CHAPTER 1
NUMERICAL METHODS

1.1 *Introduction*

I believe that, when I was a young student, I had some vague naive belief that every equation had as its solution an explicit algebraic formula, and every function to be integrated had a similar explicit analytical function for the answer. It came as quite an eye-opener to me when I began to realize that this was far from the case. There are many mathematical operations for which there is no explicit formula, and yet more for which numerical solutions are either easier, or quicker or more convenient than algebraic solutions. I also remember being impressed as a student with the seemingly endless number of "special functions" whose properties were listed in textbooks and which I feared I would have to memorize and master. Of course we now have computers, and over the years I have come to realize that it is often easier to generate numerical solutions to problems rather than try to express them in terms of obscure special functions with which few people are honestly familiar. Now, far from believing that every problem has an explicit algebraic solution, I suspect that algebraic solutions to problems may be a minority, and numerical solutions to many problems are the norm.

This chapter is not intended as a comprehensive course in numerical methods. Rather it deals, and only in a rather basic way, with the very common problems of numerical integration and the solution of simple (and not so simple!) equations. Specialist astronomers today can generate most of the planetary tables for themselves; but those who are not so specialized still have a need to look up data in tables such as *The Astronomical Almanac*, and I have therefore added a brief section on interpolation, which I hope may be useful. While any of these topics could be greatly expanded, this section should be useful for many everyday computational purposes.

I do not deal in this introductory chapter with the huge subject of differential equations. These need a book in themselves. Nevertheless, there is an example I remember from student days that has stuck in my mind ever since. In those days, calculations were done by hand-operated mechanical calculators, one of which I still fondly possess, and speed and efficiency, as well as accuracy, were a prime concern - as indeed they still are today in an era of electronic computers of astonishing speed. The problem was this: Given the differential equation

$$\frac{dy}{dx} = \frac{x+y}{x-y} \qquad 1.1.1$$

with initial conditions $y = 0$ when $x = 1$, tabulate y as a function of x.. It happens that the differential equation can readily be solved analytically:

$$\ln\left(x^2 + y^2\right) = 2\tan^{-1}\left(y/x\right) \qquad 1.1.2$$

Yet it is far quicker and easier to tabulate y as a function of x using numerical techniques directly from the original differential equation 1.1.1 than from its analytical solution 1.1.2.

1.2 *Numerical Integration*

There are many occasions when one may wish to integrate an expression numerically rather than analytically. Sometimes one cannot find an analytical expression for an integral, or, if one can, it is so complicated that it is just as quick to integrate numerically as it is to tabulate the analytical expression. Or one may have a table of numbers to integrate rather than an analytical equation. Many computers and programmable calculators have internal routines for integration, which one can call upon (at risk) without having any idea how they work. It is assumed that the reader of this chapter, however, wants to be able to carry out a numerical integration without calling upon an existing routine that has been written by somebody else.

There are many different methods of numerical integration, but the one known as Simpson's Rule is easy to program, rapid to perform and usually very accurate. (Thomas Simpson, 1710 - 1761, was an English mathematician, author of *A New Treatise on Fluxions*.)

Suppose we have a function $y(x)$ that we wish to integrate between two limits. We calculate the value of the function at the two limits and halfway between, so we now know three points on the curve. We then fit a parabola to these three points and find the area under that.

In the figure I.1, $y(x)$ is the function we wish to integrate between the limits $x_2 - \delta x$ and $x_2 + \delta x$. In other words, we wish to calculate the area under the curve. y_1, y_2 and y_3 are the values of

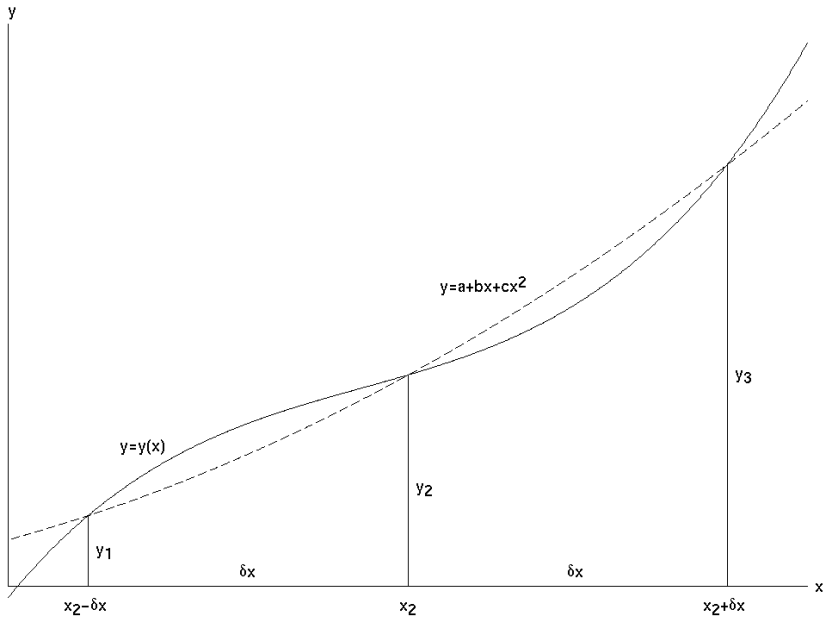


FIGURE I.1 Simpson's Rule gives us the area under the parabola (dashed curve) that passes through three points on the curve $y = y(x)$. This is approximately equal to the area under $y = y(x)$.

the function at $x_2 - \delta x$, x_2 and $x_2 + \delta x$, and $y = a + bx + cx^2$ is the parabola passing through the points $(x_2 - \delta x,\ y_1)$, (x_2, y_2) and $(x_2 + \delta x,\ y_3)$.

If the parabola is to pass through these three points, we must have

$$y_1 = a + b(x_2 - \delta x) + c(x_2 - \delta x)^2 \qquad 1.2.1$$

$$y_2 = a + bx + cx^2 \qquad 1.2.2$$

$$y_3 = a + b(x_2 + \delta x) + c(x_2 + \delta x)^2 \qquad 1.2.3$$

We can solve these equations to find the values of *a*, *b* and *c*. These are

$$a = y_2 - \frac{x_2(y_3 - y_1)}{2\delta x} + \frac{x_2^2(y_3 - 2y_2 + y_1)}{2(\delta x)^2} \qquad 1.2.4$$

$$b = \frac{y_3 - y_1}{2\delta x} - \frac{x_2(y_3 - 2y_2 + y_1)}{(\delta x)^2} \qquad 1.2.5$$

$$c = \frac{y_3 - 2y_2 + y_1}{2(\delta x)^2} \qquad 1.2.6$$

Now the area under the parabola (which is taken to be approximately the area under $y(x)$) is

$$\int_{x_2-\delta x}^{x_2+\delta x} (a + bx + cx^2)dx = 2\left[a + bx_2 + cx_2^2 + \tfrac{1}{3}c(\delta x)^2\right]\delta x \qquad 1.2.7$$

On substituting the values of *a*, *b* and *c*, we obtain for the area under the parabola

$$\tfrac{1}{3}(y_1 + 4y_2 + y_3)\delta x \qquad 1.2.8$$

and this is the formula known as Simpson's Rule.

For an example, let us evaluate $\int_0^{\pi/2} \sin x dx$.

We shall evaluate the function at the lower and upper limits and halfway between. Thus

$x = 0, \qquad y = 0$

$x = \pi/4, \quad y = 1/\sqrt{2}$

$x = \pi/2, \quad y = 1$

The interval between consecutive values of x is $\delta x = \pi/4$.

Hence Simpson's Rule gives for the area

$$\tfrac{1}{3}\left(0 + \tfrac{4}{\sqrt{2}} + 1\right)\tfrac{\pi}{4}$$

which, to three significant figures, is 1.00. Graphs of $\sin x$ and $a + bx + cx^2$ are shown in figure I.2a. The values of a, b and c, obtained from the formulas above, are

$$a = 0, \quad b = \frac{\sqrt{32} - 2}{\pi}, \quad c = \frac{8 - \sqrt{128}}{\pi^2}$$

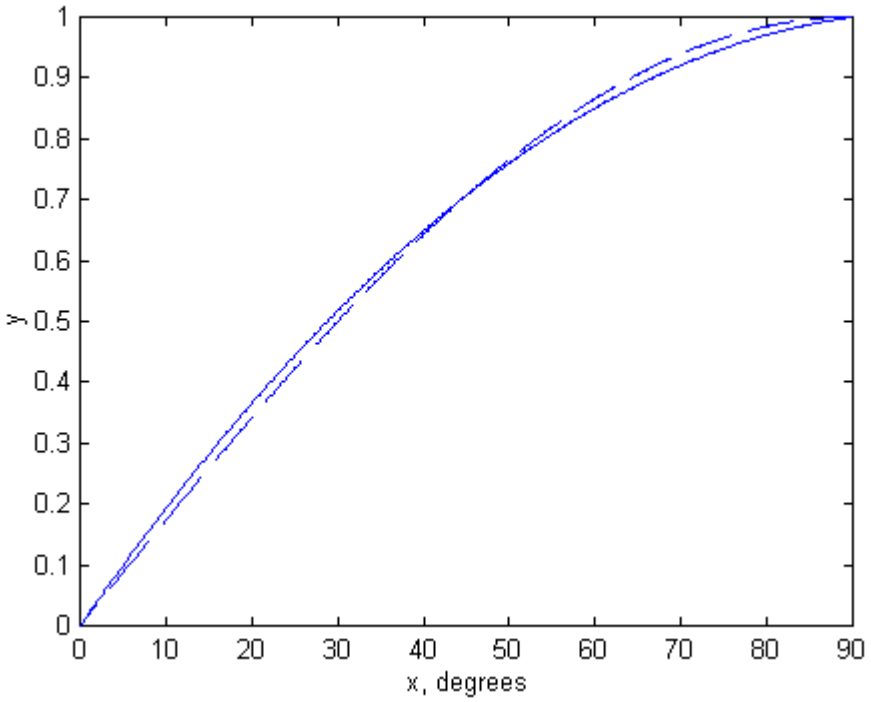


FIGURE I.2a

The result we have just obtained is quite spectacular, and we are not always so lucky. Not all functions can be approximated so well by a parabola. But of course the interval $\delta x = \pi/4$ was

ridiculously coarse. In practice we subdivide the interval into numerous very small intervals. For example, consider the integral

$$\int_0^{\pi/4} \cos^{\frac{3}{2}} 2x \sin x dx.$$

Let us subdivide the interval 0 to π/4 into ten intervals of width π /40 each. We shall evaluate the function at the end points and the nine points between, thus:

x	$\cos^{\frac{3}{2}} 2x \sin x dx$
0	y_1 = 0.000 000 000
π/40	y_2 = 0.077 014 622
2π/40	y_3 = 0.145 091 486
3π/40	y_4 = 0.196 339 002
4π/40	y_5 = 0.224 863 430
5π/40	y_6 = 0.227 544 930
6π/40	y_7 = 0.204 585 473
7π/40	y_8 = 0.159 828 877
8π/40	y_9 = 0.100 969 971
9π/40	y_{10} = 0.040 183 066
10π/40	y_{11} = 0.000 000 000

The integral from 0 to 2π /40 is $\frac{1}{3}(y_1 + 4y_2 + y_3)\delta x$, δx being the interval π/40. The integral from 3π/40 to 4π/40 is $\frac{1}{3}(y_3 + 4y_4 + y_5)\delta x$. And so on, until we reach the integral from 8π/40 to 10π/40. When we add all of these up, we obtain for the integral from 0 to π /4,

$$\frac{1}{3}(y_1 + 4y_2 + 2y_3 + 4y_4 + 2y_5 + \ldots \ldots + 4y_{10} + y_{11})\delta x$$

$$= \frac{1}{3}[y_1 + y_{11} + 4(y_2 + y_4 + y_6 + y_8 + y_{10}) + 2(y_3 + y_5 + y_7 + y_9)]\delta x,$$

which comes to 0.108 768 816.

We see that the calculation is rather quick, and it is easily programmable (try it!). But how good is the answer? Is it good to three significant figures? Four? Five?

Since it is fairly easy to program the procedure for a computer, my practice is to subdivide the interval successively into 10, 100, 1000 subintervals, and see whether the result converges. In the present example, with N subintervals, I found the following results:

N	integral
10	0.108 768 816
100	0.108 709 621
1000	0.108 709 466
10000	0.108 709 465

This shows that, even with a course division into ten intervals, a fairly good result is obtained, but you do have to work for more significant figures. I was using a mainframe computer when I did the calculation with 10000 intervals, and the answer was displayed on my screen in what I would estimate was about one fifth of a second.

There are two more lessons to be learned from this example. One is that sometimes a change of variable will make things very much faster. For example, if one makes one of the (fairly obvious?) trial substitutions $y = \cos x$, $y = \cos 2x$ or $y^2 = \cos 2x$, the integral becomes

$$\int_{1/\sqrt{2}}^{1} (2y^2 - 1)^{3/2}\, dy, \quad \int_0^1 \sqrt{\frac{y^3}{8(1+y)}}\, dy \quad \text{or} \quad \int_0^1 \frac{y^4}{\sqrt{2(1+y^2)}}\, dy.$$

Not only is it very much faster to calculate any of these integrands than the original trigonometric expression, but I found the answer 0.108 709 465 by Simpson's rule on the third of these with only 100 intervals rather than 10,000, the answer appearing on the screen apparently instantaneously. (The first two required a few more intervals.)

To gain about one fifth of a second may appear to be of small moment, but in truth the computation went faster by a factor of several hundred. One sometimes hears of very large computations involving massive amounts of data requiring overnight computer runs of eight hours or so. If the programming speed and efficiency could be increased by a factor of a few hundred, as in this example, the entire computation could be completed in less than a minute.

The other lesson to be learned is that the integral does, after all, have an explicit algebraic form. You should try to find it, not only for integration practice, but to convince yourself that there are indeed occasions when a numerical solution can be found faster than an analytic one! The answer, by the way, is $\dfrac{\sqrt{18}\ln\left(1+\sqrt{2}\right) - 2}{16}$.

You might now like to perform the following integration numerically, either by hand calculator or by computer.

$$\int_0^2 \frac{x^2 dx}{\sqrt{2-x}}$$

At first glance, this may look like just another routine exercise, but you will very soon find a small difficulty and wonder what to do about it. The difficulty is that, at the upper limit of integration, the integrand becomes infinite. This sort of difficulty, which is not infrequent, can often be overcome by means of a change of variable. For example, let $x = 2 \sin^2 \theta$, and the integral becomes

$$8\sqrt{2}\int_0^{\pi/2} \sin^5 \theta d\theta$$

and the difficulty has gone. The reader should try to integrate this numerically by Simpson's rule, though it may also be noted that it has an exact analytic answer, namely $\sqrt{8192}/15$.

Here is another example. It can be shown that the period of oscillation of a simple pendulum of length l swinging through 90^o on either side of the vertical is

$$P = \sqrt{\frac{8l}{g}}\int_0^{\pi/2} \sqrt{\sec\theta} d\theta \ .$$

As in the previous example, the integrand becomes infinite at the upper limit. I leave it to the reader to find a suitable change of variable such that the integrand is finite at both limits, and then to integrate it numerically. (If you give up, see Section 1.13.) Unlike the last example, this one has no simple analytic solution in terms of elementary functions. It can be written in terms of special functions (elliptic integrals) but they have to be evaluated numerically in any case, so that is of little help. I make the answer

$$P \ = \ 2.3607\pi\sqrt{\frac{l}{g}}.$$

For another example, consider

$$\int_0^\infty \frac{dx}{x^5\left(e^{1/x}-1\right)}$$

This integral occurs in the theory of blackbody radiation. To help you to visualize the integrand, it and its first derivative are zero at $x \ = \ 0$ and $x = \infty,$ and it reaches a maximum value of 21.201435 at $x \ = \ 0.201405$. The difficulty this time is the infinite upper limit. But, as in the

previous two examples, we can overcome the difficulty by making a change of variable. For example, if we let $x = \tan\theta$, the integral becomes

$$\int_0^{\pi/2} \frac{c^3\left(c^2+1\right)d\theta}{e^c - 1}, \quad \text{where } c = \cot\theta = 1/x.$$

The integrand is zero at both limits and is easily calculable between, and the value of the integral can now be calculated by Simpson's rule in a straightforward way. It also has an exact analytic solution, namely $\pi^4/15$, though it is hard to say whether it is easier to arrive at this by analysis or by numerical integration.

