer at x = ½, they rapidly fall behind the exponentially increasing solution by x = 1. As was suggested, the triangular solution is over all slightly better that then Fredholm solution with the discontinuous kernel.

When applying quadrature schemes directly to Volterra equations, we generate a solution with variable accuracy. The quadrature scheme can initially have a degree of precision no greater than one. While this improves as one crosses the interval the truncation error incurred in the first several points accumulates in the solution. This was not a problem with Fredholm equations as the truncation error was spread across the interval perhaps weighted to some degree by the behavior of the kernel. In addition, there is no opportunity to use the highly efficient Gaussian schemes directly as the points of the quadrature must be equally spaced. Thus we will consider an indirect application of quadrature schemes to the solution of both types of integral equations.

By using a quadrature scheme, we are tacitly assuming that the integrand is well approximated by a polynomial. Let us instead assume that the solution itself can be approximated by a polynomial of the form

$$\phi(x_i) = \sum_{j=0}^{n} \alpha_j \xi_j(x) \ . \qquad (5.2.28)$$

Substitution of this polynomial into the integral of either Fredholm or Volterra equations yields

$$\int K(x,t)\phi(t)\,dt = \sum_{j=0}^{n} \alpha_j \int K(x,t)\xi_j(t)\,dt + R = \sum_{j=0}^{n} \alpha_j H_j(x) + R \ . \qquad 5.2.29)$$

Now the entire integrand of the integral is known and may be evaluated to generate the functions $H_j(x)$. It

should be noted that the function $H_j(x)$ will depend on the limits for both classes of equations, but its evaluation poses a separate problem from the solution of the integral equation. In some cases it may be evaluated analytically and in others it will have to be computed numerically for any chosen value of x. However, once that is done, type one equations of both classes can be written as

$$F(x_i) = \sum_{j=0}^{n} \alpha_j H_j(x_i) \quad i=0,1,2, \cdots, n \quad , \tag{5.2.30}$$

which constitute a linear system of (n+1) algebraic equations in the (n+1) unknowns α_j. These, and equation (5.2.28) supply the desired solution $\phi(x)$. Solution for the type 2 equations is only slightly more complicated as equation (5.2.28) must be directly inserted into the integral equation an evaluated at $x=x_i$. However, the resulting linear equations can still be put into standard form so that the α_js can be solved for to generate the solution $\phi(x)$.

We have said nothing about the functions $\xi_j(x)$ that appear in the approximation equation (5.2.28). For nominal polynomial approximation these might be x^j, but for large n such a choice tends to develop instabilities. Thus the same sort of care that was used in developing interpolation formulae should be employed here. One might even wish to employ a rational function approach to approximating $\phi(x)$ as was done in section 3.2. Such care is justified as we have introduced an additional source of truncation error with this approach. Not only will there be truncation error resulting from the quadrature approximation for the entire integral, but there will be truncation error from the approximation of the solution itself [i.e. equation (5.2.28)]. While each of these truncation errors is separately subject to control, their combined effect is less predictable.

Finally, we should consider the feasibility of iterative approaches in conjunction with quadrature schemes for finding solutions to these equations. The type 2 equations immediately suggest an iterative function of the form

$$\phi^{(k)}(x) = F(x) + \lambda \int_a^b K(x,t)\phi^{(k-1)}(t)\,dt \quad . \tag{5.2.31}$$

Remembering that it is $\phi(x)$ that we are after, we can use equation (2.3.20) and the linearity of the integral equations with respect to $\phi(x)$ to establish that the iterative function will converge to a fixed point as long as

$$\left| \lambda \int_a^b K(x,t)\,dt < 1 \right| \quad . \tag{5.2.32}$$

Equation (5.2.17) shows us that a Volterra equation is more likely to converge by iteration than a Fredholm equation with a similar kernel. If λ is small, then not only is the iterative sequence likely to converge, but an initial guess of

$$\phi^{(0)}(x) = F(x) \quad . \tag{5.2.33}$$

suggests itself. In all cases integration required for the iteration can be accomplished by any desireable quadrature scheme as the preliminary value for the solution $\phi^{(k-1)}(x)$ is known.

d. The Influence of the Kernel on the Solution

Although the linearity of the integral operator and its inverse relationship to the differential operator tends to make one think that integral equations are no more difficult than differential equations, there are some subtle differences. For example, one would never attempt a numerical solution of a differential equation that could be shown to have no solution, but that can happen with integral equations if one is not careful. The presence of the kernel under the operator makes the behavior of these equations less transparent than differential equations. Consider the apparently benign kernel

$$K(x,t) = \cos(x)\cos(t) \quad , \tag{5.2.34}$$

and an associated Fredholm equation of the first type

$$F(x) = \int_{-a}^{+a} \cos(x)\cos(t)\phi(t)dt = \cos(x)Z(a) \quad . \tag{5.2.35}$$

Clearly this equation has a solution if and only if F(x) has the form given by the right hand side. Indeed, any kernel that is separable in the independent variables so as to have the form

$$K(x,t) = P(x)Q(t) \quad , \tag{5.2.36}$$

places constraints on the form of F(x) for which the equation has a solution. Nevertheless, it is conceivable that someone could try to solve equation (5.2.35) for functional forms of F(x) other than then those which allow for a value of $\phi(x)$ to exist. Undoubtedly the numerical method would provide some sort of answer. This probably prompted Baker[9], as reported by Craig and Brown[10], to remark 'without care we may well find ourselves computing approximate solutions to problems that have no true solutions'. Clearly the form of the kernel is crucial to nature of the solution, indeed, to its very existence. Should even the conditions imposed on F(x) by equation (5.2.35) be met, any solution of the form

$$\phi(x) = \phi_0(x) + \zeta(x) \quad , \tag{5.2.37}$$

where $\phi_0(x)$ is the initial solution and $\zeta(x)$ is any anti-symmetric function will also satisfy the equation. Not only are we not guaranteed existence, we are not even guaranteed uniqueness when existence can be shown. Fortunately, these are often just mathematical concerns and equations that arise from scientific arguments will generally have unique solutions if they are properly formulated. However, there is always the risk that the formulation will insert the problem in a class with many solutions only one of which is physical. The investigator is then faced with the added problem of finding all the solutions and deciding which ones are physical. That may prove to be more difficult than the numerical problem of finding the solutions.

There are other ways in which the kernel can influence the solution. Craig and Brown[11] devote most of their book to investigating the solution of a class of integral equations which represent inversion problems in astronomy. They show repeatedly that the presence of an inappropriate kernel can cause the numerical methods for the solution to become wildly unstable. Most of their attention is directed to the effects of random error in F(x) on the subsequent solution. However, the truncation error in equation (5.2.3) can combine with F(x) to simulate such errors. The implications are devastating. In Fredholm equations of Type 2, if λ is large and the kernel a weak function of t, then the solution is liable to be extremely unstable. The reason for this can be seen in the role of the kernel in determining the solution $\phi(x)$. K(x,t) behaves like a

filter on the contribution of the solution at all points to the local value of the solution. If K(x,t) is large only for $\tilde{x}t$ then the contribution of the rest of the integral is reduced and $\phi(x)$ is largely determined by the local value of x [i.e. F(x)]. If the Kernel is broad then distant values of $\phi(t)$ play a major role in determining the local value of $\phi(x)$. If λ is large, then the role of F(x) is reduced and the equation becomes more nearly homogeneous. Under these conditions $\phi(x)$ will be poorly determined and the effect of the truncation error on F(x) will be disproportionately large. Thus one should hope for non-separable Kernels that are strongly peaked at x = t.

What happens at the other extreme when the kernel is so strongly peaked at x=t that it exhibits a singularity. Under many conditions this can be accommodated within the quadrature approaches we have already developed. Consider the ultimately peaked kernel

$$K(x,t) = \delta(x-t) \quad , \tag{5.2.38}$$

where $\delta(x)$ is the Dirac delta function. This reduces all of the integral equations discussed here to have solutions

$$\left.\begin{aligned} \phi(x) &= F(x) && \text{type 1} \\ \phi(x) &= F(x)(1-\lambda)^{-1} && \text{type 2} \end{aligned}\right\} \quad . \tag{5.2.39}$$

Thus, even though the Dirac delta function is "undefined" for zero argument, the integrals are well defined and the subsequent solutions simple. For kernels that have singularities at x = t, but are defined elsewhere we can remove the singularity by the simple expedient of adding and subtracting the answer from the integrand so that

$$\phi^{(k)}(x) = F(x) + \lambda\int_a^b K(x,t)[\phi(t) - \phi(x)]\,dt + \lambda\phi(x)\int_a^b K(x,t)\,dt \quad . \tag{5.2.40}$$

We may use the standard quadrature techniques on this equation if the following conditions are met:

$$\left.\begin{aligned} &\left|\int_a^b K(x,t)\,dt\right| < \infty \,, \quad \forall x \\ &\lim_{t\to x}\{K(x,t)[\phi(t) - \phi(x)]\} = 0 \end{aligned}\right\} \quad . \tag{5.2.41}$$

The first of these is a reasonable constraint of the kernel. If that is not met it is unlikely that the solution can be finite. The second condition will be met if the kernel does not approach the singularity faster than linearly and the solution satisfies a Lipshitz condition. Since this is true of all continuous functions, it is likely to be true for any equation that arises from modeling the real world. If this condition is met then the contribution to the quadrature sum from the terms where (i = j) can be omitted (or assigned weight zero). With that slight modification all the previously described schemes can be utilized to solve the resulting equation. Although some additional algebra is required, the resulting linear algebraic equations can be put into standard form and solved using the formalisms from Chapter 2.

In this chapter we have considered the solution of differential and integral equations that arise so often in the world of science. What we have done is but a brief survey. One could devote his or her life to the study of these subjects. However, these techniques will serve the student of science who wishes simply to use them as tools to arrive at an answer. As problems become more difficult, algorithms may need to become more sophisticated, but these fundamentals always provide a good beginning.

Chapter 5 Exercises

1. Find the solution to the following differential equation

   $y' = 3y$,

   in the range $0 \rightarrow 3$. Let the initial value be

   $y(0) = 1$.

   Use the following methods for your solution:

   a. a second order Runge-Kutta
   b. a 3-point predictor-corrector.
   c. Picard's method with 10 steps.
   d. Compare your answer to the analytic solution.

2. Find the solution for the differential equation

   $x^2y'' + xy' + (x^2-6)y = 0$,

   in the range $0 \rightarrow 10$ with initial values of $y'(0)=y(0)=0$. Use any method you like,but explain why it was chosen.

3. Find the numerical solution to the integral equation

   $y(x) = 2 + \int_0^1 y(t)(x^2t+xt^3+5t^5)dt$, $0 \le x \le 2$.

   Comment on the accuracy of your solution and the reason for using the numerical method you chose.

4. Find a closed form solution to the equation in problem 3 of the form

   $y(x) = ax^2 + bx + c$,

   and specifically obtain the values for a,b, and c.

5. How would you have numerically obtained the values for a, b, and c of problem 4 had you only known the numerical solution to problem 3? How would the compare to the values obtained from the closed form solution?

6. We wish to find an approximate solution to the following integral equation:

$$y(x) = 1 + x + 2\int_0^1 t\,x^2\,y(t)\,dt\ .$$

   a. First assume we shall use a quadrature formula with a degree of precision of two where the points of evaluation are specified to be x_1=0.25, x_2=0.5, and x_3=0.75. Use Lagrange interpolation to find the weights for the quadrature formula and use the results to find a system of linear algebraic equations that represent the solution at the quadrature points.

   b. Solve the resulting linear equations by means of Gauss-Jordan elimination and use the results to find a interpolative solution for the integral equation. Comment on the accuracy of the resulting solution over the range $0 \rightarrow \infty$.

7. Solve the following integral equation:

$$B(x) = 1/2\int_0^\infty B(t)E_1\left|\,t-x\,\right|dt\ ,$$

where

$$E_1(x) = \int_0^\infty e^{-xt}dt/t\ .$$

   a. First solve the equation by treating the integral as a Gaussian sum. Note that

$$\lim_{x\rightarrow 0} E_1\left|\,x\,\right| = \infty\ ,$$

   b. Solve the equation by expanding B(t) in a Taylor series about x and thereby changing the integral equation into an nth order linear differential equation. Convert this equation into a system of n first order linear differential equations and solve the system subject to the boundary conditions

$$B(0) = B_0\ ,\ B'(\infty) = B''(\infty) = B^{(n)}(\infty) = 0.$$

Note that the integral equation is a homogeneous equation. Discuss how that influenced your approach to the problem.

Chapter 5 References and Supplemental Reading

1. Hamming, R.W., "Numerical Methods for Scientists and Engineers" (1962) McGraw-Hill Book Co., Inc., New York, San Francisco, Toronto, London, pp. 204-207.

2. Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T., "Numerical Recipies the Art of Scientific Computing" (1986), Cambridge University Press, Cambridge, pp. 569.

3. Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T., "Numerical Recipies the Art of Scientific Computing" (1986), Cambridge University Press, Cambridge, pp. 563-568.

4. Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T., "Numerical Recipies the Art of Scientific Computing" (1986), Cambridge University Press, Cambridge, pp. 563.

5. Day, J.T., and Collins, G.W.,II, "On the Numerical Solution of Boundary Value Problems for Linear Ordinary Differential Equations", (1964), Comm. A.C.M. 7, pp 22-23.

6. Fox, L., "The Numerical Solution of Two-point Boundary Value Problems in Ordinary Differential Equations", (1957), Oxford University Press, Oxford.

7. Sokolnikoff, I.S., and Redheffer, R.M., "Mathematics of Physics and Modern Engineering" (1958) McGraw-Hill Book Co., Inc. New York, Toronto, London, pp. 425-521.

8. Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T., "Numerical Recipies the art of scientific computing" (1986), Cambridge University Press, Cambridge, pp. 615-657.

9. Baker, C.T.N., "The Numerical Treatment of Integral Equations", (1977), Oxford University Press, Oxford.

10. Craig, I.J.D., and Brown, J.C., (1986), "Inverse Problems in Astronomy -A Guide to Inversion Strategies for Remotely Sensed Data", Adam Hilger Ltd. Bristol and Boston, pp. 51.

11 Craig, I.J.D., and Brown, J.C., (1986), "Inverse Problems in Astronomy -A Guide to Inversion Strategies for Remotely Sensed Data", Adam Hilger Ltd. Bristol and Boston.

6

Least Squares, Fourier Analysis, and Related Approximation Norms

• • •

Up to this point we have required that any function we use to represent our 'data' points pass through those points exactly. Indeed, except for the predictor-corrector schemes for differential equations, we have used all the information available to determine the approximating function. In the extreme case of the Runge-Kutta method, we even made demands that exceeded the available information. This led to approximation formulae that were under-determined. Now we will consider approaches for determining the approximating function where some of the information is deliberately ignored. One might wonder why such a course would ever be followed. The answer can be found by looking in two rather different directions.

Remember, that in considering predictor-corrector schemes in the last chapter, we deliberately ignored some of the functional values when determining the parameters that specified the function. That was done to avoid the rapid fluctuations characteristic of high degree polynomials. In short, we felt that we knew something about extrapolating our approximating function that transcended the known values of specific points. One can imagine a number of situations where that might be true. Therefore we ask if there is a general approach whereby some of the functional values can be deliberately ignored when determining the parameters that represent the approximating function. Clearly, anytime the form of the function is known this can be done. This leads directly to the second direction where such an approach will be useful. So far we have treated the functional values that constrain the approximating function as if they were known with absolute precision. What should we do if this is not the case? Consider the situation where the functional values resulted from observation or experimentation and are characterized by a certain amount of error. There would be no reason to demand exact agreement of the functional form at each of the data points. Indeed, in such cases the functional form is generally considered to be known *a priori* and we wish to test some hypothesis by seeing to what extent the imprecise data are represented by the theory. Thus the two different cases for this approach to approximation can be summarized as:

a. *the data is exact but we desire to represent it by an approximating function with fewer parameters than the data.*

b. *the approximating function can be considered to be "exact" and the data which represents that function is imprecise.*

There is a third situation that occasionally arises wherein one wishes to approximate a table of empirically determined numbers which are inherently imprecise and the form of the function must also be assumed. The use of any method in this instance must be considered suspect as there is no way to separate the errors of observation or experimentation from the failure of the assumed function to represent the data.

However, all three cases have one thing in common. They will generate systems that will be over-determined since there will, in general, be more constraining data than there are free parameters in the approximating function. We must then develop some criterion that will enable us to reduce the problem to one that is exactly determined. Since the function is not required to match the data at every point, we must specify by how much it should miss. That criterion is what is known as an *approximation norm* and we shall consider two popular ones, but devote most of our effort to the one known as the *Least Square Norm*.

6.1 Legendre's Principle of Least Squares

Legendre suggested that an appropriate criterion for fitting data points with a function having fewer parameters than the data would be to minimize the square of the amount by which the function misses the data points. However, the notion of a "miss" must be quantified. For least squares, the "miss" will be considered to result from an error in the dependent variable alone. Thus, we assume that there is no error in the independent variable. In the event that each point is as important as any other point, we can do this by minimizing the sum-square of those errors. The use of the square of the error is important for it eliminates the influence of its sign. This is the lowest power dependence of the error ε between the data point and the

approximating function that neglects the sign. Of course one could appeal to the absolute value function of the error, but that function is not continuous and so may produce difficulties as one tries to develop an algorithm for determining the adjustable free parameters of the approximating function.

Least Squares is a very broad principle and has special examples in many areas of mathematics. For example, we shall see that if the approximating functions are sines and cosines that the Principle of Least Squares leads to the determination of the coefficients of a Fourier series. Thus Fourier analysis is a special case of Least Squares. The relationship between Least Squares and Fourier analysis suggests a broad approximation algorithm involving orthogonal polynomials known as the *Legendre Approximation* that is extremely stable and applicable to very large data bases. With this in mind, we shall consider the development of the Principle of Least Squares from several different vantage points.

There are those who feel that there is something profound about mathematics that makes this the "correct" criterion for approximation. Others feel that there is something about nature that makes this the appropriate criterion for analyzing data. In the next two chapters we shall see that there are conditions where the Principle of Least Squares does provide the most probable estimate of adjustable parameters of a function. However, in general, least squares is just one of many possible approximation norms. As we shall see, it is a particularly convenient one that leads to a straightforward determination of the adjustable free parameters of the approximating function.

a. The Normal Equations of Least Squares

Let us begin by considering a collection of N data points (x_i,Y_i) which are to be represented by an approximating function $f(a_j,x)$ so that

$$f(a_j, x_i) = Y_i \ . \tag{6.1.1}$$

Here the (n+1) a_j's are the parameters to be determined so that the sum-square of the deviations from Y_i are a minimum. We can write the deviation as

$$\varepsilon_i = Y_i - f(a_j,x_i) \ . \tag{6.1.2}$$

The conditions that the sum-square error be a minimum are just

$$\frac{\partial \sum_i^N \varepsilon_i^2}{\partial a_i} = 2\sum_{i=1}^{N}[Y_i - f(a_j, x_i)]\frac{\partial f(a_j,x_i)}{\partial a_j} = 0 , \quad j = 0,1, 2,\cdots, n \ . \tag{6.1.3}$$

There is one of these equations for each of the adjustable parameters a_j so that the resultant system is uniquely determined as long as (n+1). N. These equations are known as the *normal equations* for the problem. The nature of the normal equations will be determined by the nature of $f(a_j,x)$. That is, should $f(a_j,x)$ be non-linear in the adjustable parameters a_j, then the normal equations will be non-linear. However, if $f(a_j,x)$ is linear in the a_j's as is the case with polynomials, then the resultant equations will be linear in the a_j's. The ease of solution of such equations and the great body of literature relating to them make this a most important aspect of least squares and one on which we shall spend some time.

b. Linear Least Squares

Consider the approximating function to have the form of a general polynomial as described in chapter 3 [equation (3.1.1)]. Namely

$$f(a_j, x) = \sum_{k=0}^{n} a_k \phi_k(x) \quad \phi \,. \tag{6.1.4}$$

Here the $\phi_k(x)$ are the basis functions which for common polynomials are just x^k. This function, while highly non-linear in the independent variable x is linear in the adjustable free parameters a_k. Thus the partial derivative in equation (6.1.3) is just

$$\frac{\partial f(a_j, x_i)}{\partial a_j} = \phi_j(x_i) \quad , \tag{6.1.5}$$

and the normal equations themselves become

$$\sum_{k=0}^{n} a_k \sum_{i=1}^{N} \phi_k(x_i)\phi_j(x_i) = \sum_{i=1}^{N} Y_i \phi_j(x_i) \,, \quad j = 0, 1, \cdots, n \,. \tag{6.1.6}$$

These are a set of linear algebraic equations, which we can write in component or vector form as

$$\left.\begin{aligned} \sum_k a_k A_{kj} &= C_j \\ \vec{a} \bullet \mathbf{A} &= \vec{C} \end{aligned}\right\} \,. \tag{6.1.7}$$

Since the $\phi_j(x)$ are known, the matrix $\mathbf{A}(x_i)$ is known and depends only on the specific values, x_i, of the independent variable. Thus the normal equations can be solved by any of the methods described in chapter 2 and the set of adjustable parameters can be determined.

There are a number of aspects of the linear normal equations that are worth noting. First, they form a symmetric system of equations since the matrix elements are $\Sigma\phi_k\phi_j$. Since $\phi_j(x)$ is presumed to be real, the matrix will be a *normal* matrix (see section 1.2). This is the origin of the name normal equations for the equations of condition for least squares. Second, if we write the approximating function $f(a_j,x)$ in vector form as

$$f(\vec{a}, x) = \vec{a} \bullet \vec{\phi}(x) \,, \tag{6.1.8}$$

then the normal equations can be written as

$$\vec{a} \bullet \sum_{i=1}^{N} \vec{\phi}(x_i)\vec{\phi}(x_i) = \sum_{i=1}^{N} Y_i \vec{\phi}(x_i) \quad . \tag{6.1.9}$$

Here we have defined a vector $\vec{\phi}(x)$ whose components are the basis functions $\phi_j(x)$. Thus the matrix elements of the normal equations can be generated simply by taking the outer (tensor) product of the basis vector with itself and summing over the values of the vector for each data point. A third way to develop the normal equations is to define a non-square matrix from the basis functions evaluated at the data points x_i as

$$\phi_{ki} = \begin{pmatrix} \phi_0(x_1) & \phi_1(x_1) & \cdots & \phi_n(x_1) \\ \phi_0(x_2) & \phi_1(x_2) & \cdots & \phi_n(x_2) \\ \vdots & \vdots & & \vdots \\ \phi_0(x_n) & \phi_1(x_n) & \cdots & \phi_n(x_n) \end{pmatrix}. \tag{6.1.10}$$

Now we could write an over determined system of equations which we would like to hold as

$$\boldsymbol{\phi}\vec{a} = \vec{Y}. \tag{6.1.11}$$

The normal equations can then be described by

$$[\boldsymbol{\phi}^T\boldsymbol{\phi}]\vec{a} = \boldsymbol{\phi}^T\vec{Y}, \tag{6.1.12}$$

where we take advantage of the matrix product to perform the summation over the data points. Equations (6.1.9) and (6.1.12) are simply different mathematical ways of expressing the same formalism and are useful in developing a detailed program for the generation of the normal equations.

So far we have regarded all of the data points to be of equal value in determining the solution for the free parameters a_j. Often this is not the case and we would like to count a specific point (x_i, Y_i) to be of more or less value than the others. We could simply include it more than once in the summations that lead to the normal equations (6.1.6) or add it to the list of observational points defining the matrix $\boldsymbol{\varphi}$ given by equation (6.1.10). This simplistic approach only yields integral weights for the data points. A far more general approach would simply assign the expression [equation (6.1.1) or equation (6.1.8)] representing the data point a weight ω_i. then equation (6.1.1) would have the form

$$f(\vec{a}, x) = \varpi\vec{a} \bullet \vec{\phi}(x_i) \approx \varpi Y_i \quad . \tag{6.1.13}$$

However, the partial derivative of f will also contain the weight so that

$$\frac{\partial f(\vec{a}, x_i)}{\partial a_j} = \varpi_i \hat{j} \bullet \vec{\phi}(x_i) = \varpi_i \phi_j(x_i) \quad . \tag{6.1.14}$$

Thus the weight will appear quadratically in the normal equations as

$$\sum_{k=0}^{n} a_k \sum_{i=1}^{N} \varpi_i^2 \phi_k(x_i)\phi_j(x_i) = \sum_{i=1}^{N} \varpi_i^2 Y_i \phi_j(x_i), \quad j = 0, 1, \cdots, n \quad . \tag{6.1.15}$$

In order to continually express the weight as a quadratic form, many authors define

$$w_i \equiv \varpi_i^2, \tag{6.1.16}$$

so that the normal equations are written as

$$\sum_{k=0}^{n} a_k \sum_{i=1}^{N} w_i \phi_k(x_i)\phi_j(x_i) = \sum_{i=1}^{N} w_i Y_i \phi_j(x_i), \quad j = 0, 1, \cdots, n \quad . \tag{6.1.17}$$

This simple substitution is often a source of considerable confusion. The weight w_i is the square of the weight assigned to the observation and is of necessity a positive number. One cannot detract from the importance of a data point by assigning a negative weight ϖ_i. The generation of the normal equations would force the square-weight w_i to be positive thereby enhancing the role of that point in determining the solution. Throughout the remainder of this chapter we shall consistently use w_i as the square-weight denoted by equation (6.1.16). However, we shall also use ϖ_i as the individual weight of a given observation. The reader should be careful not to confuse the two.

Once generated, these linear algebraic equations can be solved for the adjustable free parameters by any of the techniques given in chapter 2. However, under some circumstances, it may be possible to produce normal equations which are more stable than others.

c. The Legendre Approximation

In the instance where we are approximating data, either tabular or experimental, with a function of our choice, we can improve the numerical stability by choosing the basis functions $\phi_j(x)$ to be members of orthogonal set. Now the majority of orthogonal functions we have discussed have been polynomials (see section 3.3) so we will base our discussion on orthogonal polynomials. But it should remain clear that this is a convenience, not a requirement. Let $\phi_j(x)$ be an orthogonal polynomial relative to the weight function w(x) over the range of the independent variable x. The elements of the normal equations (6.1.17) then take the form

$$A_{kj} = \sum_{i=1}^{N} w_i \phi_k(x_i)\phi_j(x_i). \tag{6.1.18}$$

If we weight the points in accordance with the weight function of the polynomial, then the weights are

$$w_i = w(x_i) \quad . \tag{6.1.19}$$

If the data points are truly independent and randomly selected throughout the range of x, then as the number of them increases, the sum will approach the value of the integral so that

$$A_{kj} = \lim_{N\to\infty}\left[\sum_{i=1}^{N} w(x_i)\phi_k(x_i)\phi_j(x_i)\right] = N\int w(x)\phi_k(x)\phi_j(x)\,dx = N\delta_{kj} \quad . \tag{6.1.20}$$

This certainly simplifies the solution of the normal equations (6.1.17) as equation (6.1.20) states that the off diagonal elements will tend to vanish. If the basis functions $\phi_j(x)$ are chosen from an *orthonormal* set, then the solution becomes

$$a_j \cong \frac{1}{N}\sum_{i=1}^{N} w(x_i)\phi_j(x_i)Y_i \ , \quad j = 0, 1, \cdots, n \quad . \tag{6.1.21}$$

Should they be merely orthogonal, then the solution will have to be normalized by the diagonal elements leading to a solution of the form

$$a_j \cong \left[\sum_{i=1}^{N} w(x_i)\phi_j(x_i)Y_i\right] \times \left[\sum_{i=1}^{N} w(x_i)\phi_j^2(x_i)\right]^{-1}, \quad j = 0, 1, \cdots, n\,. \tag{6.1.22}$$

The process of using an orthogonal set of functions $\phi_j(x)$ to describe the data so as to achieve the simple result of equations (6.1.21) and (6.1.22) is known as the *Legendre approximation*. It is of considerable utility when the amount of data is vast and the process of forming and solving the full set of normal equations would be too time consuming. It is even possible that in some cases, the solution of a large system of normal equations could introduce greater round-off error than is incurred in the use of the Legendre approximation. Certainly the number of operations required for the evaluation of equations (6.1.21) or (6.1.22) are of the order of (n+1)N where for the formation and solution of the normal equations (6.1.17) themselves something of the order of $(n+1)^2(N+n+1)$ operations are required.

One should always be wary of the time required to carry out a Least Squares solution. It has the habit of growing rapidly and getting out of hand for even the fastest computers. There are many problems where n may be of the order of 10^2 while N can easily reach 10^6. Even the Legendre approximation would imply 10^8 operations for the completion of the solution, while for a full solution of the normal equations 10^{10} operations would need to be performed. For current megaflop machines the Legendre approximation would only take several minutes, while the full solution would require several hours. There are problems that are considerably larger than this example. Increasing either n or N by an order of magnitude could lead to computationally prohibitive problems unless a faster approach can be used. To understand the origin of one of the most efficient approximation algorithms, let us consider the relation of least squares to Fourier analysis.

6.2 Least Squares, Fourier Series, and Fourier Transforms

In this section we shall explicitly explore the relationship between the Principle of least Squares and Fourier series. Then we extend the notion of Fourier series to the Fourier integral and finally to the Fourier transform of a function. Lastly, we shall describe the basis for an extremely efficient algorithm for numerically evaluating a discrete Fourier transform.

a. Least Squares, the Legendre Approximation, and Fourier Series

In section 3.3e we noted that the trigonometric functions sine and cosine formed orthonormal sets in the interval $0 \rightarrow +1$, not only for the continuous range of x but also for a discrete set of values as long as the values were equally spaced. Equation (3.3.41) states that

$$\left.\begin{aligned} &\sum_{i=0}^{N} \sin(k\pi x_i)\sin(j\pi x_i) = \sum_{i=0}^{N} \cos(k\pi x_i)\cos(j\pi x_i) = N\delta_{kj} \\ &x_i = (2i - N)/N\,, \quad i = 0, 1, \cdots N \end{aligned}\right\}\,. \tag{6.2.1}$$

Here we have transformed x into the more familiar interval $-1 \le x \le +1$. Now consider the normal equations that will be generated should the basis functions be either $\cos(j\pi x)$ or $\sin(j\pi x)$ and the data points are spaced in accord with the second of equations (6.2.1). Since the functional sets are orthonormal we may employ the Legendre approximation and go immediately to the solution given by equation (6.1.21) so that the coefficients of the sine and cosine series are

$$\left.\begin{aligned} a_j &= \frac{1}{N+1}\sum_{i=1}^{N} f(x_i)\cos(j\pi x_i) \\ b_j &= \frac{1}{N+1}\sum_{i=1}^{N} f(x_i)\sin(j\pi x_i) \end{aligned}\right\}\,. \tag{6.2.2}$$

Since these trigonometric functions are strictly orthogonal in the interval, as long as the data points are equally spaced, the Legendre approximation is not an approximation. Therefore the equal signs in equations (6.2.2) are strictly correct. The orthogonality of the trigonometric functions with respect to equally spaced data and the continuous variable means that we can replace the summations in equation (6.2.2) with integral

signs without passing to the limit given in equation (6.1.20) and write

$$\left.\begin{aligned} a_j &= \int_{-1}^{+1} f(x)\cos(j\pi x)\,dx \\ b_j &= \int_{-1}^{+1} f(x)\sin(j\pi x)\,dx \end{aligned}\right\}, \tag{6.2.3}$$

which are the coefficients of the *Fourier series*

$$f(x) = \tfrac{1}{2}a_0 + \sum_{k=1}^{\infty} a_k \cos(k\pi x) + b_k \sin(k\pi x) \quad . \tag{6.2.4}$$

Let us pause for a moment to reflect on the meaning of the series given by equation (6.2.4). The function f(x) is represented in terms of a linear combination of periodic functions. The coefficients of these functions are themselves determined by the periodically weighted behavior of the function over the interval. The coefficients a_k and b_k simply measure the periodic behavior of the function itself at the period $(1/\pi k)$. Thus, a Fourier series represents a function in terms of its own periodic behavior. It is as if the function were broken into pieces that exhibit a specific periodic behavior and then re-assembled as a linear combination of the relative strength of each piece. The coefficients are then just the weights of their respective contribution. This is all accomplished as a result of the orthogonality of the trigonometric functions for both the discrete and continuous finite interval.

We have seen that Least Squares and the Legendre approximation lead directly to the coefficients of a finite Fourier series. This result suggests an immediate solution for the series approximation when the data is not equally spaced. Namely, do not use the Legendre approximation, but keep the off-diagonal terms of the normal equations and solve the complete system. As long as N and n are not so large as to pose computational limits, this is a perfectly acceptable and rigorous algorithm for dealing with the problem of unequally spaced data. However, in the event that the amount of data (N) is large there is a further development that can lead to efficient data analysis.

b. The Fourier Integral

The functions that we discussed above were confined to the interval $-1 \rightarrow +1$. However, if the functions meet some fairly general conditions, then we can extend the series approximation beyond that interval. Those conditions are known as the *Dirichlet conditions* which are that the function satisfy *Dirichlet's theorem*. That theorem states:

> *Suppose that* f(x) *is well defined and bounded with a finite number of maxima, minima, and discontinuities in the interval* $-\pi \le x \le +\pi$. *Let* f(x) *be defined beyond this region by* $f(x+2\pi) = f(x)$. *Then the Fourier series for* f(x) *converges absolutely for all* x.