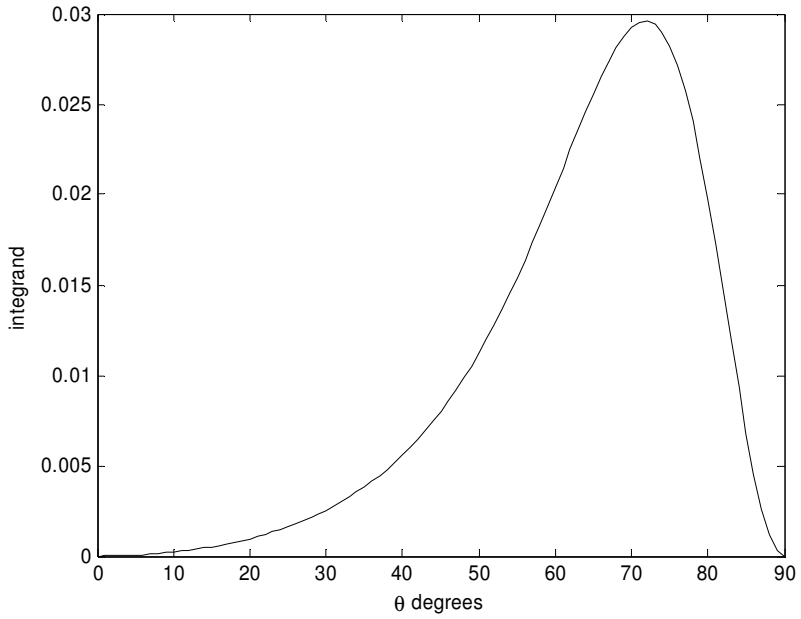
Here's another:

$$\int_0^{\infty} \frac{x^2 dx}{(x^2 + 9)(x^2 + 4)^2}$$

The immediate difficulty is the infinite upper limit, but that is easily dealt with by making a change of variable: $x = \tan\theta$. The integral then becomes

$$\int_{\theta=0}^{\pi/2} \frac{t(t+1)d\theta}{(t+9)(t+4)^2}$$

in which $t = \tan^2\theta$. The upper limit is now finite, and the integrand is easy to compute - except, perhaps, at the upper limit. However, after some initial hesitation the reader will probably agree that the integrand is zero at the upper limit. The integrand looks like this:



It reaches a maximum of 0.029 5917 at $\theta = 71°.789\ 962$. Simpson's rule easily gave me an answer of 0.015 708. The integral has an analytic solution (try it) of $\pi/200$.

There are, of course, methods of numerical integration other than Simpson's rule. I describe one here without proof. I call it "seven-point integration". It may seem complicated, but once you have successfully programmed it for a computer, you can forget the details, and it is often even faster and more accurate than Simpson's rule. You evaluate the function at $6n + 1$ points, where n is an integer, so that there are $6n$ intervals. If, for example, $n = 4$, you evaluate the function at 25 points, including the lower and upper limits of integration. The integral is then:

$$\int_a^b f(x)dx = 0.3\times(\Sigma_1 + 2\Sigma_2 + 5\Sigma_3 + 6\Sigma_4)\delta x, \qquad 1.2.9$$

where δx is the size of the interval, and

$$\Sigma_1 = f_1 + f_3 + f_5 + f_9 + f_{11} + f_{15} + f_{17} + f_{21} + f_{23} + f_{25}, \qquad 1.2.10$$

$$\Sigma_2 = f_7 + f_{13} + f_{19}, \qquad 1.2.11$$

$$\Sigma_3 = f_2 + f_6 + f_8 + f_{12} + f_{14} + f_{18} + f_{20} + f_{24} \qquad 1.2.12$$

and

$$\Sigma_4 = f_4 + f_{10} + f_{16} + f_{22}. \qquad 1.2.13$$

Here, of course, $f_1 = f(a)$ and $f_{25} = f(b)$. You can try this on the functions we have already integrated by Simpson's rule, and see whether it is faster.

Let us try one last integration before moving to the next section. Let us try

$$\int_0^{10} \frac{1}{1+8x^3}\,dx\,.$$

This can easily (!) be integrated analytically, and you might like to show that it is

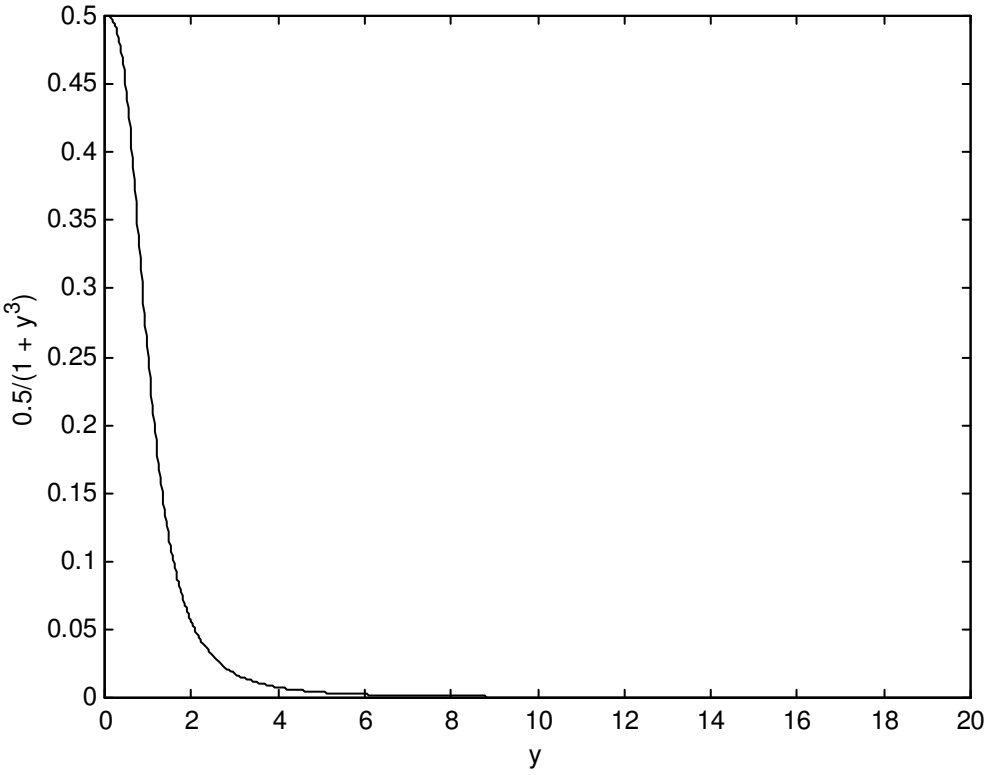
$$\tfrac{1}{12}\ln\tfrac{147}{127} + \tfrac{1}{\sqrt{12}}\tan^{-1}\sqrt{507} + \tfrac{\pi}{\sqrt{432}} = 0.6039748\,.$$

However, our purpose in this section is to learn some skills of numerical integration. Using Simpson's rule, I obtained the above answer to seven decimal places with 544 intervals. With seven-point integration, however, I used only 162 intervals to achieve the same precision, a reduction of 70%. Either way, the calculation on a fast computer was almost instantaneous. However, had it been a really lengthy integration, the greater efficiency of the seven point integration might have saved hours. It is also worth noting that $x \times x \times x$ is faster to compute than x^3. Also, if we make the substitution $y = 2x$, the integral becomes

$$\int_0^{20} \frac{0.5}{1+y^3}\,dy.$$

This reduces the number of multiplications to be done from 489 to 326 – i.e. a further reduction of one third. But we have still not done the best we could do. Let us look at the function $\dfrac{0.5}{1+y^3}$, in figure I.2b:

FIGURE I2b



We see that beyond $y = 6$, our efforts have been largely wasted. We don't need such fine intervals of integration. I find that I can obtain the same level of precision – i.e. an answer of 0.6039748 – using 48 intervals from $y = 0$ to 6 and 24 intervals from $y = 6$ to 20. Thus, by various means we have reduced the number of times that the function had to be evaluated from our original 545 to 72, as well as reducing the number of multiplications each time by a third, a reduction of computing time by 91%.

This last example shows that it often advantageous to use fine intervals of integration only when the function is rapidly changing (i.e. has a large slope), and to revert to coarser intervals where the function is changing only slowly.

The *Gaussian quadrature* method of numerical integration is described in Sections 1.15 and 1.16.

1.3 *Quadratic equations*

Any reader of this book will know that the solutions to the quadratic equation

$$ax^2 + bx + c = 0 \qquad 1.3.3$$

are

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \qquad 1.3.2$$

and will have no difficulty in finding that the solutions to

$$2.9x^2 - 4.7x + 1.7 = 0$$

are

$$x = 1.0758 \text{ or } 0.5449.$$

We are now going to look, largely for fun, at two alternative iterative numerical methods of solving a quadratic equation. One of them will turn out not to be very good, but the second will turn out to be sufficiently good to merit our serious attention.

In the first method, we re-write the quadratic equation in the form

$$x = \frac{-\left(ax^2 + c\right)}{b}$$

We guess a value for one of the solutions, put the guess in the right hand side, and hence calculate a new value for x. We continue iterating like this until the solution converges.

For example, let us guess that a solution to the equation $2.9x^2 - 4.7x + 1.7 = 0$ is $x = 0.55$. Successive iterations produce the values

0.54835	0.54501
0.54723	0.54498
0.54648	0.54496
0.54597	0.54495
0.54562	0.54494
0.54539	0.54493
0.54524	0.54493
0.54513	0.54494
0.54506	0.54492

We did eventually arrive at the correct answer, but it was very slow indeed even though our first guess was so close to the correct answer that we would not have been likely to make such a good first guess accidentally.

Let us try to obtain the second solution, and we shall try a first guess of 1.10, which again is such a good first guess that we would not be likely to arrive at it accidentally. Successive iterations result in

1.10830
1.11960
1.13515

and we are getting further and further from the correct answer!

Let us try a better first guess of 1.05. This time, successive iterations result in

1.04197
1.03160
1.01834

Again, we are getting further and further from the solution.

No more need be said to convince the reader that this is not a good method, so let us try something a little different.

We start with $ax^2 + bx = -c$ 1.3.3

Add ax^2 to each side:

$2ax^2 + bx = ax^2 - c$ 1.3.4

or $(2ax+b)x = ax^2 - c$ 1.3.5

Solve for x: $x = \dfrac{ax^2 - c}{2ax+b}$ 1.3.6

This is just the original equation written in a slightly rearranged form. Now let us make a guess for x, and iterate as before. This time, however, instead of making a guess so good that we are unlikely to have stumbled upon it, let us make a very stupid first guess, for example $x = 0$. Successive iterations then proceed as follows.

0.00000
0.36170
0.51751
0.54261
0.54491
0.54492

and the solution converged rapidly in spite of the exceptional stupidity of our first guess. The reader should now try another very stupid first guess to try to arrive at the second solution. I tried $x = 100$, which is very stupid indeed, but I found convergence to the solution 1.0758 after just a few iterations.

Even although we already know how to solve a quadratic equation, there is something intriguing about this. What was the motivation for adding ax^2 to each side of the equation, and why did the resulting minor rearrangement lead to rapid convergence from a stupid first guess, whereas a simple direct iteration either converged extremely slowly from an impossibly good first guess or did not converge at all?

Read on.

1.4 *The solution of f(x) = 0*

The title of this section is intended to be eye-catching. Some equations are easy to solve; others seem to be more difficult. In this section, we are going to try to solve any equation at all of the form $f(x) = 0$ (which covers just about everything!) and we shall in most cases succeed with ease.

Figure I.3 shows a graph of the equation $y = f(x)$. We have to find the value (or perhaps values) of x such that $f(x) = 0$.

We guess that the answer might be x_g , for example. We calculate $f(x_g)$. It won't be zero, because our guess is wrong. The figure shows our guess x_g, the correct value x, and $f(x_g)$. The tangent of the angle θ is the derivative $f'(x)$, but we cannot calculate the derivative there because we do not yet know x. However, we can calculate $f'(x_g)$, which is close. In any case $\tan\theta$, or $f'(x_g)$, is approximately equal to $f(x_g)/(x_g - x)$, so that

$$x \approx x_g - \frac{f(x_g)}{f'(x_g)} \qquad 1.4.1$$

will be much closer to the true value than our original guess was. We use the new value as our next guess, and keep on iterating until

$$\left| \frac{x_g - x}{x_g} \right|$$

is less than whatever precision we desire. The method is usually extraordinarily fast, even for a wildly inaccurate first guess. The method is known as Newton-Raphson iteration. There are some cases where the method will not converge, and stress is often placed on these exceptional cases in mathematical courses, giving the impression that the Newton-Raphson process is of

limited applicability. These exceptional cases are, however, often artificially concocted in order to illustrate the exceptions (we do indeed cite some below), and in practice Newton-Raphson is usually the method of choice.

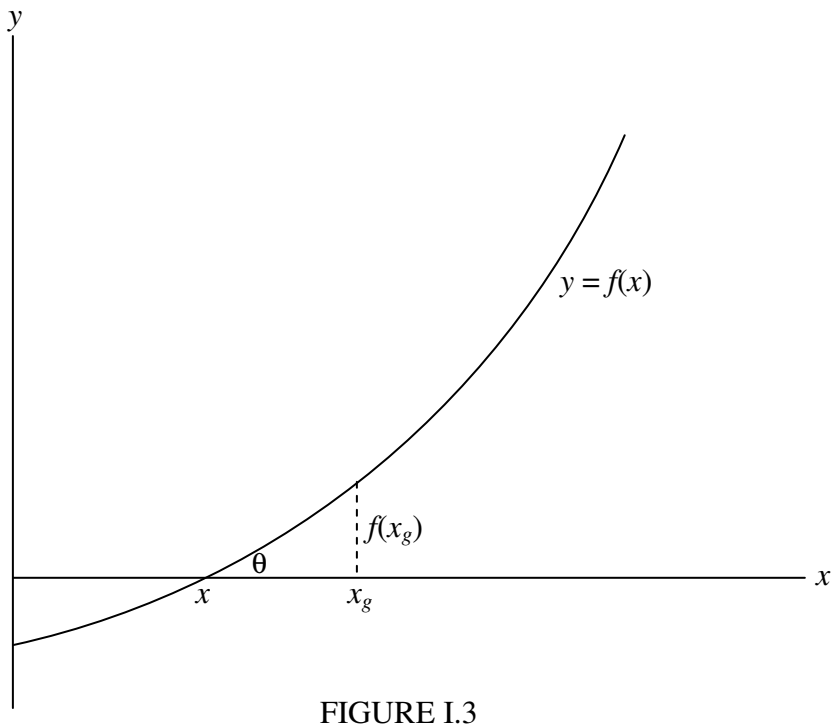


FIGURE I.3

I shall often drop the clumsy subscript *g*, and shall write the Newton-Raphson scheme as

$$x = x - f(x)/f'(x), \qquad 1.4.2$$

meaning "start with some value of x, calculate the right hand side, and use the result as a new value of x". It may be objected that this is a misuse of the = symbol, and that the above is not really an "equation", since x cannot equal x minus something. However, when the correct solution for x has been found, it will satisfy $f(x) = 0$, and the above is indeed a perfectly good equation and a valid use of the = symbol.

A few quick examples.

i. Solve the equation $1/x = \ln x$

We have $f = 1/x - \ln x = 0$

And $f' = -(1 + x)/x^2,$

from which $x - f/f'$ becomes, after some simplification,

$$\frac{x[2 + x(1 - \ln x)]}{1+x},$$

so that the Newton-Raphson iteration is

$$x = \frac{x[2 + x(1 - \ln x)]}{1+x}.$$

There remains the question as to what should be the first guess. We know (or should know!) that ln 1 = 0 and ln 2 = 0.6931, so the answer must be somewhere between 1 and 2. If we try $x = 1.5$, successive iterations are

1.735 081 403
1.762 915 391
1.763 222 798
1.763 222 834
1.763 222 835

This converged quickly from a fairly good first guess of 1.5. Very often the Newton-Raphson iteration will converge, even rapidly, from a very stupid first guess, but in this particular example there are limits to stupidity, and the reader might like to prove that, in order to achieve convergence, the first guess must be in the range

$$0 < x < 4.319\ 136\ 566$$

ii. Solve the unlikely equation $\sin x = \ln x$.

We have $f = \sin x - \ln x$ and $f' = \cos x - 1/x$,

and after some simplification the Newton-Raphson iteration becomes

$$x = x\left[1 + \frac{\ln x - \sin x}{x\cos x - 1}\right].$$

Graphs of sin x and ln x will provide a first guess, but in lieu of that and without having much idea of what the answer might be, we could try a fairly stupid $x = 1$. Subsequent iterations produce

2.830 487 722
2.267 902 211
2.219 744 452
2.219 107 263
2.219 107 149
2.219 107 149

iii. Solve the equation $x^2 = a$ (A new way of finding square roots!)

$$f = x^2 - a, \quad f' = 2x.$$

After a little simplification, the Newton-Raphson process becomes

$$x = \frac{x^2 + a}{2x}.$$

For example, what is the square root of 10? Guess 3. Subsequent iterations are

3.166 666 667
3.162 280 702
3.162 277 661
3.162 277 661

iv. Solve the equation $ax^2 + bx + c = 0$ (A new way of solving quadratic equations!)

$$f = ax^2 + bx + c = 0,$$

$$f' = 2ax + b.$$

Newton-Raphson:

$$x = x - \frac{ax^2 + bx + c}{2ax + b},$$

which becomes, after simplification,

$$x = \frac{ax^2 - c}{2ax + b}.$$

This is just the iteration given in the previous section, on the solution of quadratic equations, and it shows why the previous method converged so rapidly and also how I really arrived at the equation (which was via the Newton-Raphson process, and not by arbitrarily adding ax^2 to both sides!)

1.5 *The Solution of Polynomial Equations*.

The Newton-Raphson method is very suitable for the solution of polynomial equations, for example for the solution of a quintic equation:

$$a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4 + a_5x^5 = 0. \qquad 1.5.1$$

Before illustrating the method, it should be pointed out that, even though it may look inelegant in print, in order to evaluate a polynomial expression numerically it is far easier and quicker to nest the parentheses and write the polynomial in the form

$$a_0 + x(a_1 + x(a_2 + x(a_3 + x(a_4 + xa_5)))). \qquad 1.5.2$$

Working from the inside out, we see that the process is a multiplication followed by an addition, repeated over and over. This is very easy whether the calculation is done by computer, by calculator, or in one's head.

For example, evaluate the following expression in your head, for $x = 4$:

$$2 - 7x + 2x^2 - 8x^3 - 2x^4 + 3x^5.$$

You couldn't? But now evaluate the following expression in your head for $x = 4$ and see how (relatively) easy it is:

$$2 + x(-7 + x(2 + x(-8 + x(-2 + 3x)))).$$

As an example of how efficient the nested parentheses are in a computer program, here is a FORTRAN program for evaluating a fifth degree polynomial. It is assumed that the value of x has been defined in a FORTRAN variable called `X`, and that the six coefficients $a_0, a_1, \ldots a_5$ have been stored in a vector as `A(1), A(2),... A(6)`.

```
      Y=0.
      DO1I=1,5
1     Y=(Y+A(7-I))*X
      Y=Y+A(1)
```

The calculation is finished!

We return now to the solution of

$$f(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4 + a_5x^5 = 0. \qquad 1.5.3$$

We have $$f'(x) = a_1 + 2a_2x + 3a_3x^2 + 4a_4x^3 + 5a_5x^4. \qquad 1.5.4$$

Now $$x = x - f / f', \qquad 1.5.5$$

and after simplification,

$$x = \frac{-a_0 + x^2(a_2 + x(2a_3 + x(3a_4 + 4a_5x)))}{a_1 + x(2a_2 + x(3a_3 + x(4a_4 + 5a_5x)))}, \qquad 1.5.6$$

which is now ready for numerical iteration.

For example, let us solve

$$205 + 111x + 4x^2 - 31x^3 - 10x^4 + 3x^5 = 0. \qquad 1.5.7$$

A reasonable first guess could be obtained by drawing a graph of this function to see where it crosses the x-axis, but, truth to tell, the Newton-Raphson process usually works so well that one need spend little time on a first guess; just use the first number that comes into your head, for example, $x = 0$. Subsequent iterations then go

$$\begin{array}{r} -1.846\,847 \\ -1.983\,713 \\ -1.967\,392 \\ -1.967\,111 \\ -1.967\,110 \end{array}$$

A question that remains is: How many solutions are there? The general answer is that an nth degree polynomial equation has n solutions. This statement needs to be qualified a little. For example, the solutions need not be real. The solutions may be imaginary, as they are, for example, in the equation

$$1 + x^2 = 0 \qquad 1.5.8$$

or complex, as they are, for example, in the equation

$$1 + x + x^2 = 0. \qquad 1.5.9$$

If the solutions are real they may not be distinct. For example, the equation

$$1 - 2x + x^2 = 0 \qquad 1.5.10$$

has two solutions at $x = 1$, and the reader may be forgiven for thinking that this somewhat stretches the meaning of "two solutions". However, if one includes complex roots and repeated real roots, it is then always true that an nth degree polynomial has n solutions. The five solutions of the quintic equation we solved above, for example, are

$$\begin{array}{r} 4.947\,845 \\ 2.340\,216 \\ -1.967\,110 \\ -0.993\,808 + 1.418\,597i \\ -0.993\,808 - 1.418\,597i \end{array}$$

Can one tell in advance how many real roots a polynomial equation has? The most certain way to tell is to plot a graph of the polynomial function and see how many times it crosses the x-axis. However, it is possible to a limited extent to determine in advance how many real roots there are. The following "rules" may help. Some will be fairly obvious; others require proof.

The number of real roots of a polynomial of odd degree is odd. Thus a quintic equation can have one, three or five real roots. Not all of these roots need be distinct, however, so this is of limited help. Nevertheless a polynomial of odd degree always has at least one real root. The number of real roots of an equation of even degree is even - but the roots need not all be distinct, and the number of real roots could be zero.

An upper limit to the number of real roots can be determined by examining the signs of the coefficients. For example, consider again the equation

$$205 + 111x + 4x^2 - 31x^3 - 10x^4 + 3x^5 = 0. \qquad 1.5.11$$

The signs of the coefficients, written in order starting with a_0 , are

$$+ + + - - +$$

Run your eye along this list, and count the number of times there is a change of sign. The sign changes twice. This tells us that there are not more than two positive real roots. (If one of the coefficients in a polynomial equation is zero, i.e. if one of the terms is "missing", this does not count as a change of sign.)

Now change the signs of all coefficients of odd powers of x:

$$+ - + + - -$$

This time there are three changes of sign. This tells us that there are not more than three negative real roots.

In other words, the number of changes of sign in $f(x)$ gives us an upper limit to the number of positive real roots, and the number of changes of sign in $f(-x)$ gives us an upper limit to the number of negative real roots.

One last "rule" is that complex roots occur in conjugate pairs. In our particular example, these rules tell us that there are not more than two positive real roots, and not more than three negative real roots. Since the degree of the polynomial is odd, there is at least one real root, though we cannot tell whether it is positive or negative.

In fact the particular equation, as we have seen, has two positive real roots, one negative real root, and two conjugate complex roots.

1.6 *Failure of the Newton-Raphson Method.*

This section is written reluctantly, for fear it may give the impression that the Newton-Raphson method frequently fails and is of limited usefulness. This is not the case; in nearly all cases encountered in practice it is very rapid and does not require a particularly good first guess.

Nevertheless for completeness it should be pointed out that there are rare occasions when the method either fails or converges rather slowly.

One example is the quintic equation that we have just encountered:

$$205 + 111x + 4x^2 - 31x^3 - 10x^4 + 5x^5 = 0 \qquad 1.6.1$$

When we chose $x = 0$ as our first guess, we reached a solution fairly quickly. If we had chosen $x = 1$, we would not have been so lucky, for the first iteration would have taken us to −281, a very long way from any of the real solutions. Repeated iteration will eventually take us to the correct solution, but only after many iterations. This is not a typical situation, and usually almost any guess will do.

Another example of an equation that gives some difficulty is

$$x = \tan x, \qquad 1.6.2$$

an equation that occurs in the theory of single-slit diffraction.

We have $$f(x) = x - \tan x = 0 \qquad 1.6.3$$

and $$f'(x) = 1 - \sec^2 x = -\tan^2 x. \qquad 1.6.4$$

The Newton-Raphson process takes the form

$$x = x + \frac{x - \tan x}{\tan^2 x}. \qquad 1.6.5$$

The solution is $x = 4.493\,409$, but in order to achieve this the first guess must be between 4.3 and 4.7 . This again is unusual, and in most cases almost any reasonable first guess results in rapid convergence.

The equation

$$1 - 4x + 6x^2 - 4x^3 + x^4 = 0 \qquad 1.6.6$$

is an obvious candidate for difficulties. The four identical solutions are $x = 1$, but at $x = 1$ not only is $f(x)$ zero, but so is $f'(x)$. As the solution $x = 1$ is approached, convergence becomes very slow, but eventually the computer or calculator will record an error message as it attempts to divide by the nearly zero $f'(x)$.

I mention just one last example very briefly. When discussing orbits, we shall encounter an equation known as Kepler's equation. The Newton-Raphson process almost always solves Kepler's equation with spectacular speed, even with a very poor first guess. However, there are

some very rare occasions. almost never encountered in practice, where the method fails. We shall discuss this equation in Chapter 9.

1.7 *Simultaneous Linear Equations, N = n*

Consider the equations

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + a_{14}x_4 + a_{15}x_5 = b_1 \qquad 1.7.1$$

$$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + a_{24}x_4 + a_{25}x_5 = b_2 \qquad 1.7.2$$

$$a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + a_{34}x_4 + a_{35}x_5 = b_3 \qquad 1.7.3$$

$$a_{41}x_1 + a_{42}x_2 + a_{43}x_3 + a_{44}x_4 + a_{45}x_5 = b_4 \qquad 1.7.4$$

$$a_{51}x_5 + a_{52}x_2 + a_{53}x_3 + a_{54}x_4 + a_{55}x_5 = b_5 \qquad 1.7.5$$

There are two well-known methods of solving these equations. One of these is called Cramer's Rule. Let D be the determinant of the coefficients. Let D_i be the determinant obtained by substituting the column vector of the constants b_1, b_2, b_3, b_4, b_5 for the ith column in D. Then the solutions are

$$x_i = D_i / D \qquad 1.7.6$$

This is an interesting theorem in the theory of determinants. It should be made clear, however, that, when it comes to the practical numerical solution of a set of linear equations that may be encountered in practice, this is probably the most laborious and longest method ever devised in the history of mathematics.

The second well-known method is to write the equations in matrix form:

$$\mathbf{A}\mathbf{x} = \mathbf{b}. \qquad 1.7.7$$

Here $\mathbf{A}$ is the matrix of the coefficients, $\mathbf{x}$ is the column vector of unknowns, and $\mathbf{b}$ is the column vector of the constants. The solutions are then given by

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}, \qquad 1.7.8$$

where $\mathbf{A}^{-1}$ is the inverse or reciprocal of $\mathbf{A}$. Thus the problem reduces to inverting a matrix. Now inverting a matrix is notoriously labour-intensive, and, while the method is not quite so long as Cramer's Rule, it is still far too long for practical purposes.

How, then, should a system of linear equations be solved?

Consider the equations

$$7x - 2y = 24$$

$$3x + 9y = 30$$

Few would have any hesitation in multiplying the first equation by 3, the second equation by 7, and subtracting. This is what we were all taught in our younger days, but few realize that this remains, in spite of knowledge of determinants and matrices, the fastest and most efficient method of solving simultaneous linear equations. Let us see how it works with a system of several equations in several unknowns.

Consider the equations

$$9x_1 - 9x_2 + 8x_3 - 6x_4 + 4x_5 = -9$$

$$5x_1 - x_2 + 6x_3 + x_4 + 5x_5 = 58$$

$$2x_1 + 4x_2 - 5x_3 - 6x_4 + 7x_5 = -1$$

$$2x_1 + 3x_2 - 8x_3 - 5x_4 - 2x_5 = -49$$

$$8x_1 - 5x_2 + 7x_3 + x_4 + 5x_5 = 42$$

We first eliminate x_1 from the equations, leaving four equations in four unknowns. Then we eliminate x_2, leaving three equations in three unknowns. Then x_3, and then x_4, finally leaving a single equation in one unknown. The following table shows how it is done.

In columns 2 to 5 are listed the coefficients of x_1, x_2, x_3, x_4 and x_5, and in column 6 are the constant terms on the right hand side of the equations. Thus columns 2 to 6 of the first five rows are just the original equations. Column 7 is the sum of the numbers in columns 2 to 6, and this is a most important column. The boldface numbers in column 1 are merely labels.

Lines 6 to 9 show the elimination of x_1. Line 6 shows the elimination of x_1 from lines 1 and 2 by multiplying line 2 by 9 and line 1 by 5 and subtracting. The operation performed is recorded in column 1. In line 7, x_1 is eliminated from equations 1 and 3 and so on.

	x_1	x_2	x_3	x_4	x_5	b	Σ
1	9	−9	8	−6	4	−9	−3
2	5	−1	6	1	5	58	74
3	2	4	−5	−6	7	−1	1
4	2	3	−8	−5	−2	−49	−59
5	8	−5	7	1	5	42	58
6=9×**2**−5×**1**		36	14	39	25	567	681
7=2×**1**−9×**3**		−54	61	42	−55	−9	−15
8=**3**−**4**		1	3	−1	9	48	60
9=4×**3**−**5**		21	−27	−25	23	−46	−54
10=3×**6**+2×**7**			164	201	−35	1 683	2 013
11=**6**−36×**8**			−94	75	−299	−1 161	−1 479
12=7×**6**−12×**9**			422	573	−101	4 521	5 415
13=47×**10**+82×**11**				15 597	−26 163	−16 101	−26 667
14=211×**11**+47×**12**				42 756	−67 836	−32 484	−57 654
15=5199×**14**−14252×**13**					20 195 712	60 587 136	80 782 848

The purpose of Σ ? This column is of great importance. Whatever operation is performed on the previous columns is also performed on Σ , and Σ must remain the sum of the previous columns. If it does not, then an arithmetic mistake has been made, and it is immediately detected. There is nothing more disheartening to discover at the very end of a calculation that a mistake has been made and that one has no idea where the mistake occurred. Searching for mistakes takes far longer than the original calculation. The Σ-column enables one to detect and correct a mistake as soon as it has been made.

We eventually reach line 15, which is

$$20\,195\,712\,x_5 = 60\,587\,136,$$

from which

$$x_5 = 3.$$

x_4 can now easily be found from either or both of lines 13 and 14, x_3 can be found from any or all of lines 10, 11 and 12, and so on. When the calculation is complete, the answers should be checked by substitution in the original equations (or in the sum of the five equations). For the record, the solutions are $x_1 = 2$, $x_2 = 7$, $x_3 = 6$, $x_4 = 4$ and $x_5 = 3$.

Of course, if you have only two simultaneous equations to solve, it is easy to write down explicit algebraic expressions for the solutions, and that may be the fastest and most efficient way of doing it. Thus, if

$$a_{11}x + a_{12}y = b_1 \qquad 1.7.9$$

and

$$a_{21}x + a_{22}y = b_2, \qquad 1.7.10$$

the solutions are

$$x = c(b_1a_{22} - b_2a_{12}) \qquad 1.7.11$$

and

$$y = c(b_2a_{11} - b_1a_{21}), \qquad 1.7.12$$

where

$$c = 1/(a_{11}a_{22} - a_{12}a_{21}). \qquad 1.7.13$$

1.8 *Simultaneous Linear Equations, N > n*

Consider the following equations

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + b_1 = 0 \qquad 1.8.1$$

$$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + b_2 = 0 \qquad 1.8.2$$

$$a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + b_3 = 0 \qquad 1.8.3$$

$$a_{41}x_1 + a_{42}x_2 + a_{43}x_3 + b_4 = 0 \qquad 1.8.4$$

$$a_{51}x_1 + a_{52}x_2 + a_{53}x_3 + b_5 = 0 \qquad 1.8.5$$

Here we have five equations in only three unknowns, and there is no solution that will satisfy all five equations exactly. We refer to these equations as the *equations of condition*. The problem is to find the set of values of x_1 , x_2 and x_3 that, while not satisfying any one of the equations exactly, will come closest to satisfying all of them with as small an error as possible. The problem was well stated by Carl Friedrich Gauss in his famous *Theoria Motus*. In 1801 Gauss was faced with the problem of calculating the orbit of the newly discovered minor planet Ceres. The problem was to calculate the six elements of the planetary orbit, and he was faced with solving more than six equations for six unknowns. In the course of this, he invented the method of least squares. It is hardly possible to describe the nature of the problem more clearly than did Gauss himself:

"...as all our observations, on account of the imperfection of the instruments and the senses, are only approximations to the truth, an orbit based only on the six absolutely necessary data may still be liable to considerable errors. In order to diminish these as much as possible, and thus to reach the greatest precision attainable, no other method will be given except to accumulate the greatest number of the most perfect observations, and to adjust the elements, not so as to satisfy this or that set of observations with absolute exactness, but so as to agree with all in the best possible manner."

If we can find some set of values of x_1 , x_2 and x_3 that satisfy our five equations fairly closely, but without necessarily satisfying any one of them exactly, we shall find that, when these values are substituted into the left hand sides of the equations, the right hand sides will not be exactly zero, but will be a small number known as the residual, *R*.

Thus:

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + b_1 = R_1 \qquad 1.8.6$$

$$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + b_2 = R_2 \qquad 1.8.7$$

$$a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + b_3 = R_3 \qquad 1.8.8$$

$$a_{41}x_1 + a_{42}x_2 + a_{43}x_3 + b_4 = R_4 \qquad 1.8.9$$

$$a_{51}x_1 + a_{52}x_2 + a_{53}x_3 + b_5 = R_5 \qquad 1.8.10$$

Gauss proposed a "best" set of values such that, when substituted in the equations, gives rise to a set of residuals such that the sum of the squares of the residuals is least. (It would in principle be possible to find a set of solutions that minimized the sum of the absolute values of the residuals, rather than their squares. It turns out that the analysis and the calculation involved is a good deal more difficult than minimizing the sum of the squares, with no very obvious advantage.) Let *S* be the sum of the squares of the residuals for a given set of values of x_1 , x_2 and x_3 . If any one of the *x*-values is changed, *S* will change - unless *S* is a minimum, in which case the derivative of *S* with respect to each variable is zero. The three equations

$$\frac{\partial S}{\partial x_1} = 0, \quad \frac{\partial S}{\partial x_2} = 0, \quad \frac{\partial S}{\partial x_3} = 0 \qquad 1.8.11$$

express the conditions that the sum of the squares of the residuals is least with respect to each of the variables, and these three equations are called the *normal equations*. If the reader will write out the value of *S* in full in terms of the variables x_1 , x_2 and x_3 , he or she will find, by differentiation of *S* with respect to x_1 , x_3 and x_3 in turn, that the three normal equations are

$$A_{11}x_1 + A_{12}x_2 + A_{13}x_3 + B_1 = 0 \qquad 1.8.12$$

$$A_{12}x_1 + A_{22}x_2 + A_{23}x_3 + B_2 = 0 \qquad 1.8.13$$

$$A_{13}x_1 + A_{23}x_2 + A_{33}x_3 + B_3 = 0 \qquad 1.8.14$$

where $A_{11} = \sum a_{i1}^2, \quad A_{12} = \sum a_{i1}a_{i2}, \quad A_{13} = \sum a_{i1}a_{i3}, \quad B_1 = \sum a_{i1}b_i,$ 1.8.15

$A_{22} = \sum a_{i2}^2, \qquad A_{23} = \sum a_{i2}a_{i3}, \quad B_2 = \sum a_{i2}b_i,$ 1.8.16

$A_{33} = \sum a_{i3}^2, \qquad B_3 = \sum a_{i3}b_i,$ 1.8.17

and where each sum is from $i = 1$ to $i = 5$.