It should be noted that these are sufficient conditions, but not necessary conditions for the convergence of a Fourier series. However, they are sufficiently general enough to include a very wide range of functions which embrace virtually all the functions one would expect to arise in science. We may use these conditions to extend the notion of a Fourier series beyond the interval $-1 \rightarrow +1$.

Let us define

$$z \equiv \xi / x \quad , \tag{6.2.5}$$

where

$$\xi > 1 \quad . \tag{6.2.6}$$

Using Dirichlet's theorem we develop a Fourier series for f(x) in terms of z so that

$$f(z\xi) = \tfrac{1}{2} a_0 + \sum_{k=1}^{\infty} a_k \cos(k\pi z) + b_k \sin(k\pi z) , \tag{6.2.7}$$

implies which will have Fourier coefficients given by

$$\left.\begin{aligned} a_k &= \int_{-1}^{+1} f(z)\cos(k\pi z)\,dz = \frac{1}{\xi}\int_{-\xi}^{+\xi} f(x)\cos(k\pi x/\xi)\,dx \\ b_k &= \int_{-1}^{+1} f(z)\sin(k\pi z)\,dz = \frac{1}{\xi}\int_{-\xi}^{+\xi} f(x)\sin(k\pi x/\xi)\,dx \end{aligned}\right\} . \tag{6.2.8}$$

Making use of the addition formula for trigonometric functions

$$\cos(\alpha-\beta) = \cos\alpha\cos\beta + \sin\alpha\sin\beta , \tag{6.2.9}$$

we can write the Fourier series as

$$f(x) = \frac{1}{2\xi}\int_{-\xi}^{+\xi} f(z)\,dz + \sum_{k=1}^{\infty}\frac{1}{\xi}\int_{-\xi}^{+\xi} f(z)\cos[k\pi(z-x)/\xi]\,dz . \tag{6.2.10}$$

Here we have done two things at once. First, we have passed from a finite Fourier series to an infinite series, which is assumed to be convergent. (i.e. the Dirichlet conditions are satisfied). Second, we have explicitly included the a_k's and b_k's in the series terms. Thus we have represented the function in terms of itself, or more properly, in terms of its periodic behavior. Now we wish to let the infinite summation series pass to its limiting form of an integral. But here we must be careful to remember what the terms of the series represent. Each term in the Fourier series constitutes the contribution to the function of its periodic behavior at some discrete period or frequency. Thus, when we pass to the integral limit for the series, the integrand will measure the frequency dependence of the function. The integrand will itself contain an integral of the function itself over space. Thus this process will transform the representation of the function from its behavior in frequency to its behavior in space. Such a transformation is known as a *Fourier Transformation*.

c. The Fourier Transform

Let us see explicitly how we can pass from the discrete summation of the Fourier series to the integral limit. To do this, we will have to represent the frequency dependence in a continuous way. This can be accomplished by allowing the range of the function (i.e. $-\xi \to +\xi$) to be variable. Let

$$\delta\alpha = 1/\xi \quad , \tag{6.2.11}$$

so that each term in the series becomes

$$\frac{1}{\xi}\int_{-\xi}^{+\xi} f(z)\cos[k\pi(z-x)/\xi]\,dz = \delta\alpha\int_{-\xi}^{+\xi} f(z)\cos[(k\delta\alpha)\pi(z-x)/\xi]\,dz \quad . \tag{6.2.12}$$

Now as we pass to the limit of letting $\delta\alpha \to 0$, or $\xi \to \infty$, each term in the series will be multiplied by an

infinitesimal dα, and the limits on the term will extend to infinity. The product kδα will approach the variable of integration α so that

$$\lim_{\substack{\delta\alpha\to 0\\ \xi\to\infty}} \sum_{k=1}^{\infty}\left[\int_{-\xi}^{+\xi} f(z)\cos[(k\delta\alpha)\pi(z-x)/\xi]\,dz\right] = \int_0^{\infty}\left[\int_{-\xi}^{+\xi} f(z)\cos[(k\delta\alpha)\pi(z-x)/\xi]\,dz\right]d\alpha \quad . \tag{6.2.13}$$

The right hand side of equation 6.2.13 is known as the *Fourier integral* which allows a function f(x) to be expressed in terms of its frequency dependence f(z). If we use the trigonometric identity (6.2.9) to re-express the Fourier integrals explicitly in terms of their sine and cosine dependence on z we get

$$\left.\begin{aligned} f(x) &= 2\int_0^{+\infty}\int_0^{+\infty} f(z)\sin(\alpha\pi z)\sin(\alpha\pi x)\,dz \\ f(x) &= 2\int_0^{+\infty}\int_0^{+\infty} f(z)\cos(\alpha\pi z)\cos(\alpha\pi x)\,dz \end{aligned}\right\} \quad . \tag{6.2.14}$$

The separate forms of the integrals depend on the symmetry of f(x). Should f(x) be an odd function, then it will cancel from all the cosine terms and produce only the first of equations (6.2.14). The second will result when f(x) is even and the sine terms cancel.

Clearly to produce a representation of a general function f(x) we shall have to include both the sine and cosine series. There is a notational form that will allow us to do that using complex numbers known as Euler's formula

$$e^{ix} = \cos(x) + i\sin(x) \quad . \tag{6.2.15}$$

This yields an infinite Fourier series of the form

$$\left.\begin{aligned} f(x) &= \sum_{k=-\infty}^{+\infty} C_k e^{ikx} \\ C_k &= \tfrac{1}{2}\int_{-1}^{+1} f(t)e^{-i\pi k t}\,dt \end{aligned}\right\} \quad , \tag{6.2.16}$$

where the complex constants C_k are related to the a_k's and b_k's of the cosine and sine series by

$$\left.\begin{aligned} C_0 &= a_0/2 \\ C_{+k} &= a_k/2 \;-\; ib_k/2 \\ C_{-k} &= a_k/2 \;+\; ib_k/2 \end{aligned}\right\} \quad . \tag{6.2.17}$$

We can extend this representation beyond the interval $-1 \rightarrow +1$ in the same way we did for the Fourier Integral. Replacing the infinite summation by an integral allows us to pass to the limit and get

$$f(x) = \int_{-\infty}^{+\infty} e^{2\pi i x z} F(z)\,dz \quad , \tag{6.2.18}$$

where

$$F(z) = \int_{-\infty}^{+\infty} f(t)e^{-2\pi i z t}\,dt \equiv T(f) \, . \tag{6.2.19}$$

The integral T(f) is known as the *Fourier Transform* of the function f(x). It is worth considering the transform of the function f(t) to simply be a different representation of the same function since

$$\left.\begin{aligned} F(z) &= \int_{-\infty}^{+\infty} f(t)e^{-2\pi i z t}\,dt = T(f) \\ f(t) &= \int_{-\infty}^{+\infty} F(z)e^{+2\pi i z t}\,dt = T(F) = T^{-1}(f) \end{aligned}\right\}. \qquad (6.2.20)$$

The second of equations (6.2.20) reverses the effect of the first, [i.e.$T(f)\times T^{-1}(f) = 1$] so the second equation is known as the *inverse Fourier transform*.

The Fourier transform is only one of a large number of integrals that transform a function from one space to another and whose repeated application regenerates the function. Any such integral is known as an *integral transform*. Next to the Fourier transform, the best known and most widely used integral transform is the Laplace transform $\mathfrak{L}(f)$ which is defined as

$$\mathfrak{L}(f) = \int_0^{\infty} f(t)e^{-pt}\,dt \quad . \qquad (6.2.21)$$

For many forms of f(t) the integral transforms as defined in both equations (6.2.20) and (6.2.21) can be expressed in closed form which greatly enhances their utility. That is, given an analytic closed-form expression for f(t), one can find analytic closed-form expression for T(f) or $\mathfrak{L}(f)$. Unfortunately the expression of such integrals is usually not obvious. Perhaps the largest collection of integral transforms, not limited to just Fourier and Laplace transforms, can be found among the Bateman Manuscripts[1] where two full volumes are devoted to the subject.

Indeed, one must be careful to show that the transform actually exists. For example, one might believe from the extremely generous conditions for the convergence of a Fourier series, that the Fourier transform must always exist and there are those in the sciences that take its existence as an axiom. However, in equation (6.2.13) we passed from a finite interval to the full open infinite interval. This may result in a failure to satisfy the Dirichlet conditions. This is the case for the basis functions of the Fourier transform themselves, the sines and cosines. Thus sin(x) or cos(x) will not have a discrete Fourier transform and that should give the healthy skeptic pause for thought. However, in the event that a closed form representation of the integral transform cannot be found, one must resort to a numerical approach which will yield a discrete Fourier transform. After establishing the existence of the transform, one may use the very efficient method for calculating it known as the Fast Fourier Transform Algorithm.

d. The Fast Fourier Transform Algorithm

Because of the large number of functions that satisfy Dirichlet's conditions, the Fourier transform is one of the most powerful analytic tools in science and considerable effort has been devoted to its evaluation. Clearly the evaluation of the Fourier transform of a function f(t) will generally be accomplished by approximating the function by a Fourier series that covers some finite interval. Therefore, let us consider a finite interval of range t_0 so that we can write the transform as

$$F(z_k) = \int_{-\infty}^{+\infty} f(t)e^{-2\pi i z_k t}\,dt = \int_{-t_0/2}^{+t_0/2} f(t)e^{2\pi i z_k t}\,dt = \sum_{j=0}^{N-1} f(t_j)\,e^{2\pi i z_k t}\,W_j \quad . \qquad (6.2.22)$$

In order to take advantage of the orthogonality of the sines and cosines over a discrete set of equally spaced data the quadrature weights W_i in equation (6.2.22) will all be taken to be equal and to sum to the range of the integral so that

$$W_i = t_0 / N = t(N)/N \equiv \delta \quad . \tag{6.2.23}$$

This means that our discrete Fourier transform can be written as

$$F(z_k) = \delta \sum_{j=0}^{N-1} f(t_j)\, e^{2\pi i z(j\delta)} \quad . \tag{6.2.24}$$

In order for the units to yield a dimensionless exponent in equation (6.2.24), $z \sim t^{-1}$. Since we are determining a *discrete* Fourier transform, we will choose a discrete set of point z_k so that

$$z_k = \pm k/t(N) = \pm\, k/(N\delta) \quad , \tag{6.2.25}$$

and the discrete transform becomes

$$F(z_k) = \delta F_k = \delta \sum_{j=0}^{N-1} f(t_j)\, e^{2\pi i\,(kj/N)} \quad . \tag{6.2.26}$$

To determine the Fourier transform of f(x) is to find N values of F_k. If we write equation (6.2.26) in vector notation so that

$$\left.\begin{aligned} \vec{F} &= \mathbf{E} \bullet \vec{f} \\ E_{kj} &= e^{2\pi i(kj/N)} \end{aligned}\right\} \quad . \tag{6.2.27}$$

It would appear that to find the N components of the vector $\vec{F}(x)$ we would have to evaluate a matrix **E** having N^2 complex components. The resulting matrix multiplication would require N^2 operations. However, there is an approach that yields a Fourier Transform in about $N\log_2 N$ steps known as the *Fast Fourier Transform algorithm* or FFT for short. This tricky algorithm relies on noticing that we can write the discrete Fourier transform of equation (6.2.26) as the sum of two smaller discrete transform involving the even and odd points of the summation. Thus

$$\begin{aligned} F_k &= \sum_{j=0}^{N-1} f(t_j) e^{2\pi i(kj/N)} = \sum_{j=0}^{N/2-1} f(t_{2j}) e^{2\pi i(kj/N)} + \sum_{j=0}^{N/2-1} f(t_{2j}) e^{2\pi i(kj/N)} \\ &= \sum_{j=0}^{N/2-1} f(t_{2j}) e^{2\pi i(kj/N)} + \; e^{2\pi i(kj/N)} \sum_{j=0}^{N/2-1} f(t_{2j+1}) e^{2\pi i(kj/N)} = F_k^{(0)} + Q_k F_k^{(1)} \end{aligned} \quad . \tag{6.2.28}$$

If we follow the argument of Press et. al.[2], we note that each of the transforms involving half the points can themselves be subdivided into two more. We can continue this process until we arrive at sub-transforms containing but a single term. There is no summation for a one-point transform so that it is simply equal to a particular value of $f(t_k)$. One need only identify which sub-transform is to be associated with which point. The answer, which is what makes the algorithm practical, is contained in the order in which a sub-transform is generated. If we denote an even sub-transform at a given level of subdivision by a superscript 0 and an odd one by a superscript of 1, the sequential generation of sub-transforms will generate a series of binary digits unique to that sub-transform. The binary number represented by the *reverse order* of those digits is the binary representation of i denoting the functional value $f(t_i)$. Now re-sort the points so that they are ordered sequentially on this new binary subscript say p. Each $f(t_p)$ represents a one point sub-transform which we can combine via equation (6.2.28) with its adjacent neighbor to form a two point sub-transform. There will of course be N of these. These can be combined to form N four-point sub-transforms and so on until the N values of the final transform are generated. Each step of combining transforms will take on the order of N operations. The process of breaking the original transform down to one-point

transforms will double the number of transforms at each division. Thus there will be m sub-divisions where

$$2^m = N \ , \tag{6.2.29}$$

so that

$$m = \mathrm{Log}_2 N \ . \tag{6.2.30}$$

Therefore the total number of operations in this algorithm will be of the order of $N\log_2 N$. This clearly suggests that N had better be a power of 2 even if it is necessary to interpolate some additional data. There will be some additional computation involved in the calculation in order to obtain the Q_k's, carry out the additions implied by equation (6.1.46), and perform the sorting operation. However, it is worth noting that at each subdivision, the values of Q_k are related to their values from the previous subdivision $e^{2k\pi i/N}$ for only the length of the sub-transform, and hence N, has changed. With modern efficient sorting algorithms these additional tasks can be regarded as negligible additions to the entire operation. When one compares N^2 to $N\log_2 N$ for $N \sim 10^6$, then the saving is of the order of 5×10^4. Indeed, most of the algorithm can be regarded as a bookkeeping exercise. There are extremely efficient packages that perform FFTs. The great speed of FFTs has lead to their wide spread use in many areas of analysis and has focused a great deal of attention on Fourier analysis. However, one should always remember the conditions for the validity of the discrete Fourier analysis. The most important of these is the existence of equally space data.

The speed of the FFT algorithm is largely derived from the repetitive nature of the Fourier Transform. The function is assumed to be represented by a Fourier Series which contains only terms that repeat outside the interval in which the function is defined. This is the essence of the Dirichlet conditions and can be seen by inspecting equation (6.2.28) and noticing what happens when k increases beyond N. The quantity $e^{2\pi ijk/N}$ simply revolves through another cycle yielding the periodic behavior of F_k. Thus when values of a sub-transform F_k^o are needed for values of k beyond N, they need not be recalculated.

Therefore the basis for the FFT algorithm is a systematic way of keeping track if the booking associated with the generation of the shorter sub-transforms. By way of an example, let us consider the discrete Fourier transform of the function

$$f(t) = e^{-|t|} \ . \tag{6.2.31}$$

We shall consider representing the function over the finite range ($-½t_0 \rightarrow +½t_0$) where $t_0 = 4$. Since the FFT algorithm requires that the calculation be carried out over a finite number of points, let us take 2^3 points to insure a sufficient number of generations to adequately demonstrate the subdivision process. With these constraints in mind the equation (6.2.22) defining the discrete Fourier Transform becomes

$$F(z) = \int_{-t_0/2}^{+t_0/2} f(t) e^{+2\pi i t z}\, dt = \int_{-2}^{+2} e^{+2\pi i t z - |t|}\, dt = \sum_{j=0}^{7} e^{-|t_j|} e^{2\pi i t_j z}\, W_j \ . \tag{6.2.32}$$

We may compare the discrete transform with the Fourier Transform for the full infinite interval (i.e. $-\infty \rightarrow +\infty$) as the integral in equation (6.2.32) may be expressed in closed form so that

$$F[f(t)] = F(z) = 2/[1+(2\pi \left| z \right|)] \ . \tag{6.2.33}$$

The results of both calculations are summarized in table 6.1. We have deliberately chosen an even function of t as the Fourier transform will be real and even. This property is shared by both the discrete and continuous transforms. However, there are some significant differences between the continuous transform

for the full infinite interval and the discrete transform. While the maximum amplitude is similar, the discrete transform oscillates while the continuous transform is monotonic. The oscillation of the discrete transform results from the truncation of the function at ½t_0. To properly describe this discontinuity in the function a larger amplitude for the high frequency components will be required. The small number of points in the transform exacerbates this. The absence of the higher frequency components that would be specified by a larger number of points forces their influence into the lower order terms leading to the oscillation. In spite of this, the magnitude of the transform is roughly in accord with the continuous transform. Figure 6.1 shows the comparison of the discrete transform with the full interval continuous transform. We have included a dotted line connecting the points of the discrete transform to emphasize the oscillatory nature of the transform, but it should be remembered that the transform is only defined for the discrete set of points z_k.

Table 6.1

Summary Results for a Sample Discrete Fourier Transform

I	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
$\mathbf{t_i}$	-2.0000	-1.5000	-1.0000	-0.5000	0.0000	+0.5000	+1.0000	+1.5000
$\mathbf{f(t_i)}$	0.1353	0.2231	0.3678	0.6065	1.0000	0.6065	0.3678	0.2231
k	0	1	2	3	4	5	6	7
$\mathbf{z_k}$	0.0000	+0.2500	+0.5000	+0.7500	+1.0000	-0.7500	-0.5000	-0.2500
$\mathbf{F(z_k)}$	+1.7648	-0.7010	+0.2002	-0.1613	+0.1056	-0.1613	+0.2002	-0.7010
$\mathbf{F_c(z_k)}$	+2.0000	+0.5768	+0.1840	+0.0863	+0.0494	+0.0863	0.1840	+0.5768

While the function we have chosen is an even function of t, we have not chosen the points representing that function symmetrically in the interval (−½t_0 → +½t_0). To do so would have included the each end point, but since the function is regarded to be periodic over the interval, the endpoints would not be linearly independent and we would not have an additionally distinct point. In addition, it is important to include the point t = 0 in the calculation of the discrete transform and this would be impossible with 2^m points symmetrically spaced about zero.

Let us proceed with the detailed implementation of the FFT. First we must calculate the weights W_j that appear in equation (6.2.22) by means of equation (6.2.23) so that

$$W_j = \delta = 4/2^3 = 1/2 \quad . \tag{6.2.34}$$

The first sub-division into sub-transforms involving the even and odd terms in the series specified by equation (6.2.22) is

$$F_k = \delta(F_k^0 + Q_k^1 F_k^1) \quad . \tag{6.2.35}$$

The sub-transforms specified by equation (6.2.35) can be further subdivided so that

$$\left.\begin{aligned} F_k^0 &= (F_k^{00} + Q_k^2 F_k^{01}) \\ F_k^1 &= (F_k^{10} + Q_k^2 F_k^{11}) \end{aligned}\right\} \quad . \tag{6.2.36}$$

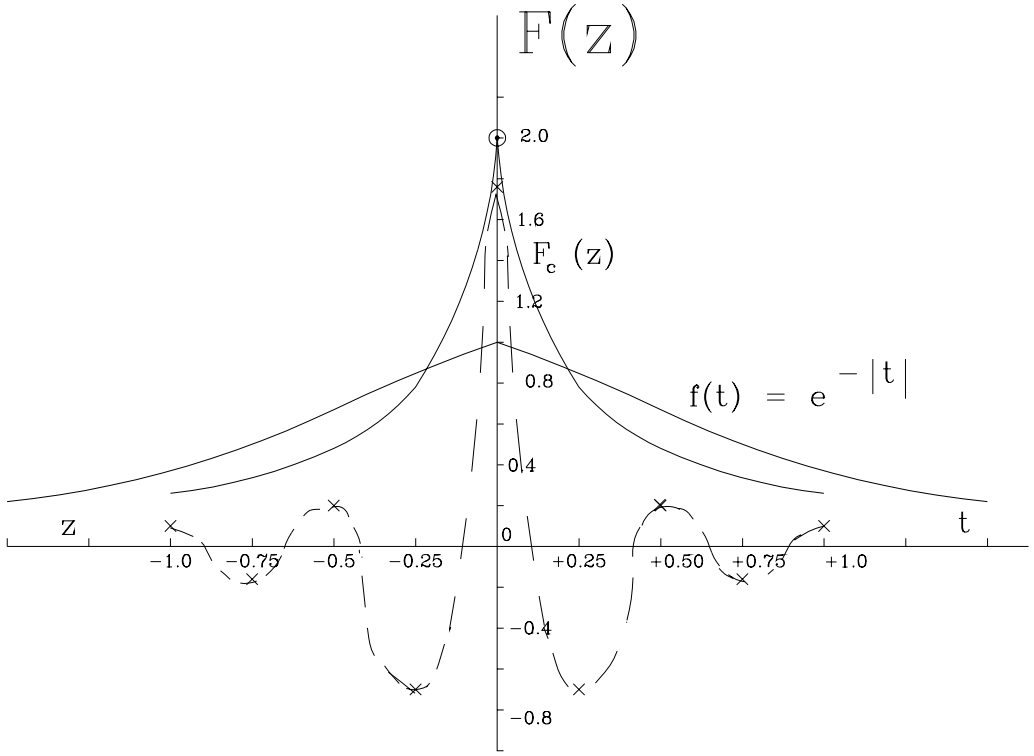


Figure 6.1 compares the discrete Fourier transform of the function $e^{-|x|}$ *with the continuous transform for the full infinite interval. The oscillatory nature of the discrete transform largely results from the small number of points used to represent the function and the truncation of the function at* $t = \pm 2$. *The only points in the discrete transform that are even defined are denoted by* $\times$ *, the dashed line is only provided to guide the reader's eye to the next point.*

The final generation of sub-division yields

$$\left.\begin{aligned} F_k^{00} &= (F_k^{000} + Q_k^3 F_k^{001}) = f_0 + Q_k^3 f_4 \\ F_k^{01} &= (F_k^{010} + Q_k^3 F_k^{011}) = f_2 + Q_k^3 f_6 \\ F_k^{10} &= (F_k^{100} + Q_k^3 F_k^{101}) = f_1 + Q_k^3 f_5 \\ F_k^{11} &= (F_k^{110} + Q_k^3 F_k^{111}) = f_3 + Q_k^3 f_7 \end{aligned}\right\}, \tag{6.2.37}$$

where

$$\left.\begin{aligned} Q_k^n &= (e^{2\pi i k / N_n})^n \\ N_n &= N / 2^{(n-1)} \\ f_j &= f(t_j) \end{aligned}\right\}. \tag{6.2.38}$$

Here we have used the "bit-reversal" of the binary superscript of the final sub-transforms to identify which of the data points $f(t_j)$ correspond to the respective one-point transforms. The numerical details of the calculations specified by equations (6.2.35) - (6.2.38) are summarized in Table 6.2.

Here we have allowed k to range from $0 \rightarrow 8$ generating an odd number of resultant answers. However, the values for k = 0 and k = 8 are identical due to the periodicity of the function. While the symmetry of the initial function $f(t_j)$ demands that the resultant transform be real and symmetric, some of the sub-transforms may be complex. This can be seen in table 6.2 in the values of $F^1_{y1,3,5,7}$. They subsequently cancel, as they must, in the final transform F_k, but their presence can affect the values for the real part of the transform. Therefore, complex arithmetic must be used throughout the calculation. As was already mentioned, the sub-transforms become more rapidly periodic as a function of k so that fewer and fewer terms need be explicitly kept as the subdivision process proceeds. We have indicated this by highlighting the numbers in table 6.2 that must be calculated. While the tabular numbers represent values that would be required to evaluate equation (6.2.22) for any specific value of k, we may use the repetitive nature of the sub-transforms when calculating the Fourier transform for all values of k. The highlighted numbers of table 6.2 are clearly far fewer that N^2 confirming the result implied by equation (6.2.30) that $N\log_2 N$ operations will be required to calculate that discrete Fourier transform. While the saving is quite noticeable for N = 8, it becomes monumental for large N.

The curious will have noticed that the sequence of values for z_k does not correspond with the values of t_j. The reason is that the particular values of k that are used are somewhat arbitrary as the Fourier transform can always be shifted by $e^{2\pi im/N}$ corresponding to a shift in k by +m. This simply moves on to a different phase of the periodic function F(z). Thus, our tabular values begin with the center point z=0, and moves to the end value of +1 before starting over at the negative end value of -0.75 (note that -1 is to be identified with +1 due to the periodicity of F_k). While this cyclical ranging of k seems to provide an endless set of values of F_k, there are only N distinctly different values because of the periodic behavior of F_k. Thus our original statement about the nature of the discrete Fourier transform - that it is defined only at a discrete set of points - remains true.

As with most subjects in this book, there is much more to Fourier analysis than we have developed here. We have not discussed the accuracy of such analysis and its dependence on the sampling or amount of the initial data. The only suggestion for dealing with data missing from an equally spaced set was to interpolate the data. Another popular approach is to add in a "fake" piece of data with $f(t_j) = 0$ on the grounds that it makes no direct contribution to the sums in equation (6.2.28). This is a deceptively dangerous argument as there is an implicit assumption as to the form of the function at that point. Interpolation, as long as it is not excessive, would appear to be a better approach.

Table 6.2

Calculations for a Sample Fast Fourier Transform

K	f_k	$F_k^{000}=f_0$	$F_k^{001}=f_4$	$F_k^{010}=f_2$	$F_k^{011}=f_6$	$F_k^{100}=f_1$	$F_k^{101}=f_5$	$F_k^{110}=f_3$	$F_k^{111}=f_7$
0	**0.1353**	**0.1353**	**1.0000**	**0.3678**	**0.3678**	**0.2231**	**0.6065**	**0.6065**	**0.2231**
1	0.1353	0.1353	1.0000	0.3678	0.3678	0.2231	0.6065	0.6065	0.2231
2	0.1353	0.1353	1.0000	0.3678	0.3678	0.2231	0.6065	0.6065	0.2231
3	0.1353	0.1353	1.0000	0.3678	0.3678	0.2231	0.6065	0.6065	0.2231
4	0.1353	0.1353	1.0000	0.3678	0.3678	0.2231	0.6065	0.6065	0.2231
5	0.1353	0.1353	1.0000	0.3678	0.3678	0.2231	0.6065	0.6065	0.2231
6	0.1353	0.1353	1.0000	0.3678	0.3678	0.2231	0.6065	0.6065	0.2231
7	0.1353	0.1353	1.0000	0.3678	0.3678	0.2231	0.6065	0.6065	0.2231
8	0.1353	0.1353	1.0000	0.3678	0.3678	0.2231	0.6065	0.6065	0.2231

k	Q_k^1	F_k^{00}	F_k^{01}	F_k^{10}	F_k^{11}	Q_k^2	F_k^0	F_k^1	Q_k^3	F_k	z_k
0	**+1**	**1.1353**	**0.7350**	**0.8296**	**0.8296**	**+1**	**1.8703**	**1.6592**	**+1**	**1.7648**	0.00
1	**-1**	**-.8647**	**0.0000**	**-.3834**	**-.3834**	**+i**	**-.8647**	**-.3834**	$(1+i)/\sqrt{2}$	**-.7010**	0.25
							0.0000 i	**+.3834 i**			
2	+1	1.1353	0.7350	0.8296	0.8296	-1	**0.4003**	**0.0000**	**+i**	**0.2002**	0.50
3	-1	-.8647	0.0000	-.3834	-.3834	-i	**-.8647**	**-.3834**	$(i-1)/\sqrt{2}$	**-.1613**	0.75
							0.0000 i	**-.3834i**			
4	+1	1.1353	0.7350	0.8296	0.8296	+1	1.8703	1.6592	**-1**	**0.1056**	1.00
5	-1	-.8647	0.0000	-.3834	-.3834	+i	-.8647	-.3834	$(1+i)/\sqrt{2}$	**-.1613**	-0.75
							0.0000 i	+.3834i			
6	+1	1.1353	0.7350	0.8296	0.8296	-1	0.4003	0.0000	**-i**	**0.2002**	-0.50
7	-1	-.8647	0.0000	-.3834	-.3834	-i	-.8647	-.3834	$(i-1)/\sqrt{2}$	**-.7010**	-0.25
							0.0000 i	-.3834i			
8	+1	1.1353	0.7350	0.8296	0.8296	+1	1.8703	1.6592	+1	1.7648	0.00

6.3 Error Analysis for Linear Least-Squares

While Fourier analysis can be used for basic numerical analysis, it is most often used for observational data analysis. Indeed, the widest area of application of least squares is probably the analysis of observational data. Such data is intrinsically flawed. All data, whether it results from direct observation of the natural world or from the observation of a carefully controlled experiment, will contain errors of observation. The equipment used to gather the information will have characteristics that limit the accuracy of that information. This is not simply poor engineering, but at a very fundamental level, the observing equipment is part of the phenomenon and will distort the experiment or observation. This, at least, is the view of modern quantum theory. The inability to carry out precise observations is a limit imposed by the very nature of the physical world. Since modern quantum theory is the most successful theory ever devised by man, we should be mindful of the limits it imposes on observation. However, few experiments and observational equipment approach the error limits set by quantum theory. They generally have their accuracy set by more practical aspects of the research. Nevertheless observational and experimental errors are always with us so we should understand their impact on the results of experiment and observation. Much of the remaining chapters of the book will deal with this question in greater detail, but for now we shall estimate the impact of observational errors on the parameters of least square analysis. We shall give this development in some detail for it should be understood completely if the formalism of least squares is to be used at all.

a. Errors of the Least Square Coefficients

Let us begin by assuming that the approximating function has the general linear form of equation (6.1.4). Now we will assume that each observation Y_i has an unspecified error E_i associated with it which, if known, could be corrected for, yielding a set of least square coefficients a_j^0. However, these are unknown so that our least square analysis actually yields the set of coefficients a_j. If we knew both sets of coefficients we could write

$$\left.\begin{aligned} E_i &= Y_i - \sum_{j=0}^{n} a_j^0 \phi_j(x_i) \\ \varepsilon_i &= Y_i - \sum_{j=0}^{n} a_j \phi_j(x_i) \end{aligned}\right\} . \tag{6.3.1}$$

Here ε_i is the normal residual error resulting from the standard least square solution.

In performing the least square analysis we weighted the data by an amount ω_i so that

$$\sum_{i=1}^{N} (\omega_i \varepsilon_i)^2 = \text{Minimum}. \tag{6.3.2}$$

We are interested in the error in a_j resulting from the errors E_i in Y_i so let us define

$$\delta a_j \equiv a_j - a_j^0 . \tag{6.3.3}$$

We can multiply the first of equations (6.3.1) by $\omega_i^2\varphi_k(x_i)$, sum over i, and get

$$\sum_{j=0}^{N} a_j^0 \sum_{i=1}^{N} \omega_i^2 \phi_j(x_i)\phi_k(x_i) = \sum_{i=1}^{N} \omega_i^2 Y_i \phi_k(x_i) - \sum_{i=1}^{N} \omega_i^2 \phi_k(x_i) E_i \ , \quad k = 0,1,\cdots, n \ , \tag{6.3.4}$$

while the standard normal equations of the problem yield

$$\sum_{j=0}^{N} a_j \sum_{i=1}^{N} \omega_i^2 \phi_j(x_i)\phi_k(x_i) = \sum_{i=1}^{N} \omega_i^2 Y_i \phi_k(x_i), \quad k = 0,1,\cdots, n \ . \tag{6.3.5}$$

If we subtract equation (6.3.4) from equation (6.3.5) we get an expression for δa_j.

$$\sum_{j=0}^{N} \delta a_j \sum_{i=1}^{N} w_i \phi_j(x_i)\phi_k(x_i) = \sum_{j=0}^{n} \delta a_j A_{jk} = \sum_{i=1}^{N} w_i \phi_k(x_i) E_i \ , \quad k = 0,1,\cdots, n \ . \tag{6.3.6}$$

Here we have replace ω^2_i with w_i as in section 1 [equation (6.1.16)]. These linear equations are basically the normal equations where the errors of the coefficients δa_j have replaced the least square coefficients a_j, and the observational errors E_i have replace the dependent variable Y_i. If we knew the individual observational errors E_i, we could solve them explicitly to get

$$\delta a_j = \sum_{k=0}^{n} [A_{jk}]^{-1} \sum_{i=1}^{N} w_i \phi_k(x_i) E_i \ , \tag{6.3.7}$$

and we would know precisely how to correct our standard answers a_j to get the "true" answers a^0_j. Since we do not know the errors E_i, we shall have to estimate them in terms of ε_i , which at least is knowable.