These three normal equations, when solved for the three unknowns x_1 , x_2 and x_3 , will give the three values that will result in the lowest sum of the squares of the residuals of the original five equations of condition.

Let us look at a numerical example, in which we show the running checks that are made in order to detect mistakes as they are made. Suppose the equations of condition to be solved are

$$7x_1 - 6x_2 + 8x_3 - 15 = 0 \qquad -6$$

$$3x_1 + 5x_2 - 2x_3 - 27 = 0 \qquad -21$$

$$2x_1 - 2x_2 + 7x_3 - 20 = 0 \qquad -13$$

$$4x_1 + 2x_2 - 5x_3 - 2 = 0 \qquad -1$$

$$9x_1 - 8x_2 + 7x_3 - 5 = 0 \qquad 3$$

$$-108 \quad -69 \quad -71$$

The column of numbers to the right of the equations is the sum of the coefficients (including the constant term). Let us call these numbers s_1 , s_2 , s_3 , s_4 , s_5 .

The three numbers below the equations are $\sum a_{i1}s_i, \quad \sum a_{i2}s_i, \quad \sum a_{i3}s_i.$

Set up the normal equations:

$$159x_1 - 95x_2 + 107x_3 - 279 = 0 \qquad -108$$

$$-95x_1 + 133x_2 - 138x_3 + 31 = 0 \qquad -69$$

$$107x_1 - 138x_2 + 191x_3 - 231 = 0 \qquad -71$$

The column of numbers to the right of the normal equations is the sum of the coefficients (including the constant term). These numbers are equal to the row of numbers below the equations of condition, and serve as a check that we have correctly set up the normal equations. The solutions to the normal equations are

$$x_1 = 2.474 \quad x_2 = 5.397 \quad x_3 = 3.723$$

and these are the numbers that satisfy the equations of condition such that the sum of the squares of the residuals is a minimum.

I am going to suggest here that you write a computer program, in the language of your choice, for finding the least squares solutions for N equations in n unknowns. You are going to need such a program over and over again in the future – not least when you come to Section 1.12 of this chapter!.

1.9 *Nonlinear Simultaneous Equations*

We consider two simultaneous equations of the form

$$f(x, y) = 0, \tag{1.9.1}$$

$$g(x, y) = 0, \tag{1.9.2}$$

in which the equations are not linear.

As an example, let us solve the equations

$$x^2 = \frac{a}{b - \cos y} \tag{1.9.3}$$

$$x^3 - x^2 = \frac{a(y - \sin y \cos y)}{\sin^3 y}, \tag{1.9.4}$$

in which a and b are constants whose values are assumed given in any particular case.

This may seem like an artificially contrived pair of equations, but in fact a pair of equations like this does appear in orbital theory.

We suggest here two methods of solving the equations.

In the first, we note that in fact x can be eliminated from the two equations to yield a single equation in y:

$$F(y) = aR^3 - R^2 - 2SR - S^2 = 0, \qquad 1.9.5$$

where $$R = 1/(b-\cos y) \qquad 1.9.5a$$

and $$S = (y-\sin y\cos y)/\sin^3 y. \qquad 1.9.5b$$

This can be solved by the usual Newton-Raphson method, which is repeated application of $y = y - F/F'$. The derivative of F with respect to y is

$$F' = 3aR^2R' - 2RR' - 2(S'R + SR') - 2SS', \qquad 1.9.6$$

where $$R' = -\frac{\sin y}{(b-\cos y)^2} \qquad 1.9.6a$$

and $$S' = \frac{\sin y(1-\cos 2y) - 3\cos y(y - \frac{1}{2}\sin 2y)}{\sin^4 y} \qquad 1.9.6b$$

In spite of what might appear at first glance to be some quite complicated equations, it will be found that the Newton-Raphson process, $y = y - F/F'$, is quite straightforward to program, although, for computational purposes, F and F' are better written as

$$F = -S^2 + R(-2S + R(-1 + aR)), \qquad 1.9.7a$$

and $$F' = 3aR^2R' - 2(R+S)(R' + S') \qquad 1.9.7.b$$

Let us look at a particular example, say with $a = 36$ and $b = 4$. We must of course, make a first guess. In the orbital application, described in Chapter 13, we suggest a first guess. In the present case, with $a = 36$ and $b = 4$, one way would be to plot graphs of equations 1.9.3 and 1.9.4 and see where they intersect. We have done this in Figure 1.4, from which we see that y must be close to 0.6.

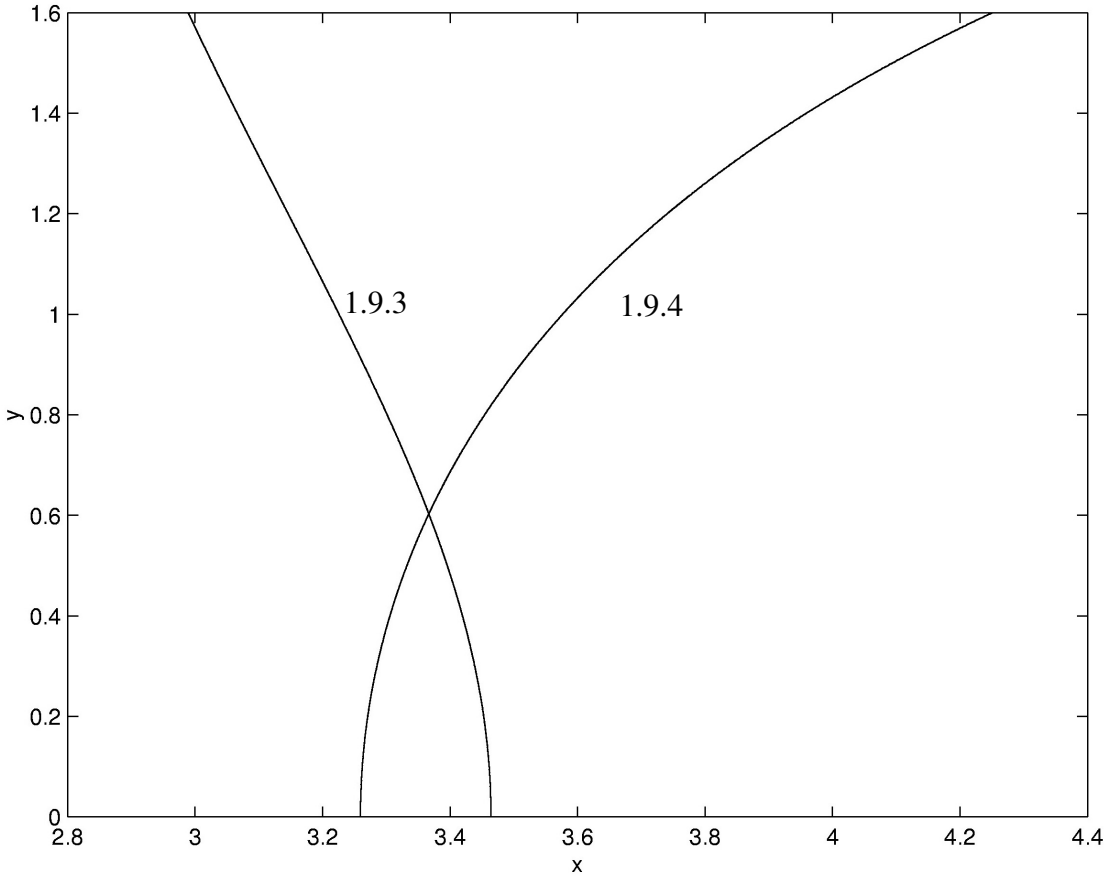


FIGURE 1.4
Equations 1.9.3 and 1.9.4 with $a = 36$ and $b = 4$.

With a first guess of $y = 0.6$, convergence to $y = 0.60292$ is reached in two iterations, and either of the two original equations then gives $x = 3.3666$.

We were lucky in this case in that we found we were able to eliminate one of the variables and so reduce the problem to a single equation on one unknown. However, there will be occasions where elimination of one of the unknowns may be considerably more difficult or, in the case of two simultaneous transcendental equations, impossible by algebraic means. The following iterative method, an extension of the Newton-Raphson technique, can nearly always be used. We describe it for two equations in two unknowns, but it can easily be extended to n equations in n unknowns.

The equations to be solved are

$$f(x,\ y)\ =\ 0 \qquad 1.9.8$$

$$g(x,\ y)\ =\ 0. \qquad 1.9.9$$

As with the solution of a single equation, it is first necessary to guess at the solutions. This might be done in some cases by graphical methods. However, very often, as is common with the Newton-Raphson method, convergence is rapid even when the first guess is very wrong.

Suppose the initial guesses are $x + h,\ y + k,$ where $x,\ y$ are the correct solutions, and h and k are the errors of our guess. From a first-order Taylor expansion (or from common sense, if the Taylor expansion is forgotten),

$$f(x+h, y+k)\ \approx\ f(x, y)+hf_x+kf_y. \qquad 1.9.10$$

Here f_x and f_y are the partial derivatives and of course $f(x,\ y) = 0$. The same considerations apply to the second equation, so we arrive at the two linear equations in the errors h, k:

$$f_x h\ +\ f_y k\ =\ f, \qquad 1.9.11$$

$$g_x h\ +\ g_y k\ =\ g. \qquad 1.9.12$$

These can be solved for h and k:

$$h\ =\ \frac{g_y f - f_y g}{f_x g_y - f_y g_x}, \qquad 1.9.13$$

$$k\ =\ \frac{f_x g - g_x f}{f_x g_y - f_y g_x}. \qquad 1.9.14$$

These values of h and k are then subtracted from the first guess to obtain a better guess. The process is repeated until the changes in x and y are as small as desired for the particular application. It is easy to set up a computer program for solving any two equations; all that will change from one pair of equations to another are the definitions of the functions f and g and their partial derivatives.

In the case of our example, we have

$$f = x^2 - \frac{a}{b - \cos y} \qquad 1.9.15$$

$$g = x^3 - x^2 - \frac{a(y - \sin y \cos y)}{\sin^3 y} \qquad 1.9.16$$

$$f_x = 2x \qquad 1.9.17$$

$$f_y = \frac{a \sin y}{(b - \cos y)^2} \qquad 1.9.18$$

$$g_x = x(3x - 2) \qquad 1.9.19$$

$$g_y = \frac{a[3(y - \sin y \cos y)\cos y - 2\sin^3 y]}{\sin^4 y} \qquad 1.9.20$$

In the particular case where $a = 36$ and $b = 4$, we can start with a first guess (from the graph - Figure I.4) of $y = 0.6$ and hence $x = 3.3$. Convergence to one part in a million is reached in three iterations, the solutions being $x = 3.3666$, $y = 0.60292$.

A simple application of these considerations arises if you have to solve a polynomial equation $f(z) = 0$, where there are no real roots, and all solutions for z are complex. You then merely write $z = x + iy$ and substitute this in the polynomial equation. Then equate the real and imaginary parts separately, to obtain two equations of the form

$$R(x, y) = 0 \qquad 1.9.21$$

$$I(x, y) = 0 \qquad 1.9.22$$

and solve them for x and y. For example, find the roots of the equation

$$z^4 - 5z + 6 = 0. \qquad 1.9.23$$

It will soon be found that we have to solve

$$R(x, y) = x^4 - 6x^2y^2 + y^4 - 5x + 6 = 0 \qquad 1.9.24$$

$$I(x, y) = 4x^3 - 4xy^2 - 5 = 0 \qquad 1.9.25$$

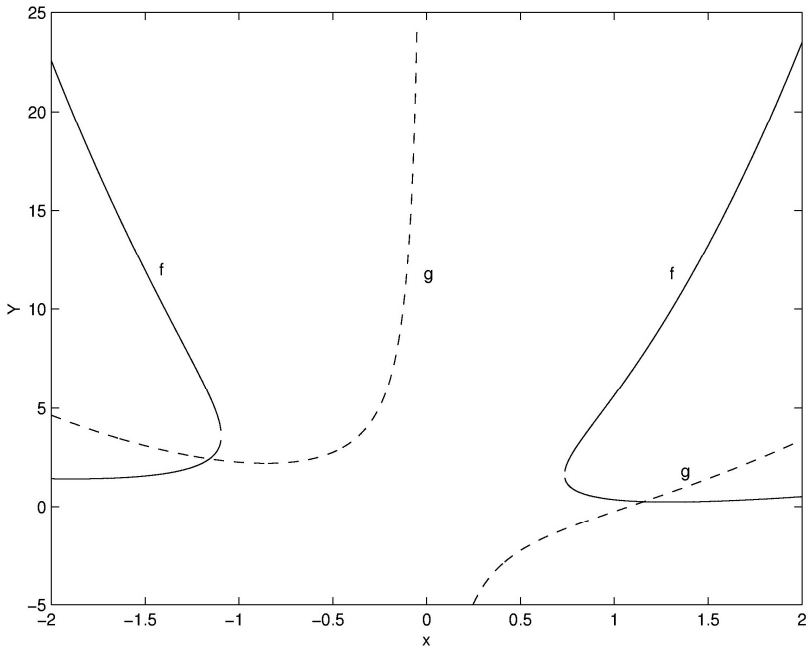
It will have been observed that, in order to obtain the last equation, we have divided through by y, which is permissible, since we know z to be complex. We also note that y now occurs only as y^2, so it will simplify things if we let $y^2 = Y$, and then solve the equations

$$f(x, Y) = x^4 - 6x^2Y + Y^2 - 5x + 6 = 0 \qquad 1.9.26$$

$$g(x, Y) = 4x^3 - 4xY - 5 = 0 \qquad 1.9.27$$

It is then easy to solve either of these for Y as a function of x and hence to graph the two functions (figure I.5):

FIGURE I.5



This enables us to make a first guess for the solutions, namely

$$x = -1.2, \quad Y = 2.4$$

and

$$x = +1.2, \quad Y = 0.3$$

We can then refine the solutions by the extended Newton-Raphson technique to obtain

$$x = -1.15697, \quad Y = 2.41899$$
$$x = +1.15697, \quad Y = 0.25818$$

so the four solutions to the original equation are

$$z = -1.15697 \pm 1.55531i$$

$$z = 1.15697 \pm 0.50812i$$

1.10 *Besselian Interpolation*

In the days before the widespread use of high-speed computers, extensive use was commonly made of printed tables of the common mathematical functions. For example, a table of the Bessel function $J_0(x)$ would indicate

$$J_0(1.7) = 0.397\,984\,859$$
$$J_0(1.8) = 0.339\,986\,411$$

If one wanted the Bessel function for $x = 1.762$, one would have to interpolate between the tabulated values.

Today it would be easier simply to calculate the Bessel function for any particular desired value of the argument x, and there is less need today for printed tables or to know how to interpolate. Indeed, most computing systems today have internal routines that will enable one to calculate the commoner functions such as Bessel functions even if one has only a hazy notion of what a Bessel function is.

The need has not entirely passed, however. For example, in orbital calculations, we often need the geocentric coordinates of the Sun. These are not trivial for the nonspecialist to compute, and it may be easier to look them up in *The Astronomical Almanac*, where it is tabulated for every day of the year, such as, for example, July 14 and July 15. But, if one needs y for July 14.395, how does one interpolate?

In an ideal world, a tabulated function would be tabulated at sufficiently fine intervals so that linear interpolation between two tabulated values would be adequate to return the function to the same number of significant figures as the tabulated points. The world is not perfect, however, and to achieve such perfection, the tabulation interval would have to change as the function changed more or less rapidly. We need to know, therefore, how to do nonlinear interpolation.

Suppose a function $y(x)$ is tabulated at $x = x_1$ and $x = x_2$, the interval $x_2 - x_1$ being δx. If one wishes to find the value of y at $x + \theta\delta x$, linear interpolation gives

$$y(x_1 + \theta\Delta x) = y_1 + \theta(y_2 - y_1) = \theta y_2 + (1-\theta)y_1, \qquad 1.10.1$$

where $y_1 = y(x_1)$ and $y_2 = y(x_2)$. Here it is assumed that is a fraction between 0 and 1; if θ is outside this range (that is negative, or greater than 1) we are extrapolating, not interpolating, and that is always a dangerous thing to do.

Let us now look at the situation where linear interpolation is not good enough. Suppose that a function $y(x)$ is tabulated for four points x_1, x_2, x_3, x_4 of the argument x, the corresponding values of the function being y_1, y_2, y_3, y_4. We wish to evaluate y for $x = x_2 + \theta\delta x$, where δx is the interval $x_2 - x_1$ or $x_3 - x_2$ or $x_4 - x_3$. The situation is illustrated in figure I.6A.

A possible approach would be to fit a polynomial to the four adjacent points:

$$y = a + bx + cx^2 + dx^3.$$

We write down this equation for the four adjacent tabulated points and solve for the coefficients, and hence we can evaluate the function for any value of x that we like in the interval between x_1 and x_4 . Unfortunately, this might well involve more computational effort than evaluating the original function itself.

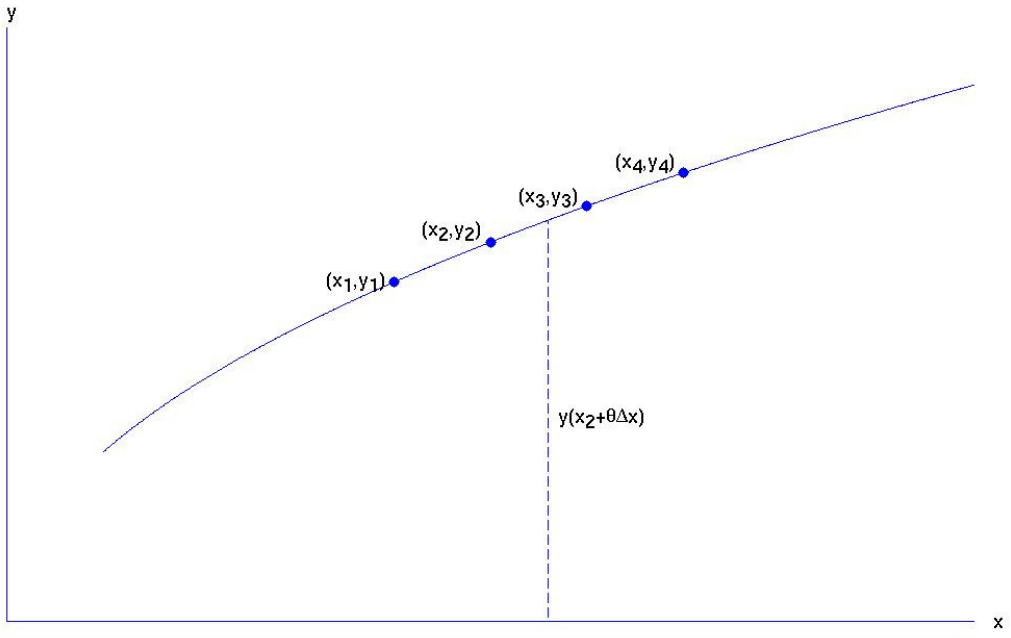
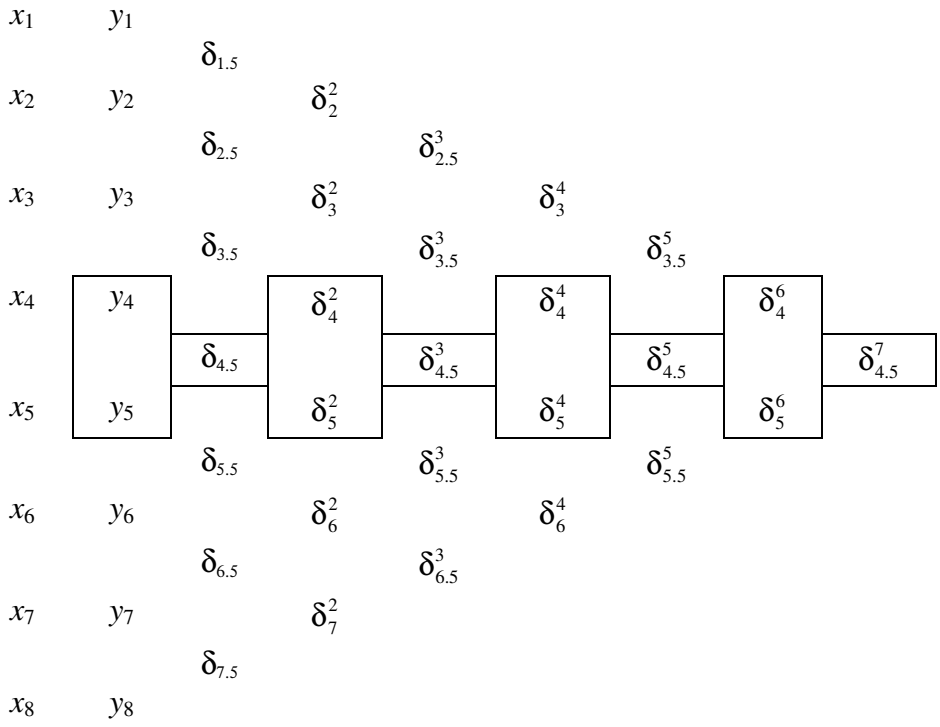


FIGURE I.6A

The problem has been solved in a convenient fashion in terms of finite difference calculus, the logical development of which would involve an additional substantial chapter beyond the intended scope of this book. I therefore just provide the method only, without proof.

The essence of the method is to set up a table of differences as illustrated below. The first two columns are x and y. The entries in the remaining columns are the differences between the two entries in the column immediately to the left. Thus, for example, $\delta_{4.5} = y_5 - y_4$, $\delta_4^2 = \delta_{4.5} - \delta_{3.5}$, etc.



Let us suppose that we want to find y for a value of x that is a fraction θ of the way from x_4 to x_5. Bessel's interpolation formula is then

$$y(x) = \tfrac{1}{2}(y_4 + y_5) + B_1\delta_{4.5} + B_2(\delta^2_4 + \delta^2_5) + B_3\delta^3_{4.5} + B_4(\delta^4_4 + \delta^4_5) + \ldots \qquad 1.10.3$$

Here the B_n are the Besselian interpolation coefficients, and the successive terms in parentheses in the expansion are the sums of the numbers in the boxes in the table.

The Besselian coefficients are

$$B_n(\theta) = \tfrac{1}{2}\binom{\theta+\frac{1}{2}n-1}{n} \qquad \text{if } n \text{ is even,} \qquad 1.10.4$$

and

$$B_n(\theta) = \frac{\theta-\frac{1}{2}}{n}\binom{\theta+\frac{1}{2}n-\frac{3}{2}}{n-1} \qquad \text{if } n \text{ is odd.} \qquad 1.10.5$$

The notation $\binom{m}{n}$ means the coefficient of x^m in the binomial expansion of $(1 + x)^n$.

Explicitly, $B_1 = \theta - \frac{1}{2}$ 1.10.6

$B_2 = \frac{1}{2}\theta(\theta-1)/2! = \theta(\theta-1)/4$ 1.10.7

$B_3 = (\theta-\frac{1}{2})\theta(\theta-1)/3! = \theta(0.5 + \theta(-1.5 + \theta))/6$ 1.10.8

$B_4 = \frac{1}{2}(\theta+1)\theta(\theta-1)(\theta-2)/4! = \theta(2 + \theta(-1 + \theta(-2 + \theta)))/48$ 1.10.9

$B_5 = (\theta-\frac{1}{2})(\theta+1)\theta(\theta-1)(\theta-2)/5! = \theta(-1 + \theta(2.5 + \theta^2(-2.5 + \theta)))/120$ 1.10.10

The reader should convince him- or herself that the interpolation formula taken as far as B_1 is merely linear interpolation. Addition of successively higher terms effectively fits a curve to more and more points around the desired value and more and more accurately reflects the actual change of *y* with *x*.

t	*y*			
1	0.920 6928			
		−26037		
2	0.918 0891		−2600	
		−28637		9
3	0.915 2254		−2591	
		−31228		8
4	0.912 1026		−2583	
		−33811		10
5	0.908 7215		−2573	
		−36384		13
6	0.905 0831		−2560	
		−38944		11
7	0.901 1887		−2549	
		−41493		
8	0.897 0394			

The above table is taken from *The Astronomical Almanac* for 1997, and it shows the *y*-coordinate of the Sun for eight consecutive days in July. The first three difference columns are tabulated, and it is clear that further difference columns are unwarranted.

If we want to find the value of *y*, for example, for July 4.746, we have $\theta = 0.746$ and the first three Bessel coefficients are

$B_1 = +0.246$
$B_2 = -0.047\,371$
$B_3 = -0.007\,768\,844$

The reader can verify the following calculations for y from the sum of the first 2, 3 and 4 terms of the Besselian interpolation series formula. The sum of the first two terms is the result of linear interpolation.

Sum of the first 2 terms, $y = 0.909\,580\,299$
Sum of the first 3 terms, $y = 0.909\,604\,723$
Sum of the first 4 terms, $y = 0.909\,604\,715$

Provided the table is not tabulated at inappropriately coarse intervals, one need rarely go past the third Bessel coefficient. In that case an alternative and equivalent interpolation formula (for $t = t_4 + \theta\,\Delta t$), which avoids having to construct a difference table, is

$$y(t_4 + \theta\Delta t) = -\tfrac{1}{6}\theta\left[(2 - \theta(3 - \theta))\, y_3 + (1 - \theta)\, y_6\right] + \tfrac{1}{2}\left[(2 + \theta(-1 + \theta(-2 + \theta)))\, y_4 + \theta(2 + \theta(1 - \theta))\, y_5\right].$$

Readers should check that this gives the same answer, at the same time noting that the nested parentheses make the calculation very rapid and they are easy to program on either a calculator or a computer.

Exercise: From the following table, construct a difference table up to fourth differences. Calculate the first four Bessel coefficients for $\theta = 0.73$. Hence calculate the value of y for $x = 0.273$.

x	y
0.0	+0.381300
0.1	+0.285603
0.2	+0.190092
0.3	+0.096327
0.4	+0.008 268
0.5	−0.067725

Answers: $B_1 = +0.23$ $B_2 = -0.049275$ $B_3 = -7.5555 \times 10^{-3}$
$B_4 = +9.021841875 \times 10^{-3}$ $y = 0.121289738$

Note: the table was calculated from a formula, and the interpolated answer is correct to nine significant figures.

Exercise: From the following table of sin x, use linear interpolation and Besselian interpolation to estimate sin 51° to three significant figures.

x^o	$\sin x$
0	0.0
30	0.5
60	$\sqrt{3}/2=0.86603$
90	1.0

Answers: By linear interpolation, $\sin 51^o = 0.634$.

By Besselian interpolation, $\sin 51^o = 0.776$.

The correct value is 0.777. You should be impressed – but there is more on interpolation to come, in Section 1.11.

For Sections 1.11 and 1.14-16 I am much indebted to the late Max Fairbairn of Wentworth Falls, Australia, who persuaded me of the advantages of Lagrangian Interpolation and of Gaussian Quadrature. Much of these sections is adapted directly from correspondence with Max.

1.11 *Fitting a Polynomial to a Set of Points. Lagrange Polynomials. Lagrange Interpolation.*

Given a set of n points on a graph, there any many possible polynomials of sufficiently high degree that go through all n of the points. There is, however, just one polynomial of degree less than n that will go through them all. Most readers will find no difficulty in determining the polynomial. For example, consider the three points (1 , 1), (2 , 2) , (3 , 2). To find the polynomial $y = a_0 + a_1x + a_2x^2$ that goes through them, we simply substitute the three points in turn and hence set up the three simultaneous equations

$$\begin{aligned} 1 &= a_0 + a_1 + a_2 \\ 2 &= a_0 + 2a_1 + 4a_2 \\ 2 &= a_0 + 3a_1 + 9a_2 \end{aligned} \qquad 1.11.1$$

and solve them for the coefficients. Thus $a_0 = -1$, $a_1 = 2.5$ and $a_3 = -0.5$. In a similar manner we can fit a polynomial of degree $n-1$ to go exactly through n points. If there are *more than n* points, we may wish to fit a *least squares polynomial* of degree $n-1$ to go close to the points, and we show how to do this in sections 1.12 and 1.13. For the purpose of this section (1.11), however, we are interested in fitting a polynomial of degree $n-1$ exactly through n points, and we are going to show how to do this by means of Lagrange polynomials as an alternative to the method described above.

While the smallest-degree polynomial that goes through n points is usually of degree $n-1$, it could be less than this. For example, we might have four points, all of which fit exactly on a parabola (degree two). However, in general one would expect the polynomial to be of degree $n-1$, and, if this is not the case, all that will happen is that we shall find that the coefficients of the highest powers of x are zero.

That was straightforward. However, what we are going to do in this section is to fit a polynomial to a set of points by using some functions called *Lagrange polynomials*. These are functions that are described by Max Fairbairn as "cunningly engineered" to aid with this task.

Let us suppose that we have a set of n points:

$$(x_1, y_1), (x_1, y_1), (x_2, y_2), \ldots \quad \ldots (x_i, y_i), \ldots \quad \ldots (x_n, y_n),$$

and we wish to fit a polynomial of degree $n-1$ to them.

I assert that the function

$$y = \sum_{i=1}^{n} y_i L_i(x) \qquad 1.11.2$$

is the required polynomial, where the n functions $L_i(x), \quad i = 1, n,$ are n *Lagrange polynomials*, which are polynomials of degree $n-1$ defined by

$$L_i(x) = \prod_{\substack{j=1 \\ j \neq i}}^{n} \frac{x - x_j}{x_i - x_j}. \qquad 1.11.3$$

Written more explicitly, the first three Lagrange polynomials are

$$L_1(x) = \frac{(x - x_2)(x - x_3)(x - x_4) \ldots \quad \ldots (x - x_n)}{(x_1 - x_2)(x_1 - x_3)(x_1 - x_4) \ldots \quad \ldots (x_1 - x_n)}, \qquad 1.11.4$$

and

$$L_2(x) = \frac{(x - x_1)(x - x_3)(x - x_4) \ldots \quad \ldots (x - x_n)}{(x_2 - x_1)(x_2 - x_3)(x_2 - x_4) \ldots \quad \ldots (x_2 - x_n)} \qquad 1.11.5$$

and

$$L_3(x) = \frac{(x - x_1)(x - x_2)(x - x_4) \ldots \quad \ldots (x - x_n)}{(x_3 - x_1)(x_3 - x_2)(x_3 - x_4) \ldots \quad \ldots (x_3 - x_n)}. \qquad 1.11.6$$

At first encounter, this will appear meaningless, but with a simple numerical example it will become clear what it means and also that it has indeed been cunningly engineered for the task.

Consider the same points as before, namely (1 , 1), (2 , 2) , (3 , 2). The three Lagrange polynomials are

$$L_1(x) = \frac{(x-2)(x-3)}{(1-2)(1-3)} = \tfrac{1}{2}(x^2 - 5x + 6), \qquad 1.11.7$$

$$L_2(x) = \frac{(x-1)(x-3)}{(2-1)(2-3)} = -x^2 + 4x - 3, \qquad 1.11.8$$

$$L_3(x) = \frac{(x-1)(x-2)}{(3-1)(3-2)} = \tfrac{1}{2}(x^2 - 3x + 2). \qquad 1.12.9$$

Equation 1.11.2 for the polynomial of degree $n - 1$ that goes through the three points is, then,

$$y = 1\times\tfrac{1}{2}(x^2 - 5x + 6) + 2\times(-x^2 + 4x - 3) + 2\times\tfrac{1}{2}(x^2 - 3x + 2); \qquad 1.11.10$$

that is

$$y = -\tfrac{1}{2}x^2 + \tfrac{5}{2}x - 1, \qquad 1.11.11$$

which agrees with what we got before.

One way or another, if we have found the polynomial that goes through the n points, we can then use the polynomial to interpolate between nontabulated points. Thus we can either determine the coefficients in $y = a_0 + a_1x^2 + a_2x^2 \ldots$ by solving n simultaneous equations, or we can use equation 1.11.2 directly for our interpolation (without the need to calculate the coefficients a_0 , a_1, etc.), in which case the technique is known as *Lagrangian interpolation.* If the tabulated function for which we need an interpolated value is a polynomial of degree less than n, the interpolated value will be exact. Otherwise it will be approximate. An advantage of this over Besselian interpolation is that it is not necessary that the function to be interpolated be tabulated at equal intervals in x. Most mathematical functions and astronomical tables, however, are tabulated at equal intervals, and in that case either method can be used.