Unfortunately, in relating E_i to ε_i it will be necessary to lose the sign information on δa_j. This is a small price to pay for determining the magnitude of the error. For simplicity let

$$\mathbf{C} = \mathbf{A}^{-1} \ . \tag{6.3.8}$$

We can then square equation (6.3.7) and write

$$\begin{aligned} (\delta a_j)^2 &= \left[\sum_{k=0}^{n} C_{jk} \sum_{i=1}^{N} w_i \phi_k(x_i) E_i\right]\left[\sum_{p=0}^{n} C_{jp} \sum_{q=1}^{N} w_q \phi_p(x_q) E_q\right] \\ &= \sum_{k=0}^{n}\sum_{p-0}^{n} C_{jk} C_{jp} \sum_{i=1}^{N}\sum_{q=1}^{N} w_i w_q \phi_k(x_i)\phi_p(x_q) E_i E_q \end{aligned} \ . \tag{6.3.9}$$

Here we have explicitly written out the product as we will endeavor to get rid of some of the terms by making reasonable assumptions. For example, let us specify the manner in which the weights should be chosen so that

$$\omega_i E_i = \text{const.} \tag{6.3.10}$$

While we do not know the value of E_i, in practice, one usually knows something about the expected error distribution. The value of the constant in equation (6.3.10) doesn't matter since it will drop out of the normal equations. Only the distribution of E_i matters and the data should be weighted accordingly.

We shall further assume that the error distribution of E_i is anti-symmetric about zero. This is a less justifiable assumption and should be carefully examined in all cases where the error analysis for least squares

is used. However, note that the distribution need only be anti-symmetric about zero, it need not be distributed like a Gaussian or normal error curve, since both the weights and the product $\phi(x_i)\ \phi(x_q)$ are symmetric in i and q. Thus if we chose a negative error, say, E_q to be paired with a positive error, say, E_i we get

$$\sum_{\substack{i=1\\ i\neq q}}^{N}\sum_{q=1}^{N} w_i w_q \phi_k(x_i)\phi_p(x_q)E_i E_q = 0\ , \quad \forall\, k = 0, 1, \cdots, n \quad , \quad p = 0, 1, \cdots, n\ . \tag{6.3.11}$$

Therefore only terms where i=q survive in equation (6.3.9) and we may write it as

$$(\delta a_j)^2 = \overline{(\omega E)^2}\sum_{k=0}^{n} C_{jk}\sum_{p=0}^{n} C_{jp}\sum_{i=1}^{N} w_i \phi_k(x_i)\phi_p(x_i) = \overline{(\omega E)^2}\sum_{k=0}^{n} C_{jk}\left[\sum_{p=0}^{n} C_{jp}A_{pk}\right]\ . \tag{6.3.12}$$

Since $\mathbf{C}=\mathbf{A}^{-1}$ [i.e. equation (6.3.8)], the term in large brackets on the far right-hand-side is the Kronecker delta δ_{jk} and the expression for $(\delta a_j)^2$ simplifies to

$$(\delta a_j)^2 = \overline{(\omega E)^2}\sum_{k=0}^{n} C_{jk}\delta_{jk} = \overline{(\omega E)^2}C_{jj}\ . \tag{6.3.13}$$

The elements C_{jj} are just the diagonal elements of the inverse of the normal equation matrix and can be found as a by product of solving the normal equations. Thus the square error in a_j is just the mean weighted square error of the data multiplied by the appropriate diagonal element of the inverse of the normal equation matrix. To produce a useful result, we must estimate $\overline{(\omega E)^2}$.

b. The Relation of the Weighted Mean Square Observational Error to the Weighted Mean Square Residual

If we subtract the second of equations (6.3.1) from the first, we get

$$E_i - \varepsilon_i = \sum_{j-0}^{n} \delta a_j \phi_j(x_i) = \sum_{j=0}^{n}\phi_j(x_i)\sum_{k=0}^{n} C_{jk}\sum_{q=1}^{N} w_q \phi_k(x_q)E_q\ . \tag{6.3.14}$$

Now multiply by $w_i\varepsilon_i$ and sum over all i. Re-arranging the summations we can write

$$\sum_{i=1}^{N} w_i\varepsilon_i E_i - \sum_{i=1}^{N}\varepsilon_i^2 = \sum_{i=1}^{N} w_i\varepsilon_i\sum_{j-0}^{n}\delta a_j\phi_j(x_i) = \sum_{j=0}^{n}\sum_{k=0}^{n}\sum_{q=1}^{N} C_{jk}w_q\phi_k(x_q)E_q\left[\sum_{i=1}^{N} w_i\varepsilon_i\phi_j(x_i)\right]. \tag{6.3.15}$$

But the last term in brackets can be obtained from the definition of least squares to be

$$\frac{\partial\sum_{i=1}^{N} w_i\varepsilon_i^2}{\partial a_j} = 2\sum_{i=1}^{N} w_i\varepsilon_i\frac{\partial\varepsilon_i}{\partial a_j} = 2\sum_{i=1}^{N}\phi_j(x_i)w_i\varepsilon_i = 0\ , \tag{6.3.16}$$

so that

$$\sum_{i=1}^{N} w_i E_i\varepsilon_i = \sum_{i=1}^{N} w_i\varepsilon_i^2\ . \tag{6.3.17}$$

Now multiply equation (6.3.14) by w_iE_i and sum over all i. Again rearranging the order of summation we get

$$\sum_{i=1}^{N} w_i E_i^2 - \sum_{i=1}^{N} w_i E_i \varepsilon_i = \sum_{i=1}^{N} w_i E_i \sum_{j-0}^{n} \delta a_j \phi_j(x_i)$$
$$= \sum_{j=0}^{n}\sum_{k=0}^{n}\sum_{q=1}^{N}\sum_{i=1}^{N} C_{jk} w_q \phi_j(x_i)\phi_k(x_q) E_q E_i = \sum_{j=0}^{n}\sum_{k=0}^{n} C_{jk} \sum_{i=1}^{N} w_i^2 E_i^2 \phi_j(x_i)\phi_k(x_i) \quad , \quad (6.3,13)$$

where we have used equation (6.3.11) to arrive at the last expression for the right hand side. Making use of equation (6.3.10) we can further simplify equation (6.3.18) to get

$$\overline{N(\omega E)^2} - \sum_{i=1}^{N} w_i E_i \varepsilon_i = \overline{(\omega E)^2} \sum_{j=0}^{n}\sum_{k=0}^{n} C_{jk} A_{jk} = n\overline{(\omega E)} \quad . \quad (6.3.19)$$

Combining this with equation (6.3.17) we can write

$$N\overline{(\omega E)} = \frac{1}{N-n} \sum_{i=1}^{N} (\omega_i \varepsilon_i)^2 \quad , \quad (6.3.20)$$

and finally express the error in a_j [see equation (6.3.13)] as

$$(\delta a_j)^2 = \left[\frac{C_{jj}}{N-n} \right] \sum_{i=1}^{N} (\omega_i \varepsilon_i)^2 \quad . \quad (6.3.21)$$

Here everything on the right hand side is known and is a product of the least square solution. However, to obtain the ε_i's we would have to recalculate each residual after the solution has been found. For problems involving large quantities of data, this would double the effort.

c. Determining the Weighted Mean Square Residual

To express the weighted mean square residual in equation (6.3.21) in terms of parameters generated during the initial solution, consider the following geometrical argument. The $\phi_j(x)$'s are all linearly independent so they can form the basis of a vector space in which the $f(a_j,x_i)$'s can be expressed (see figure 6.1).

The values of $f(a_j,x_i)$ that result from the least square solution are a linear combination of the $\phi_j(x_i)$'s where the constants of proportionality are the a_j's. However, the values of the independent variable are also independent of each other so that the length of any vector is totally uncorrelated with the length of any other and its location in the vector space will be random [note: the space is linear in the a_j's , but the component lengths depend on $\phi_j(x)$]. Therefore the magnitude of the square of the vector sum of the $\vec{f}_i$'s will grow as the square of the individual vectors. Thus, if $\vec{F}$ is the vector sum of all the individual vectors $\vec{f}_i$ then its magnitude is just

$$\left|\vec{F}\right|^2 = \sum_{i=1}^{N} f^2(a_j, x_i) \quad . \quad (6.3.22)$$

The observed values for the independent variable Y_i are in general not equal to the corresponding $f(a_j,x_i)$ so they cannot be embedded in the vector space formed by the $\phi_j(x_i)$'s. Therefore figure 6.1 depicts them lying above (or out of) the vector space. Indeed the difference between them is just ε_i. Again, the Y_i's are

independent so the magnitude of the vector sum of the $\vec{Y}_i$'s and the $\vec{\varepsilon}_i$'s is

$$\left.\begin{aligned} \left|\vec{Y}\right|^2 &= \sum_{i=1}^{N} Y_i^2 \\ \left|\vec{\varepsilon}\right|^2 &= \sum_{i=1}^{N} \varepsilon_i^2 \end{aligned}\right\} . \tag{6.3.23}$$

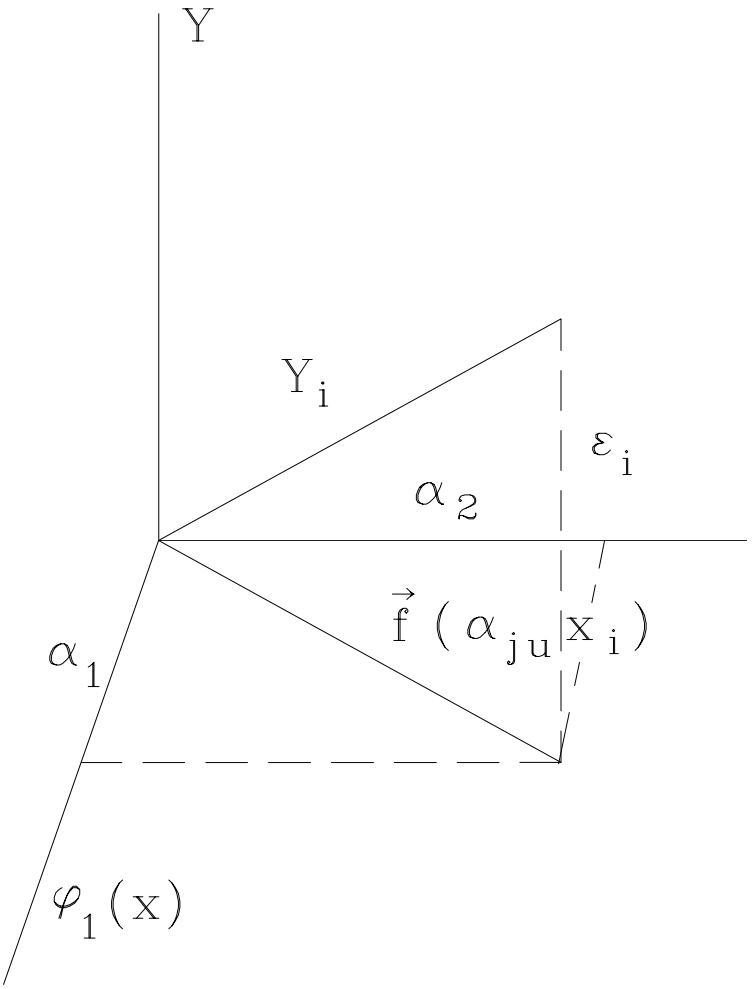


Figure 6.2 shows the parameter space defined by the $\phi_j(x)$*'s. Each* $f(a_j,x_i)$ *can be represented as a linear combination of the* $\phi_j(x_i)$ *where the* a_j *are the coefficients of the basis functions. Since the observed variables* Y_i *cannot be expressed in terms of the* $\phi_j(x_i)$*, they lie out of the space.*

Since least squares seeks to minimize $\Sigma\varepsilon_i^2$, that will be accomplished when the tip of $\vec{Y}$ lies over the tip of $\vec{F}$ so that $\vec{\varepsilon}$ is perpendicular to the $\phi_j(x)$ vector space. Thus we may apply the theorem of Pythagoras (in n-dimensions if necessary) to write

$$\sum_{i=1}^{N} w_i \varepsilon_i^2 = \sum_{i=1}^{N} w_i Y_i^2 - \sum_{i=1}^{N} w_i f^2(a_j, x_i) \quad . \tag{6.3.24}$$

Here we have included the square weights w_i as their inclusion in no way changes the result. From the definition of the mean square residual we have

$$\sum_{i=1}^{N} (w_i \varepsilon_i)^2 = \sum_{i=1}^{N} w_i [Y_i - f(a_j, x_i)]^2 = \sum_{i=1}^{N} w_i Y_i^2 - 2\sum_{i=1}^{N} w_i Y_i f(a_j, x_i) + \sum_{i=1}^{N} w_i f^2(a_j, x_i) \ , \tag{6.3.25}$$

which if we combine with equation (6.3.24) will allow us to eliminate the quadratic term in f^2 so that equation (6.3.21) finally becomes

$$(\delta a_j)^2 = \left[\frac{C_{jj}}{N-n}\right]\left(\left[\sum_{i=1}^{N} w_i Y_i^2\right] - \sum_{k=0}^{n} a_k \left[\sum_{i=1}^{N} w_i Y_i \phi_k(x_i)\right]\right) \quad . \tag{6.3.26}$$

The term in the square brackets on the far right hand side is the constant vector of the normal equations. Then the only unknown term in the expression for δa_j is the scalar term $[\Sigma w_i Y_i^2]$, which can easily be generated during the formation of the normal equations. Thus it is possible to estimate the effect of errors in the data on the solution set of least square coefficients using nothing more than the constant vector of the normal equations, the diagonal elements of the inverse matrix of the normal equations, the solution itself, and the weighted sum squares of the dependent variables. This amounts to a trivial calculation compared to the solution of the initial problem and should be part of any general least square program.

d. The Effects of Errors in the Independent Variable

Throughout the discussion in this section we have investigated the effects of errors in the dependent variable. We have assumed that there is no error in the independent variable. Indeed the least square norm itself makes that assumption. The "best" solution in the least square sense is that which minimizes the sum square of the residuals. Knowledge of the independent variable is assumed to be precise. If this is not true, then real problems emerge for the least square algorithm. The general problem of uncorrelated and unknown errors in both x and Y has never been solved. There do exist algorithms that deal with the problem where the ratio of the errors in Y to those in x is known to be a constant. They basically involve a coordinate rotation through an angle $\alpha = \tan(x/y)$ followed by the regular analysis. If the approximating function is particularly simple (e.g. a straight line), it may be possible to invert the defining equation and solve the problem with the role of independent and dependent variable interchanged. If the solution is the same (allowing for the transformation of variables) within the formal errors of the solution, then some confidence may be gained that a meaningful solution has been found. Should they differ by more than the formal error then the analysis is inappropriate and no weight should be attached to the solution.

Unfortunately, inversion of all but the simplest problems will generally result in a non-linear system of equations if the inversion can be found at all. So in the next section we will discuss how one can approach a least square problem where the normal equations are non-linear.

6.4 Non-linear Least Squares

In general, the problem of non-linear least squares is fraught with all the complications to be found with any non-linear problem. One must be concerned with the uniqueness of the solution and the non-linear propagation of errors. Both of these basic problems can cause great difficulty with any solution. The simplest approach to the problem is to use the definition of least squares to generate the normal equations so that

$$\sum_{i=1}^{N} w_i [Y_i - f(a_j, x_i)] \frac{\partial f(a_j, x_i)}{\partial a_j} = 0 , \quad j = 0, 1, \cdots, n \quad . \tag{6.4.1}$$

These n+1 non-linear equations must then be solved by whatever means one can find for the solution of non-linear systems of equations. Usually some sort of fixed-point iteration scheme, such as Newton-Raphson, is used. However, the error analysis may become as big a problem as the initial least square problem itself. Only when the basic equations of condition will give rise to stable equations should the direct method be tried. Since one will probably have to resort to iterative schemes at some point in the solution, a far more common approach is to linearize the non-linear equations of condition and solve them iteratively. This is generally accomplished by linearizing the equations in the vicinity of the answer and then solving the linear equations for a solution that is closer to the answer. The process is repeated until a sufficiently accurate solution is achieved. This can be viewed as a special case of a fixed-point iteration scheme where one is required to be relatively near the solution.

In order to find appropriate starting values it is useful to understand precisely what we are trying to accomplish. Let us regard the sum square of the residuals as a function of the regression coefficients a_j so that

$$\sum_{i=1}^{N} w_i [Y_i - f(a_j, x_i)]^2 = \sum_{i=1}^{N} w_i \varepsilon_i^2 = \chi^2(a_j) \quad . \tag{6.4.2}$$

For the moment, we shall use the short hand notation of χ^2 to represent the sum square of the residuals. While the function $f(a_j,x)$ is no longer linear in the a_j's they may be still regarded as independent and therefore can serve to define a space in which χ^2 is defined. Our non-linear least square problem can be geometrically interpreted to be finding the minimum in the χ^2 hypersurface (see figure 6.2). If one has no prior knowledge of the location of the minima of the χ^2 surface, it is best to search the space with a coarse multidimensional grid. If the number of variables a_j is large, this can be a costly search, for if one picks m values of each variable a_j, one has m^n functional evaluations of equation (6.4.2) to make. Such a search may not locate all the minima and it is unlikely to definitively locate the deepest and therefore most desirable minimum. However, it should identify a set(s) of parameters a_k^0 from which one of the following schemes will find the true minimum.

We will consider two basic approaches to the problem of locating these minima. There are others, but they are either logically equivalent to those given here or very closely related to them. Basically we shall assume that we are near the true minimum so that first order *changes* to the solution set a_k^0 will lead us to that minimum. The primary differences in the methods are the manner by which the equations are formulated.

a. The Method of Steepest Descent

A reasonable way to approach the problem of finding a minimum in χ^2-space would be to change the values of a_j so that one is moving in the direction, which yields the largest change in the value of χ^2. This will occur in the direction of the gradient of the surface so that

$$\left.\begin{aligned} \nabla\chi^2 &= \sum_{i=1}^{N} \frac{\partial\chi^2}{\partial a_j}\hat{a}_j \\ \frac{\partial\chi^2}{\partial a_j} &= \frac{\chi^2(a_j^0 + \Delta a_j) - \chi^2(a_j^0)}{\Delta a_j} \end{aligned}\right\} . \tag{6.4.3}$$

We can calculate this by making small changes Δa_j in the parameters and evaluating the components of the gradient in accordance with the second of equations (6.4.3). Alternately, we can use the definition of least squares and calculate

$$\nabla_j\chi^2 = \frac{\partial\chi^2}{\partial a_j} = 2\sum_{i=1}^{N} w_i[Y_i - f(a_j, x_i)]\frac{\partial f(a_j, x_i)}{\partial a_j} . \tag{6.4.4}$$

If the function $f(a_j,x)$ is not too complicated and has closed form derivatives, this is by far the preferable manner to obtain the components of $\nabla\chi^2$. However, we must exercise some care as the components of $\nabla\chi^2$ are not dimensionless. In general, one should formulate a numerical problem so that the units don't get in the way. This means normalizing the components of the gradient in some fashion. For example we could define

$$\xi_i = \frac{[a_j\nabla_j\chi^2/\chi^2]}{\sum_{j=0}^{n} a_j\nabla_j\chi^2/\chi^2} = \frac{a_j\nabla_j\chi^2}{\sum_{j=0}^{n} a_j\nabla_j\chi^2} , \tag{6.4.5}$$

which is a sort of normalized gradient with unit magnitude. The next problem is how far to apply the gradient in obtaining the next guess, A conservative possibility is to use Δa_j from equation (6.4.3) so that

$$\delta a_j = \Delta a_j/\xi_j . \tag{6.4.6}$$

In order to minimize computational time, the direction of the gradient is usually maintained until χ^2 begins to increase. Then it is time to re-evaluate the gradient. One of the difficulties of the method of steepest descent is that the values of the gradient of χ^2 vanish as one approaches a minimum. Therefore the method becomes unstable as one approaches the answer in the same manner and for the same reasons that Newton-Raphson fixed-point iteration became unstable in the vicinity of multiple roots. Thus we shall have to find another approach.

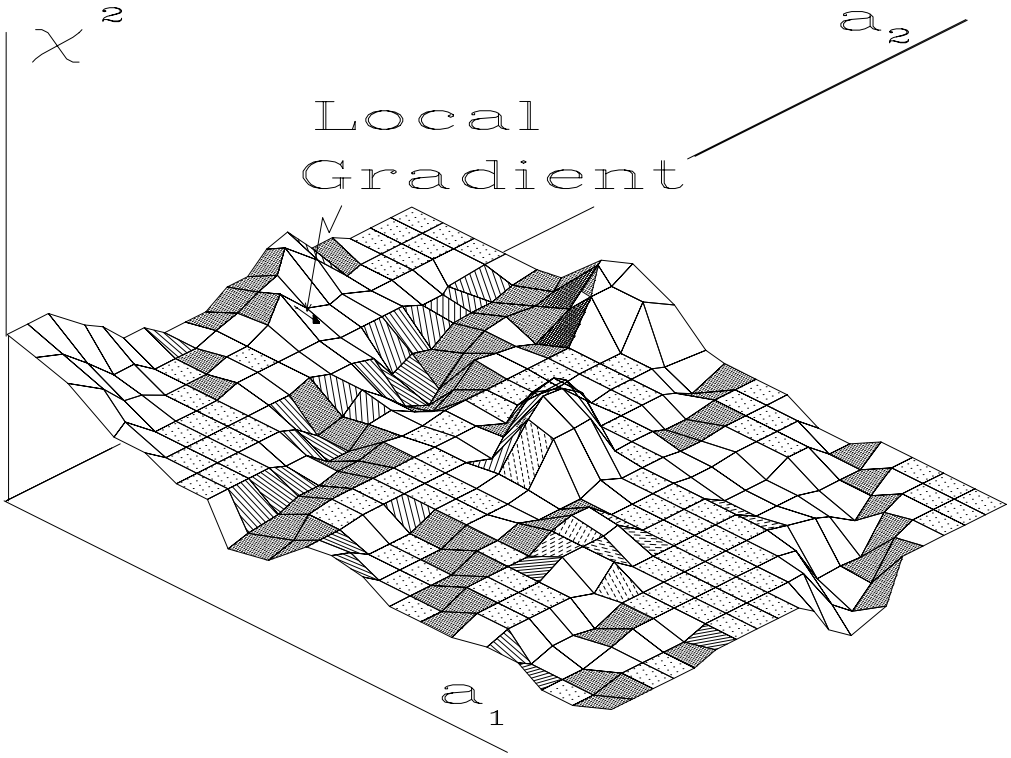


Figure 6.3 shows the χ^2 hypersurface defined on the a_j *space. The non-linear least square seeks the minimum regions of that hypersurface. The gradient method moves the iteration in the direction of steepest decent based on local values of the derivative, while surface fitting tries to locally approximate the function in some simple way and determines the local analytic minimum as the next guess for the solution.*

b. Linear approximation of $\mathbf{f(a_j,x)}$

Let us consider approximating the non-linear function $f(a_j,x)$ by a Taylor series in a_j. To the extent that we are near the solution, this should yield good results. A multi-variable expansion of $f(a_j,x)$ around the present values a_j^0 of the least square coefficients is

$$f(a_j,x) = f(a_j^0,x) + \sum_{ki=0}^{n} \frac{\partial f(a_k^0,x)}{\partial a_k}\delta a_k \quad . \tag{6.4.7}$$

If we substitute this expression for $f(a_j,x)$ into the definition for the sum-square residual χ^2, we get

$$\chi^2 = \sum_{i=1}^{N} w_i[Y_i - f(a_j,x_i)]^2 = \sum_{i=1}^{N} w_i\left[Y_i - f(a_j^0,x_i) - \sum_{k=0}^{n}\frac{\partial f(a_j^0,x_i)}{\partial a_k}\delta a_k\right]^2 \quad . \tag{6.4.8}$$

This expression is linear in δa_j so we can use the regular methods of linear least squares to write the normal

equations as

$$\frac{\partial \chi^2}{\partial \delta a_p} = 2\sum_{i=1}^{N} w_i \left[Y_i - f(a_j^0, x_i) - \sum_{k=0}^{n} \frac{\partial f(a_j^0, x_i)}{\partial a_k} \delta a_k \right] \frac{\partial f(a_j^0, x_i)}{\partial a_p} = 0 , \quad p = 0, 1, \cdots, n \quad , \qquad (6.4.9)$$

which can be put in the standard form of a set of linear algebraic equations for δa_k so that

$$\left.\begin{aligned} &\sum_{k=0}^{n} \delta a_k A_{kp} = B_p , \quad p = 0, 1, \cdots, n \\ &A_{kp} = \sum_{i=1}^{N} w_i \frac{\partial f(a_j^0, x_i)}{\partial a_k} \frac{\partial f(a_j^0, x_i)}{\partial a_p} , \quad k = 0, 1, \cdots, n , \quad p = 0, 1, \cdots, n \\ &B_p = \sum_{i=1}^{N} w_i [Y_i - f(a_j^0, x_i)] \frac{\partial f(a_j^0, x_i)}{\partial a_p} , \quad p = 0, 1, \cdots, n \end{aligned}\right\} . \qquad (6.4.10)$$

The derivative of $f(a_j,x)$ that appears in equations (6.4.9) and (6.4.10) can either be found analytically or numerically by finite differences where

$$\frac{\partial f(a_j, x_i)}{\partial a_p} = \frac{f[a_j^0, (a_p^0 + \Delta a_p), x_i] - f(a_j^0, a_p^0, x_i)}{\Delta a_p} \quad . \qquad (6.4.11)$$

While the equations (6.4.10) are linear in δa_k, they can be viewed as being quadratic in a_k. Consider any expansion of a_k in terms of χ^2 such as

$$a_k = q_0 + q_1 \chi^2 + q_2 \chi^4 \quad . \qquad (6.4.12)$$

The variation of a_k will then have the form

$$\delta a_k = q_1 + 2 q_2 \chi^2 \quad , \qquad (6.4.13)$$

which is clearly linear in χ^2. This result therefore represents a parabolic fit to the hypersurface χ^2 with the condition that δa_k is zero at the minimum value of χ^2. The solution of equations (6.4.10) provides the location of the minimum of the χ^2 hypersurface to the extent that the minimum can locally be well approximated by a parabolic hypersurface. This will certainly be the case when we are near the solution which is precisely where the method of steepest descent fails.

It is worth noting that the constant vector of the normal equations is just half of the components of the gradient given in equation (6.4.4). Thus it seems reasonable that we could combine this approach with the method of steepest descent. One approach to this is given by Marquardt[4]. Since we were somewhat arbitrary about the distance we would follow the gradient in a single step we could modify the diagonal elements of equations (6.4.10) so that

$$\left.\begin{aligned} A'_{kk} &= A_{kk}(1 + \lambda) , \quad k = 0, 1, \cdots, n \\ A'_{kp} &= A_{kp} , \quad k \neq p \end{aligned}\right\} . \qquad (6.4.14)$$

Clearly as λ increases, the solution approaches the method of steepest descent since

$$\lim_{\lambda\to\infty} \delta a_k = B_k/\lambda A_{kk} \quad . \tag{6.4.15}$$

All that remains is to find an algorithm for choosing λ. For small values of λ, the method approaches the first order method for δa_k. Therefore we will choose λ small (say about 10^{-3}) so that the δa_k's are given by the solution to equations (6.4.10). We can use that solution to re-compute χ^2. If

$$\chi^2(\vec{a} + \delta\vec{a}) > \chi^2(\vec{a}), \tag{6.4.16}$$

then increase λ by a factor of 10 and repeat the step. However, if condition (6.4.16) fails and the value of χ^2 is decreasing, then decrease λ by a factor of 10, adopt the new values of a_k and continue. This allows the analytic fitting procedure to be employed where it works the best - near the solution, and utilizes the method of steepest descent where it will give a more reliable answer - well away from the minimum. We still must determine the accuracy of our solution.

c. Errors of the Least Squares Coefficients

The error analysis for the non-linear case turns out to be incredibly simple. True, we will have to make some additional assumptions to those we made in section 6.3, but they are reasonable assumptions. First, we must assume that we have reached a minimum. Sometimes it is not clear what constitutes a minimum. For example, if the minimum in χ^2 hyperspace is described by a valley of uniform depth, then the solution is not unique, as a wide range of one variable will minimize χ^2. The error in this variable is large and equal at least to the length of the valley. While the method we are suggesting will give reliable answers to the formal errors for a_j when the approximation accurately matches the χ^2 hypersurface, when it does not the errors will be unreliable. The error estimate relies on the linearity of the approximating function in δa_j.

In the vicinity of the χ^2 minimum

$$\delta a_j = a_j - a_j^0 \quad . \tag{6.4.17}$$

For the purposes of the linear least squares solution that produces δa_j, the initial value a_j^0 is a constant devoid of any error. Thus when we arrive at the correct solution, the error estimates for δa_j will provide the estimate for the error in a_j itself since

$$\Delta(\delta a_j) = \Delta a_j - \Delta[a_j^0] = \Delta a_j \quad . \tag{6.4.18}$$

Thus the error analysis we developed for linear least squares in section 6.3 will apply here to finding the error estimates for δa_j and hence for a_j itself. This is one of the virtues of iterative approaches. All past sins are forgotten at the end of each iteration. Any iteration scheme that converges to a fixed-point is in some real sense a good one. To the extent that the approximating function at the last step is an accurate representation of the χ^2 hypersurface, the error analysis of the linear least squares is equivalent to doing a first order perturbation analysis about the solution for the purposes of estimating the errors in the coefficients representing the coordinates of the hyperspace function. As we saw in section 6.3, we can carry out that error analysis for almost no additional computing cost.

One should keep in mind all the caveats that apply to the error estimates for non-linear least squares. They are accurate only as long as the approximating function fits the hyperspace. The error distribution of the independent variable is assumed to be anti-symmetric. In the event that all the conditions are met, the

errors are just what are known as the formal errors and should be taken to represent the *minimum* errors of the parameters.

6.5 Other Approximation Norms

Up to this point we have used the Legendre Principle of Least Squares to approximate or "fit" our data points. As long as this dealt with experimental data or other forms of data which contained intrinsic errors, one could justify the Least Square norm on statistical grounds (as long as the error distribution met certain criteria). However, consider the situation where one desires a computer algorithm to generate, say, sin(x) over some range of x such as $0 \le x \le \pi/4$. If one can manage this, then from multiple angle formulae, it is possible to generate sin(x) for any value of x. Since at a very basic level, digital computers only carry out arithmetic, one would need to find some approximating function that can be computed arithmetically to represent the function sin(x) accurately over that interval. A criterion that required the average error of computation to be less than $\bar{\varepsilon}$ is not acceptable. Instead, one would like to be able to guarantee that the computational error would *always* be less than ε_{max}. An approximating norm that will accomplish this is known as the *Chebyschev norm* and is sometimes called the "mini-max" norm. Let us define the maximum value of a function h(x) over some range of x to be

$$h_{max} \equiv \text{Max} \left| h(x) \right| \quad \forall \text{ allowed } x \ . \tag{6.5.1}$$

Now assume that we have a function Y(x) which we wish to approximate by $f(a_j,x)$ where a_j represents a set of free parameters that may be adjusted to provide the "best" approximation in some sense. Let h(x) be the difference between those two functions so that

$$h(x) = \varepsilon(x) = Y(x) - f(a_j,x) \ . \tag{6.5.2}$$

The least square approximation norm would say that the "best" set of a_j's is found from

$$\text{Min} \int \varepsilon^2(x)dx \ . \tag{6.5.3}$$

However, an approximating function that will be the best function for computational approximation will be better given by

$$\text{Min} \left| h_{max} \right| = \text{Min} \left| \varepsilon_{max} \right| = \text{Min} \left| \text{Max} \left| Y(x)-f(a_j,x) \right| \right| . \tag{6.5.4}$$

A set of adjustable parameters a_j that are obtained by applying this norm will guarantee that

$$\varepsilon(x) \le \varepsilon_{max} \quad \forall x \ , \tag{6.5.5}$$

and that ε_{max} is the smallest possible value that can be found for the given function $f(a_j,x)$. This guarantees the investigator that any numerical evaluation of f(x) will represent Y(x) within an amount ε_{max}. Thus, by minimizing the maximum error, one has obtained an approximation algorithm of known accuracy throughout the entire range. Therefore this is the approximation norm used by those who generate high quality functional subroutines for computers. Rational functions are usually employed for such computer algorithms instead of ordinary polynomials. However, the detailed implementation of the norm for determining the free parameters in approximating rational functions is well beyond the scope of this book. Since we have emphasized polynomial approximation throughout this book, we will discuss the implementation of this norm with polynomials.

a. The Chebyschev Norm and Polynomial Approximation

Let our approximating function $f(a_j,x)$ be of the form given by equation (3.1.1) so that

$$f(a_j, x) = \sum_{j=0}^{n} a_j \phi_j(x) \quad . \tag{6.5.6}$$

The choice of $f(a_j,x)$ to be a polynomial means that the free parameters a_j will appear linearly in any analysis. So as to facilitate comparison with our earlier approaches to polynomial approximation and least squares, let us choose ϕ_j to be x^j and we will attempt to minimize $\varepsilon_{max}(x)$ over a discrete set of points x_i. Thus we wish to find a set of a_j so that

$$\text{Min}(\varepsilon_i)_{max} = \text{Min}\left| Y_i - \sum_{j=0}^{n} a_j x_j^i \right|_{max} \qquad \forall x \quad . \tag{6.5.7}$$

Since we have (n+1) free parameters, a_j, we will need at least N = n+1 points in our discrete set x_i. Indeed, if n+1 = N then we can fit the data exactly so that ε_{max} will be zero and the a_j's could be found by any of the methods in chapter 3. Consider the more interesting case where N >> n+1. For the purposes of an example let us consider the cases where n = 0, and 1 . For n = 0 the approximating function is a constant, represented by a horizontal line in Figure 6.4

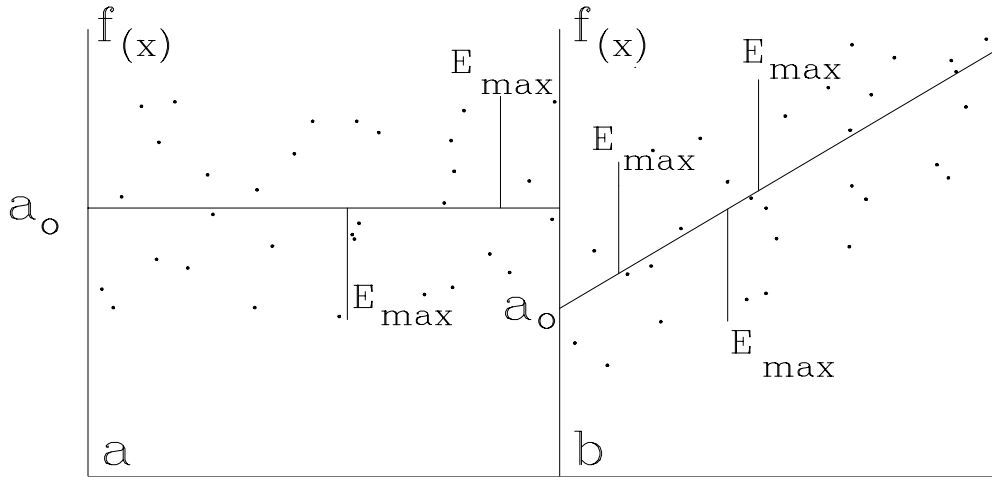


Figure 6.4 shows the Chebyschev fit to a finite set of data points. In panel a the fit is with a constant a_0 *while in panel b the fit is with a straight line of the form* $f(x) = a_1x+a_0$*. In both cases, the adjustment of the parameters of the function can only produce* (n+2) *maximum errors for the* (n+1) *free parameters.*

By adjusting the horizontal line up or down in figure 6.3a we will be able to get two points to have the same largest value of $|\varepsilon_i|$ with one change in sign between them. For the straight line in Figure 6.3b, we will be able to adjust both the slope and intercept of the line thereby making the three largest values of $|\varepsilon_i|$ the same. Among the extreme values of ε_i there will be at least two changes in sign. In general, as long as N > (n+1), one can adjust the parameters a_j so that there are n+2 extreme values of ε_i all equal to ε_{max} and there

will be (n+1) changes of sign along the approximating function. In addition, it can be shown that the a_j's will be unique. All that remains is to find them.

b. The Chebyschev Norm, Linear Programming, and the Simplex Method

Let us begin our search for the "best" set of free-parameters a_j by considering an example. Since we will try to show graphically the constraints of the problem, consider an approximating function of the first degree which is to approximate three points (see figure 6.3b). We then desire

$$\left.\begin{aligned} Y_1 - (a_0 + a_1 x_1) &\le \varepsilon_{max} \\ Y_2 - (a_0 + a_1 x_2) &\le \varepsilon_{max} \\ Y_3 - (a_0 + a_1 x_3) &\le \varepsilon_{max} \\ |\varepsilon_{max}| &= \mathrm{Min}|\varepsilon_{max}| \end{aligned}\right\} . \tag{6.5.8}$$

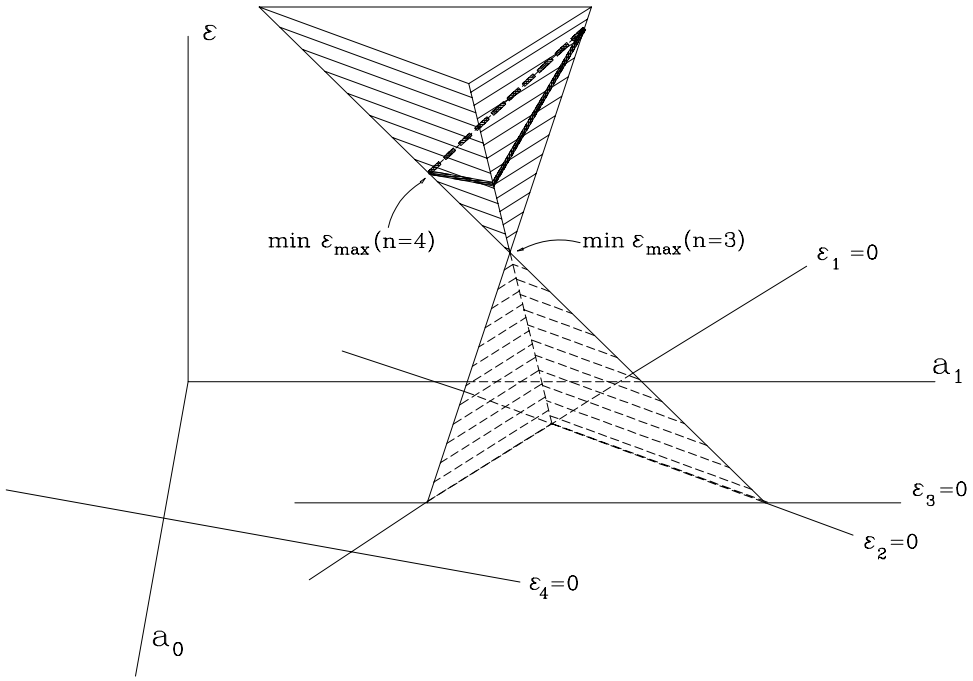


Figure 6.5 shows the parameter space for fitting three points with a straight line under the Chebyschev norm. The equations of condition denote half-planes which satisfy the constraint for one particular point.

These constraints constitute the basic minimum requirements of the problem. If they were to be plotted in parameter space (see Figure 6.4), they would constitute semi-planes bounded by the line for $\varepsilon = 0$. The half of the semi-plane that is permitted would be determined by the sign of ε. However, we have used the result

from above that there will be three extreme values for ε_i all equal to ε_{max} and having opposite sign. Since the value of ε_{max} is unknown and the equation (in general) to which it is attached is also unknown, let us regard it as a variable to be optimized as well. The semi-planes representing the constraints are now extended out of the a_0-a_1 plane in the direction of increasing $|\varepsilon_{max}|$ with the semi-planes of the constraints forming an inverted irregular pyramid. The variation of the sign of ε_{max} guarantees that the planes will intersect to form a convex solid. The solution to our problem is trivial, as the lower vertex of the pyramid represents the minimum value of the maximum error, which will be the same for each constraint. However, it is nice that the method will tell us that without it being included in the specification of the problem. Since the number of extrema for this problem is 1+2, this is an expected result. The inclusion of a new point produces an additional semi-constraint plane which will intersect the pyramid producing a triangular upper base. The minimum value of the maximum error will be found at one of the vertices of this triangle. However since the vertex will be defined by the intersection of three lines, there will still be three extrema as is required by the degree of the approximating polynomial. Additional points will increase the number of sides as they will cut the initial pyramid forming a multi-sided polygon. The vertices of the polygon that is defined in parameter-ε_{max} space will still hold the optimal solution. In this instance the search is simple as we simply wish to know which ε_{max} is the smallest in magnitude. Thus we look for the vertex nearest the plane of the parameters. An increase in the number of unknowns a_i's will produce figures in higher dimensions, but the analysis remains essentially the same.

The area of mathematics that deals with problems that can be formulated in term of linear constraints (including inequalities) is known as *Linear Programming* and it has nothing to do with computer programming. It was the outgrowth of a group of mathematicians working in a broader area of mathematics known as operations research. The inspiration for its development was the finding of solutions to certain optimization problems such as the efficient allocation of scarce resources (see Bland[4]).

Like many of the subjects we have introduced in this book, linear programming is a large field of study having many ramifications far beyond the scope of this book. However, a problem that is formulated in terms of constraint inequalities will consist of a collection of semi-spaces that define a *polytope* (a figure where each side is a polygon) in multidimensional parameter space. It can be shown that the optimum solution lies at one of the vertices of the polytope. A method for sequentially testing each vertex so that the optimal one will be found in a deterministic way is known as the *simplex method*. Starting at an arbitrary vertex one investigates the adjacent vertices finding the one which best satisfies the optimal conditions. The remaining vertices are ignored and one moves to the new "optimal" vertex and repeats the process.

When one can find no adjacent vertices that better satisfy the optimal condition that vertex is the most optimal of the entire polytope and represents the optimal solution to the problem. In practice, the simplex method has been found to be far more efficient than general theoretical considerations would lead one to expect. So, while there are other approaches to linear programming problems, the one that still attracts most attention is the simplex method.

c. The Chebyschev Norm and Least Squares

At the beginning of this chapter, we justified the choice of the Least Square approximation norm on the grounds that it yielded linear equations of condition and was the lowest power of the deviation ε that was guaranteed to be positive. What about higher powers? The desire to keep the error constraints positive should limit us to even powers of ε. Thus consider a norm of the form

$$\text{Min} \sum_i \varepsilon_i^{2n} = \text{Min} \sum_i [Y_i - f(a_j, x_i)]^{2n} , \tag{6.5.9}$$

which lead to the non-linear equations

$$\frac{\partial}{\partial a_j}\left(\sum_i \left[y_i - f(a_j, x_i) \right]^{2n} \right) = \sum_i 2n \left[y_i - f(a_j, x_i) \right]^{2n-1} \frac{\partial f(a_j, x_i)}{\partial a_j} = 0 \quad . \tag{6.5.10}$$

Now one could solve these non-linear equations, but there is no reason to expect that the solution would be "better" in any real sense than the least square solution. However, consider the limit of equation (6.5.9) as $n \to \infty$.

$$\lim_{n\to\infty}\left(\text{Min} \sum_i \varepsilon_i^{2n} \right) = \text{Min}\left(\lim_{n\to\infty} \sum_i \varepsilon_i^{2n} \right) = \text{Min} \left| \varepsilon_{max} \right|^{2n} \quad . \tag{6.5.11}$$

The solution that is found subject to the constraint that ε_{max}^{2n} is a minimum will be the same solution that is obtained when ε_{max} is a minimum. Thus the limit of the 2nth norm as n goes to infinity is the Chebyschev norm.

In this chapter we have made a transition from discussing numerical analysis where the basic inputs to a problem are known with arbitrary accuracy tp those where the basic data contained errors. In earlier chapters the only errors that occur in the calculation result from round-off of arithmetic processes or truncation of the approximation formulae. However, in section 6.3 we allowed for the introduction of "flawed" inputs, with inherent errors resulting from experiment or observation. Since any interaction with the real world will involve errors of observation, we shall spend most of the remainder of the book discussing the implication of these errors and the manner by which they can be managed.

Chapter 6 Exercises

1. Develop normal equations for the functions:

   a. $f(x) = a_0 e^{a_1 x}$

   b. $f(x) = a_0 + a_1 \sin(a_2 \pi x + a_3)$.

   Which expressions could be replaced with a linear function with no loss of accuracy? What would the error analysis of that function fit to observational data say about the errors of the original coefficients a_j?

2. Using least squares find the "best" straight-line fit and the error estimates for the slope and intercept of that line for the following set of data.

x_i	Y_i
1	1.5
2	2.0
3	2.8
4	4.1
5	4.9
6	6.3
7	5.0
8	11.5

3. Fit the following table with a polynomial of the form

   $f(a_j,x) = \sum_k \phi_k(x)$, where $\phi_k(x) = \cos(k\pi x)$

x_i	$f(a_j,x_i)$
0.00000	0.00000
0.17453	0.17101
0.34907	0.32139
0.41888	0.37157
0.62839	0.47553
0.78540	0.49970
1.0123	0.44940
1.0821	0.41452
1.2915	0.26496
1.5010	0.06959

   How many terms are required to fit the table accurately? Discuss what you mean by "accurately" and why you have chosen that meaning.

4. Given the following two sets of data to be fit by straight lines.

$x_{1,i}$	$Y_{1,i}$	$x_{2,i}$	$Y_{2,i}$
1	9.1	1	0.5
2	8.5	2	3.2
3	7.6	3	2.5
4	3.5	4	4.6
5	4.2	5	5.1
6	2.1	6	6.9
7	0.2	7	6.8

find the "best" value for the intersection of the straight lines and an estimate for the error in Y. How would you confirm the assumption that there is no error in x?

5. Determine the complex Fourier transform of

a. e^{-t^2} $-\infty < t < +\infty$.

b. $e^{-t}\cos(t)$, $0 < t < +\infty$.

6. Find the FFT for the functions in problem 5 where the function is sampled every .01 in t and the total number of points is 1024. Calculate the inverse transform of the result and compare the accuracy of the process.

Chapter 6 References and Supplementary Reading

1. Bateman, H., "Tables of Integral Transforms" (1954) Ed. A. Erdélyi, Volumes 1,2, McGraw-Hill Book Co., Inc. New York, Toronto, London.

2. Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T., "Numerical Recipies the Art of Scientific Computing" (1986), Cambridge University Press, Cambridge, pp. 390-394.

3. Marquardt, D.W., "An Algorithm for Least-Squares Estimation of Nonlinear Parameters", (1963), J. Soc. Ind. Appl. Math., Vol.11, No. 2, pp.431-441.

4. Bland, R.G., "The Allocation of Resources by Linear Programming", (1981) Sci. Amer. Vol. 244, #6, pp.126-144.

Most books on numerical analysis contain some reference to least squares. Indeed most freshmen calculus courses deal with the subject at some level. Unfortunately no single text contains a detailed description of the subject and its ramifications.

1. Hildebrand, F.B., "Introduction to Numerical Analysis" (1956) McGraw-Hill Book Co., Inc., New York, Toronto, London, pp. 258-311,

This book presents a classical discussion and much of my discussion in section 6.3 is based on his presentation. The error analysis for non-linear least squares in section 6.4 is dealt with in considerable detail in

2. Bevington, P.R., "Data Reduction and Error Analysis for the Physical Sciences", (1969), McGraw-Hill Book Co. Inc., New York, San Francisco, St. Louis, Toronto, London, Sydney, pp. 204-246.

Nearly any book that discusses Fourier series and transforms contains useful information elaborating on the uses and extended theory of the subject. An example would be

3. Sokolnikoff, I.S., and Redheffer, R.M., "Mathematics of Physics and Modern Engineering", (1958) McGraw-Hill Book Co., Inc. New York, Toronto, London, pp. 175-211.

Two books completely devoted to Fourier analysis and the transforms particularly are:

4. Brigham, E.O., "The Fast Fourier Transform", (1974) Prentice-Hall, Inc. Englewood Cliffs, N.J.,

and

5. Bracewell, R.N., "The Fourier Transform and its Applications", 2nd Ed., (1978), McGraw-Hill Book Company, New York N.Y.

A very compressed discussion, of Linear Programming, which covers much more that we can, is to be found in

6. Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T., "Numerical Recipies the Art of Scientific Computing" (1986), Cambridge University Press, Cambridge. pp. 274-334,

but a more basic discussion is given by

7. Gass, S.T., "Linear Programming" (1969), 3rd ed. McGraw-Hill, New York.

7

Probability Theory and Statistics

• • •

In the last chapter we made the transition from discussing information which is considered to be error free to dealing with data that contained intrinsic errors. In the case of the former, uncertainties in the results of our analysis resulted from the failure of the approximation formula to match the given data and from round-off error incurred during calculation. Uncertainties resulting from these sources will always be present, but in addition, the basic data itself may also contain errors. Since all data relating to the real world will have such errors, this is by far the more common situation. In this chapter we will consider the implications of dealing with data from the real world in more detail.

Philosophers divide data into at least two different categories, observational, historical, or empirical data and experimental data. Observational or historical data is, by its very nature, non-repeatable. Experimental data results from processes that, in principle, can be repeated. Some[1] have introduced a third type of data labeled hypothetical-observational data, which is based on a combination of observation and information supplied by theory. An example of such data might be the distance to the Andromeda galaxy since a direct measurement of that quantity has yet to be made and must be deduced from other aspects of the physical world. However, in the last analysis, this is true of all observations of the world. Even the determination of repeatable, experimental data relies on agreed conventions of measurement for its unique interpretation. In addition, one may validly ask to what extent an experiment is precisely repeatable. Is there a fundamental difference between an experiment, which can be repeated and successive observations of a phenomenon that apparently doesn't change? The only difference would appear to be that the scientist has the option in the case of the former in repeating the experiment, while in the latter case he or she is at the mercy of nature. Does this constitute a fundamental difference between the sciences? The hard sciences such as physics and chemistry have the luxury of being able to repeat experiments holding important variables constant, thereby lending a certain level of certainty to the outcome. Disciplines such as Sociology, Economics and Politics that deal with the human condition generally preclude experiment and thus must rely upon observation and "historical experiments" not generally designed to test scientific hypotheses. Between these two extremes are sciences such as Geology and Astronomy which rely largely upon observation but are founded directly upon the experimental sciences. However, all sciences have in common the gathering of data about the real world. To the analyst, there is little difference in this data. Both experimental and observational data contain intrinsic errors whose effect on the sought for description of the world must be understood.

However, there is a major difference between the physical sciences and many of the social sciences and that has to do with the notion of cause and effect. Perhaps the most important concept driving the physical sciences is the notion of causality. That is the physical biological, and to some extent the behavioral sciences, have a clear notion that event A causes event B. Thus, in testing a hypothesis, it is always clear which variables are to be regarded as the dependant variables and which are to be considered the independent variables. However, there are many problems in the social sciences where this luxury is not present. Indeed, it may often be the case that it is not clear which variables used to describe a complex phenomenon are even related. We shall see in the final chapter that even here there are some analytical techniques that can be useful in deciding which variables are possibly related. However, we shall also see that these tests do not prove cause and effect, rather they simply suggest where the investigator should look for causal relationships. In general data analysis may guide an investigator, but cannot substitute for his or her insight and understanding of the phenomena under investigation.

During the last two centuries a steadily increasing interest has developed in the treatment of large quantities of data all representing or relating to a much smaller set of parameters. How should these data be combined to yield the "best" value of the smaller set of parameters? In the twentieth century our ability to collect data has grown enormously, to the point where collating and synthesizing that data has become a scholarly discipline in itself. Many academic institutions now have an entire department or an academic unit devoted to this study known as statistics. The term statistics has become almost generic in the language as it can stand for a number of rather different concepts. Occasionally the collected data itself can be referred to as statistics. Most have heard the reference to reckless operation of a motor vehicle leading to the operator "becoming a statistic". As we shall see, some of the quantities that we will develop to represent large

amounts of data or characteristics of that data are also called statistics. Finally, the entire study of the analysis of large quantities of data is referred to as the study of statistics. The discipline of statistics has occasionally been defined as providing a basis for decision-making on the basis of incomplete or imperfect data. The definition is not a bad one for it highlights the breadth of the discipline while emphasizing it primary function. Nearly all scientific enterprises require the investigator to make some sort of decisions and as any experimenter knows, the data is always less than perfect.

The subject has its origins in the late 18th and early 19th century in astronomical problems studied by Gauss and Legendre. Now statistical analysis has spread to nearly every aspect of scholarly activity. The developing tools of statistics are used in the experimental and observational sciences to combine and analyze data to test theories of the physical world. The social and biological sciences have used statistics to collate information about the inhabitants of the physical world with an eye to understanding their future behavior in terms of their past performance. The sampling of public opinion has become a driving influence for public policy in the country. While the market economies of the world are largely self-regulating, considerable effort is employed to "guide" these economies based on economic theory and data concerning the performance of the economies. The commercial world allocates resources and develops plans for growth based on the statistical analysis of past sales and surveys of possible future demand. Modern medicine uses statistics to ascertain the efficacy of drugs and other treatment procedures. Such methods have been used, not without controversy, to indicate man made hazards in our environment. Even in the study of language, statistical analysis has been used to decide the authorship of documents based on the frequency of word use as a characteristic of different authors.

The historical development of statistics has seen the use of statistical tools in many different fields long before the basis of the subject were codified in the axiomatic foundations to which all science aspires. The result is that similar mathematical techniques and methods took on different designations. The multi-discipline development of statistics has lead to an uncommonly large amount of jargon. This jargon has actually become a major impediment to understanding. There seems to have been a predilection, certainly in the nineteenth century, to dignify shaky concepts with grandiose labels. Thus the jargon in statistics tends to have an excessively pretentious sound often stemming from the discipline where the particular form of analysis was used. For example, during the latter quarter of the nineteenth century, Sir Francis Galton analyzed the height of children in terms of the height of their parents[2]. He found that if the average height of the parents departed from the general average of the population by an amount x, then the average height of the children would depart by, say, 2x/3 from the average for the population. While the specific value of the fraction (2/3) may be disputed all now agree that it is less than one. Thus we have the observation that departures from the population average of any sub group will *regress* toward the population average in subsequent generations. Sir Francis Galton used Legendre's Principle of Least Squares to analyze his data and determine the *coefficient of regression* for his study. The use of least squares in this fashion has become popularly known as *regression analysis* and the term is extended to problems where the term regression has absolutely no applicability. However, so wide spread has the use of the term become, that failure to use it constitutes a barrier to effective communication.

Statistics and statistical analysis are ubiquitous in the modern world and no educated person should venture into that world without some knowledge of the subject, its strengths and limitations. Again we touch upon a subject that transcends even additional courses of inquiry to encompass a lifetime of study. Since we may present only a bare review of some aspects of the subject, we shall not attempt a historical development.

Rather we will begin by giving some of the concepts upon which most of statistics rest and then developing some of the tools which the analyst needs.

7.1 Basic Aspects of Probability Theory

We can find the conceptual origins of statistics in probability theory. While it is possible to place probability theory on a secure mathematical axiomatic basis, we shall rely on the commonplace notion of probability. Everyone has heard the phrase "the probability of snow for tomorrow 50%". While this sounds very quantitative, it is not immediately clear what the statement means. Generally it is interpreted to mean that on days that have conditions like those expected for tomorrow, snow will fall on half of them. Consider the case where student A attends a particular class about three quarters of the time. On any given day the professor could claim that the probability of student A attending the class is 75%. However, the student knows whether or not he is going to attend class so that he would state that the probability of his attending class on any particular day is either 0% or 100%. Clearly the probability of the event happening is dependent on the prior knowledge of the individual making the statement. There are those who define *probability as a measure of ignorance*. Thus we can define two events to be equally likely if we have no reason to expect one event over the other. In general we can say that if we have n equally likely cases and any m of them will generate an event E, then the probability of E occurring is

$$P(E) = m/n \quad . \tag{7.1.1}$$

Consider the probability of selecting a diamond card from a deck of 52 playing cards. Since there are 13 diamonds in the deck, the probability is just 13/52 = ¼. This result did not depend on there being 4 suits in the standard deck, but only on the ratio of 'correct' selections to the total number of possible selections. It is always assumed that the event will take place if all cases are selected so that the probability that an event E will *not* happen is just

$$Q(\tilde{E}) = 1 - P(E) \quad . \tag{7.1.2}$$

In order to use equation (7.1.1) to calculate the probability of event E taking place, it is necessary that we correctly enumerate all the possible cases that can give rise to the event. In the case of the deck of cards, this seems fairly simple. However, consider the tossing of two coins where we wish to know the probability of two 'heads' occurring. The different possibilities would appear to be each coin coming up 'heads', each coin coming up 'tails', and one coin coming up 'heads' while the other is 'tails'. Thus naïvely one would think that the probability of obtaining two 'heads' would be 1/3. However, since the coins are truly independent events, each coin can be either 'heads' or 'tails'. Therefore there are two separate cases where one coin can be 'head' and the other 'tails' yielding four possible cases. Thus the correct probability of obtaining two 'heads' is 1/4. The set of all possible cases is known as the *sample set*, or *sample space*, and in statistics is sometimes referred to as the *parent population*.

a. The Probability of Combinations of Events

It is possible to view our coin tossing even as two separate and independent events where each coin is tossed separately. Clearly the result of tossing each coin and obtaining a specific result is 1/2. Thus the result of tossing two coins *and* obtaining a specific result (two heads) will be 1/4, or (1/2)×(1/2). In general, the probability of obtaining event E *and* event F, [P(EF)], will be

$$P(EF) = P(E) \times P(F) \quad . \tag{7.1.3}$$

Requiring of the occurrence of event E *and* event F constitutes the use of the *logical and* which always results in a multiplicative action. We can ask what will be the total, or joint, probability of event E *or* event F occurring. Should events E and F be mutually exclusive (i.e. there are no cases in the sample set that result in both E and F), then $P(E_{or}F)$ is given by

$$P(E_{or}F) = P(E) + P(F) \quad . \tag{7.1.4}$$

This use of addition represents the *logical 'or'*. In our coin tossing exercise obtaining one 'head' and one 'tail' could be expressed as the probability of the first coin being 'heads' *and* the second coin being 'tails' *or* the first coin being 'tails' *and* the second coin being 'heads' so that

$$P(HT) = P(H)P(T) + P(T)P(H) = (1/2)\times(1/2) + (1/2)\times(1/2) = 1/2 \quad . \tag{7.1.5}$$

We could obtain this directly from consideration of the sample set itself and equation (7.1.1) since m = 2, and n = 4. However, in more complicated situations the laws of combining probabilities enable one to calculate the combined probability of events in a clear and unambiguous way.

In calculating $P(E_{or}F)$ we required that the events E and F be mutually exclusive and in the coin exercise, we guaranteed this by using separate coins. What can be done if that is not the case? Consider the situation where one rolls a die with the conventional six faces numbered 1 through 6. The probability of any particular face coming up is 1/6. However, we can ask the question what is the probability of a number less than three appearing *or* an even number appearing. The cases where the result is less than three are 1 and 2, while the cases where the result is even are 2, 4, and 6. Naïvely one might think that the correct answer 5/6. However, these are not mutually exclusive cases for the number 2 is both an even number and it is also less than three. Therefore we have counted 2 twice for the only distinct cases are 1, 2, 4, and 6 so that the correct result is 4/6. In general, this result can be expressed as

$$P(E_{or}F) = P(E) + P(F) - P(EF) \quad , \tag{7.1.6}$$

or in the case of the die

$$P(<3_{or}\text{even}) = [(1/6)+(1/6)] + [(1/6)+(1/6)+(1/6)] - [(1/3)\times(1/2)] = 2/3 \quad . \tag{7.1.7}$$

We can express these laws graphically by means of a Venn diagram as in figure 7.1. The simple sum of the dependent probabilities counts the intersection on the Venn diagram twice and therefore it must be removed from the sum.

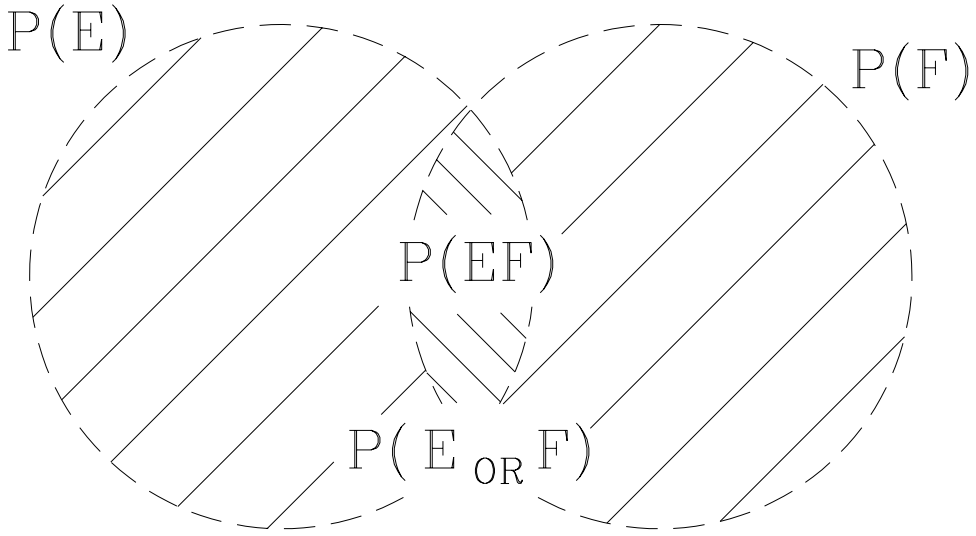


Figure 7.1 shows a sample space giving rise to events E *and* F. *In the case of the die,* E *is the probability of the result being less than three and* F *is the probability of the result being even. The intersection of circle* E *with circle* F *represents the probability of* E *and* F *[i.e.* P(EF)*]. The union of circles* E *and* F *represents the probability of* E *or* F. *If we were to simply sum the area of circle* E *and that of* F *we would double count the intersection.*

b. Probabilities and Random Variables

We can define a random process as one where the result of the process cannot be predicted. For example, the toss of a coin will produce either 'heads' or 'tails', but which will occur as the result of flipping the coins cannot be predicted with complete certainty. If we assign a 1 to 'heads' and a 0 to 'tails', then a succession of coin flips will generate a series of 1's and 0's having no predictable order. If we regard a finite sequence of flips as a binary number, then we can call it a random number since its value will not be predictable. Any secession of finite sequences of the same length will produce a secession of random binary numbers where no number can be predicted from the earlier numbers. We could carry out the same experiment with the die where the results would range from 1 to 6 and the sequences would form base six random numbers.

Now the sequence that produces our random number could be of arbitrary length even though the sample set is finite, but it will always have some numerical value. We can define a *random variable* as any numerically valued function that is defined on the sample set. In the case we have picked, it could be, say, all numbers with five digits or less. Let us define the elements of the sample set to have numerical values x_i. In the case of the coin these would be the 1's and 0's we assigned to 'heads' and 'tails'. For the die, they are simply the values of the faces. Then any random variable, which would appear through its definition as a

random process, would have a result $X_j(x_i) = X_j$ that depends on the values of the sample set x_i. The probability P_j that any particular value of X_j will appear will depend on the probabilities p_i associated with the values x_i that produce the numerical value of the random variable X_j. We could then ask "If we generate n values of the random variable X_j from the sample set, what is the most likely value of X_j that we should expect?". We will call that value of X_j the *expected* or *expectation value* of X and it will be given by

$$E(\mathrm{X}) = \sum_{\mathrm{j=1}}^{\mathrm{N}} \mathrm{P_j X_j} \quad . \tag{7.1.8}$$

Consider the simple case of tossing coins and ask "What is the expectation value for obtaining one 'head' in any given trial of tossing the two coins?". The possibilities are that both coins could turn up 'tails' yielding no 'heads', or one coin could be 'heads' and the other 'tails', or both could be 'heads'. The probabilities of the first and last occurring is ¼, but since either coin can be 'heads' while the other is 'tails' the middle possibility represents two separate cases. Thus the expected value for the number of 'heads' is just

$$E(\mathrm{H}) = 0\times(¼) + 1\times(¼) + 1\times(¼) + 2\times(¼) = 1 \quad . \tag{7.1.9}$$

The first term is made up of the number of heads that result for each trial times the probability of that trial while the other representation of that sum show the distinctly different values of X_j multiplied by the combined probability of those values occurring. The result is that we may expect one 'head' with the toss of two coins. The expectation value of a random variable is sort of an average value or more properly the most likely value of that variable.

c. Distributions of Random Variables

It is clear from our analysis of the coin tossing experiment that not all values of the random variable (eg. the number of 'heads') are equally likely to occur. Experiments that yield one 'head' are twice as likely to occur as either no 'heads' or two 'heads'. The frequency of occurrence will simply be determined by the total probability of the random variable. The dependence of the probability of occurrence on the value of the random variable is called a *probability distribution*. In this instance there is a rather limited number of possibilities for the value of the random variable. Such cases are called discrete probability distributions. If we were to define our random variable to be the value expected from the roll of two dice, then the values could range from 2-12, and we would have a more extensive discrete probability distribution. In general, measured values contain a finite set of digits for the random variables and their probability distributions are always discrete.

However, it is useful to consider continuous random variables as they are easier to use analytically. We must be careful in our definition of probability. We can follow the standard practice of limits used in the differential calculus and define the differential probability of the continuous random variable x occurring within the interval between x and x+Δx to be

$$\mathrm{dP(x)} = \lim_{\Delta x \to 0} [\mathrm{f(x+\Delta x)-f(x)}]/\Delta \mathrm{x} \quad . \tag{7.1.10}$$

Thus the probability that the value of the random variable will lie between a and b will be

$$P(a,b) = \int_a^b f(x)\,dx \quad . \tag{7.1.11}$$

The function f(x) is known as the *probability density distribution function* while P(a,b) is called the *probability distribution function*. The use of probability density functions and their associated probability distribution functions constitute a central tool of analysis in science.