Let us recall the example that we had in Section 1.10 on Besselian interpolation, in which we were asked to estimate the value of sin 51° from the table

x°	sin x
0	0.0
30	0.5
60	√3/2=0.86603
90	1.0

The four Lagrangian polynomials, evaluated at $x = 51$, are

$$L_1(51) = \frac{(51-30)(51-60)(51-90)}{(0-30)(0-60)(0-90)} = -0.0455,$$

$$L_2(51) = \frac{(51-0)(51-60)(51-90)}{(30-0)(30-60)(30-90)} = +0.3315,$$

$$L_3(51) = \frac{(51-0)(51-30)(51-90)}{(60-0)(60-30)(60-90)} = +0.7735,$$

$$L_4(51) = \frac{(51-0)(51-30)(51-60)}{(90-0)(90-30)(90-60)} = -0.0595.$$

Equation 1.11.2 then gives us

$$\sin 51^\circ = 0 \times (-0.0455) + 0.5 \times 0.3315 + 0.86603 \times 0.7735 + 1 \times (-0.0595)$$
$$= 0.776.$$

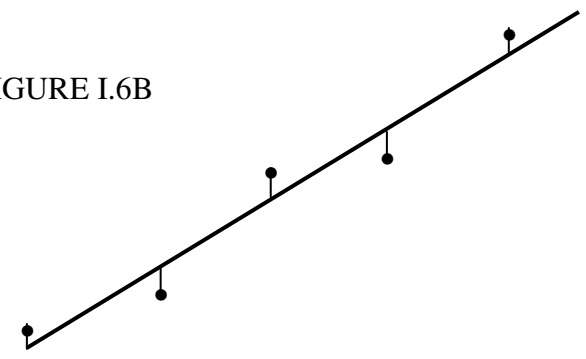
This is the same as we obtained with Besselian interpolation, and compares well with the correct value of 0.777. I point out again, however, that the Lagrangian method can be used if the function is not tabulated at equal intervals, whereas the Besselian method requires tabulation at equal intervals.

1.12 *Fitting a Least Squares Straight Line to a set of Observational Points*

Very often we have a set of observational points (x_i , y_i), $i = 1$ to N, that seem to fall roughly but not quite on a straight line, and we wish to draw the "best" straight line that passes as close as possible to all the points. Even the smallest of scientific hand calculators these days have programs for doing this – but it is well to understand precisely what it is that is being calculated.

Very often the values of x_i are known "exactly" (or at least to a high degree of precision) but there are appreciable errors in the values of y_i. In figure I.6B I show a set of points and a plausible straight line that passes close to the points.

FIGURE I.6B



Also drawn are the vertical distances from each point from the straight line; these distances are the *residuals* of each point.

It is usual to choose as the "best" straight line that line such that the sum of the squares of these residuals is least. You may well ask whether it might make at least equal sense to choose as the "best" straight line that line such that the sum of the absolute values of the residuals is least. That certainly does make good sense, and in some circumstances it may even be the appropriate line to choose. However, the "least squares" straight line is rather easier to calculate and is readily amenable to statistical analysis. Note also that using the *vertical* distances between the points and the straight line is appropriate only if the values of x_i are known to much higher precision than the values of y_i. In practice, this is often the case – but it is not always so, in which case this would not be the appropriate "best" line to choose.

The line so described – i.e. the line such that the sum of the squares of the vertical residuals is least is often called loosely the "least squares straight line". Technically, it is the *least squares linear regression of y upon x.* It might be under some circumstances that it is the values of y_i that are known with great precision, whereas there may be appreciable errors in the x_i. In that case we want to minimize the sum of the squares of the *horizontal* residuals, and we then calculate the *least squares linear regression of x upon y*. Yet again, we may have a situation in which the errors in x and y are comparable (not necessarily exactly equal). In that case we may want to minimize the sum of the squares of the *perpendicular* residuals of the points from the line. But then there is a difficulty of drawing the x- and y-axes to equal scales, which would be problematic if, for example, x were a time and y a distance.

To start with, however, we shall assume that the errors in x are negligible and we want to calculate the least squares regression of y upon x. We shall also make the assumption that all points have *equal weight.* If they do not, this is easily dealt with in an obvious manner; thus, if a point has twice the weight of other points, just count that point twice.

So, let us suppose that we have N points, (x_i , y_i), $i = 1$ to N, and we wish to fit a straight line that goes as close as possible to all the points. Let the line be $y = a_1x + a_0$. The *residual* R_i of the ith point is

$$R_i = y_i - (a_1x_i + a_0). \qquad 1.12.1$$

We have N simultaneous linear equations of this sort for the two unknowns a_1 and a_0, and, for the least squares regression of y upon x, we have to find the values of a_1 and a_0 such that the sum of the squares of the residuals is least. *We already know how to do this* from Section 1.8, so the problem is solved. (Just make sure that you understand that, in Section 1.8 we were using x for the unknowns and a for the coefficients; here we are doing the opposite!)

Now for an *Exercise*. Suppose our points are as follows:

x	y
1	1.00
2	2.50
3	2.75
4	3.00
5	3.50

i.) Draw these points on a sheet of graph paper and, using your eye and a ruler, draw what you think is the best straight line passing close to these points.

ii.) Write a computer program for calculating the least squares regression of y upon x. You've got to do this sooner or later, so you might as well do it now. In fact you should already (after you read Section 1.8) have written a program for solving N equations in n unknowns, so you just incorporate that program into this.

iii.) Now calculate the least squares regression of y upon x. I make it $y = 0.55x + 0.90$. Draw this on your graph paper and see how close your eye-and-ruler estimate was!

iv.) How are you going to calculate the least squares regression of x upon y? Easy! Just use the same program, but read the x-values for y and the y-values for x! No need to write a second program! I make it $y = 0.645x + 0.613$. Draw that on your graph paper and see how it compares with the regression of y upon x.

The two regression lines intersect at the centroid of the points, which in this case is at (3.00, 2.55). If the errors in x and y are comparable, a reasonable best line might be one that passes through the centroid, and whose slope is the mean (arithmetic? geometric?) of the regressions of y upon x and x upon y. However, in Section 1.12 I shall give a reference to where this question is treated more thoroughly.

If the regressions of y upon x and x upon y are respectively $y = a_1x + a_0$ and $y = b_1x + b_0$, the quantity $\sqrt{a_1 / b_1}$ is called the *correlation coefficient* r between the *variates* x and y. If the points are exactly on a straight line, the correlation coefficient is 1. The correlation coefficient is often used to show how well, or how badly, two variates are correlated, and it is often averred that they are highly correlated if r is close to 1 and only weakly correlated if r is close to zero. I am not intending to get bogged down in formal statistics in this chapter, but a word of warning here is in order. If you have just two points, they are necessarily on a straight line, and the correlation coefficient is necessarily 1 – but there is no evidence whatever that the variates are in any way correlated. The correlation coefficient by itself does not tell how closely correlated two variates are. The *significance* of the correlation coefficient depends on the number of points, and the significance is something that can be calculated numerically by precise statistical tests.

1.13 *Fitting a Least Squares Polynomial to a Set of Observational Points*

I shall start by assuming that the values of x are known to a high degree of precision, and all the errors are in the values of y. In other words, I shall calculate a least squares polynomial regression of y upon x. In fact I shall show how to calculate a least squares *quadratic* regression of y upon x, a quadratic polynomial representing, of course, a parabola. What we want to do is to calculate the coefficients a_0, a_1, a_2 such that the sum of the squares of the residual is least, the residual of the ith point being

$$R_i = y_i - (a_0 + a_1 x_i + a_2 x_1^2). \qquad 1.13.1$$

You have N simultaneous linear equations of this sort for the three unknowns a_0, a_1 and a_2. You already know how to find the least squares solution for these, and indeed, after having read Section 1.8, you already have a program for solving the equations. (Remember that here the unknowns are a_0, a_1 and a_2 – not x! You just have to adjust your notation a bit.) Thus there is no difficulty in finding the least squares quadratic regression of y upon x, and indeed the extension to polynomials of higher degree will now be obvious.

As an *Exercise*, here are some points that I recently had in a real application:

x	y
395.1	171.0
448.1	289.0
517.7	399.0
583.3	464.0
790.2	620.0

Draw these on a sheet of graph paper and draw by hand a nice smooth curve passing as close as possible to the point. Now calculate the least squares parabola (quadratic regression of y upon x) and see how close you were. I make it $y = -961.34 + 3.7748x - 2.247\times10^{-3}x^2$. It is shown in Figure I.6C.

I now leave you to work out how to fit a least squares cubic (or indeed any polynomial) regression of y upon x to a set of data points. For the above data, I make the cubic fit to be

$$y = -2537.605 + 12.4902x - 0.017777x^2 + 8.89\times10^{-6}x^3.$$

This is shown in Figure I.6D, and, on the scale of this drawing it cannot be distinguished (within the range covered by x in the figure) from the quartic equation that would go exactly through all five points.

The cubic curve is a "better" fit than either the quadratic curve or a straight line in the sense that, the higher the degree of polynomial, the closer the fit and the less the residuals. But higher degree polynomials have more "wiggles", and you have to ask yourself whether a high-degree

polynomial with lots of "wiggles" is really a realistic fit, and maybe you should be satisfied with a quadratic fit. Above all, it is important to understand that it is very dangerous to use the curve that you have calculated to *extrapolate* beyond the range of x for which you have data – and this is especially true of higher-degree polynomials.

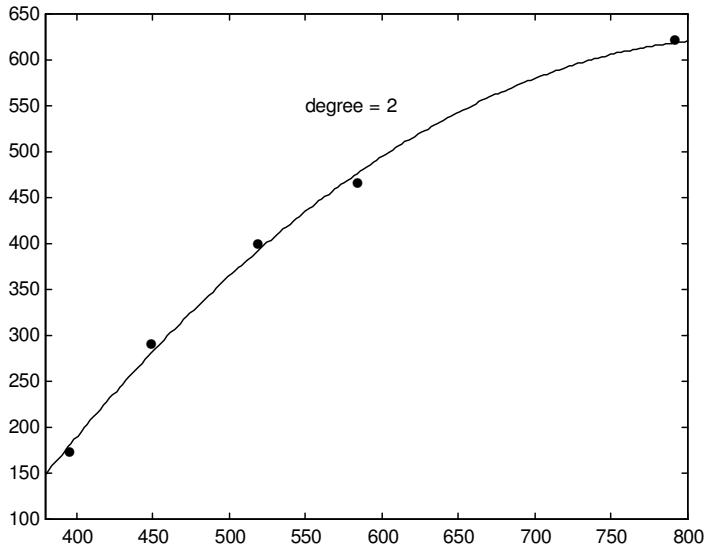


FIGURE I.6C

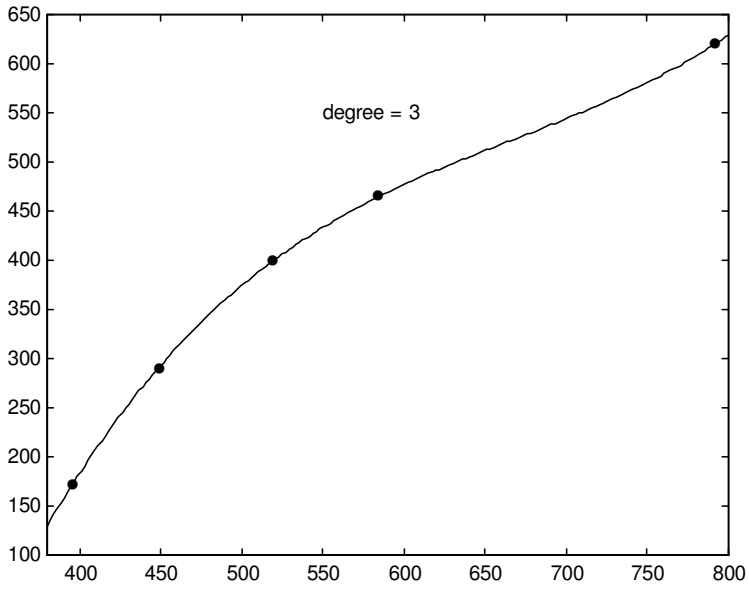


FIGURE I.6D

What happens if the errors in x are not negligible, and the errors in x and y are comparable in size? In that case you want to plot a graph of y against x on a scale such that the unit for x is equal to the standard deviation of the x-residuals from the chosen polynomial and the unit for y is equal to the standard deviation of the y-residuals from the chosen polynomial. For a detailed and

thorough account of how to do this, I refer you to a paper by D. York in *Canadian Journal of Physics,* **44**, 1079 (1966).

1.14 *Legendre Polynomials*

Consider the expression

$$(1 - 2rx + r^2)^{-1/2}, \qquad 1.14.1$$

in which $|x|$ and $|r|$ are both less than or equal to one. Expressions similar to this occur quite often in theoretical physics - for example in calculating the gravitational or electrostatic potentials of bodies of arbitrary shape. See, for example, Chapter 5, Sections 5.11 and 5.12.

Expand the expression 1.14.1 by the binomial theorem as a series of powers of r. This is straightforward, though not particularly easy, and you might expect to spend several minutes in obtaining the coefficients of the first few powers of r. You will find that the coefficient of r^l is a polynomial expression in x of degree l. Indeed the expansion takes the form

$$(1 - 2rx + r^2)^{-1/2} = P_0(x) + P_1(x)r + P_2(x)r^2 + P_3(x)r^3 \ldots \qquad 1.14.2$$

The coefficients of the successive power of r are the *Legendre polynomials*; the coefficient of r^l, which is $P_l(x)$, is the Legendre polynomial of order l, and it is a polynomial in x including terms as high as x^l. We introduce these polynomials in this section because we shall need them in Section 1.15 on the derivation of Gaussian Quadrature.

If you have conscientiously tried to expand expression 1.14.1, you will have found that

$$P_0(x) = 1, \qquad P_1(x) = x, \qquad P_2(x) = \tfrac{1}{2}(3x^2 - 1), \qquad 1.14.3$$

though you probably gave up with exhaustion after that and didn't take it any further. If you look carefully at how you derived the first few polynomials, you may have discovered for yourself that you can obtain the next polynomial as a function of two earlier polynomials. You may even have discovered for yourself the following *recursion relation*:

$$P_{l+1} = \frac{(2l+1)xP_l - lP_{l-1}}{l+1}. \qquad 1.14.4$$

This enables us very rapidly to obtain higher order Legendre polynomials, whether numerically or in algebraic form. For example, put $l = 1$ and show that equation 1.12.4 results in $P_2 = \frac{1}{2}(3x^2 - 1)$. You will then want to calculate P_3, and then P_4, and more and more and more. Another way to generate them is form the equation

$$P_{l+1} = \frac{1}{2^l l!}\frac{d^l}{dx^l}(x^2 - 1)^l . \qquad 1.14.5$$

Here are the first eleven Legendre polynomials:

$$\begin{aligned}
P_0 &= 1 \\
P_1 &= x \\
P_2 &= \tfrac{1}{2}(3x^2 - 1) \\
P_3 &= \tfrac{1}{2}(5x^3 - 3x) \\
P_4 &= \tfrac{1}{8}(35x^4 - 30x^2 + 3) \\
P_5 &= \tfrac{1}{8}(63x^5 - 70x^3 + 15x) \\
P_6 &= \tfrac{1}{16}(231x^6 - 315x^4 + 105x^2 - 5) \\
P_7 &= \tfrac{1}{16}(429x^7 - 693x^5 + 315x^3 - 35x) \\
P_8 &= \tfrac{1}{128}(6435x^8 - 12012x^6 + 6930x^4 - 1260x^2 + 35) \\
P_9 &= \tfrac{1}{128}(12155x^9 - 25740x^7 + 18018x^5 - 4620x^3 + 315x) \\
P_{10} &= \tfrac{1}{256}(46189x^{10} - 109395x^8 + 90090x^6 - 30030x^4 + 3465x^2 - 63)
\end{aligned} \qquad 1.14.6$$

The polynomials with argument cos θ are given in Section 5.11 of Chapter 5.

In what follows in the next section, we shall also want to know the roots of the equations $P_l = 0$ for $l > 1$. Inspection of the forms of these polynomials will quickly show that all odd polynomials have a root of zero, and all nonzero roots occur in positive/negative pairs. Having read Sections 1.4 and 1.5, we shall have no difficulty in finding the roots of these equations. The roots $x_{l,i}$ are in the following table, which also lists certain coefficients $c_{l,i}$ that will be explained in Section 1.15.