7.2 Common Distribution Functions

From our discussion of random variables, let us consider how certain widely used distribution functions arise. Most distribution functions are determined for the discrete case before generalizing to their continuous counterparts and we will follow this practice. Consider a sample space where each event has a constant probability p of occurring. We will let the random variable be represented by a sequence of sampling events. We then wish to know what the probability of obtaining a particular sequence might be. If we assign each sequence a numerical value, then the probability values of the sequences form a probability distribution function. Let us sample the set of equally probable events n times with m occurrences of an event that has probability p so that we obtain the sequence with total probability

$$P(S) = ppqqq \cdots pqqqppqp = p^m q^{n-m}\,, \tag{7.2.1}$$

where

$$q = 1 - p\ , \tag{7.2.2}$$

is the probability that the sampling did not result in the event. One can think of an event as getting a head from a coin toss.

Since the sampling events are considered independent, one is rarely interested in the probability of the occurrence of a particular sequence. That is, a sequence ppq will have the same probability as the sequence pqp, but one generally wishes to know the probability that one *or* the other *or* some equivalent (i.e. one having the same number of p's and q's) sequence will occur. One could add all the individual probabilities to obtain the probability of all equivalent sequences occurring, or, since each sequence has the same probability, we may simply find the number of such sequences and multiply by the probability associated with the sequence.

a. Permutations and Combinations

The term *permutation* is a special way of describing an arrangement of items. The letters in the word *cat* represent a sequence or permutation, but so do *act*, *tac*, *tca*, *atc*, and *cta*. All of these represent permutations of the same letters. By enumeration we see that there are 6 such permutations in the case of the word *cat*. However, if there are N elements in the sequence, then there will be N! different permutations that can be formed. A simple way to see this is to go about constructing the most general permutation possible. We can begin by selecting the first element of the sequence from any of the n-elements. That means that we would have at least n permutations that begin with one of the n first elements. However, having selected a

first element, there are only (n-1) elements left. Thus we will have only (n-1) new permutations for each of our initial n permutations. Having chosen twice only (n-2) elements will remain. each of the n(n-1) permutations generated by the first two choices will yield (n-2) new permutations. This process can be continued until there are no more elements to select at which point we will have constructed n! distinct permutations.

Now let us generalize this argument where we will pick a sequence of m elements from the original set of n. How many different permutations of m-elements can we build out of n-elements? Again, there are n-ways to select the first element in the permutation leaving (n-1) remaining elements. However, now we do not pick all n-elements, we repeat this process only m-times. Therefore the number of permutations, P^n_m, of n-elements taken m at a time is

$$P^n_m = n(n-1)(n-2) \cdots (n-m+1) = n!/(n-m)! \quad . \tag{7.2.3}$$

A combination is a very different thing than a permutation. When one selects a combination of things, the order of selection is unimportant. If we select a combination of four elements out of twenty, we don't care what order they are selected in only that we ended up with four elements. However, we can ask a question similar to that which we asked for permutations. How many combinations with m-elements can we make from n-elements? Now it is clear why we introduced the notion of a permutation. We may use the result of equation (7.2.3) to answer the question about combinations. Each permutation that is generated in accordance with equation (7.2.3) is a combination. However, since the order in which elements of the combination are selected is unimportant, all permutations with those elements can be considered the same combination. But having picked the m elements, we have already established that there will be m! such permutations. Thus the number of combinations C^n_m of n-elements taken m at a time can be written in terms of the number of permutations as

$$C^n_m = P^n_m/m! = n!/[(n-m)!m!] \quad \binom{n}{m} \quad . \tag{7.2.4}$$

These are often known as the binomial coefficients since they are the coefficients of the binomial series

$$(x+y)^n = C^n_0 x^n + C^n_1 x^{n-1}y + \cdots + C^n_{n-2} x^2 y^{n-1} + C^n_n y^n \quad . \tag{7.2.5}$$

As implied by the last term in equation (7.2.4), the binomial coefficients are often denoted by the symbol $\binom{n}{m}$.

b. The Binomial Probability Distribution

Let us return to the problem of finding the probability of equivalent sequences. Each sequence represents a permutation of the samplings producing events m-times. However, since we are not interested in the order of the sampling, the distinctly different number of sequences is the number of combinations of n-samplings producing m-events. Thus the probability of achieving m-events in n-samplings is

$$P_B(m) = C^n_m p^m q^{n-m} = C^n_m p^m (1-p)^{n-m} \quad , \tag{7.2.6}$$

and is known as the *binomial frequency function*. The probability of having *at least* m-events in n-tries is

$$F(m) = \sum_{i=1}^{m} P(i) = C_0^n(1\text{-}p)^n + C_1^n p(1\text{-}p)^{n-1} + \cdots + C_m^n p^m(1\text{-}p)^{n-m} \quad . \tag{7.2.7}$$

and is known as the *binomial distribution*.

Equations (7.2.6) and (7.2.7) are discrete probability functions. Since a great deal of statistical analysis is related to sampling populations where the samples are assumed to be independent of one another, a great deal of emphasis is placed on the binomial distribution. Unfortunately, it is clear from equation (7.2.4) that there will be some difficulties encountered when n is large. Again since many problems involve sampling very large populations, we should pay some attention to this case. In reality, the case when n is large should be considered as two cases; one where the total sample, n, and the product of the sample size and the probability of a single event, np, are both large, and one where n is large but np is not. Let us consider the latter.

c. The Poisson Distribution

By assuming that n is large but np is not we are considering the case where the probability of obtaining a successful event from any particular sampling is very small (i.e. p<<1). A good example of this is the decay of radioactive isotopes. If one focuses on a particular atom in any sample, the probability of decay is nearly zero for any reasonable time. While p is considered small, we will assume both n and m to be large. If m is large, then the interval between m and m+1 (i.e. 1) will be small compared to m and we can replace m with a continuous variable x. Now

$$\frac{n!}{(n-x)!} = n(n\text{-}1)(n\text{-}2) \cdots (n\text{-}x\text{+}1) \cong n^x, \quad x >> 1, n >> x \quad . \tag{7.2.8}$$

With this approximation we can write equation (7.2.6) as

$$P_B(x) = \frac{n!}{x!(n-x)!} p^x q^{n-x} \cong \frac{n^x}{x!} p^x (1-p)^n \quad . \tag{7.2.9}$$

The last term can be written as

$$(1\text{-}p)^n = (1\text{-}p)^{\mu/p} = [(1\text{-}p)^{1/p}]^{\mu} \quad , \tag{.10}$$

where

$$\mu = np \quad . \tag{7.2.11}$$

The meaning of the parameter μ will become apparent later. For the moment it is sufficient to note that it results from the product of a very large number and a very small number. If expand the quantity on the right in brackets by means of the binomial theorem and take the limit as $p \to 0$, we get

$$\lim_{p\to 0}[(1-p)^{1/p} = \lim_{p\to 0}\left(1 - \left[\frac{1}{p}\right]p + \left[\frac{1}{p}\right]\left[\frac{1}{p}-1\right]\left[\frac{p^2}{2!}\right] - \left[\frac{1}{p}\right]\left[\frac{1}{p}-1\right]\left[\frac{1}{p}-2\right]\left[\frac{p^3}{3!}\right] + \cdots + \right) = e^{-1} \quad . \tag{7.2.12}$$

Therefore in the limit of vanishing p equation (7.2.9) becomes

$$\lim_{p\to 0} P_B(x) \equiv P_P(x,\mu) = \mu^x e^{-\mu}/x! \quad . \tag{7.2.13}$$

$P_P(x,\mu)$ is known as the *Poisson* probability density distribution function. From equation (7.1.8) and equation (7.2.13) one can show that μ is the expected value of x. However, one can see that intuitively from the definition in equation (7.2.11). Surely if one has a large number of samples n and the probability p that any one of them will produce an event, then the expected number of events will simply be np = μ. The

Poisson distribution function is extremely useful in describing the behavior of unlikely events in large populations. However, in the case where the event is much more likely so that np is large, the situation is somewhat more complicated.

d. The Normal Curve

By assuming that both n and np are large, we move into the realm where all the elements of the binomial coefficients are large. So although the variables are still technically discrete, the unit interval between them remains small compared to their value and we can again replace m by the continuous variable x and np by the continuous variable μ. We can summarize the situation by

$$n >> x \approx np \tilde{} \ \mu >> 1 \ . \tag{7.1.14}$$

Now we may use Sterling's approximation formula,

$$k! \approx e^{-k} k^k \sqrt{2\pi k} \quad , \tag{7.2.15}$$

for large factorials to simplify the binomial coefficients in equation (7.2.9) to get

$$P_B(x) = \frac{n^n p^x q^{(n-x)}}{x^x (n-x)^{(n-x)}} \left(\frac{n}{2\pi k(n-x)} \right)^{1/2} = \left(\frac{np}{x} \right)^x \left(\frac{nq}{n-x} \right)^{n-x} \left(\frac{n}{2\pi k(n-x)} \right)^{1/2} \quad , \tag{7.2.16}$$

Now we add the further restriction that

$$0 < p < 1 \ . \tag{7.2.17}$$

As in the case of the Poisson distribution, np will be the expectation value of x and it is near that value that we will be most interested in the probability distribution. Thus let us describe x in the vicinity of np by defining a small quantity δ so that

$$\delta = x - np \ , \tag{7.2.18}$$

and

$$n-x = n(1-p) - \delta = nq-\delta \ . \tag{7.2.19}$$

Expressing the binomial distribution function given by equation (7.2.16) in terms of δ, we get

$$P_b(x) = \left(1 + \frac{\delta}{np} \right)^{-(\delta+np)} \left(1 - \frac{\delta}{np} \right)^{+(\delta-np)} \left(\frac{n}{2\pi(nq-\delta)(np+\delta)} \right)^{1/2} \quad , \tag{7.2.20}$$

which in terms of logarithms can be written as

$$\ln [P_B(x)Q] \approx - (\delta+np)\ln(1+\delta/np) - (nq-\delta)\ln(1-\delta/nq) \ , \tag{7.2.21}$$

where

$$Q = \sqrt{2\pi pq(1 - \delta/nq)(1 + \delta/np)} \ . \tag{7.2.22}$$

Now we choose to investigate the region in the immediate vicinity of the expected value of x, namely near np. Therefore δ will remain small so that

$$|\delta| < npq \ . \tag{7.2.23}$$

This implies that

$$\left.\begin{aligned} \left|\frac{\delta}{np}\right| &< 1 \\ \left|\frac{\delta}{nq}\right| &< 1 \end{aligned}\right\} , \tag{7.2.24}$$

and the terms in equations (7.2.21) and (7.2.22) can be approximated by

$$\left.\begin{aligned} Q &\cong \sqrt{2\pi npq} \\ \ln\left[1+\frac{\delta}{np}\right] &\cong \frac{\delta}{np} - \frac{\delta^2}{2n^2p^2} + \cdots + \\ \ln\left[1+\frac{\delta}{nq}\right] &\cong \frac{\delta}{nq} - \frac{\delta^2}{2n^2q^2} + \cdots + \end{aligned}\right\} . \tag{7.2.25}$$

Keeping all terms through second order in δ for the logarithmic expansions, equation (7.2.21) becomes

$$\ln[P_B(x)Q] \approx -(\delta+np)(\delta/np)(1-\delta/2np)+(nq-\delta)(\delta/nq)(1-\delta/2nq) \approx -\delta^2/2npq \ , \tag{7.2.26}$$

so that the binomial distribution function becomes

$$f_B(x) \approx \frac{e^{-\delta^2/2npq}}{\sqrt{2\pi npq}} \ . \tag{7.2.27}$$

Replacing np by μ as we did with the Poisson distribution and defining a new quantity σ by

$$\left.\begin{aligned} \sigma^2 &\equiv 2npq = 2np(1-p) \\ \delta &= x - \mu \end{aligned}\right\} , \tag{7.2.28}$$

we can write equation (7.2.27) as

$$f_N(x) \approx \frac{e^{-(x-\mu)^2/\sigma^2}}{\sqrt{2\pi\sigma}} . \tag{7.2.29}$$

This distribution function is known as the *normal distribution function* or just the *normal curve*. Some texts refer to it as the "Bell-shaped" curve. In reality it is a probability density distribution function since, in considering large n, we have passed to the limit of the continuous random variable. While the normal curve is a function of the continuous random variable x, the curve also depends on the expectation value of x (that is μ) and the probability p of a single sampling yielding an event. The sample set n is assumed to be very

much larger than the random variable x which itself is assumed to be very much greater than 1. The meaning of the parameters μ and σ can be seen from Figure 7.2.

Although the normal curve is usually attributed to Laplace, it is its use by Gauss for describing the distribution of experimental or observational error that brought the curve to prominence. It is simply the

large number limit of the discrete binomial probability function. If one makes a series of independent measurements where the error of measurement is randomly distributed about the "true" value, one will obtain an expected value of x equal to μ and the errors will produce a range of values of x having a characteristic width of σ. Used in this context the normal curve is often called the *Gaussian error curve*.

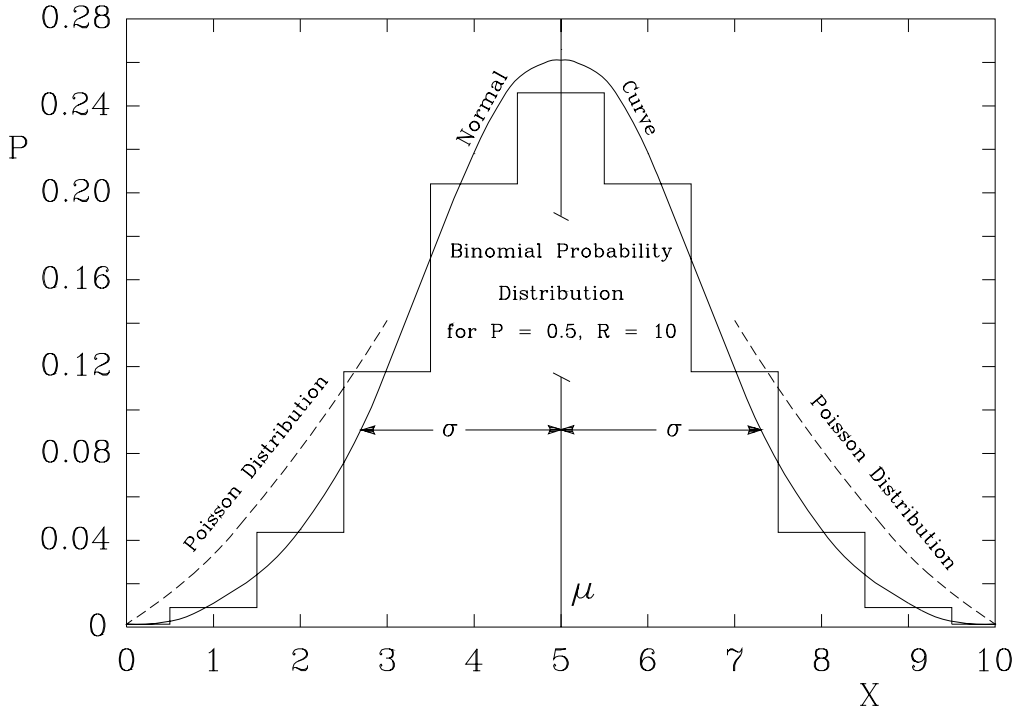


Figure 7.2 shows the normal curve approximation to the binomial probability distribution function. We have chosen the coin tosses so that p = 0.5. *Here* μ *and* σ *can be seen as the most likely value of the random variable* x *and the 'width' of the curve respectively. The tail end of the curve represents the region approximated by the Poisson distribution.*

Because of the basic nature of the sampling assumptions on which it is based, the normal curve plays a major role in testing. This is the curve that students hope will be used when they ask "Will the course be curved?". Of course there are many reasons why a test sample will depart from the normal curve and we will explore some of them in the next chapter. One of the most obvious is that the sample size is small. It should always be remembered that the continuous distribution functions such as the normal curve and the

Poisson distribution are approximations which only approach validity when the sample set is very large. Also, these are not the only distribution functions that arise from probability theory. To demonstrate this point, let us consider some important ones that occur in the physical world.

e. Some Distribution Functions of the Physical World

The foundations of statistical mechanics devote considerable effort to describing the distribution functions for particles that make up our physical world. The random variable that is used turns out to be the total energy of the particles. Most of the details of the derivations are related to the manner by which experiment effectively samples the set of available particles. In the realm of the quantum, the nature of the particles also plays a major role in determining the resulting probability distribution functions. Since the physical world can be viewed as being made up of atomic, or if necessary nuclear, particles, the number of particles in the sample set is usually huge. Therefore the derived distribution functions are usually expressed in terms of functions of the continuous random variable.

Consult a book on statistical mechanics, and you will immediately encounter the terms *microstate*, and *macrostate*. A macrostate is basically a physical distribution of particles with respect to the random variable. A microstate is an artificial concept developed to aid in enumerating the various possible macrostates in the same spirit that permutations aided in the calculation of combinations. The concept of a microstate specifically assumes that the particles are distinguishable. The detailed arrangement of which particles have which values of the random variable determines the microstate. Based on the sampling assumptions, one attempts to find the most probable macrostate which corresponds to the expectation value of the system of particles. In addition, one searches for the number of microstates within a particular macrostate. Since the relative probability of a particular macrostate occurring will be proportional to the number of microstates yielding that macrostate, finding that number is equivalent to finding the probability distribution of macrostates. The most probable macrostate is the one most likely to occur in nature. The basic differences of the distribution functions (i.e. most probable macrostates) that occur can be traced to properties attributed to the particles themselves and to the nature of the space in which they occur.

Consider the total number of particles (N) to be arranged sequentially among m volumes of some space. The total number of sequences or permutations is simply N!. However, within each volume (say the ith volume), there will be N_i particles which yield $N_i!$ indistinguishable sequences which must be removed. If we take the 'volumes' in which we are arranging the particles to be energy w_i then we get the distribution function to be

$$N_i = a_1 e^{-w_i/kT} . \tag{7.2.30}$$

Here T is the temperature of the gas, w_i is the energy of the particles, the constant a_1 depends on the detailed physical makeup of the gas, and k is the Boltzmann constant.

The statistical distribution of particles within the m 'spatial' volumes given by equation (7.2.30) is known as Maxwell-Boltzmann statistics and gives excellent results for a classical gas where the particles can be regarded as distinguishable. In the world of classical physics, the position and momentum of a particle are sufficient to make it distinguishable from all other particles. However, the quantum-mechanical picture of the physical world is quite different and results in different distribution functions. In the world of the

quantum, as a consequence of the Heisenberg uncertainty principle, there is a small volume of 'space' within which particles are indistinguishable. Thus, one may loose any number of particles into one of these 'volumes' and they would all be considered the same kind of particle. Earlier, the sampling order produced permutations that were different from combinations where the sampling order didn't matter. This affected

the probability distributions through the difference between P_m^n and C_m^n. In a similar manner we would expect the distinguishability of particles to affect the nature of the most probable macrostate. In this case the resultant distribution function has the form

$$N_i = a_2 (e^{w_i/kT} - 1) \ , \tag{7.2.31}$$

where the parameter a_2 can be determined in terms of the energy of the particles N_i. This is the distribution function that is suitable for the particles of light called photons and any particles that behave like photons. The distribution function is known as the Bose-Einstein distribution function.

Finally if one invokes the Pauli Exclusion Principle that says you can put no more than two of certain kinds of nuclear particles in the minimum volume designated by the Heisenberg uncertainty principle, then the particle distribution function has the form

$$N_i = a_3 (e^{w_i/kT} + 1) \ , \tag{7.2.32}$$

This is known as the Fermi-Dirac distribution function and again a_3 is determined by the detailed nature of the particles.

Equations (7.2.30 - 32) are just examples of the kinds of probability distribution functions that occur in nature. There are many more. Clearly the knowledge of the entire distribution function provides all the available information about the sample set. However, much of the important information can be obtained from simpler properties of the distribution function.

7.3 Moments of Distribution Functions

Let us begin by defining what is meant by the moment of a function. The moment of a function is the integral of some property of interest, weighted by its probability density distribution function, over the space for which the distribution function is defined. Common examples of such moments can be found in statistics. The mean, or average of a distribution function is simply the first moment of the distribution function and what is called the variance can be simply related to the second moment. In general, if the distribution function is analytic, all the information contained in the function is also contained in the moments of that function.

One of the most difficult problems in any type of analysis is to know what information is unnecessary for the understanding of the basic nature of a particular phenomenon. In other words, what information can be safely thrown away? The complete probability density distribution function representing some phenomenon contains much more information about the phenomenon than we usually wish to know. The process of integrating the function over its defined space in order to obtain a specific moment removes or averages out much of the information about the function. However, it results in parameters which are much easier to interpret. Thus one trades off information for the ability to utilize the result and obtain some explicit properties of the phenomenon. This is a standard 'trick' of mathematical analysis.

We shall define the kth moment of a function f(x) as

$$M_k = \int x^k f(x)dx \Big/ \int f(x)dx \quad , k \geq 1 \; . \tag{7.3.1}$$

The kth moment then is the kth power of the independent variable averaged over all allowed values of the that variable and weighted by the probability density distribution function. Clearly M_0 is unity as we have chosen to normalize the moment by $\int f(x)dx$. This has the practical advantage of making the units of M_k the same as the units and magnitude of an average of x^k in the occasional situation where f(x) is not a normalized probability density function. If the function f(x) is defined for a range of the independent variable $a \leq x \leq b$, then the moments can be written as

$$\left.\begin{aligned} < x > \equiv M_1 &= \int_a^b xf(x)\,dx \Big/ \int_a^b f(x)\,dx \\ < x^2 > \equiv M_2 &= \int_a^b x^2 f(x)\,dx \Big/ \int_a^b f(x)\,dx \\ &\vdots \qquad \vdots \qquad\qquad \vdots \\ < x^k > \equiv M_k &= \int_a^b x^k f(x)\,dx \Big/ \int_a^b f(x)\,dx \end{aligned}\right\} \; . \tag{7.3.2}$$

In equations (7.3.1) and (7.3.2) we have chosen to define moments of the continuous random variable x which is represented by a probability density distribution function f(x). However, we could just as easily define a set of discrete moments where the integral is replaced by a sum and the probability density distribution function is replaced by the probability of obtaining the particular value of the random variable itself. Such moments would then be written as

$$\overline{x^k} \equiv \sum_{i=1}^{N} x_i^k P(x_i) \Big/ \sum_{i=1}^{N} P(x_i) \; . \tag{7.3.3}$$

If the case where the probability of obtaining the random variable is uniform (which it should be if x is really a random variable), equation (7.3.3) becomes

$$\overline{x^k} \equiv \sum_{i=1}^{N} x_i^k P(x_i) \Big/ N \; . \tag{7.3.4}$$

As we shall see, much of statistical analysis is concerned with deciding when the finite or discrete moment can be taken to represent the continuous moment (i.e. when $\overline{x^k} = <x^k>$).

While a complete knowledge of the moments of a analytic function will enable one to specify the function and hence all the information it contains, it is usually sufficient to specify only a few of the

moments in order to obtain most of that information. Indeed, this is the strength and utility of the concept of moments. Four parameters which characterize a probability density distribution function, and are

commonly used in statistics are the *mean*, *variance*, *skewness*, and *kurtosis*. Figure 7.3 shows a graphical representation of these parameters for an arbitrary distribution function.

These four parameters provide a great deal of information about the probability density distribution function f(x) and they are related to the first four moments of the distribution function. Indeed, the *mean of a function* is simply defined as the first moment and is often denoted by the symbol μ. We have already used the symbol σ to denote the 'width' of the normal curve and it is called *the standard deviation* [see equation (7.2.29) and figure 7.2]. In that instance, the 'width' was a measure of the root-mean-square of the departure of the random variable from the mean. The quantity σ^2 is formally called the *variance of the function* and is defined as

$$\sigma^2 \equiv \int (x-\mu)^2 f(x)\,dx = \int x^2 f(x)\,dx - 2\mu\int x f(x)\,dx + \mu^2 \int f(x)\,dx = < x^2 > -\mu^2 \quad . \qquad (7.3.5)$$

Thus the variance clearly contains the information supplied by the second moment of f(x) and is just the *mean-square minus the square of the mean*. We can define a dimensionless parameter, the *skewness of a function*, as a measure of the cube of the departure of f(x) from its mean value so that

$$s^3 \equiv \int (x-\mu)^3 f(x)\,dx \Big/ \sigma^3 \quad = \quad [< x^3 > -\mu(3 < x^2 > -2\mu^2)]/\sigma^3 = [< x^3 > -\mu(< x^2 > -2\sigma^2)]/\sigma^3 . \qquad (7.3.6)$$

The name skewness given s^3 describes what it measures about the function f(x). If the distribution function is symmetric about μ, then the integrand of the integral in equation (7.3.6) is anti-symmetric and $s = 0$. If the skewness is positive then on average f(x) > f(-x), and the distribution function is 'skewed' to the right. The situation is reversed for $s^3 < 0$. Since this parameter describes an aspect of the relative shape of the distribution function, it should be normalized so that it carries no units. This is the reason for the presence of σ^3 in the denominator of equation (7.3.6).

As one would expect, the kurtosis involves information from the fourth moment of the probability density distribution function. Like the skewness, the kurtosis is dimensionless as it is normalized by the square of the variance. Therefore the *kurtosis of a function* is defined as

$$\beta = \int (x-\mu)^4 f(x)dx \Big/ (\sigma^2)^2 \int f(x)dx = [<x^4>-4\mu<x^3>+6\mu^2<x^2>-3\mu^4] . \qquad (7.3.7)$$

For the normal curve given by equation (7.2.29), $\beta = 3$. Thus if $\beta < 3$ the distribution function f(x) is 'flatter' in the vicinity of the maximum than the normal curve while $\beta > 3$ implies a distribution function that is more sharply peaked. Since a great deal of statistical analysis deals with ascertaining to what extent a sample of events represents a normal probability distribution function, these last two parameters are very helpful tools.

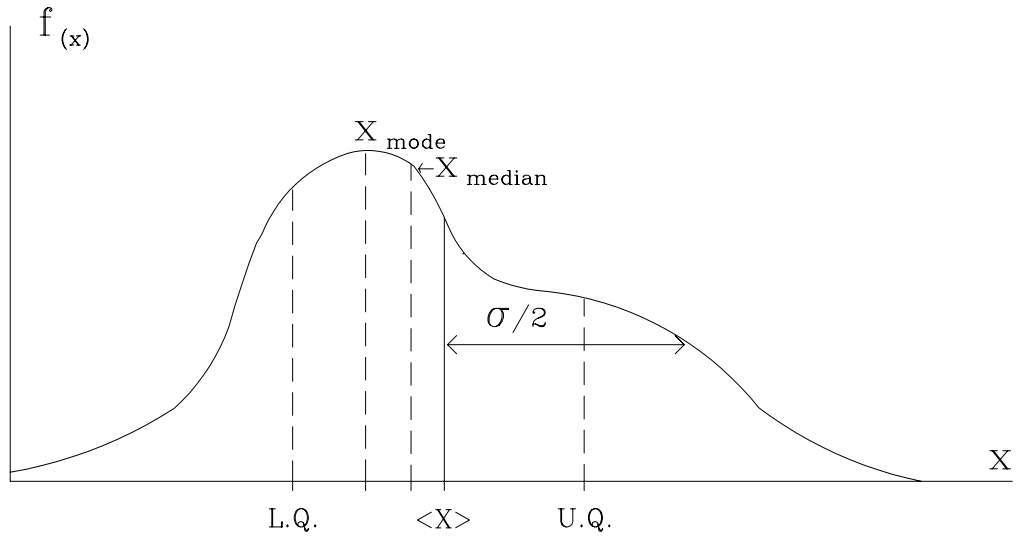


Figure 7.3 shows the mean of a function f(x) *as* <x>. *Note this is not the same as the most likely value of* x *as was the case in Figure 7.2. However, in some real sense* σ *is still a measure of the width of the function. The skewness is a measure of the asymmetry of* f(x) *while the kurtosis represents the degree to which the* f(x) *is 'flattened' with respect to a normal curve. We have also marked the location of the values for the upper and lower quartiles, median and mode.*

There are two other quantities that are often used to characterize a distribution function. These are the median and mode. To understand the notion of median, let us consider the more general concept of a percentile. Consider a probability density function defined for values of the random variable in the interval a ≤ x ≤ b. Now let α represent that fraction of the interval corresponding to x_α so that

$$\alpha = (x_\alpha - a)/(b-a) \quad . \tag{7.3.8}$$

Now we can define the αth *percentile* by

$$\alpha = \int_a^{x_a} f(x)\,dx \Big/ \int_a^b f(x)\,dx \quad . \tag{7.3.9}$$

The value of α is often given in terms of the percentage of the interval a → b, hence the name for x_α. x_α is a measure of the probability that the event will occur in α-percent of the sample tries. When α is given as a fraction 1/4 or 3/4, x_α is known as a *quartile* Q_α. Specifically $x_{1/4}$ is called the *lower quartile*, while $x_{3/4}$ is called the *upper quartile*. The parameter $x_{1/2}$ acquires the special name of *median*. Thus the median is that value of the random variable x for which it is equally probable that an event will occur with x greater or less than $x_{1/2}$. Thus the median is defined by

$$\frac{1}{2} = \int_a^{x_{1/2}} f(x)\,dx \Big/ \int_a^b f(x)\,dx \quad . \tag{7.3.10}$$

Finally, the term *mode* is reserved for the most frequently occurring value of x. This parameter is similar to the expectation value of x discussed in section 7.1 [see equation (7.1.8)]. For continuous distribution functions, this will clearly occur where the curve has a maximum. Thus we may define the *mode of a function* as

$$\left.\frac{df(x)}{dx}\right|_{x=x_m} = 0 \quad . \tag{7.3.11}$$

In this section we have made all of the definitions in terms of the continuous probability density distribution function f(x). The reason for generating these specific parameters is to provide ways of characterizing that function without enumerating it for all values of x. These parameters allow us to compare f(x) to other distribution functions within certain limits and thereby to ascertain the extent to which the conditions that give rise to f(x) correspond to the conditions that yield known probability density distribution functions. Usually one does not have a complete continuous probability density distribution function available for analysis. Instead, one deals with finite samples and attempts to ascertain the nature of the distribution function that governs the results of the sampling. All the parameters defined in this section can be defined for finite samples. Usually the transformation is obvious for those parameters based on moments. Equations (7.3.3) and (7.3.4) give suitable definitions of their discrete definitions. However, in the case of the mode, no simple mathematical formula can be given. It will simply be the most frequently occurring value of the sampled events.

When dealing with finite samples, it is common to define skewness in terms of other more easily calculated parameters of the sample distribution. Some of these definitions are

$$\left.\begin{aligned} s_1 &\equiv (\mu - x_m)/\sigma \\ s_2 &\equiv 3(\mu - x_{1/2})/\sigma \\ s_3 &\equiv 2(x_{3/4} + x_{1/4} - 2x_{1/2})/(x_{3/4} + x_{1/4}) \end{aligned}\right\} \quad . \tag{7.3.12}$$

There are practical reasons for picking any particular one of these definitions, but they are not equivalent so that the user should be careful and consistent when using them.

Let us close this section by considering a hypothetical case of a set of grades given in a course. Suppose that there is a class of ten students who take a twenty-question test with the results given in Table 7.1. Here we encounter a common problem with the use of statistics on small samples. The values for the percentiles do not come out to be integer values so that it is necessary to simply assign them to the nearest integer value. At first look, we find that the median and mode are the same which is required if the scores are to follow the normal curve. However, we might suspect that the curve departs somewhat from the statistically desired result as there are a number of grades that equal the maximum allowed. Therefore let us consider the moments of the grade distribution as given in Table 7.2

Table 7.1

Grade Distribution for Sample Test Results

Student No.	Percentage Grade	Percentile Scores
1	100	
2	100	
3	95	Upper Quartile
4	90	
5	85	Median
6	85	Mode
7	85	
8	70	Lower Quartile
9	60	
10	40	

Table 7.2

Examination Statistics for the Sample Test

STATISTIC	VALUE
Mode	85
$\overline{x}$	81
$\overline{x^2}$	6890
$\overline{x^3}$	605175
$\overline{x^4}$	54319250
Standard Deviation σ	18.138
Skewness s	-1.041
s_1	-0.221
s_2	0.000
s_3	-0.061
Kurtosis β	3.087

Here we see that the mean is somewhat below the median and mode indicating that there are more extreme negative scores than there are positive ones. Or conversely that a larger fraction of the class has scores above the mean than below then mean. This is supported by the value for the skewness. However, here we have four different choices to choose from. The values s_i are often used to allow for the small number statistics.

While they would tend to imply that the curve is skewed somewhat toward negative numbers in the sense suggested by the relative values of the median and mean, the magnitude is not serious. The value of the Kurtosis is obtained from equation (7.3.7) and suggests that the curve is very similar to a normal curve in its flatness.

Thus the instructor responsible for this test could feel confident that the test grades represent a sample of the parent population. In the next chapter we will investigate quantitatively how secure he or she may be in that regard. However, this begs the issue as to whether or not this is a good test. With the mean at 81, one finds 70% of the class with grades between the mean and the top possible grade of 100. Thus 20% of the grading range has been used to evaluate 70% of the class. Excellent discrimination has been obtained for the lower 30% of the class as their grades are spread over 80% of the possible test range. If the goal of the test is to evaluate the relative performance of the class, the spread in scores indicates that this was not done in a very efficient way. Indeed, for the two students who scored 100, no upper limit on their ability has been established. The examiner when establishing the degree of difficulty of the examination so that uniform discrimination is obtained for all segments of the class should consider such factors.

7.4 The Foundations of Statistical Analysis

In making the transition to finite sample sizes we also make the transition from the theoretical realm of probability theory to the more practical world of statistical analysis. Thus we should spend some time understanding the basic tenets of statistics before we use the results.

In science we never prove a theory or hypothesis correct, we simple add confirmatory evidence to an existing body of evidence that supports the theory or hypothesis. However, we may prove a theory or hypothesis to be incorrect or at least invalid for a particular set of circumstances. We investigate the validity of a hypothesis by carrying out experiments or observations. In its purest form, the act of experimentation can be viewed as the measurement of the values of two supposedly related quantities. The relationship is said to be a functional relationship when the quantities are theoretically related [for example y=f(x)] where the relationship involves parameters that are to be determined by the experiment. The entire point of the dual measurement of y and x is to determine those parameters and thereby test the validity of the statement y=f(x). In the physical world no measurement can be carried out with arbitrary precision and therefore there will be errors inherent in both y and x. One of the important roles of statistics is to objectively establish the extent to which the errors affect the determination of the parameters in f(x) and thereby place limits on the extent to which the experiment confirms or rejects the hypothesis. Most statistical analysis is focused on answering the question "To what extent is this experimental result a matter of chance?".

In general, we assume that experiments sample some aspect of the real world producing values of y_i and x_i. We further assume that this sampling could in principle be carried out forever yielding an arbitrarily large set of values of y_i and x_i. In other words there exists an infinite sample space or set which is often called the *parent population*. As a result of sampling error, our sample values will deviate from those of the parent population by an amount, say ε. Each measured value of x_i departs from its 'true' value by some unknown value ε_i. However, we have already seen that if the errors ε_i are not correlated with each other, then ε_i will be distributed in accordance with the binomial distribution. The notion that we are unbiasedly sampling the parent population basically assumes that our error sample will follow the binomial distribution

and this is a central assumption of most statistical analysis. To be sure there are ways we may check the validity of this assumption, but most of the tests comprising statistical inference rely on the assumption being true. It is essentially what we mean when we address the question "To what extent is this experimental result a matter of chance?".

Many students find the terminology of statistics to be a major barrier to understanding the subject. As with any discipline, the specific jargon of the discipline must be understood before any real comprehension can take place. This is particularly true with statistics where the terminology has arisen from many diverse scientific disciplines. We have already noted how a study in population genetics gave rise to the term "regression analysis" to describe the use of Legendre's principle of least squares. Often properly phrased statistical statements will appear awkward in their effort to be precise. This is important for there are multitudinous ways to deceive using statistics badly. This often results from a lack of precision in making a statistical statement or failure to properly address the question "To what extent is this experimental result a matter of chance?".

a. Moments of the Binomial Distribution

Since the binomial distribution, and its associated large sample limit, the normal curve, play such a central role in statistical analysis, we should consider the meaning of the moments of this distribution. As is clear from figure 7.2, the binomial distribution is a symmetric function about its peak value. Thus the mean of the distribution [as given by the first of equations (7.3.2)] will be the peak value of the distribution. From the symmetric nature of the curve, the median will also be the peak value which, in turn, is the mode by definition. Therefore, for the normal curve the median, mean and mode are all equal or

$$\mu_N \equiv \langle x \rangle_N = (x_{½})_N = (x_m)_N \ . \tag{7.4.1}$$

Similarly the various percentiles will be symmetrically placed about the mean. We have already seen that the fourth moment about the mean called the kurtosis takes on the particular value of 3 for the normal curve and it is clear from the symmetry of the normal curve that the skewness will be zero.

The variance σ^2, is simply the square of a characteristic half-width of the curve called the standard deviation σ. Since any area under a normalized probability density distribution function represents the probability that an observation will have a value of x defined by the limits of the area, σ corresponds to the probability that x will lie within σ of μ_N. We may obtain that probability by integrating equation 7.2.29 so that

$$P_N(\sigma) = \frac{1}{\sqrt{2\pi}\sigma}\int_{\mu-\sigma}^{\mu+\sigma} e^{-(x-\mu_N)^2/2\sigma^2}\,dx - \frac{1}{\sqrt{\pi}}\int_{-1}^{+1} e^{-y^2}\,dy = \mathrm{erf}(1) = 0.68269 \ . \tag{7.4.2}$$

Thus the probability that a particular randomly sampled value of x will fall within σ of the mean value μ, is about 68%. Since this argument applies to the error distribution ε, σ is sometime called the *standard error of estimate*. One could ask "What is the range in x corresponding to a 50% probability of x being within that value of the mean"? This will clearly be a smaller number than σ since we wish

$$P_N(x_p) = \frac{1}{\sqrt{2\pi\sigma}} \int_{\mu - x_p}^{\mu + x_p} e^{-(x-\mu_N)^2 / 2\sigma^2} dx = \frac{1}{2} \quad . \tag{7.4.3}$$

The quantity x_p is usually called the *probable error*.

$$x_p = 0.6745\sigma \quad . \tag{7.4.4}$$

The use of the probable error is discouraged since it has become associated with statistical arguments here the author chooses the smaller probable error over the more common standard error simply for its psychological effect.

b. Multiple Variables, Variance, and Covariance

We have discussed the behavior of events that can be characterized by a single random variable distributed according to f(x). What are we to do when the event is the result of two or more variables each characterized by their own probability density distribution functions? Say the event y is related to two variables v and w by

$$y = g(v,w) \quad . \tag{7.4.5}$$

If only two variables are involved y is said to have a *bivariant* distribution. Should the event depend on more than two variables, it has a *multivariant* distribution. Such a situation can result from an experiment where more than one variable must be measured simultaneously in order to characterize the result. Consider the Hall effect in physics where a current flowing perpendicular to a magnetic field will generate a voltage in the direction of the field. In order to investigate this effect one must simultaneously measure the strength of the field and the current as well as the resulting voltage. Each of the independent variables v and w will be characterized by probability density distribution functions that reflect the errors of measurement. Each distribution function will be characterized by the moments we developed for the single random variable. Measurement error will affect the values of both the current and magnetic field and it is a fair question to ask how those errors of measurement affect the expected value of the voltage through the function g(v,w).

Let any variation from the means of y, v, and w be denoted by δ. Then the chain rule of calculus guarantees that

$$(\delta y)^2 = \left[\delta v \frac{\partial y}{\partial v} + \delta w \frac{\partial y}{\partial w} \right]^2 = (\delta v)^2 \left[\frac{\partial g}{\partial v} \right]^2 + 2\delta v \delta w \left[\frac{\partial g}{\partial v} \right] \left[\frac{\partial g}{\partial w} \right] + (\delta w)^2 \left[\frac{\partial g}{\partial w} \right]^2 \quad . \tag{7.4.6}$$

Therefore

$$\sigma_y^2 = \sigma_v^2 \left[\frac{\partial g}{\partial v} \right]^2 + 2\sigma_{vw}^2 \left[\frac{\partial g}{\partial v} \right] \left[\frac{\partial g}{\partial w} \right] + \sigma_w^2 \left[\frac{\partial g}{\partial w} \right]^2 \quad . \tag{7.4.7}$$

Here we have introduced the parameter σ_{vw}^2 which is called the *coefficient of covariance*, or just the *covariance*, as it measures the combined variations from the mean of the variables v and w. For continuous random variables v and w, the coefficient of covariance is defined by

$$\sigma_{vw}^2 \equiv \int\int (v-\mu_v)(w-\mu_w) f(v) h(w)\, dv dw \quad . \tag{7.4.8}$$

Here f(v) and h(v) are the normalized probability density distribution functions of v and w respectively. The coefficient of covariance can be defined over a finite data set as

$$\sigma_{vw}^2 = \sum_{i=1}^{N} (v_i - \mu_v)(w_i - \mu_w) \Big/ N \quad . \tag{7.4.9}$$

Unlike the variance, which in some sense measures the variation of a single y variable against itself, the terms that make up the covariance can be either positive or negative. Indeed, if the probability density distribution functions that govern v and w are symmetric about the mean, then $\sigma_{vw}^2 = 0$. If this is true for a multivariant distribution function, then all the covariances will be zero and

$$\sigma_y^2 = \sum_{i=1}^{N} \sigma_{x_k}^2 \left(\frac{\partial g}{\partial x_k} \right)^2 \quad . \tag{7.4.10}$$

This is a result similar to that obtained in section 6.3 [see equations (6.3.9) - (6.3.11)] for the errors of the least square coefficients and rests on the same assumption of error symmetry. Indeed, we shall see in the next chapter that there is a very close relation between linear least squares, and the statistical methods of regression analysis and analysis of variance.

When one is discussing the moments and properties of the normal curve, there is no question as to the their value. This is a result of the infinite sample size and therefore is not realized for actual cases where the sample is finite. Thus there will be an uncertainty resulting from the error of the sampled items in the mean as well as other moments and it is a fair question to ask how that uncertainty can be estimated. Let us regard the determination of the mean from a finite sample to be the result of a multivariant analysis where

$$\mu = g(x_i) = \sum_{i=1}^{N} x_i \Big/ N \quad . \tag{7.4.11}$$

The partial derivative required by equation (7.4.10) will then yield

$$\frac{\partial g}{\partial x_k} = \frac{1}{N} \quad , \tag{7.4.12}$$

and taking $y = \mu$ we get the variance of the mean to be

$$\sigma_\mu^2 = \sum_{i=1}^{N} \frac{\sigma_{x_k}^2}{N^2} = \frac{\sigma^2}{N} \quad ; \tag{7.4.13}$$

the different observations are all of the same parameter x, and the values of $\sigma^2_{x_k}$ will all be equal.

In order to evaluate the variance of the mean σ^2_μ directly, we require an expression for the variance of a single observation for a finite sample of data. Equation (7.3.5) assumes that the value of the mean is known with absolute precision and so its generalization to a finite data set will underestimate the actual spread in the finite distribution function. Say we were to use one of our observations to specify the value of the mean. That observation would no longer be available to determine other statistical parameters as it could no longer be regarded as independent. So the total number of independent observations would now be N-1 and we could write the variance of a single observation as

$$\sigma_x^2 = \sum_{i=1}^{N} \frac{(x_i - \mu)^2}{(N-1)} \quad . \tag{7.4.14}$$

Therefore, the variance of the mean becomes

$$\sigma_\mu^2 = \sum_{i=1}^{N} \frac{(x_i - \mu)^2}{N(N-1)} \quad . \tag{7.4.15}$$

The factor of (N-1) in the denominator results from the uncertainty of the mean itself. The number of independent observations that go into a statistical analysis are often referred to as the number of *degrees of freedom* of the analysis. Since the equivalent of one observation is required to specify the mean, one degree of freedom is removed from further analysis. It is that degree of freedom required to specify the value of the mean. At any point in a statistical analysis one should always be concerned with the number of degrees of freedom available to specify the solution to the problem. In some real sense, the number of degrees of freedom represents the extent to which the problem is over-determined in the absence of error. Thus in a least square problem with n coefficients to be determined from N data points, there are only (N-n) degrees of freedom. This is the statistical origin of the factor of (N-n) in equation (6.3.26) that specifies the error in the least square coefficients.

c. Maximum Likelihood

Most of statistics is devoted to determining the extent to which a sample population represents the parent population. A corollary to this task is the problem of determining the extent to which the parent population is represented by a normal distribution. We have already seen that the mean, mode, and median are all equal for a normal distribution. This means that the most probable value (i.e. the expectation value) of x is obtained from the mean, median, or mode. For a finite population, these three will not, in general be equal. Is there some way to decide if the differences result simply from chance and a finite random sample, or whether the parent population is not represented by the normal curve? One approach is to reverse the question and ask, "What is the likelihood that the finite sample will result in a particular value for the mean, median, mode or any other statistic?". To answer this question assumes that the probability density distribution for the parent population is known. If this is the case, then one can calculate the probability that a sample of known size (and characteristics) will result from sampling that distribution. Indeed the logarithm of that probability is known as the *likelihood* of the statistic. The value of the likelihood will depend on the particular value of the statistic, which should not be regarded as a variable, as well as the nature of the probability distribution of the parent population. Maximum likelihood algorithms are those that adjust the sampling procedure within the constraints imposed by the definition of the statistic so as to maximize the likelihood of obtaining a particular statistic when sampling the parent population.

Assume that we are interested in determining the most probable value of an event from a sample of a parent population, which does not follow the normal curve. If the distribution function is not symmetric about the mean, then the arithmetic mean will not, in general, be the most probable result (see figure 7.3). However, if we knew the nature of the distribution function of the parent population (i.e. its shape, not its exact values) we could devise a sampling procedure that yielded an accurate value for the mode, which then would be the most probable value for the sampled event. If the probability density function of the parent population is the normal curve, then the mean is that value. In the case of multivariant analysis, least-squares

yields the maximum likelihood values for the coefficients when the parent populations of the various variables are represented by the normal curve.

In the next chapter we will consider some specific ways of determining the nature of the parent population and the extent to which we can believe that the values of the moments accurately sample the parent population. In addition, we will also deal with the problem of multivariant analysis, small sample size and other practical problems of statistical analysis.

Chapter 7 Exercises

1. Find the probability that, from a deck of 52 playing cards, a person can draw exactly:

    a. a pair,
    b. three of a kind,
    c. four of a kind.

2. Calculate the probability that a person sitting third from the dealer in a four person game will be dealt five cards containing:

    a. a pair,
    b. three of a kind,
    c. four of a kind.

    What is the effect of having additional players in the game? Does it matter where the player is located with respect to the other players? If so, why?

3. What is the probability that a single person can draw a five-card straight *or* a flush from a single deck of cards?

4. Calculate the binomial probability distribution function of obtaining "heads" for ten throws of an unbiased coin.

5. Show explicitly how the skewness and the kurtosis are related to the third and fourth moments of the distribution function. Express them in terms of these moments and the mean and variance. Re-express the kurtosis in terms of the fourth moment, the mean variance and skewness.

6. Show that the value for the kurtosis of the normal curve is 3.

7. Obtain expressions for:

    a. the variance of the skewness of a finite sample,
    b. the variance of the kurtosis of a finite sample.

Chapter 7 References and Supplemental Reading

1. Eddington, Sir A.S. "The Philosophy of Physical Science" (1939)

2. Smith, J.G., and Duncan, A.J. "Elementary Statistics and Applications: Fundamentals of the Theory of Statistics", (1944), Mc Graw-Hill Book Company Inc., New York, London, pp. 323.

The basics of probability theory and statistics can be found is a very large number of books. The student should try to find one that is slanted to his/her particular area of interest. Below are a few that he/she may find useful.

1. DeGroot, M.H., "Probability and Statistics" (1975), Addison-Wesley Pub. Co. Inc., Reading, Mass.

2. Miller, I.R., Freund, J.E., and Johnson,R., "Probability and Statistics for Engineers", 4th ed., (1990), Prentice-Hall, Inc. Englewood Cliffs, N.J.

3. Rice, J.A. "Mathematical Statistics and Data Analysis", (1988), Wadsworth and Brooks/Cole Advanced Books and Software, Pacific Grove Cal.

4. Devore, J.L., "Probability and Statistics for Engineering and the Sciences", 2nd ed., (1987), Brooks/Cole Publishing Co. Inc. Monterey Cal.

5. Larsen, R.J., and Marx, M.L., An Introduction to Mathematical Statistics and Its Applications", 2nd ed., (1986) Prentice-Hall, Englewood Cliffs, N.J.

8

Sampling Distributions of Moments, Statistical Tests, and Procedures

• • •

The basic function of statistical analysis is to make judgments about the real world on the basis of incomplete information. Specifically, we wish to determine the nature of some phenomenon based on a finite sampling of that phenomenon. The sampling procedure will produce a distribution of values, which can be characterized by various moments of that distribution. In the last chapter we saw that the distribution of a random variable is given by the binomial distribution function, which under certain limiting conditions can be represented by the normal probability density distribution function and the Poisson distribution function. In addition, certain physical phenomena will follow distribution functions that are non-normal in nature. We shall see that the characteristics, or statistics, of the distribution functions themselves can be characterized by sampling probability density distribution functions. Generally these distribution functions are also non-normal particularly in the small sample limit.

In section 7.4 we determined the variance of the mean which implied that the moments of any sampling could themselves be regarded as sample that would be characterized by a distribution. However, the act of forming the moment is a decidedly non-random process so that the distribution of the moments may not be represented by the normal distribution. Let us consider several distributions that commonly occur in statistical analysis.

8.1 The t, χ^2, and F Statistical Distribution Functions

In practice, the moments of any sampling distribution have values that depend on the sample size. If we were to repeat a finite sample having N values a large number of times, then the various moments of that sample will vary. Since sampling the same parent population generates them all, we might expect the sampling distribution of the moments to approach that of the parent population as the sample size increases. If the parent population is represented by a random variable, its moments will approach those of the normal curve and their distributions will also approach that of the normal curve. However, when the sample size N is small, the distribution functions for the mean, variance and other statistics that characterize the distribution will depart from the normal curve. It is these distribution functions that we wish to consider.

a. The t-Density Distribution Function

Let us begin by considering the range of values for the mean $\bar{x}$ that we can expect from a small sampling of the parent population N. Let us define the amount that the mean $\bar{x}$ of any particular sample departs from the mean of the parent population x_p as

$$t \equiv (\bar{x} - \bar{x}_p)/\sigma_{\bar{x}}. \tag{8.1.1}$$

Here we have normalized our variable t by the best un-biased estimate of the standard deviation of the mean $\sigma_{\bar{x}}$ so as to produce a dimensionless quantity whose distribution function we can discuss without worrying about its units. Clearly the distribution function of t will depend on the sample size N. The differences from the normal curve are represented in Figure 8.1. The function is symmetric with a mean, mode, and skewness equal to zero. However, the function is rather flatter than the normal curve so the kurtosis is greater than three, but will approach three as N increases. The specific form of the t-distribution is

$$f(t) = \frac{\Gamma[\tfrac{1}{2}(N+1)]}{\sqrt{\pi N}\,\Gamma(\tfrac{1}{2}N)}\left[1+\frac{t^2}{N}\right]^{-(N+1)/2}, \tag{8.1.2}$$

which has a variance of

$$\sigma^2{}_t = N/(N-2)\ . \tag{8.1.3}$$

Generally, the differences between the t-distribution function and the normal curve are negligible for N > 30, but even this difference can be reduced by using a normal curve with a variance given by equation (8.1.3) instead of unity. At the out set we should be clear about the difference between the number of samples N and the number of degrees of freedom v contained in the sample. In Chapter 7 (section 7.4) we introduced the concept of "degrees of freedom" when determining the variance. The variance of both a single observation and the mean was expressed in terms of the mean itself. The determination of the mean

reduced the number of independent information points represented by the data by one. Thus the factor of (N-1) represented the remaining independent pieces of information, known as the degrees of freedom, available for the statistic of interest. The presence of the mean in the expression for the t-statistic [equation (8.1.1)] reduces the number of degrees of freedom available for t by one.

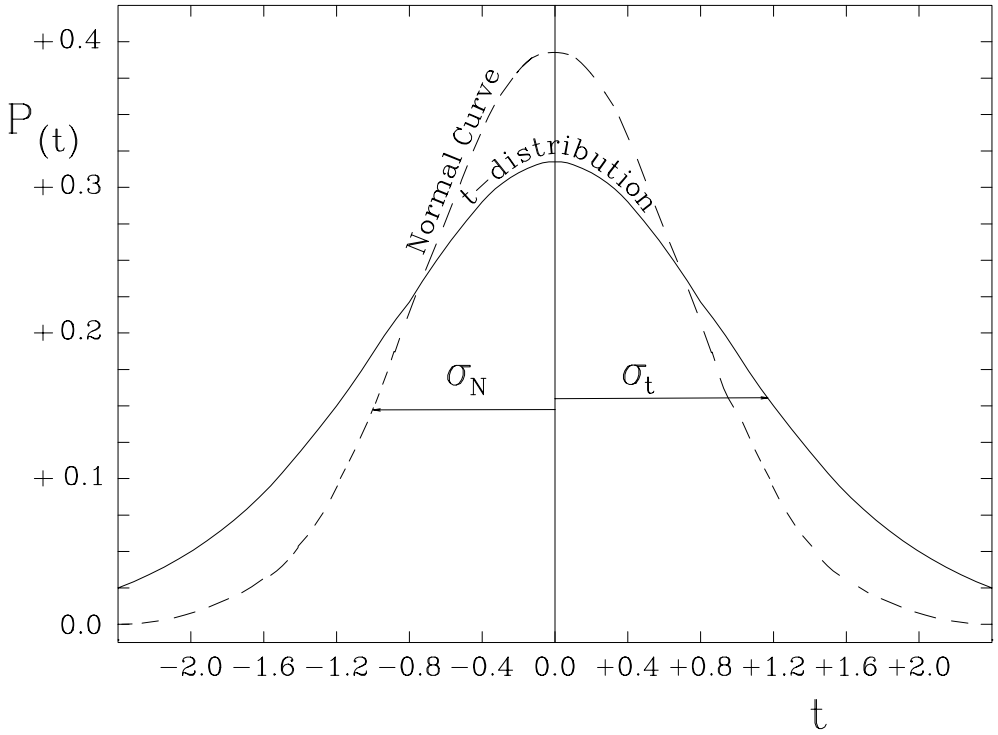


Figure 8.1 shows a comparison between the normal curve and the t*-distribution function for* N=8*. The symmetric nature of the* t*-distribution means that the mean, median, mode, and skewness will all be zero while the variance and kurtosis will be slightly larger than their normal counterparts. As* $N \to \infty$*, the* t*-distribution approaches the normal curve with unit variance.*

b. The χ^2 -Density Distribution Function

Just as we inquired into the distribution of means x that could result from various samples, so we could ask what the distribution of variances might be. In chapter 6 (section 6.4) we introduced the parameter χ^2 as a measure of the mean square error of a least square fit to some data. We chose that symbol with the current use in mind. Define

$$\chi^2 = \sum_{j=1}^{N} (x_j - \overline{x}_j)^2 / \sigma_j^2 \quad , \tag{8.1.4}$$

where σ^2_j is the variance of a single observation. The quantity χ^2 is then sort of a normalized square error. Indeed, in the case where the variance of a single observation is constant for all observations we can write

$$\chi^2 = N\overline{\varepsilon^2 / \sigma^2}\,, \tag{8.1.5}$$

where. ε^2 is the mean square error. However, the value of χ^2 will continue to grow with N so that some authors further normalize χ^2 so that

$$\chi_\nu^2 = \chi^2 / \nu\,. \tag{8.1.6}$$

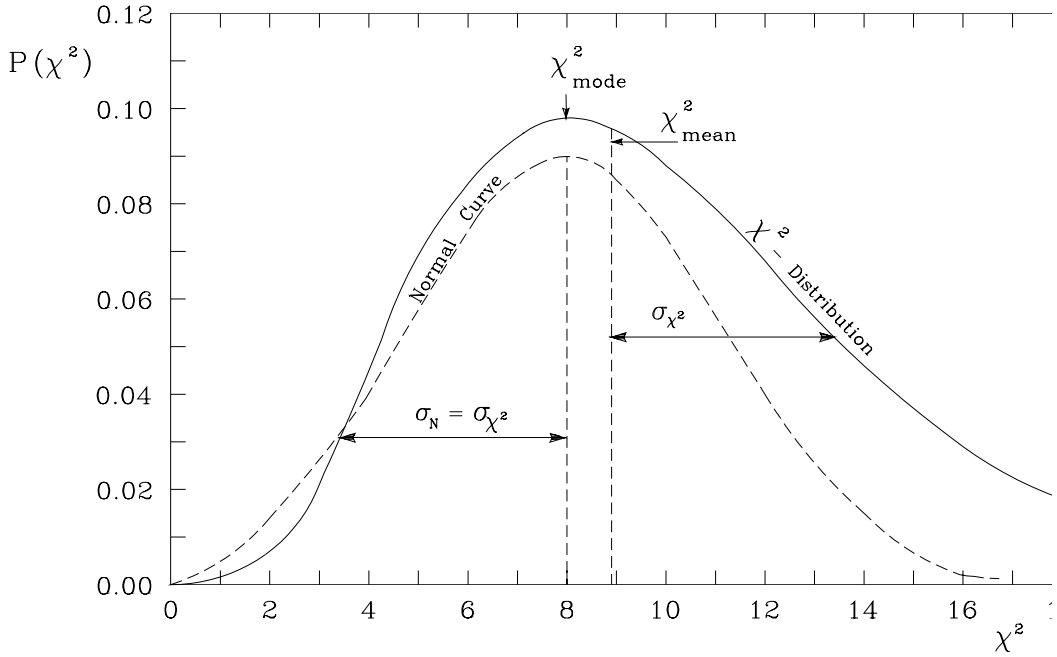


Figure 8.2 compares the χ^2*- distribution with the normal curve. For* N = 10 *the curve is quite skewed near the origin with the mean occurring past the mode* (χ^2 = 8)*. The Normal curve has* μ = 8 *and* σ^2 = 20*. For large* N*, the mode of the* χ^2*- distribution approaches half the variance and the distribution function approaches a normal curve with the mean equal the mode.*

Here the number of degrees of freedom (i.e. the sample size N reduced by the number of independent moments present in the expression) does not appear explicitly in the result. Since χ^2 is intrinsically positive, its distribution function cannot be expected to be symmetric. Figure 8.2 compares the probability density distribution function for χ^2, as given by

$$f(\chi^2) = [2^{N/2}\,\Gamma(\tfrac{1}{2}N)]^{-1}\; e^{-\chi^2/2}(\chi^2)^{\frac{1}{2}(N-2)}\,, \tag{8.1.7}$$

with the normal distribution function.

The moments of the χ^2 density distribution function yield values of the variance, mode, and skewness of

$$\left.\begin{aligned} \sigma^2_{\chi^2} &= 2N \\ \chi^2_m &= N-2 \\ s &= \sqrt{2/N} \end{aligned}\right\} . \tag{8.1.8}$$

As N increases, the mode increases approaching half the variance while the skewness approaches zero. Thus, this distribution function will also approach the normal curve as N becomes large.

c. *The* F-*Density Distribution Function*

So far we have considered cases where the moments generated by the sampling process are all generated from samples of the same size (i.e. the same value of N). We can ask how the sample size could affect the probability of obtaining a particular value of the variance. For example, the χ^2 distribution function describes how values of the variance will be distributed for a particular value of N. How could we expect this distribution function to change *relatively* if we changed N? Let us inquire into the nature of the probability density distribution of the ratio of two variances, or more specifically define F to be

$$F_{12} \equiv \left((\chi_1^2/\nu_1) \Big/ (\chi_2^2/\nu_2) \right) = \left(\chi^2_{\nu_1} \Big/ \chi^2_{\nu_2} \right) . \tag{8.1.9}$$

This can be shown to have the rather complicated density distribution function of the form

$$f(F) = \frac{\Gamma[\frac{1}{2}(N_1+N_2)]\, N_1^{\frac{1}{2}N_1}\, N_2^{\frac{1}{2}N_2}\, F_{12}^{\frac{1}{2}(N_1-1)}}{\Gamma(\frac{1}{2}N_1)\Gamma(\frac{1}{2}N_2)(N_1F+N_2)^{\frac{1}{2}(N_1+N_2)}} = \frac{\Gamma[\frac{1}{2}(\nu_1+\nu_2)]}{\Gamma(\frac{1}{2}\nu_1)\Gamma(\frac{1}{2}\nu_2)} \left[\frac{\nu_1}{\nu_2}\right]^{\nu_1/2} \frac{F_{12}^{(\nu_1-1)/2}}{(1+F_{12}\nu_1/\nu_2)^{(\nu_1+\nu_2)/2}} , \tag{8.1.10}$$

where the degrees of freedom ν_1 and ν_2 are N_1 and N_2 respectively. The shape of this density distribution function is displayed in Figure 8.3.

The mean, mode and variance of F-probability density distribution function are

$$\left.\begin{aligned} \bar{F} &= N_2/(N_2-2) \\ F_{m0} &= \frac{N_2(N_1-2)}{N_1(N_2-2)} \\ \sigma_F^2 &= \frac{2(N_2+N_1-2)N_2^2}{N_1(N_2-4)(N_2-2)^2} \end{aligned}\right\} . \tag{8.1.11}$$

As one would expect, the F-statistic behaves very much like a χ^2 except that there is an additional parameter involved. However, as N_1 and N_2 both become large, the F-distribution function becomes indistinguishable from the normal curve. While N_1 and N_2 have been presented as the sample sizes for two different samplings of the parent population, they really represent the number of independent pieces of information (i.e. the number of degrees of freedom give or take some moments) entering into the determination of the variance σ^2_n or alternately, the value of χ^2_n. As we saw in chapter 6, should the statistical analysis involve a more complicated function of the form $g(x,a_i)$, the number of degrees of freedom will depend on the number of values of a_i. Thus the F-statistic can be used to provide the distribution of variances resulting from a change in the number of values of a_i thereby changing the number of degrees of freedom as well as a change in the sample size N. We shall find this very useful in the next section.

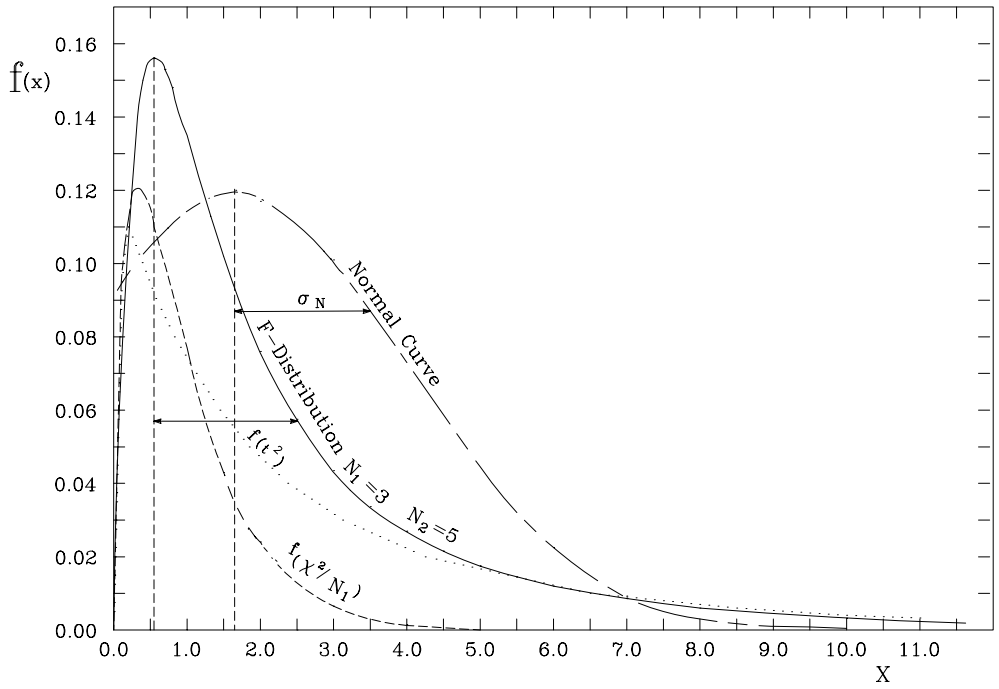


Figure 8.3 shows the probability density distribution function for the F*-statistic with values of* $N_1 = 3$ *and* $N_2 = 5$ *respectively. Also plotted are the limiting distribution functions* $f(\chi^2/N_1)$ *and* $f(t^2)$*. The first of these is obtained from* f(F) *in the limit of* $N_2 \to \infty$*. The second arises when* $N_1 \to 1$*. One can see the tail of the* $f(t^2)$ *distribution approaching that of* f(F) *as the value of the independent variable increases. Finally, the normal curve which all distributions approach for large values of* N *is shown with a mean equal to* $\bar{F}$ *and a variance equal to the variance for* f(F)*.*

Since the t, χ^2, and F density distribution functions all approach the normal distribution function as $N \to \infty$, the normal curve may be considered a special case of the three curves. What is less obvious is that the t- and χ^2 density distribution functions are special cases of the F density distribution. From the defining

equations for t [equation (8.1.1)] and χ^2 [equation(8.1.4)] we see that

$$\lim_{N \to 1} t^2 = \chi^2 \quad , \tag{8.1.12}$$

From equations (8.1.5) and (8.1.6) the limiting value of the normalized or reduced χ^2 is given by

$$\lim_{v \to \infty} \chi_v^2 = 1 \quad , \tag{8.1.13}$$

so that

$$\lim_{\substack{N_1 \to N \\ N_2 \to \infty}} F = \chi^2/N \quad . \tag{8.1.14}$$

Finally t can be related to F in the special case where

$$\lim_{\substack{N_1 \to 1 \\ N_2 \to N}} F = t^2 \quad . \tag{8.1.15}$$

Thus we see that the F probably density distribution function is the general generator for the density distribution functions for t and χ^2 and hence for the normal density distribution function itself.