Roots of $P_l = 0$

l	$x_{l,i}$	$c_{l,i}$
2	$\pm$0.577 350 269 190	1.000 000 000 00
3	$\pm$0.774 596 669 241	0.555 555 555 56
	0.000 000 000 000	0.888 888 888 89
4	$\pm$0.861 136 311 594	0.347 854 845 14
	$\pm$0.339 981 043 585	0.652 145 154 86
5	$\pm$0.906 179 845 939	0.236 926 885 06
	$\pm$0.538 469 310 106	0.478 628 670 50
	0.000 000 000 000	0.568 888 888 89
6	$\pm$0.932 469 514 203	0.171 324 492 38
	$\pm$0.661 209 386 466	0.360 761 573 05
	$\pm$0.238 619 186 083	0.467 913 934 57
7	$\pm$0.949 107 912 343	0.129 484 966 17
	$\pm$0.741 531 185 599	0.279 705 391 49
	$\pm$0.405 845 151 377	0.381 830 050 50
	0.000 000 000 000	0.417 959 183 68

l	$x_{l,i}$	$c_{l,i}$
8	±0.960 289 856 498	0.101 228 536 29
	±0.796 666 477 414	0.222 381 034 45
	±0.525 532 409 916	0.313 706 645 88
	±0.183 434 642 496	0.362 683 783 38
9	±0.968 160 239 508	0.081 274 388 36
	±0.836 031 107 327	0.180 648 160 69
	±0.613 371 432 701	0.260 610 696 40
	±0.324 253 423 404	0.312 347 077 04
	0.000 000 000 000	0.330 239 355 00
10	±0.973 906 528 517	0.066 671 343 99
	±0.865 063 366 689	0.149 451 349 64
	±0.679 409 568 299	0.219 086 362 24
	±0.433 395 394 129	0.269 266 719 47
	±0.148 874 338 982	0.295 524 224 66
11	±0.978 228 658 146	0.055 668 567 12
	±0.887 062 599 768	0.125 580 369 46
	±0.730 152 005 574	0.186 290 210 93
	±0.519 096 129 207	0.233 193 764 59
	±0.269 543 155 952	0.262 804 544 51
	0.000 000 000 000	0.272 925 086 78
12	±0.981 560 634 247	0.047 175 336 39
	±0.904 117 256 370	0.106 939 325 99
	±0.769 902 674 194	0.160 078 328 54
	±0.587 317 954 287	0.203 167 426 72
	±0.367 831 498 998	0.233 492 536 54
	±0.125 233 408 511	0.249 147 045 81
13	±0.984 183 054 719	0.040 484 004 77
	±0.917 598 399 223	0.092 121 499 84
	±0.801 578 090 733	0.138 873 510 22
	±0.642 349 339 440	0.178 145 980 76
	±0.448 492 751 036	0.207 816 047 54
	±0.230 458 315 955	0.226 283 180 26
	0.000 000 000 000	0.232 551 553 23

14	±0.986 283 808 697	0.035 119 460 33
	±0.928 434 883 664	0.080 158 087 16
	±0.827 201 315 070	0.121 518 570 69
	±0.687 292 904 812	0.157 203 167 16
	±0.515 248 636 358	0.185 538 397 48
	±0.319 112 368 928	0.205 198 463 72
	±0.108 054 948 707	0.215 263 853 46
15	±0.987 992 518 020	0.030 753 242 00
	±0.937 273 392 401	0.070 366 047 49
	±0.848 206 583 410	0.107 159 220 47
	±0.724 417 731 360	0.139 570 677 93
	±0.570 972 172 609	0.166 269 205 82
	±0.394 151 347 078	0.186 161 000 02
	±0.201 194 093 997	0.198 431 485 33
	0.000 000 000 000	0.202 578 241 92
16	±0.989 400 934 992	0.027 152 459 41
	±0.944 575 023 073	0.062 253 523 94
	±0.865 631 202 388	0.095 158 511 68
	±0.755 404 408 355	0.124 628 971 26
	±0.617 876 244 403	0.149 595 988 82
	±0.458 016 777 657	0.169 156 519 39
	±0.281 603 550 779	0.182 603 415 04
	±0.095 012 509 838	0.189 450 610 46
17	±0.990 575 475 315	0.024 148 302 87
	±0.950 675 521 769	0.055 459 529 38
	±0 880 239 153 727	0.085 036 148 32
	±0.781 514 003 897	0.111 883 847 19
	±0.657 671 159 217	0.135 136 368 47
	±0.512 690 537 086	0.154 045 761 08
	±0.351 231 763 454	0.168 004 102 16
	±0.178 484 181 496	0.176 562 705 37
	0.000 000 000 000	0.179 446 470 35

For interest, I draw graphs of the Legendre polynomials in figures I.7 and I.8.

Figure I.7. Legendre polynomials for even l

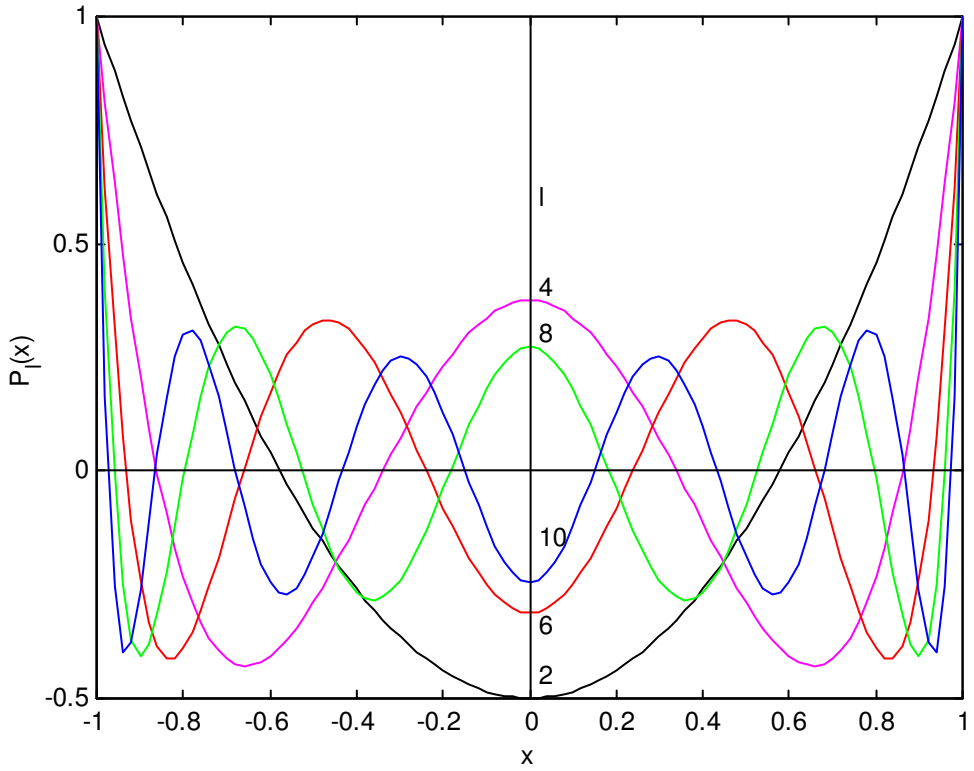
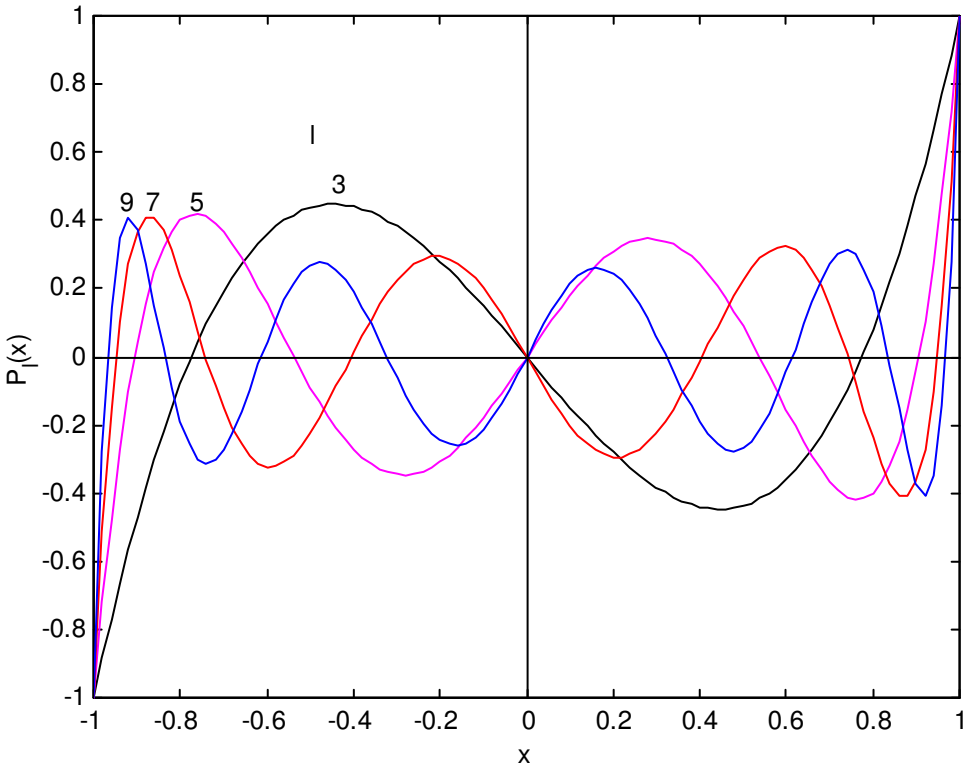


Figure I.8. Legendre polynomials for odd l



For further interest, it should be easy to verify, by substitution, that the Legendre polynomials are solutions of the differential equation

$$(1 - x^2)y'' - 2xy' + l(l+1)y = 0. \qquad 1.14.7$$

The Legendre polynomials are solutions of this and related equations that appear in the study of the vibrations of a solid sphere (spherical harmonics) and in the solution of the Schrödinger equation for hydrogen-like atoms, and they play a large role in quantum mechanics.

1.15 *Gaussian Quadrature – The Algorithm*

Gaussian quadrature is an alternative method of numerical integration which is often much faster and more spectacular than Simpson's rule. Gaussian quadrature allows you to carry out the integration

$$\int_{-1}^{1} f(x)dx. \qquad 1.15.1$$

But what happens if your limits of integration are not ±1? What if you want to integrate

$$\int_{a}^{b} F(t)dt\,? \qquad 1.15.2$$

That is no problem at all – you just make a change of variable. Thus, let

$$x = \frac{2t - a - b}{b - a}\ , \quad t = \tfrac{1}{2}[(b-a)x + a + b], \qquad 1.15.3$$

and the new limits are then $x = \pm 1$.

At the risk of being pedagogically unsound I'll describe first, without any theoretical development, just what you do, with an example – as long as you promise to look at the derivation afterwards, in Section 1.16.

For our example, let's try to evaluate

$$I = \int_{0}^{\pi/2} \sin\theta\, d\theta. \qquad 1.15.4$$

Let us make the change of variable given by equation 1.15.3 (with $t = \theta$, $a = 0$, $b = \pi/2$), and we now have to evaluate

$$I = \int_{-1}^{1} \tfrac{\pi}{4} \sin \tfrac{\pi}{4}(x+1)\, dx. \qquad 1.15.5$$

For a 5-point Gaussian quadrature, you evaluate the integrand at five values of x, where these five values of x are the solutions of $P_5(x) = 0$ given in Section 1.14, P_5 being the Legendre polynomial. That is, we evaluate the integrand at $x = \pm 0.906\ 469\ 514\ 203$, $\pm 0.538\ 469\ 310\ 106$ and 0.

I now assert, without derivation (until later), that

$$I = \sum_{i=1}^{5} c_{5,i}\, f(x_{5,i}), \qquad 1.15.6$$

where the coefficients $c_{l,i}$ (all positive) are listed with the roots of the Legendre polynomials in Section 1.14.

Let's try it.

$x_{5,i}$	$f(x_{5,i})$	$c_{5,i}$
+0.906 179 845 939	0.783 266 908 39	0.236 926 885 06
+0.538 469 310 106	0.734 361 739 69	0.478 628 670 50
0.000 000 000 000	0.555 360 367 27	0.568 888888 89
−0.538 469 310 006	0.278 501 544 60	0.478 628 670 50
−0.906 179 845 939	0.057 820 630 35	0.236 926 885 06

and the expression 1.15.6 comes to 1.000 000 000 04, and might presumably have come even closer to 1 had we given $x_{l,i}$ and $c_{l,i}$ to more significant figures.

You should now write a computer program for Gaussian quadrature – you will have to store the $x_{l,i}$ and $c_{l,i}$, of course. You have presumably already written a program for Simpson's rule.

In a text on integration, the author invited the reader to evaluate the following integrals by Gaussian quadrature:

(a) $\int_1^{1.5} x^2 \ln x\, dx$

(b) $\int_0^1 x^2 e^{-x}\, dx$

(c) $\int_0^{0.35} \frac{2}{x^2 - 4}\, dx$

(d) $\int_0^{\pi/4} x^2 \sin x\, dx$

(e) $\int_0^{\pi/4} e^{3x} \sin 2x\, dx$

(f) $\int_1^{1.6} \frac{2x}{x^2 - 4}\, dx$

(g) $\int_3^{3.5} \frac{x}{\sqrt{x^2 - 4}}\, dx$

(h) $\int_0^{\pi/4} \cos^2 x\, dx$

All of these can be integrated analytically, so I am going to invite the reader to evaluate them first analytically, and then numerically by Simpson's rule and again by Gaussian quadrature, and to see at how many points the integrand has to be evaluated by each method to achieve nine or ten figure precision. I tried, and the results are as follows. The first column is the answer, the second column is the number of points required by Simpson's rule, and the third column is the number of points required by Gaussian quadrature.

(*a*)	0.192 259 358	33	4
(*b*)	0.160 602 794	99	5
(*c*)	–0.176 820 020	19	4
(*d*)	0.088 755 284 4	111	5
(*e*)	2.588 628 633	453	7
(*f*)	–0.733 969 175	143	8
(*g*)	0.636 213 346	31	5
(*h*)	0.642 699 082	59	5

Let us now have a look at four of the integrals that we met in Section 1.2.

1. $\int_0^1 \frac{x^4 dx}{\sqrt{2(1+x^2)}}$. This was straightforward. It has an analytic solution of $\frac{\sqrt{18}\ln\left(1+\sqrt{2}\right)-2}{16}$ = 0.108 709 465. I needed to evaluate the integral at 89 points in order to get this answer to nine significant figures using Simpson's rule. To use Gaussian quadrature, we note that integrand contains only even powers of x and so it is symmetric about $x = 0$, and therefore the integral is equal to $\frac{1}{2}\int_{-1}^1 \frac{x^4 dx}{\sqrt{2(1+x^2)}}$, which makes it immediately convenient for Gaussian quadrature! I give below the answers I obtained for 3- to 7-point Gaussian quadrature.

3	0.108 667 036
4	0.108 711 215
5	0.108 709 441
6	0.108 709 463
7	0.108 709 465
Correct answer	0.108 709 465

2. $\int_0^2 \frac{y^2\,dy}{\sqrt{2-y}}$. This had the difficulty that the integrand is infinite at the upper limit. We got round this by means of the substitution $y = 2\sin^2\theta$, and the integral becomes $\sqrt{128}\int_0^{\pi/2}\sin^5\theta\,d\theta$. This has an analytic solution of $\sqrt{8192}/15 = 6.033977866$. I needed 59 points to get this answer to ten significant figures using Simpson's rule. To use Gaussian quadrature we can let $y = 1 + x$, so that the integral becomes $\int_{-1}^{1}\frac{(1+x)^2\,dy}{\sqrt{1-x}}$, which seems to be immediately suitable for Gaussian quadrature. Before we proceed, we recall that the integrand becomes infinite at the upper limit, and it still does so after our change of variable. We note, however, that with Gaussian quadrature, *we do not evaluate the integrand at the upper limit*, so that this would appear to be a great advantage of the method over Simpson's method. Alas! – this turns out not to be the case. If, for example, we use a 17-point quadrature, the largest value of x for which we evaluate the integrand is equal to the largest solution of $P_{17}(x) = 0$, which is 0.9906. We just cannot ignore the fact that the integrand shoots up to infinity beyond this, so we *have left behind a large part of the integral*. Indeed, with a 17-point Gaussian quadrature, I obtained an answer of 5.75, which is a long way from the correct answer of 6.03.

Therefore we have to make a change of variable, as we did for Simpson's method, so that the upper limit is finite. We chose $y = 2\sin^2\theta$, which changed the integral to $\sqrt{128}\int_0^{\pi/2}\sin^5\theta\,d\theta$. To make this suitable for Gaussian quadrature, we must now make the further substitution (see equation 1.15.3) $x = 4\theta/\pi - 1$, $\theta = \frac{\pi}{4}(x+1)$. If we wish to impress, we can make the two substitutions in one step, thus: Let $y = 2\sin^2\frac{\pi}{4}(1+x)$, $x = \frac{4}{\pi}\sin^{-1}\sqrt{\frac{y}{2}} - 1$. The integral becomes $\sqrt{8}\pi\int_{-1}^{1}\sin^5\frac{\pi}{4}(1+x)\,dx$, and there are no further difficulties. With a 9-point integration, I obtained the answer, correct to ten significant figures, 6.033 977 866. Simpson's rule required 59 points.

3. $\int_0^{\pi/2}\sqrt{\sec\theta}\,d\theta$. This integral occurs in the theory of a simple pendulum swinging through 90^o. As far as I can tell it has no simple analytical solution unless we have recourse to unfamiliar elliptic integrals, which we would have to evaluate numerically in any case. The integral has the difficulty that the integrand is infinite at the upper limit. We get round this by means of a substitution. Thus let $\sin\phi = \sqrt{2}\sin\frac{1}{2}\theta$. (Did you not think of this?) The integral becomes $\sqrt{2}\int_0^{\pi/2}\frac{d\phi}{\sqrt{1-\frac{1}{2}\sin^2\phi}}$. I needed 13 points by Simpson's rule to get the answer to ten significant figures, 2.622 057 554.