8.2 The Level of Significance and Statistical Tests

Much of statistical analysis is concerned with determining the extent to which the properties of a sample reflect the properties of the parent population. This could be re-stated by obtaining the probability that the particular result differs from the corresponding property of the parent population by an amount ε. These probabilities may be obtained by integrating the appropriate probability density distribution function over the appropriate range. Problems formulated in this fashion constitute a statistical test. Such tests generally test hypotheses such as "this statistic does not differ from the value of the parent population". Such a hypothesis is often called *null hypothesis* for it postulates no difference between the sample and the value for the parent population. We test this hypothesis by ascertaining the probability that the statement is true or possibly the probability that the statement is false. Statistically, one never "proves" or "disproves" a hypothesis. One simply establishes the probability that a particular statement (usually a null hypothesis) is true or false. If a hypothesis is sustained or rejected with a certain probability p the statement is often said to be *significant* at a percent level corresponding to the probability multiplied by 100. That is, a particular statement could be said to be significant at the 5% level if the probability that the event described could occur by chance is .05.

a. The "Students" t-Test

Say we wish to establish the extent to which a particular mean value $\bar{x}$ obtained from a sampling of N items from some parent population actually represents the mean of the parent population. To do this we must establish some tolerances that we will accept as allowing the statement that $\bar{x}$ is indeed "the same" as $\bar{x}_p$. We can do this by first deciding how often we are willing to be wrong. That is, what is the acceptable probability that the statement is false? For the sake of the argument, lets us take that value to be 5%. We can re-write equation (8.1.1) as

$$\bar{x} = \bar{x}_p \pm \sigma_{\bar{x}} t \quad , \tag{8.2.1}$$

and thereby establish a range δ in $\bar{x}$ given by

$$\delta = \left|\bar{x} - \bar{x}_p\right| = \sigma_{\bar{x}} t \quad , \tag{8.2.2}$$

or for the 5% level as

$$\delta_{(5\%)} = \sigma_{\bar{x}} t_{5\%} \quad , \tag{8.2.3}$$

Now we have already established that the t-distribution depends only on the sample size N so that we may find $t_{5\%}$ by integrating that distribution function over that range of t that would allow for it to differ from the expected value with a probability of 5%. That is

$$0.05 = 2\int_{t_{5\%}}^{\infty} f(t)\,dt = 2\left(1 - \int_0^{t_{5\%}} f(t)\,dt\right) \quad . \tag{8.2.4}$$

The value of t will depend on N and the values of δ that result and are known as the *confidence limits of the 5% level*. There are numerous books that provide tables of t for different levels of confidence for various values of N (e.g. Croxton et al[1]). For example if N is 5, then the value of t corresponding to the 5% level is 2.571. Thus we could say that there is only a 5% chance that $\bar{x}$ differs from $\bar{x}_p$ by more than $2.571\sigma_{\bar{x}}$. In the case where the number of samples increases to $\bar{x}_p$, the same confidence limits drop to 1.96 $\sigma_{\bar{x}}$. We can obtain the latter result simply by integrating the 'tails' of the normal curve until we have enclosed 5% of the total area of the curve. Thus it is important to use the proper density distribution function when dealing with small to moderate sample sizes. There integrals set the confidence limit appropriate for the small sample sizes.

We may also use this test to examine additional hypotheses about the nature of the mean. Consider the following two hypotheses:

a. *The measured mean is greater than the mean of the parent population (i.e* $\bar{x} > \bar{x}_p$*),*

and

b. *The measured mean is less than the mean of the parent population (i.e* $\bar{x} < \bar{x}_p$*)* .

While these hypotheses resemble the null hypothesis, they differ subtly. In each case the probability of meeting the hypothesis involves the frequency distribution of t on just one side of the mean. Thus the factor of two that is present in equation (8.2.4) allowing for both "tails" of the t-distribution in establishing the probability of occurrence is absent. Therefore the confidence limits at the p-percentile are set by

$$\left.\begin{aligned} p_a &= \int_{t_p}^{\infty} f(t)\,dt = 1 - \int_0^{t_p} f(t)\,dt \\ p_b &= \int_{-\infty}^{-t_p} f(t)\,dt = 1 - \int_{-t_p}^{0} f(t)\,dt \end{aligned}\right\} . \tag{8.2.5}$$

Again one should be careful to remember that one never "proves" a hypothesis to be correct, one simply finds that it is not necessarily false. One can say that the data are consistent with the hypothesis at the p-percent level.

As the sample size becomes large and the t density distribution function approaches the normal curve, the integrals in equations (8.2.4) and (8.2.5) can be replaced with

$$\left.\begin{aligned} p &= 2\,\mathrm{erfc}(t_p) = 2[1 - \mathrm{erf}(t_p)] \\ p_{a,b} &= \mathrm{erfc}(\pm t_p) = 1 - \mathrm{erf}(\pm t_p) \end{aligned}\right\} , \tag{8.2.6}$$

where erf(x) is called the error function and erfc(x) is known as the complimentary error function of x respectively. The effect of sample sizes on the confidence limits, or alternately the levels of significance, when estimating the accuracy of the mean was first pointed out by W.S. Gossett who used the pseudonym "Student" when writing about it. It has been known as "Students's t-Test" ever since. There are many other uses to which the t-test may be put and some will be discussed later in this book, but these serve to illustrate its basic properties.

b. The χ^2-test

Since χ^2 is a measure of the variance of the sample mean compared with what one might expect, we can use it as a measure of how closely the sampled data approach what one would expect from the sample of a normally distributed parent population. As with the t-test, there are a number of different ways of expressing this, but perhaps the simplest is to again calculate confidence limits on the value of χ^2 that can be expected from any particular sampling. If we sample the entire parent population we would expect a χ_v^2 of unity. For any finite sampling we can establish the probability that the actual value of χ^2 should occur by chance. Like the t-test, we must decide what probability is acceptable. For the purposes of demonstration, let us say that a 5% probability that χ^2 did occur by chance is a sufficient criteria. The value of χ^2 that represents the upper limit on the value that could occur by chance 5% of the time is

$$0.05 = 2\int_{\chi^2_{5\%}}^{\infty} f(\chi^2, N)\,d\chi^2 = N - \int_0^{\chi^2_{5\%}} f(\chi^2, N)\,d\chi^2 \quad , \tag{8.2.7}$$

which for a general percentage is

$$p = \int_{\chi^2_p}^{\infty} f(\chi^2, N)\,d\chi^2 \quad , \tag{8.2.8}$$

Thus an observed value of χ^2 that is greater than χ^2_p would suggest that the parent population is not represented by the normal curve or that the sampling procedure is systematically flawed.

The difficulty with the χ^2-test is that the individual values of σ^2_i must be known before the calculations implied by equation (8.1.4) can be carried out. Usually there is an independent way of

estimating them. However, there is usually also a tendency to under estimate them. Experimenters tend believe their experimental apparatus performs better than it actually does. This will result in too large a value of an observed chi-squared (i.e. χ^2_o). Both the t-test and the χ^2-test as described here test specific properties of a single sample distribution against those expected for a randomly distributed parent population. How may we compare two different samples of the parent population where the variance of a single observation may be different for each sample?

c. The F*-test*

In section 8.1 we found that the ratio of two different χ^2's would have a sampling distribution given by equation (8.1.10). Thus if we have two different experiments that sample the parent population differently and obtain two different values of χ^2, we can ask to what extent are the two experiments different. Of course the expected value of F would be unity, but we can ask "what is the probability that the actual value occurred by chance?" Again we establish the confidence limits on F_{12} by integrating the probability density distribution function so that

$$p = \int_{F_{12}^{(p)}}^{\infty} f(F)\, dF \quad . \tag{8.2.9}$$

Thus if the observed value of F_{12} exceeds $F_{12}^{(p)}$, then we may suspect that one of the two experiments did not sample the parent population in an unbiased manner. However, satisfying the condition that $F_{12} < F_{12}^{(p)}$ is not sufficient to establish that the two experiments did sample the parent population in the same way. F_{12} might be too small. Note that from equation (8.1.9) we can write

$$F_{12} = 1/F_{21} \quad . \tag{8.2.10}$$

One must then compare F_{21} to its expected value $F_{21}^{(p)}$ given by

$$p = \int_{F_{21}^{(p)}}^{\infty} f(F)\, dF \quad . \tag{8.2.11}$$

Equations (8.2.9) and (8.2.11) are not exactly symmetric so that only in the limit of large ν_1 and ν_2 can we write

$$F > F_{12} > 1/F \quad . \tag{8.2.12}$$

So far we have discussed the cases where the sampled value is a direct measure of some quantity found in the parent population. However, more often than not the observed value may be some complicated function of the random variable x. This was certainly the case with our discussion of least squares in chapter 6. Under these conditions, the parameters that relate y and x must be determined by removing degrees of freedom needed to determine other parameters of the fit from the statistical analysis. If we were to fit N data points with a function having n independent coefficients, then we could, in principle, fit n of the data points exactly leaving only (N-n) points to determine, say, ε^2. Thus there would only be (N-n) degrees of freedom left for statistical analysis. This is the origin of the (N-n) term in the denominator of equation (6.3.26) for the errors (variances) of the least square coefficients that we found in chapter 6. Should the mean be required in subsequent analysis, only (N-n-1) degrees of freedom would remain. Thus we must be careful in determining the number of degrees of freedom when dealing with a problem having multiple parameters. This includes the use of the t-test and the χ^2-test. However, such problems suggest a very powerful application of the F-test. Assume that we have fit some data with a function of n parameters. The χ^2-test and perhaps other considerations suggest that we have not achieved the best fit to the data so that we consider a function with an additional parameter so that there are now a total of (n+1) independent parameters. Now we know that including an additional parameter will remove one more degree of freedom from the analysis and that the

mean square error ε^2 should decrease. The question then becomes, whether or not the decrease in ε^2 represents an amount that we would expect to happen by chance, or by including the additional parameter have we matched some systematic behavior of the parent population. Here the F-test can provide a very useful answer. Both samples of the data are "observationally" identical so that the σ_i^2's for the two χ^2's are identical. The only difference between the two χ's is the loss on one degree of freedom. Under the conditions that σ_i^2s are all equal, the F-statistic takes on the fairly simple form of

$$F = \frac{(n - n - 1)\overline{\varepsilon_n^2}}{(N - n)\overline{\varepsilon_{n=1}^2}} \quad . \tag{8.2.13}$$

However, now we wish to know if F_{12} is greater that what would be expected by chance (i.e. is $F_{12} > F_{12}^{(p)}$). Or answering the question "What is the value of p for which $F_{12} = F_{12}^{(p)}$?" is another way of addressing the problem. This is a particularly simple method of determining when the addition of a parameter in an approximating function produces an improvement which is greater than that to be expected by chance. It is equivalent to setting confidence limits for the value of F and thereby establishing the significance of the additional parameter. Values of the probability integrals that appear in equations (8.2.5), (8.2.6), (8.2.8), (8.2.9), and (8.2.11) can be found in the appendices of most elementary statistics books[1] or the CRC Handbook of tables for Probability and Statistics[2]. Therefore the F-test provides an excellent criterion for deciding when a particular approximation formula, lacking a primary theoretical justification, contains a sufficient number of terms.

d. Kolmogorov-Smirnov Tests

Virtually all aspects of the statistical tests we have discussed so far have been based on ascertaining to what extent a particular property or statistic of a sample population can be compared to the expected statistic for the parent population. One establishes the *"goodness of fit"* of the sample to the parent population on the basis of whether or not these statistics fall within the expected ranges for a random sampling. The parameters such as skewness, kurtosis, t, χ^2, or F, all represent specific properties of the distribution function and thus such tests are often called parametric tests of the sample. Such tests can be definitive when the sample size is large so that the actual value of the parameter represents the corresponding value of the parent population. When the sample size is small, even when the departure of the sampling distribution function from a normal distribution is allowed for, the persuasiveness of the statistical argument is reduced. One would prefer tests that examined the entire distribution in light of the expected parent distribution. Examples of such tests are the Kolmogorov-Smirnov tests.

Let us consider a situation similar to that which we used for the t-test and χ^2-test where the random variable is sampled directly. For these tests we shall used the observed data points, x_i, to estimate the *cumulative probability* of the probability density distribution that characterizes the parent population. Say we construct a histogram of the values of x_i that are obtained from the sampling procedure (see figure 8.4). Now we simply sum the number of points with $x < x_i$, normalized by the total number of points in the sample. This number is simply the probability of obtaining $x < x_i$ and is known as the cumulative probability distribution $S(x_i)$. It is reminiscent of the probability integrals we had to evaluate for the parametric tests [eg. equations (8.2.5),(8.2.8), and (8.2.9)] except that now we are using the sampled probability distribution itself instead of one obtained from an assumed binomial distribution. Therefore we can define $S(x_i)$ by

$$S(x_i) = \frac{1}{N}\sum_{j=1}^{i} n(x_j < x) \quad . \tag{8.2.14}$$

This is to be compared with the cumulative probability distribution of the parent population, which is

$$p(x) = \int_0^x f(z)\,dz \quad . \tag{8.2.15}$$

The statistic which is used to compare the two cumulative probability distributions is the largest departure D_0 between the two cumulative probability distributions, or

$$D_0 \equiv \text{Max} \left| S(x_i) - p(x_i) \right|, \forall x_i \quad . \tag{8.2.16}$$

If we ask what is the probability that the two probability density distribution functions are different (i.e. disproof of the null hypothesis), then

$$\left.\begin{aligned} P_{D_0} &= Q(D_0\sqrt{N}) \\ P_{D_0} &= Q[D_0\sqrt{N_1 N_2/(N_1+N_2)}\,] \end{aligned}\right\}, \tag{8.2.17}$$

where Press et al[3] give

$$Q(x) = 2\sum_{j=1}^{\infty}(-1)^{j-1} e^{-2j^2x^2} \quad . \tag{8.2.18}$$

Equations (8.2.17) simply state that if the measured value of D is D_0 then p is the probability that the null hypothesis is false. The first of equations (8.2.17) applies to the case where the probability density distribution function of the parent population is known so that the cumulative probability required to compute D_0 from equations (8.2.15) and (8.2.16) is known *a priori*. This is known as the Kolmogorov-Smirnov Type 1 test. If one has two different distributions $S_1(x_i)$ and $S_2(x_i)$ and wishes to know if they originate from the same distribution, then one uses the second of equations (8.2.17) and obtains D_0 from $\text{Max} \left| S_1(x_i) - S_2(x_i) \right|$. This is usually called the Kolmogorov-Smirnov Type 2 test.

Note that neither test assumes that the parent population is given by the binomial distribution or the normal curve. This is a major strength of the test as it is relatively independent of the nature of the actual probability density distribution function of the parent population. All of the parametric tests described earlier compared the sample distribution with a normal distribution which may be a quite limiting assumption. In addition, the cumulative probability distribution is basically an integral of the probability density distribution function which is itself a probability that x lies in the range of the integral. Integration tends to smooth out local fluctuations in the sampling function. However, by considering the entire range of the sampled variable x, the properties of the whole density distribution function go into determining the D_0-statistic. The combination of these two aspects of the statistic makes it particularly useful in dealing with small samples. This tends to be a basic property of the non-parametric statistical tests such as the Kolmogorov- Smirnov tests.

We have assumed throughout this discussion of statistical tests that a single choice of the random variable results in a specific sample point. In some cases this is not true. The data points or samples could themselves be averages or collections of data. This data may be treated as being collected in groups or bins. The treatment of such data becomes more complicated as the number of degrees of freedom is no longer calculated as simply as for the cases we have considered. Therefore we will leave the statistical analysis of grouped or binned data to a more advanced course of study in statistics.

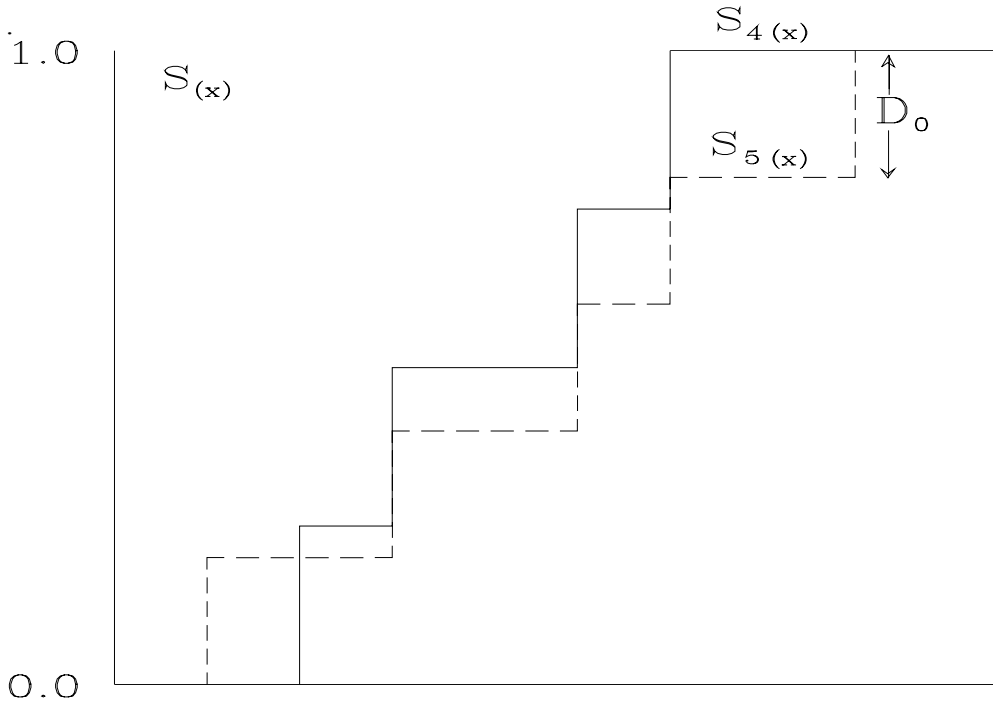


Figure 8.4 shows a histogram of the sampled points x_i *and the cumulative probability of obtaining those points. The Kolmogorov-Smirnov tests compare that probability with another known cumulative probability and ascertain the odds that the differences occurred by chance.*

8.3 Linear Regression, and Correlation Analysis

In Chapter 6 we showed how one could use the principle of least squares to fit a function of several variables and obtain a maximum likelihood or most probable fit under a specific set of assumptions. We also noted in chapter 7 that the use of similar procedures in statistics was referred to as regression analysis. However, in many statistical problems it is not clear which variable should be regarded as the dependent variable and which should be considered as the independent variable. In this section we shall describe some of the techniques for approaching problems where cause and effect cannot be determined.

Let us begin by considering a simple problem involving just two variables, which we will call X_1 and X_2. We have reason to believe that these variables are related, but have no *a priori* reason to believe that either should be regarded as causally dependent on the other. However, in writing any algebraic formalism it is necessary to decide which variables will be regarded as functions of others. For example, we could write

$$X_1 = a_{1.2} + X_2 b_{1.2} \,, \tag{8.3.1}$$

or

$$X_2 = a_{2.1} + X_1 b_{2.1} \quad . \tag{8.3.2}$$

Here we have introduced a notation commonly used in statistics to distinguish the two different sets of a's and b's. The subscript m.n indicates which variable is regarded as being dependent (i.e. the m) and which is to be regarded as being independent (i.e. the n).

a. The Separation of Variances and the Two-Variable Correlation Coefficient

In developing the principle of least squares in chapter 6, we regarded the uncertainties to be confined to the dependent variable alone. We also indicated some simple techniques to deal with the case where there was error in each variable. Here where the very nature of dependency is uncertain, we must extend these notions. To do so, let us again consider the case of just two variables X_1 and X_2. If we were to consider these variables individually, then the distribution represented by the sample of each would be characterized by moments such as $\overline{X}_1$, σ^2_1, $\overline{X}_2$, σ^2_2, etc. However, these variables are suspected to be related. Since the simplest relationship is linear, let us investigate the linear least square solutions where the roles of independence are interchanged. Such analysis will produce solutions of the form

$$\left.\begin{aligned} X_2^c &= a_{1.2} + X_1 b_{2.1} \\ X_1^c &= a_{1.2} + X_2 b_{1.2} \end{aligned}\right\} . \tag{8.3.3}$$

Here we have denoted the values of the dependent variable resulting from the solution by the superscript ^{c}. The lines described by equations (8.3.3) resulting from a least square analysis are known in statistics as *regression lines*. We will further define the departure of any data value X_i from its mean value as a *deviation* x_i. In a similar manner let x_i^c be the calculated deviation of the ith variable. This variable measures the spread in the ith variable as given by the regression equation. Again the subscript denotes the dependent variable. Thus, for a regression line of the form of the first of equations (8.3.3), $(x_2 - x_2^c)$ would be the same as the error ε that was introduced in chapter 6 (see figure 8.5). We may now consider the statistics of the deviations x_i. The mean of the deviations is zero since $a_{m.n} = X_n$, but the variances of the deviations will not be. Indeed they are just related to what we called the mean square error in chapter 6. However, the value of these variances will depend on what variable is taken to be the dependent variable. For our situation, we may write the variances of x_i as

$$\left.\begin{aligned} \sigma^2_{2.1} &= \left(\sum X_2^2 - a_{2.1}\sum X_2 - b_{2.1}\sum X_1 X_2\right)/N \\ \sigma^2_{1.2} &= \left(\sum X_1^2 - a_{1.2}\sum X_1 - b_{1.2}\sum X_1 X_2\right)/N \end{aligned}\right\} \quad . \tag{8.3.4}$$

Some authors[4] refer to these variances as *first-order variances*. While the origin of equations (8.3.4) is not immediately obvious, it can be obtained from the analysis we did in chapter 6 (section 6.3). Indeed, the right hand side of the first of equations (8.3.4) can be obtained by combining equations (6.3.24) and (6.3.25) to get the term in the large parentheses on the right hand side of equation (6.3.26). From that expression it is clear that

$$\sigma^2_{1.2} = \overline{w\varepsilon^2} \quad . \tag{8.3.5}$$

The second of equations (8.3.4) can be obtained from the first by symmetry. Again, the mean of x_i^c is clearly zero but its variance will not be. It is simple a measure in the spread of the computed values of the

dependent variable. Thus the total variance σ^2_i will be the sum of the variance resulting from the relation between X_1 and X_2 (i.e. $\sigma^2\, x^c_i$) and the variance resulting from the failure of the linear regression line to accurately represent the data. Thus

$$\left.\begin{aligned}\sigma_1^2 &= \sigma^2_{x_1^c} + \sigma^2_{1.2}\\ \sigma_2^2 &= \sigma^2_{x_2^c} + \sigma^2_{2.1}\end{aligned}\right\}\ . \tag{8.3.6}$$

The division of the total variance σ^2_i into parts resulting from the relationship between the variables X_1 and X_2 and the failure of the relationship to fit the data allow us to test the extent to which the two variables are related. Let us define

$$r_{12} = \frac{\sum X_1 X_2}{N\sigma_1\sigma_2} = \pm\left(\frac{\sigma^2_{x_1^c}}{\sigma_1^2}\right)^{1/2} = \pm\left(\frac{\sigma^2_{x_2^c}}{\sigma_2^2}\right)^{1/2} = \pm\left(1-\frac{\sigma^2_{1.2}}{\sigma_1^2}\right)^{1/2} = \pm\left(1-\frac{\sigma^2_{2.1}}{\sigma_2^2}\right)^{1/2} = r_{21}\ . \tag{8.3.7}$$

The quantity r_{ij} is known as the Pearson correlation coefficient after Karl Pearson who made wide use of it. This simple correlation coefficient r_{12} measures the way the variables X_1 and X_2 change with respect to their means and is normalized by the standard deviations of each variable. However, the meaning is perhaps more clearly seen from the form on the far right hand side of equation (8.3.7). Remember σ_2 simply measures the scatter of X_{2j} about the mean X_2, while $\sigma_{2.1}$ measures the scatter of X_{2j} about the regression line. Thus, if the variance $\sigma^2_{2.1}$ accounts for the entire variance of the dependent variable X_2, then the correlation coefficient is zero and a plot of X_2 against X_1 would simply show a random scatter diagram. It would mean that the variance $\sigma^2\, x_2^c$ would be zero meaning that none of the total variance resulted from the regression relation. Such variables are said to be uncorrelated. However, if the magnitude of the correlation coefficient is near unity then $\sigma^2_{2.1}$ must be nearly zero implying that the total variance of X_2 is a result of the regression relation. The definition of r as given by the first term in equation (8.3.7) contains a sign which is lost in the subsequent representations. If an increase in X_1 results in a decrease in X_2 then the product of the deviations will be negative yielding a negative value for r_{12}. Variables which have a correlation coefficient with a large magnitude are said to be highly correlated or anti-correlated depending on the sign of r_{12}. It is worth noting that $r_{12} = r_{21}$, which implies that it makes no difference which of the two variables is regarded as the dependent variable.

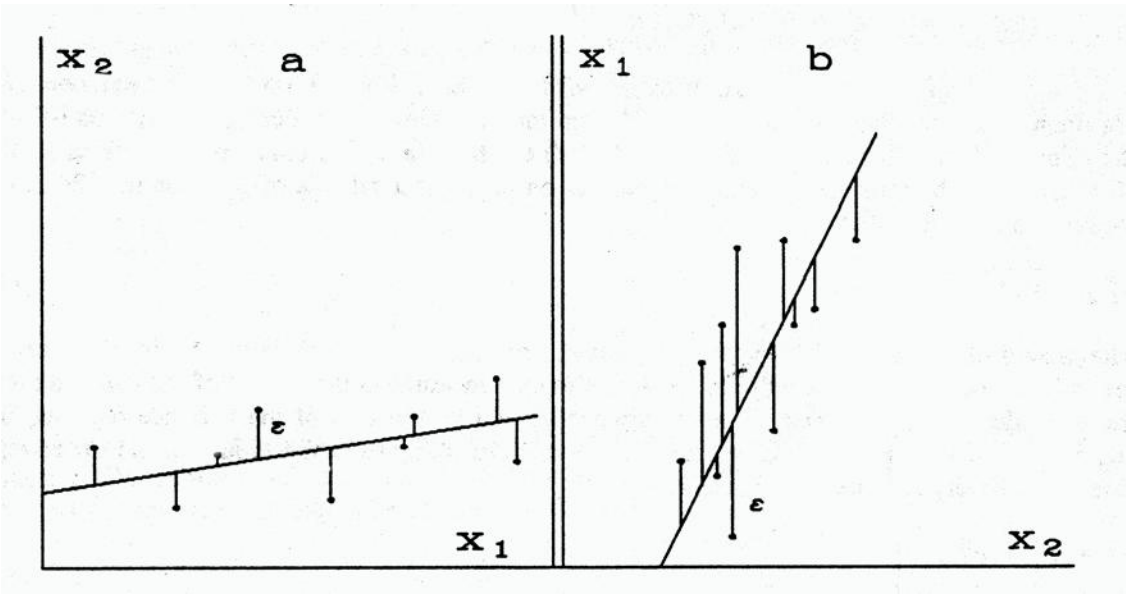


Figure 8.5 shows the regression lines for the two cases where the variable X_2 is regarded as the dependent variable (panel a) and the variable X_1 is regarded as the dependent variable (panel b).

b. The Meaning and Significance of the Correlation Coefficient

There is a nearly irresistible tendency to use the correlation coefficient to imply a causal relationship between the two variables X_1 and X_2. The symmetry of $r_{12}=r_{21}$ shows that this is completely unjustified. The correlation statistic r_{12} does not distinguish which variable is to be considered the dependent variable and which is to be considered the independent variable. But this is the very basis of causality. One says that A causes B, which is very different than B causing A. The correlation coefficient simply measures the relation between the two. That relation could be direct, or result from relations that exist between each variable and additional variables, or simply be a matter of the chance sampling of the data. Consider the following experiment. A scientist sets out to find out how people get from where they live to a popular beach. Researchers are employed to monitor all the approaches to the beach and count the total number of people that arrive on each of a number of days. Say they find the numbers given in Table 8.1.

Table 8.1

Sample Beach Statistics for Correlation Example

Day	Total # Going to the Beach	# Taking the Ferry	# Taking the Bus
1	10000	100	1000
2	20000	200	500
3	5000	50	2000
4	40000	400	250

If one carries out the calculation of the correlation coefficient between the number taking the Ferry and the number of people going to the beach one would get $r_{12}=1$. If the researcher didn't understand the meaning of the correlation coefficient he might be tempted to conclude that all the people who go to the beach take the Ferry. That, of course, is absurd since his own research shows some people taking the bus. However, a correlation between the number taking the bus and the total number of people on the beach would be negative. Should one conclude that people only take the bus when they know nobody else is going to the beach? Of course not. Perhaps most people drive to the beach so that large beach populations cause such congestion so that busses find it more difficult to get there. Perhaps there is no causal connection at all. Can we at least rule out the possibility that the correlation coefficient resulted from the chance sampling? The answer to this question is yes and it makes the correlation coefficient a powerful tool for ascertaining relationships.

We can quantify the interpretation of the correlation coefficient by forming hypotheses as we did with the mono-variant statistical tests and then testing whether the data supports or rejects the hypotheses. Let us first consider the null hypothesis that there is no correlation in the parent population. If this hypothesis is discredited, then the correlation coefficient may be considered significant. We may approach this problem by means of a t-test. Here we are testing the probability of the occurrence of a correlation coefficient r_{12} that is significantly different from zero and

$$t = r_{12}\left(\frac{(n-2)}{1-r_{12}^2}\right) \quad . \tag{8.3.8}$$

The factor of (N-2) in the numerator arises because we have lost two degrees of freedom to the constants of the linear regression line. We can then use equations (8.2.5) to determine the probability that this value of t (and hence r_{12}) would result from chance. This will of course depend on the number of degrees of freedom (in this case N-2) that are involved in the sample. Conversely, one can turn the problem around and find a value of t for a given p and v that one considers significant and that sets a lower limit to the value for r_{12} that would support the hypothesis that r_{12} occurred by chance. For example, say we had 10 pairs of data points which we believed to be related, but we would only accept the probability of a chance occurrence of .1% as being significant. Then solving equation (8.3.8) for r_{12} we get

$$r_{12} = t(v+t^2)^{-½} \quad . \tag{8.3.9}$$

Consulting tables[2] that solve equations (8.2.5) we find the boundary value for t is 4.587 which leads to a minimum value of r = 0.851. Thus, small sample sizes can produce rather large values for the correlation coefficient simply from the chance sampling. Most scientists are very circumspect about moderate values of the correlation coefficient. This probably results from the fact that causality is not guaranteed by the correlation coefficient and the failure of the null hypothesis is not generally taken as strong evidence of significance.

A second hypothesis, which is useful to test, is appraising the extent to which a given correlation coefficient represents the value present in the parent population. Here we desire to set some confidence limits as we did for the mean in section 8.2. If we make the transformation

$$z = ½\,\ell n\,[(1+r_{12})/(1-r_{12})] = \tanh^{-1}(r_{12}) \quad , \tag{8.3.10}$$

then the confidence limits on z are given by

$$\delta z = t_p \sigma_z \ , \tag{8.3.11}$$

where

$$\sigma_z \tilde{} \ [N-(8/3)]^{1/2} \ . \tag{8.3.12}$$

If for our example of 10 pairs of points we ask what are the confidence limits on a *observed* value of r_{12}=0.851 at the 5% level, we find that t=2.228 and that δz=0.8227. Thus we can expect the value of the parent population correlation coefficient to lie between $0.411<r_{12}<0.969$. The mean of the z distribution is

$$z = \tfrac{1}{2}\{ \ell n \ [(1+r_p)/(1-r_p)] + r_p/(N-1)\} \ . \tag{8.3.13}$$

For our example this leads to the best unbiased estimator of $r_p = 0.837$. This nicely illustrates the reason for the considerable skepticism that most scientists have for small data samples. To significantly reduce these limits, σ_z should be reduced at least a factor of three which implies an increase in the sample size of a factor of ten. In general, many scientists place little faith in a correlation analysis containing less than 100 data points for reasons demonstrated by this example. The problem is two-fold. First small sample correlation coefficients must exhibit a magnitude near unity in order for it to represent a statistically significant relationship between the variables under consideration. Secondly, the probability that the correlation coefficient lies near the correlation coefficient of the parent population is small for a small sample. For the correlation coefficient to be meaningful, it must not only represent a relationship in the sample, but also a relationship for the parent population.

c. Correlations of Many Variables and Linear Regression

Our discussion of correlation has so far been limited to two variables and the simple Pearson correlation coefficient. In order to discuss systems of many variables, we shall be interested in the relationships that may exist between any two variables. We may continue to use the definition given in equation (8.3.7) in order to define a correlation coefficient between any two variables X_i and X_j as

$$r_{ij} = \Sigma \ X_i X_j \ / \ N\sigma_i\sigma_j \ . \tag{8.3.14}$$

Certainly the correlation coefficients may be evaluated by brute force after the normal equations of the least square solution have been solved. Given the complete multi-dimensional regression line, the deviations required by equation (8.3.14) could be calculated and the standard deviations of the individual variables obtained. However, as in finding the error of the least square coefficients in chapter 6 (see section 6.3), most of the require work has been done by the time the normal equations have been solved. In equation (6.3.26) we estimated the error of the least square coefficients in terms of parameters generated during the establishment and solution of the normal equations. If we choose to weight the data by the inverse of the experimental errors ε_i, then the errors can be written in terms of the variance of a_j as

$$\sigma^2(a_j) = C_{jj} = \sigma_j^2 \ . \tag{8.3.15}$$

Here C_{jj} is the diagonal element of the inverse matrix of the normal equations. Thus it should not be surprising that the off-diagonal elements of the inverse matrix of the normal equations are the covariances

$$\sigma_{ij}^2 = C_{ij} \quad . \tag{8.3.16}$$

of the coefficients a_i and a_j as defined in section 7.4 [see equation (7.4.9)]. An inspection of the form of equation (7.4.9) will show that much of what we need for the general correlation coefficient is contained in the definition of the covariance. Thus we can write

$$r_{ij} = \sigma_{ij}^2 / \sigma_i \sigma_j \quad . \tag{8.3.17}$$

This allows us to solve the multivariant problems of statistics that arise in many fields of science and investigate the relationships between the various parameters that characterize the problem. Remember that the matrix of the normal equations is symmetric so that the inverse is also symmetric. Therefore we find that

$$r_{ij} = r_{ji} \quad . \tag{8.3.18}$$

Equation (8.3.18) generalizes the result of the simple two variable correlation coefficient that no cause and effect result is implied by the value of the coefficient. A large value of the magnitude of the coefficient simply implies a relationship may exist between the two variables in question. Thus correlation coefficients only test the relations between each set of variables. But we may go further by determining the statistical significance of those correlation coefficients using the t-test and confidence limits given earlier by equations (8.3.8)-(8.3.13).

d Analysis of Variance

We shall conclude our discussion of the correlation between variables by briefly discussing a discipline known as the *analysis of variance*. This concept was developed by R.A. Fisher in the 1920's and is widely used to search for variables that are correlated with one another and to evaluate the reliability of testing procedures. Unfortunately there are those who frequently make the leap between correlation and causality and this is beyond what the method provides. However, it does form the basis on which to search for causal relationships and for that reason alone it is of considerable importance as an analysis technique.

Since its introduction by Fisher, the technique has been expanded in many diverse directions that are well beyond the scope of our investigation so we will only treat the simplest of cases in an attempt to convey the flavor of the method. The name analysis the variance is derived from the examination of the variances of collections of different sets of observed data values. It is generally assumed from the outset that the observations are all obtained from a parent population having a normal distribution and that they are all independent of one another. In addition, we assume that the individual variances of each single observation are equal. We will use the method of least squares in describing the formalism of the analysis, but as with many other statistical methods different terminology is often used to express this venerable approach.

The simplest case involves one variable or "factor", say y_i. Let there be m experiments that each collect a set of n_j values of y. Thus we could form m average values of $\bar{y}$ for each set of values that we shall label $\bar{y}_j$. It is a fair question to ask if the various means $\bar{y}_j$ differ from one another by more than chance. The general approach is not to compare the individual means with one another, but rather to consider the means as a group and determine their variance. We can then compare the variance of the means with the estimated variances of each member within the group to see if that variance departs from the overall variance of the group by more than we would expect from chance alone.

First we wish to find the maximum likelihood values of these estimates of $\bar{y}_j$ so we shall use the formalism of least squares to carry out the averaging. Lets us follow the notation used in chapter 6 and denote the values of $\bar{y}_j$ that we seek as a_j. We can then describe our problem by stating the equations we would like to hold using equations (6.1.10) and (6.1.11) so that

$$\phi \vec{a} = \vec{y}\,, \tag{8.3.19}$$

where the non-square matrix ϕ has the rather special and restricted form

$$\phi_{ik} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 1 & 0 & \cdots & 0 \end{pmatrix} n_1 \quad \begin{pmatrix} 0 & 1 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 1 & \cdots & 0 \end{pmatrix} n_1 \quad \vdots \quad \begin{pmatrix} 0 & 0 & \cdots & 1 \\ 0 & 0 & \cdots & 1 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} n_1 \quad . \tag{8.3.20}$$

This matrix is often called the *design matrix* for analysis of variance. Now we can use equation (6.1.12) to generate the normal equations, which for this problem with one variable will have the simple solution

$$a_j = n_j^{-1} \sum_{i=1}^{n_j} y_{ij} \quad . \tag{8.3.21}$$

The over all variance of y will simply be

$$\sigma^2(y) = n^{-1} \sum_{j=i}^{m} \sum_{i=1}^{n_j} (y_{ij} - \bar{y}_j)^2 \quad , \tag{8.3.22}$$

by definition, and

$$n = \sum_{j=1}^{m} n_j \quad . \tag{8.3.23}$$

We know from least squares that under the assumptions made regarding the distribution of the y_j's that the a_j's are the best estimate of the value of y_j (i.e.y^0_j), but can we decide if the various values of y^0_j are all equal? This is a typical statistical hypothesis that we would like to confirm or reject. We shall do this by investigating the variances of a_j and comparing them to the over-all variance. This procedure is the source of the name of the method of analysis.

Let us begin by dividing up the over-all variance in much the same way we did in section 8.3a so that

$$\sum_{j=1}^{m}\sum_{i=1}^{n_j}\frac{(y_{ij}-\bar{y}_j^{0})^2}{\sigma^2}=\sum_{j=1}^{m}\left[\left(\sum_{i=1}^{n_j}\frac{(y_{ij}-\bar{y}_j)^2}{\sigma^2}\right)+\frac{n_j(\bar{y}_j-\bar{y}_j^{0})^2}{\sigma^2}\right] . \tag{8.3.24}$$

The term on the left is just the sum of square of n_j independent observations normalized by σ^2 and so will follow a χ^2 distribution having n degrees of freedom. This term is nothing more than the total variation of the observations of each experiment set about their true means of the parent populations (i.e. the variance if the true mean weighted by the inverse of the variance of the observed mean). The two terms of the right will also follow the χ^2 distribution function but have n-m and m degree of freedom respectively. The first of these terms is the total variation of the data about the observed sample means while the last term represents the variation of the sample means themselves about their true means. Now define the overall means for the observed data and parent populations to be

$$\left.\begin{aligned}\bar{y}&=\frac{1}{n}\sum_{j=1}^{m}\sum_{i=1}^{n_j}y_{ij}=\frac{1}{n}\sum_{j=1}^{m}n_j\bar{y}_j\\ \bar{y}^{0}&=\frac{1}{n}\sum_{j=1}^{m}n_j\bar{y}_j^{0}\end{aligned}\right\} . \tag{8.3.25}$$

respectively. Finally define

$$a_j^{0}\equiv\bar{y}_j^{0}-\bar{y}^{0} , \tag{8.3.26}$$

which is usually called the *effect* of the factor y^0 and is estimated by the least square procedure to be

$$a_j=\bar{y}_j-\bar{y} . \tag{8.3.27}$$

We can now write the last term on the right hand side of equation (8.3.24) as

$$\sum_{j=1}^{m}\frac{n_j(\bar{y}_j-\bar{y}_j^{0})^2}{\sigma^2}=\sum_{j=1}^{m}\frac{n_j(\bar{y}_j-\bar{y}-a_j^{0})^2}{\sigma^2}+\frac{n(\bar{y}-\bar{y}^{0})}{\sigma^2} , \tag{8.3.28}$$

and the first term on the right here is

$$\sum_{j=1}^{m}\frac{n_j(\bar{y}_j-\bar{y}-a_j^{0})^2}{\sigma^2}=\sum_{j=1}^{m}\frac{n_j(a_j-a_j^{0})^2}{\sigma^2} , \tag{8.3.29}$$

and the definition of α_j allows us to write that

$$\sum_{j=1}^{m}a_j=0 . \tag{8.3.30}$$

However, should any of the α_j^0's not be zero, then the results of equation (8.3.29) will not be zero and the assumptions of this derivation will be violated. That basically means that one of the observation sets does not sample a normal distribution or that the sampling procedure is flawed.

We may determine if this is the case by considering the distribution of the first term on the right hand side of equation (8.3.28). Equation (8.3.28) represents the further division of the variation of the first term on the right of equation (8.3.24) into two new terms. This term was the total variation of the

observations about their sample means and so would follow a χ^2-distribution having n-m degrees of freedom. As can be seen from equation (8.3.29), the first term on the right of equation (8.3.28) represents the variation of the sample effects about their true value and therefore should also follow a χ^2-distribution with m-1 degrees of freedom. Thus, if we are looking for a single statistic to test the assumptions of the analysis, we can consider the statistic

$$Q = \left[\sum_{j=1}^{m} n_j(\bar{y}_j - \bar{y})/(m-1)\right] \Bigg/ \left[\sum_{j=1}^{m}\sum_{i=1}^{n_j}(y_{ij} - \bar{y}_j)^2/(n-m)\right], \tag{8.3.31}$$

which, by virtue of being the ratio of two terms having χ^2-distributions, will follow the distribution of the F-statistic and can be written as

$$Q = \frac{(n-m)\sum_{j=1}^{m}(n_j\bar{y}_j^2 - n\bar{y})/(m-1)}{\left[\sum_{j=1}^{m}\sum_{i=1}^{n_j} y_{ij}^2\right] - \sum_{j=1}^{m} n_j\bar{y}_j^2}. \tag{8.3.32}$$

Thus we can test the hypothesis that all the effects α^0_j are zero by comparing the results of calculating Q[(n-m),(m-1)] with the value of F expected for any specified level of significance. That is, if $Q>F_c$, where F_c is the value of F determined for a particular level of significance, then one knows that the α_j^0's are not all zero and at least one of the sets of observations is flawed.

In development of the method for a single factor or variable, we have repeatedly made use of the additive nature of the variances of normal distributions [i.e. equations (8.3.24) and (8.3.28)]. This is the primary reason for the assumption of "normality" on the parent population and forms the foundation for analysis of variance. While this example of an analysis of variance is for the simplest possible case where the number of "factors" is one, we may use the technique for much more complicated problems employing many factors. The philosophy of the approach is basically the same as for one factor, but the specific formulation is lengthy and beyond the scope of this book.

This just begins the study of correlation analysis and the analysis of variance. We have not dealt with multiple correlation, partial correlation coefficients, or the analysis of covariance. All are of considerable use in exploring the relationship between variables. We have again said nothing about the analysis of grouped or binned data. The basis for analysis of variance has only been touched on and the testing of nonlinear relationships has not been dealt with at all. We will leave further study in these areas to courses specializing in statistics. While we have discussed many of the basic topics and tests of statistical analysis, there remains one area to which we should give at least a cursory look.

8.4 The Design of Experiments

In the last section we saw how one could use correlation techniques to search for relationships between variables. We dealt with situations where it was even unclear which variable should be regarded as the dependent variable and which were the independent variables. This is a situation unfamiliar to the

physical scientist, but not uncommon in the social sciences. It is the situation that prevails whenever a new phenomenology is approached where the importance of the variables and relationships between them are totally unknown. In such situations statistical analysis provides the only reasonable hope of sorting out and identifying the variables and ascertaining the relationships between them. Only after that has been done can one begin the search for the causal relationships which lead to an understanding upon which theory can be built.

Generally, physical experimentation sets out to test some theoretical prediction and while the equipment design of the experiment may be extremely sophisticated and the interpretation of the results subtle and difficult, the philosophical foundations of such experiments are generally straightforward. Where there exists little or no theory to guide one, experimental procedures become more difficult to design. Engineers often tread in this area. They may know that classical physics could predict how their experiments should behave, but the situation may be so complex or subject to chaotic behavior, that actual prediction of the outcome is impossible. At this point the engineer will find it necessary to search for relationships in much the same manner as the social scientist. Some guidance may come from the physical sciences, but the final design of the experiment will rely on the skill and wisdom of the experimenter. In the realm of medicine and biology theoretical description of phenomena may be so vague that one should even relax the term variable which implies a specific relation to the result and use the term "*factor*" implying a parameter that may, or may not, be relevant to the result. Such is the case in the experiments we will be describing.

Even the physical sciences, and frequently the social and biological sciences undertake surveys of phenomena of interest to their disciplines. A survey, by its very nature, is investigating factors with suspected but unknown relationships and so the proper layout of the survey should be subject to considerable care. Indeed, Cochran and Cox[5] have observed

> *"Participation in the initial stages of an experiment in different areas of research leads to the strong conviction that too little time and effort is put into the planning of experiments. The statistician who expects that his contribution to the planning will involve some technical matter in statistical theory finds repeatedly that he makes a much more valuable contribution simply by getting the investigator to explain clearly why he is doing the experiment, to justify experimental treatments whose effects he expects to compare and to defend his claim that the completed experiment will enable his objectives to be realized. ..."*

Therefore, it is appropriate that we spend a little time discussing the language and nature of experimental design.

At the beginning of chapter 7, we drew the distinction between data that were obtained by observation and those obtained by experimentation. Both processes are essentially *sampling* a parent population. Only in the latter case, does the scientist have the opportunity to partake in the specific outcome. However, even the observer can arrange to carry out a well designed survey or a badly designed survey by choosing the nature and range of variables or factors to be observed and the equipment with which to do the observing.

The term experiment has been defined as "a considered course of action aimed at answering one or

more carefully framed questions". Therefore any experiment should meet certain criteria. It should have a specific and well defined mission or objective. The list of relevant variables, or factors, should be complete. Often this latter condition is difficult to manage. In the absence of some theoretical description of the phenomena one can imagine that a sequence of experiments may be necessary simply to establish what are the relevant factors. As a corollary to this condition, every attempt should be made to exclude or minimize the effect of variables beyond the scope or control of the experiment. This includes the bias of the experimenters themselves. This latter consideration is the source of the famous "double-blind" experiments so common in medicine where the administers of the treatment are unaware of the specific nature of the treatment they are administrating at the time of the experiment. Which patients received which medicines is revealed at a later time. Astronomers developed the notion of the "personal equation" to attempt to allow for the bias inadvertently introduced by observers where personal judgement is required in making observations. Finally the experiment should have the internal precision necessary to measure the phenomena it is investigating. All these conditions sound like "common sense", but it is easy to fail to meet them in specific instances. For example, we have already seen that the statistical validity of any experiment is strongly dependent on the number of degrees of freedom exhibited by the sample. When many variables are involved, and the cost of sampling the parent population is high, it is easy to short cut on the sample size usually with disastrous results.

While we have emphasized the two extremes of scientific investigation where the hypothesis is fully specified to the case where the dependency of the variables is not known, the majority of experimental investigations lie somewhere in between. For example, the quality of milk in the market place could depend on such factors as the dairies that produce the milk, the types of cows selected by the farmers that supply the dairies, the time of year when the milk is produced, supplements used by the farmers, etc. Here causality is not firmly established, but the order of events is so there is no question that the quality of the milk determines the time of year, but the relevance of the factors is certainly not known. It is also likely that there are other unspecified factors that may influence the quality of the milk that are inaccessible to the investigator. Yet, assuming the concept of milk quality can be clearly defined, it is reasonable to ask if there is not some way to determine which of the known factors affect the milk quality and design an experiment to find out. It is in these middle areas that experimental design and techniques such as analysis of variance are of considerable use.

The design of an experiment basically is a program or plan for the manner in which the data will be sampled so as to meet the objectives of the experiment. There are three general techniques that are of use in producing a well designed experiment. First, data may be grouped so that unknown or inaccessible variables will be common to the group and therefore affect all the data within the group in the same manner. Consider an experiment where the one wishes to determine the factors that influence the baking of a type of bread. Let us assume that there exists an objective measure of the quality of the resultant loaf. We suspect that the oven temperature and duration of baking are relevant factors determining the quality of the loaf. It is also likely that the quality depends on the baker mixing and kneading the loaf. We could have all the loaves produced by all the bakers at the different temperatures and baking times measured for quality without keeping track of which baker produced which loaf. In our subsequent analysis the variations introduced by the different bakers would appear as variations attributed to temperature and baking time reducing the accuracy of our test. But the simple expedient of grouping the data according to each baker and separately analyzing the group would isolate the effect of variations among bakers and increase the accuracy of the experiment regarding the primary factors of interest.

Second, variables which cannot be controlled or "blocked out" by grouping the data should be reduced in significance by randomly selecting the sampled data so that the effects of these remaining variables tend to cancel out of the final analysis. Such randomization procedures are central to the design of a well-conceived experiment. Here it is not even necessary to know what the factors may be, only that their effect can be reduced by randomization. Again, consider the example of the baking of bread. Each baker is going to be asked to bake loaves at different temperatures and for varying times. Perhaps as the baker bakes more and more bread fatigue sets in affecting the quality of the dough he produces. If each baker follows the same pattern of baking the loaves (i.e. all bake the first loaves at temperature T_1 for a time t_1 etc.) then systematic errors resulting from fatigue will appear as differences attributable to the factors of the experiment. This can be avoided by assigning random sequences of time and temperature to each baker. While fatigue may still affect the results, it will not be in a systematic fashion.

Finally, in order to establish that the experiment has the precision necessary to answer the questions it poses, it may be necessary to repeat the sampling procedure a number of times. In the parlance of statistical experiment design the notion of repeating the experiment is called *replication* and can be used to help achieve proper randomization and well as establish the experimental accuracy.

Thus the concepts of data grouping, randomization and repeatability or replication are the basic tools one has to work with in designing an experiment. As in other areas of statistics, a particular jargon has been developed associated with experiment design and we should identify these terms and discuss some of the basic assumptions associated with experiment design.

a. The Terminology of Experiment Design

Like many subjects in statistics, the terminology of experiment design has its origin in a subject where statistical analysis was developed for the specific analysis of the subject. As the term *regression analysis* arose form studies in genetics, so much of experimental design formalism was developed for agriculture. The term *experimental area* used to describe the scope or environment of the experiment was initially a area of land on which an agricultural experiment was to be carried out. The terms *block* and *plot* meant subdivisions of this area. Similarly the notion of a *treatment* is known as a *factor* in the experiment and is usually the same as what we have previously meant by a variable. A *treatment level* would then refer to the value of the variable. (However, remember the caveats mentioned above relating to the relative role of variables and factors.) Finally the term *yield* was just that for an agricultural experiment. It was the results of a treatment being applied to some plot. Notice that here there is a strong causal bias in the use of the term yield. For many experiments this need not be the case. One factor may be chosen as the yield, but its role as dependent variable can be changed during the analysis. Perhaps a somewhat less prejudicial term might be *result*.

All these terms have survived and have taken on very general meanings for experiment design. Much of the mystery of experiment design is simply relating the terms of agricultural origin to experiments set in far different contexts. For example, the term *factorial experiment* refers to any experiment design where the levels (values) of several factors (i.e. variables) are controlled at two or more levels so as to investigate their effects on one another. Such an analysis will result in the presence of terms involving each factor in combination with the remaining factors. The expression of the number of combinations of n thing

taken m at a time does involve factorials [see equation (7.2.4)] but this is a slim excuse for calling such systems "factorial designs". Nevertheless, we shall follow tradition and do so.

Before delving into the specifics of experiment designs, let us consider some of the assumptions upon which their construction rests. Underlying any experiment there is a model which describes how the factors are assumed to influence the result or yield. This is not a full blown detailed equation such as the physical scientist is used to using to frame a hypothesis. Rather, it is a statement of additivity and linearity. All the factors are assumed to have a simple proportional effect on the result and the contribution of all factors is simply additive. While this may seem, and in some cases may be, an extremely restrictive assumption, it is the simplest non-trivial behavior and in the absence of other information provides a good place to begin any investigation. In the last section we divided up the data for an analysis of variance into sets of experiments each of which contained individual data entries. For the purposes of constructing a model for experiment design we will similiarly divide the observed data so that i represents the treatment level, and j represents the block containing the factor, and we may need a third subscript to denote the order of the treatment within the block. We could then write the mathematical model for such an experiment as

$$y_{ij\,k} = \langle y \rangle + f_i + b_j + \varepsilon_{ij\,k} \ . \tag{8.4.1}$$

Here $y_{ij\,k}$ is the yield or results of the ith treatment or factor-value contained in the jth block subject to an experimental error $\varepsilon_{i\,j\,k}$. The auusmption of additivity means that the block effect b_j will be the same for all treatments within the same block so that

$$y_{1j\,k_1} - y_{2j\,k_2} = f_1 - f_2 + \varepsilon_{1j\,k_1} - \varepsilon_{2\,j\,k_2} \ . \tag{8.4.2}$$

In addition, as was the case with the analysis of variance it is further assumed that the errors $\varepsilon_{i\,j\,k}$ are normally distributed.

By postulating a linear relation between the factors of interest and the result, we can see that only two values of the factors would be necessary to establish the dependence of the result on that factor. Using the terminology of experiment design we would say that only two treatment levels are necessary to establish the effect of the factor on the yield. However, we have already established that the order in which the treatments are applied should be randomized and that the factors should be grouped or blocked in some rational way in order for the experiment to be well designed. Let us briefly consider some plans for the acqusition of data which constitute an experiment design.

b. Blocked Designs

So far we have studiously avoided discussing data that is grouped in bins or ranks etc. However, the notion is central to experiment design so we will say just enough about the concept to indicate the reasons for involving it and indicate some of the complexities that result. However, we shall continue to avoid discussing the statistical analysis that results from such groupings of the data and refer the student to more complete courses on statistics. To understand the notion of grouped or blocked data, it is useful to return to the agricultural origins of experiment design.

If we were to design an experiment to investigate the effects of various fertilizers and insecticides on the yield of a particular species of plant, we would be foolish to treat only one plant with a particular combination of products. Instead, we would set out a block or plot of land within the experimental area and treat all the plants within that block in the same way. Presumably the average for the block is a more reliable measure of the behavior of plants to the combination of products than the results from a single plant. The data obtained from a single block would then be called grouped data or blocked data. If we can completely isolate a non-experimental factor within a block, the data can be said to be *completely blocked* with respect to that data. If the factor cannot be completely isolated by the grouping, the data is said to be *incompletely blocked*. The subsequent statistical analysis for these different types of blocking will be different and is beyond the scope of this discussion.

Now we must plan the arrangements of blocks so that we cover all combinations of the factors. In addition, we would like to arrange the blocks so that variables that we can't allow for have a minimal influence on our result. For example, soil conditions in our experimental area are liable to be similar for blocks that are close together than for blocks that are widely separated. We would like to arrange the blocks so that variations in the field conditions will affect all trials in a random manner. This is similiar to our approach with the bread where having the bakers follow a random sequence of allowed factors (i,e, T_i, and t_j) was used to average out fatgue factors. Thus randomization can take place in a time sequence as well as a spatial layout. This will tend to minimize the effects of these unknown variables.

The reason this works is that if we can group our treatments (levels or factor values) so that each factor is exposed to the same unspecified influence in a random order, then the effects of that influence should tend to cancel out over the entire run of the experiment. Unfortunately one pays a price for the grouping or blocking of the experimental data. The arrangement of the blocks may introduce an effect that appears as an interaction between the factors. Usually it is a high level interaction and it is predictable from the nature of the design. An interaction that is liable to be confused with an effect arising strictly from the arrangement of the blocks is said to be *confounded* and thus can never be considered as significant. Should that interaction be the one of interest, then one must change the design of the experiment. Standard statistical tables[2] give the arrangements of factors within blocks and the specific interactions that are confounded for a wide range of the number of blocks and factors for two treatment-level experiments.

However, there are other ways of arranging the blocks or the taking of the data so that the influence of inaccessible factors or sources of variation are reduced by randomization. By way of example consider the agricultural situation where we try to minimize the systematic effects of the location of the blocks. One possible arrangement is known as a *Latin square* since it is a square of Latin letters arranged in a specific way. The rule is that no row or column shall contain any particular letter more than once. Thus a 3×3 Latin square would have the form:

$$\begin{pmatrix} \text{ABC} \\ \text{BCA} \\ \text{CAB} \end{pmatrix} .$$

Let the Latin letters A, B, and C represent three treatments to be investigated. Each row and each column represents a complete experiment (i.e. replication). Thus the square symbolically represents a way of randomizing the order of the treatments within each replication so that variables depending on the order are

averaged out. In general, the rows and columns represent two variables that one hopes to eliminate by randomization. In the case of the field, they are the x-y location within the field and the associated soil variations etc. In the case of the baking of bread, the two variables could have been the batch of flour and time. The latter would then eliminate the fatigue factor which was a concern. Should there have been a third factor, we might have used a Greco-Latin square where a third dimension is added to the square by the use of Greek subscripts so that the arrangement becomes:

$$\begin{pmatrix} A_\alpha B_\delta C_\beta \\ B_\beta C_\alpha A_\delta \\ C_\delta A_\beta B_\alpha \end{pmatrix} .$$

Here the three treatments are grouped into replicates in three different ways with the result three sources of variation can be averaged out.

A Latin or Greco-Latin square design is restrictive in that it requires that the number of "rows" and "columns" corresponding to the two unspecified systematic parameters, be the same. In addition, the number of levels or treatments must equal the number of rows and columns. The procedure for use of such a design is to specify a trial by assigning the levels to the letters randomly and then permuting the rows and columns of the square until all trials are completed. One can find larger squares that allow for the use of more treatments or factors in books on experiment design[6] or handbooks of statistics[7]. These squares simply provide random arrangements for the application of treatments or the taking of data which will tend to minimize the effects of phenomena or sources of systematic error which cannot be measures, but of which the experimenter is aware. While their use may increase the amount of replication above the minimum required by the model, the additional effort is usually more than compensated by the improvement in the accuracy of the result.

While the Latin and Greco-Latin squares provide a fine design for randomizing the replications of the experiment, they are by no means the only method for doing so. Any reasonable modern computer will provide a mechanism for generating random numbers which can be used to design the plan for an experiment. However, one must be careful about the confounding between blocked data that can result in any experiment and be sure to identify those regions of the experiment in which it is likely to occur.

c. Factorial Designs

As with all experimental designs, the primary purpose of the factorial design is to specify how the experiment is to be run and the data sampling carried out. The main purpose of this protocol is to insure that all combinations of the factors (variables) are tested at the required treatment levels (values). Thus the basic model for the experiment is somewhat different from that suggested by equations (8.4.1) and (8.4.2). One looks for *effects* which are divided into *main effects* on the yield (assumed dependent variable) resulting from changes in the level of a specific factor, and *interaction effects* which are changes in the yield that result from the simultaneous change of two or more factors. In short, one looks for correlations between the factors and the yield and between the factors themselves. An experiment that has n factors each of which is allowed to have m levels will be required to have m^n trials or replications. Since most of the statistical analysis that is done on such experimental data will assume that the relationships are linear, m is usually taken to be two. Such an experiment would be called a 2^n *factorial experiment*. This simply means that it is an experiment with n-factors requires 2^n trials.

A particularly confusing notation is used to denote the order and values of the factors in the experiment. While the factors themselves are denoted by capital letters with subscripts starting at *zero* to denote their level (i.e. A_0, B_1, C_0, etc.), a particular trial is given a combination of lower case letters. If the letter is present it implies that the corresponding factor has the value with the subscript 1. Thus a trial where the factors A,B, and C have the values A_0, B_1, and C_1 would be labeled simply bc. A special representation is reserved for the case A_0, B_0, C_0, where by convention nothing would appear. The symbology is that this case is represented by (1). Thus all the possible combinations of factors which give rise to the interaction effects requiring the 2^n trials for a 2^n factorial experiment are given in Table 8.2

Table 8.2

Factorial Combinations for Two-level Experiments with n = 2 – 4

No. of Levels	Combinations of factors in standard notation
2 factors	(1), a, b, ab
3 factors	(1), a, b, ab, c, ac, bc, abc
4 factors	(1), a, b, ab, c, ac, bc, abc, d, ad, bd, cd, acd, bcd, abcd.

Tables[2] exist of the possible combinations of the interaction terms for any number of factors and reasonable numbers of treatment-levels.

As an example, let us consider the model for two factors each having the two treatments (i.e. values) required for the evaluation of linear effects

$$y_i = \langle y \rangle + a_i + b_i + a_i b_i + \varepsilon_i \ . \tag{8.4.3}$$

The subscript $_i$ will take on values of 0 and 1 for the two treatments given to a and b. Here we see that the cross term ab appears as an additional unknown. Each of the factors A and B will have a main effect on y. In addition the cross term AB which is known as the interaction term, will produce an interaction effect. These represent three unknowns that will require three independent pieces of information (i.e. trials, replications, or repetitions) for their specification. If we also require the determination of the grand mean <y> then an additional independent piece of information will be needed bringing the total to 2^2. In order to determine all the cross terms arising from an increased number of factors many more independent pieces of information are needed. This is the source of the 2^n required number of trials or replications given above. In carrying out the trials or replications required by the factorial design, it may be useful to make use of the blocked data designs including the Latin and Greco-latin squares to provide the appropriate randomization which reduces the effect of inaccessible variables.

There are additional designs which further minimize the effects of suspected influences and allow more flexibility in the number of factors and levels to be used, but they are beyond the scope of this book.

The statistical design of an experiment is extremely important when dealing with an array of factors or variables whose interaction is unpredictable from theoretical considerations. There are many pitfalls to be encountered in this area of study which is why it has become the domain of specialists. However, there is no substitute for the insight and ingenuity of the researcher in identifying the variables to be investigated. Any statistical study is limited in practice by the sample size and the systematic and unknown effects that may plague the study. Only the knowledgeable researcher will be able to identify the possible areas of difficulty. Statistical analysis may be able to confirm those suspicions, but will rarely find them without the foresight of the investigator. Statistical analysis is a valuable tool of research, but it is not meant to be a substitute for wisdom and ingenuity. The user must also always be aware that it is easy to phrase statistical inference so that the resulting statement says more than is justified by the analysis. Always remember that one does not "prove" hypotheses by means of statistical analysis. At best one may reject a hypothesis or add confirmatory evidence to support it. But the sample population is not the parent population and there is always the chance that the investigator has been unlucky.

Chapter 8 Exercises

1. Show that the variance of the t-probability density distribution function given by equation (8.1.2) is indeed σ^2_t as given by equation (8.1.3).

2. Use equation (8.1.7) to find the variance, mode , and skewness of the χ^2-distribution function. Compare your results to equation (8.1.8).

3. Find the mean, mode and variance of the F-distribution function given by equation (8.1.11).

4. Show that the limiting relations given by equations (8.1.13) - (8.1.15) are indeed correct.

5. Use the numerical quadrature methods discussed in chapter 4 to evaluate the probability integral for the t-test given by equation (8.2.5) for values of p=.1, 0.1, 0.01, and N=10, 30, 100. Obtain values for t_p and compare with the results you would obtain from equation (8.2.6).

6. Use the numerical quadrature methods discussed in chapter 4 to evaluate the probability integral for the χ^2-test given by equation (8.2.8) for values of p=.1, 0.1, 0.01, and N=10, 30, 100. Obtain values for χ^2_p and compare with the results you would obtain from using the normal curve for the χ^2-probability density distribution function.

7. Use the numerical quadrature methods discussed in chapter 4 to evaluate the probability integral for the F-test given by equation (8.2.9) for values of p=.1, 0.1, 0.01, N_1=10, 30, 100, and N_2=1, 10, 30. Obtain values for F_p.

8. Show how the various forms of the correlation coefficient given by equation (8.3.7) can be obtained from the definition given by the second term on the left.

9. Find the various values of the 0.1% marginally significant correlation coefficients when n= 5, 10, 30, 100, 1000.

10. Find the correlation coefficient between X_1 and Y_1, and Y_1 and Y_2 in problem 4 of chapter 6.

11. Use the F-test to decide when you have added enough terms to represent the table given in problem 3 of chapter 6.

12. Use analysis of variance to show that the data in Table 8.1 imply that taking the bus and taking the ferry are important factors in populating the beach.

13. Use analysis of variance to determine if the examination represented by the data in Table 7.1 sampled a normal parent population and at what level of confidence on can be sure of the result.

14. Assume that you are to design an experiment to find the factors that determine the quality of bread baked at 10 different bakeries. Indicate what would be your central concerns and how you would go about addressing them. Identify four factors that are liable to be of central significance in determining the quality of bread. Indicate how you would design an experiment to find out if the factors are indeed important.

Chapter 8 References and Supplemental Reading

1 Croxton, F.E., Cowden, D.J., and Klein, S., "Applied General Statistics", (1967), Prentice-Hall, Inc., Englewood Cliffs, N.J.

2. Weast, R.C., "CRC Handbook of Tables for Probability and Statistics", (1966), (Ed. W.H.Beyer), The Chemical Rubber Co. Cleveland.

3. Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T., "Numerical Recipies the art of scientific computing" (1986), Cambridge University Press, Cambridge.

4. Smith, J.G., and Duncan, A.J., "Sampling Statistics and Applications: Fundementals of the Theory of Statistics", (1944), McGraw-Hill Book Company Inc., New York, London, pp.18.

5. Cochran , W.G., and Cox, G.M., "Experimental Designs" (1957) John Wiley and Sons, Inc., New York, pp 10.

6. Cochran , W.G., and Cox, G.M., "Experimental Designs" (1957) John Wiley and Sons, Inc., New York, pp 145-147.

7. Weast, R.C., "CRC Handbook of Tables for Probability and Statistics", (1966), (Ed. W.H.Beyer), The Chemical Rubber Co. Cleveland, pp63-65.

Index

A

B

C

E

F

K

L

M

Q

R

S

T

U

V

W

Y

Z