In order to make the limits ±1, suitable for Gaussian quadrature, we can make the second substitution (as in example 2), $\phi = \frac{\pi}{4}(x+1)$. If we wish truly to impress our friends, we can

make the two substitutions in one step, thus: Let $\sin\frac{\pi}{4}(1+x) = \sqrt{2}\sin\frac{1}{2}\theta$. (No one will ever guess how we thought of that!) The integral becomes $\frac{\pi}{2}\int_{-1}^{1}\frac{dx}{\sqrt{2-\sin^2\frac{\pi}{4}(x+1)}}$, which is now ready for Gaussian quadrature. I obtained the answer 2.622 057 554 in a 10-point Gaussian quadrature, which is only a little faster than the 13 points required by Simpson's rule.

4. $\int_0^\infty \frac{dy}{y^5\left(e^{1/y}-1\right)}$. This integral occurs in the theory of blackbody radiation. It has the difficulty of an infinite upper limit. We get round this by means of a substitution. Thus let $y = \tan\theta$. The integral becomes $\int_0^{\pi/2}\frac{c^3(c^2+1)}{e^c-1}d\theta$, where $c = \cot\theta$. It has an analytic solution of $\pi^4/15 = 6.493\ 939\ 402$. I needed 261 points by Simpson's rule to get the answer to ten significant figures. To prepare it for Gaussian quadrature, we can let $\theta = \frac{\pi}{4}(x+1)$, as we did in example 2, so that the integral becomes $\frac{\pi}{4}\int_{-1}^{1}\frac{c^3(c^2+1)}{e^c-1}dx$, where $c = \cot\frac{\pi}{4}(x+1)$. Using 16-point Gaussian quadrature, I got 6.48. Thus we would need to extend our table of constants for the Gaussian method to much higher order in order to use the method successfully. Doubtless the Gaussian method would then be faster than the Simpson method – but we do not need an extensive (and difficult-to-calculate) set of constants for the latter. A further small point: You may have noticed that it is not immediately obvious that the integrand is zero at the end points, and that some work is needed to prove it. But with the Gaussian method you don't evaluate the integrand at the end points, so that is one less thing to worry about!

Thus we have found that in most cases the Gaussian method is far faster than the Simpson method. In some cases it is only marginally faster. In yet others it probably would be faster than Simpson's rule, but higher-order constants are needed to apply it. Whether we use Simpson's rule or Gaussian quadrature, we have to carry out the integration with successively higher orders until going to higher orders results in no further change to the number of significant figures desired.

1.16 *Gaussian Quadrature - Derivation*

In order to understand why Gaussian quadrature works so well, we first need to understand some properties of polynomials in general, and of Legendre polynomials in particular. We also need to remind ourselves of the use of Lagrange polynomials for approximating an arbitrary function.

First, a statement concerning polynomials in general: Let *P* be a polynomial of degree *n*, and let *S* be a polynomial of degree less than 2*n*. Then, if we divide *S* by *P*, we obtain a quotient *Q* and a remainder *R*, each of which is a polynomial of degree less than *n*.

That is to say: $$\frac{S}{P} = Q + \frac{R}{P}. \qquad 1.16.1$$

What this means is best understood by looking at an example, with $n = 3$. For example,

let $$P = 5x^3 - 2x^2 + 3x + 7 \qquad 1.16.2$$

and $$S = 9x^5 + 4x^4 - 5x^3 + 6x^2 + 2x - 3. \qquad 1.16.3$$

If we carry out the division $S \div P$ by the ordinary process of long division, we obtain

$$\frac{9x^5 + 4x^4 - 5x^3 + 6x^2 + 2x - 3}{5x^3 - 2x^2 + 3x + 7} = 1.8x^2 + 1.52x - 1.472 - \frac{14.104x^2 + 4.224x - 7.304}{5x^3 - 2x^2 + 3x + 7}. \qquad 1.16.4$$

For example, if $x = 3$, this becomes

$$\frac{2433}{133} = 19.288 - \frac{132.304}{133}.$$

The theorem given by equation 1.16.1 is true for any polynomial P of degree l. In particular, it is true if P is the Legendre polynomial of degree l.

Next an important property of the Legendre polynomials, namely, if P_n and P_m are Legendre polynomials of degree n and m respectively, then

$$\int_{-1}^{1} P_n P_m dx = 0 \quad \text{unless } m = n. \qquad 1.16.5$$

This property is called the *orthogonal* property of the Legendre polynomials.

I give here a proof. Although it is straightforward, it may look formidable at first, so, on first reading, you might want to skip the proof and go on the next part (after the next short horizontal dividing line).

From the symmetry of the Legendre polynomials (see figure I.7), the following are obvious:

$$\int_{-1}^{1} P_n P_m dx \neq 0 \quad \text{if } m = n$$

and $$\int_{-1}^{1} P_n P_m dx = 0 \quad \text{if one (but not both) of } m \text{ or } n \text{ is odd.}$$

In fact we can go further, and, as we shall show,

$$\int_{-1}^{1} P_n P_m dx = 0 \quad \text{unless } m = n, \text{ whether } m \text{ and } n \text{ are even or odd.}$$

Thus P_m satisfies the differential equation (see equation 1.14.7)

$$(1 - x^2)\frac{d^2 P_m}{dx^2} - 2x\frac{dP_m}{dx} + m(m+1)P_m = 0, \tag{1.16.6}$$

which can also be written

$$\frac{d}{dx}\left[(1 - x^2)\frac{dP_m}{dx}\right] + m(m+1)P_m = 0. \tag{1.16.7}$$

Multiply by P_n:

$$P_n \frac{d}{dx}\left[(1 - x^2)\frac{dP_m}{dx}\right] + m(m+1)P_m P_n = 0, \tag{1.16.8}$$

which can also be written

$$\frac{d}{dx}\left[(1 - x^2)P_n \frac{dP_m}{dx}\right] - (1 - x^2)\frac{dP_n}{dx}\frac{dP_m}{dx} + m(m+1)P_m P_n = 0. \tag{1.16.9}$$

In a similar manner, we have

$$\frac{d}{dx}\left[(1 - x^2)P_m \frac{dP_n}{dx}\right] - (1 - x^2)\frac{dP_n}{dx}\frac{dP_m}{dx} + n(n+1)P_m P_n = 0. \tag{1.16.10}$$

Subtract one from the other:

$$\frac{d}{dx}\left[(1 - x^2)\left(P_n \frac{dP_m}{dx} - P_m \frac{dP_n}{dx}\right)\right] + [m(m+1) - n(n+1)]P_m P_n = 0. \tag{1.16.11}$$

Integrate from −1 to +1:

$$\left[(1 - x^2)\left(P_n \frac{dP_m}{dx} - P_m \frac{dP_n}{dx}\right)\right]_{-1}^{1} = [n(n+1) - m(m+1)]\int_{-1}^{1} P_m P_n dx. \tag{1.16.12}$$

The left hand side is zero because $1 - x^2$ is zero at both limits.

Therefore, unless $m = n$,

$$\int_{-1}^{1} P_m P_n dx = 0. \qquad \text{Q.E.D.} \qquad 1.16.13$$

I now assert that, if P_l is the Legendre polynomial of degree l, and if Q is any polynomial of degree less than l, then

$$\int_{-1}^{1} P_l Q dx = 0. \qquad 1.16.14$$

I shall first prove this, and then give an example, to see what it means.

To start the proof, we recall the recursion relation (see equation 1.14.4 – though here I am substituting $l - 1$ for l) for the Legendre polynomials:

$$lP_l = (2l - 1)xP_{l-1} - (l - 1)P_{l-2}. \qquad 1.16.15$$

The proof will be by induction.

Let Q be any polynomial of degree less than l. Multiply the above relation by Qdx and integrate from −1 to +1:

$$l\int_{-1}^{1} P_l Q dx = (2l - 1)\int_{-1}^{1} xP_{l-1} Q dx - (l - 1)\int_{-1}^{1} P_{l-2} Q dx. \qquad 1.16.16$$

If the right hand side is zero, then the left hand side is also zero.

A correspondent has suggested to me a much simpler proof. He points out that you could in principle expand Q in equation 1.16.14 as a sum of Legendre polynomials for which the highest degree is $l - 1$. Then, by virtue of equation 1.16.13, every term is zero.

For example, let $l = 4$, so that

$$P_{l-2} = P_2 = \tfrac{1}{2}(3x^2 - 1) \qquad 1.16.17$$

and

$$xP_{l-1} = xP_3 = \tfrac{1}{2}(5x^4 - 3x^2), \qquad 1.16.18$$

and let

$$Q = 2(a_3 x^3 + a_2 x^2 + a_1 x + a_0). \qquad 1.16.19$$

It is then straightforward (and only slightly tedious) to show that

$$\int_{-1}^{1} P_{l-2} Q dx = \left(\tfrac{6}{5} - \tfrac{2}{3}\right) a_2 \qquad 1.16.20$$

and that

$$\int_{-1}^{1} x P_{l-1} Q dx = \left(\tfrac{10}{7} - \tfrac{6}{5}\right) a_2. \qquad 1.16.21$$

But $$7\left(\tfrac{10}{7} - \tfrac{6}{5}\right) a_2 - 3\left(\tfrac{6}{5} - \tfrac{2}{3}\right) a_2 = 0, \qquad 1.16.22$$

and therefore $$\int_{-1}^{1} P_4 Q dx = 0. \qquad 1.16.23$$

We have shown that

$$l \int_{-1}^{1} P_l Q dx = (2l-1) \int_{-1}^{1} x P_{l-1} Q dx - (l-1) \int_{-1}^{1} P_{l-2} Q dx = 0 \qquad 1.16.24$$

for $l = 4$, and therefore it is true for all positive integral l.

You can use this property for a parlour trick. For example, you can say: "Think of any polynomial. Don't tell me what it is – just tell me its degree. Then multiply it by (here give a Legendre polynomial of degree more than this). Now integrate it from −1 to +1. The answer is zero, right?" (Applause.)

Thus: Think of any polynomial. $3x^2 - 5x + 7$. Now multiply it by $5x^3 - 3x$. OK, that's $15x^5 - 25x^4 - 2x^3 + 15x^2 - 21x$. Now integrate it from −1 to +1. The answer is zero.

Now, let S be any polynomial of degree less than $2l$. Let us divide it by the Legendre polynomial of degree l, P_l, to obtain the quotient Q and a remainder R, both of degree less than l. Then I assert that

$$\int_{-1}^{1} S dx = \int_{-1}^{1} R dx. \qquad 1.16.25$$

This follows trivially from equations 1.16.1 and 1.16.14. Thus

$$\int_{-1}^{1} S dx = \int_{-1}^{1} (Q P_l + R) dx = \int_{-1}^{1} R dx. \qquad 1.16.26$$

Example: Let $S = 6x^5 - 12x^4 + 4x^3 + 7x^2 - 5x + 7$. The integral of this from -1 to $+1$ is $13.8\dot{6}$. If we divide S by $\frac{1}{2}(5x^3 - 3x)$, we obtain a quotient of $2.4x^2 - 4.8x + 3.04$ and a remainder of $-0.2x^2 - 0.44x + 7$. The integral of the latter from -1 to $+1$ is also $13.8\dot{6}$.

I have just described some properties of Legendre polynomials. Before getting on to the rationale behind Gaussian quadrature, let us remind ourselves from Section 1.11 about Lagrange polynomials. We recall from that section that, if we have a set of n points, the following function:

$$y = \sum_{i=1}^{n} y_i L_i(x) \qquad 1.16.27$$

(in which the n functions $L_i(x)$, $i = 1, n$, are Lagrange polynomials of degree $n-1$) is the polynomial of degree $n-1$ that passes exactly through the n points. Also, if we have some function $f(x)$ which we evaluate at n points, then the polynomial

$$y = \sum_{i=1}^{n} f(x_i) L_i(x) \qquad 1.16.28$$

is a jolly good approximation to $f(x)$ and indeed may be used to interpolate between nontabulated points, even if the function is tabulated at irregular intervals. In particular, if $f(x)$ is a polynomial of degree $n - 1$, then the expression 1.16.28 is an exact representation of $f(x)$.

We are now ready to start talking about quadrature. We wish to approximate $\int_{-1}^{1} f(x)dx$ by an n-term finite series

$$\int_{-1}^{1} f(x)dx \approx \sum_{i=1}^{n} c_i f(x_i), \qquad 1.16.29$$

where $-1 < x_i < 1$. To this end, we can approximate $f(x)$ by the right hand side of equation 1.16.28, so that

$$\int_{-1}^{1} f(x)dx \approx \int_{-1}^{1} \sum_{i=1}^{n} f(x_i) L_i(x)\, dx = f(x_i) \int_{-1}^{1} \sum_{i=1}^{n} L_i(x)\, dx. \qquad 1.16.30$$

Recall that the Lagrange polynomials in this expression are of degree $n - 1$.

The required coefficients for equation 1.16.29 are therefore

$$c_i = \int_{-1}^{1} L_i(x)\,dx. \qquad 1.16.31$$

Note that at this stage the values of the x_i have not yet been chosen; they are merely restricted to the interval [−1 , 1].

Now let's consider $\int_{-1}^{1} S(x)\,dx$, where S is a polynomial of degree less than $2n$, such as, for example, the polynomial of equation 1.16.3. We can write

$$\int_{-1}^{1} S(x)\,dx = \int_{-1}^{1} \sum_{i=1}^{n} S(x_i)L_i(x)\,dx = \int_{-1}^{1} \sum_{i=1}^{n} L_i(x)[Q(x_i)P(x_i) + R(x_i)]\,dx. \qquad 1.16.32$$

Here, as before, P is a polynomial of degree n, and Q and R are of degree less than n.

If we now choose the x_i to be the roots of the Legendre polynomials, then

$$\int_{-1}^{1} S(x)\,dx = \int_{-1}^{1} \sum_{i=1}^{n} L_i(x)R(x_i)\,dx. \qquad 1.16.33$$

Note that the integrand on the right hand side of equation 1.16.33 is an *exact representation of* $R(x)$. But we have already shown (equation 1.16.26) that $\int_{-1}^{1} S(x)dx = \int_{-1}^{1} R(x)dx$, and therefore

$$\int_{-1}^{1} S(x)dx = \int_{-1}^{1} R(x)dx = \sum_{i=1}^{n} c_i R(x_i) = \sum_{i=1}^{n} c_i S(x_i). \qquad 1.16.34$$

It follows that the Gaussian quadrature method, if we choose the roots of the Legendre polynomials for the n abscissas, will yield exact results for any polynomial of degree less than $2n$, and will yield a good approximation to the integral if $S(x)$ is a polynomial representation of a general function $f(x)$ obtained by fitting a polynomial to several points on the function.

1.17 *Frequently-needed Numerical Procedures*

Many years ago I gradually became aware that there were certain mathematical equations and procedures that I found myself using over and over again. I therefore set aside some time to write short computer programs for dealing with each of them, so that whenever in the future I needed, for example, to evaluate a determinant, I had a program already written to do it. I show here a partial list of the programs I have for instant use by myself whenever needed. I would suggest that the reader might consider compiling for him- or herself a similar collection of small programs. I have found over the years that they have saved me an immense amount of time and

effort. Most programs are very short and required only a few minutes to write (although this depends, of course, on how much programming experience one has), though a few required a bit more effort. Some programs are so short – consisting of a few lines only - that they might be thought to be too trivial to be worth writing. These include, for example, programs for solving a quadratic equation or for solving two simultaneous linear equations. Yet I have perhaps used these particularly simple ones more than any others, and they have been of use out of all proportion to the almost negligible effort required to write them. Here, then, is a partial list, and I do suggest that the reader will be repaid enormously over the years if he takes a short time to write similar programs. Of course many or even most of them are readily available in pre-packaged programs. But there are enormous advantages in writing your own programs. Quite apart from the extra programming practice that they provide, you know exactly what your own programs do, you can tailor them exactly to your own requirements, you know their strengths and their weaknesses or limitations, and you don't have to struggle for hours over an instruction manual trying to understand how to use them, only to find in the end that they don't do exactly what you want.

Solve quadratic equation
Solve cubic equation
Solve quintic equation
Solve $f(x) = 0$ by Newton-Raphson
Solve $f(x , y) = 0$, $g(x , y) = 0$ by Newton-Raphson
Tabulate $y = f(x)$
Tabulate $y = f(x , a)$
Fit least-squares straight line to data
Fit least-squares cubic equation to data
Solve two simultaneous linear equations
Solve three simultaneous linear equations
Solve four simultaneous linear equations
Solve N (>4) simultaneous linear equations in two, three or four unknowns by least squares
Multiply column vector by square matrix
Invert matrix
Diagonalize matrix
Find eigenvectors and eigenvalues of matrix
Test matrix for orthogonality
Evaluate determinant
Convert between rectangular and polar coordinates
Convert between rectangular and spherical coordinates
Convert between direction cosines and Euler angles
Fit a conic section to five points
Numerical integration by Simpson's rule
Gaussian quadrature
Given any three elements of a plane triangle, calculate the remaining elements
Given any three elements of a spherical triangle, calculate the remaining elements

In addition to these common procedures, there are many others that I have written and have readily to hand that are of more specialized use tailored to my own particular interests, such as

Solve Kepler's equation
Convert between wavelength and wavenumber
Calculate *LS*-coupling line strengths
Convert between relativity factors such as $\gamma = 1/\sqrt{1 - \beta^2}$

Likewise, you will be able to think of many formulas special to your own interests that you use over and over again, and it would be worth your while to write short little programs for them.