Lecture Notes in

Numerical Analysis

prepared by

Yvette Fajardo-Lim, Ph.D.
Mathematics Department
De La Salle University
Manila

Chapter 1

Solutions of Nonlinear Equations

1.1 Introduction

The numerical methods discussed in this chapter are used to find approximations to solutions of nonlinear equations when the exact solutions cannot be obtained by algebraic methods. We will discuss one of the most basic problems in numerical analysis, the **root-finding problem**. This problem consists of finding values of x on a closed interval $[a, b]$ that satisfy the given equation $f(x) = 0$, for a continuous function f. A solution to this problem is called a **zero** of f or a **root** of $f(x) = 0$.

1.2 Bracketing Methods

1.2.1 Bisection Method

This method is based on the Intermediate Value Theorem.

Theorem 1.1. (Intermediate Value Theorem) *Let f be continuous over a closed interval $[a, b]$ and suppose $f(a) \neq f(b)$. Then for all k between $f(a)$ and $f(b)$, there exists $c \in (a, b)$ for which $f(c) = k$.*

Suppose a continuous function f defined on the interval $[a, b]$, is given, with $f(a)$ and $f(b)$ of opposite sign. Then by theorem 1.1, there exists s, $a < s < b$ for which $f(s) = 0$. Although the procedure will work for the case when $f(a)$ and $f(b)$ have opposite signs and there is more than one root in the interval $[a, b]$, it will be assumed for simplicity that the root in this interval is unique. The method calls for a repeated halving of subintervals of $[a, b]$ and at each step, locating the"half" containing s. To begin, set $a_1 = a$ and $b_1 = b$, and let x_1 be the midpoint of $[a, b]$; that is, $x_1 = \frac{1}{2}(a_1 + b_1)$. If $f(x_1) = 0$ then $s = x_1$; if not, then $f(x_1)$ has the same sign as either $f(a_1)$ or $f(b_1)$. If $f(x_1)$ and $f(a_1)$ have the same sign then $s \in (x_1, b_1)$ and we set $a_2 = x_1$ and $b_2 = b_1$. If $f(x_1)$ and $f(b_1)$ have the same sign then $s \in (a_1, x_1)$ and we set $a_2 = a_1$ and $b_2 = x_1$.

Now we reapply the process to the interval $[a_2, b_2]$. The process is repeated until a good approximation is obtained.

Theorem 1.2. *Let f be a continuous function on the interval $[a, b]$ and suppose $f(a) \cdot f(b) < 0$. The bisection method generates a sequence $\{x_n\}$ of approximations converging to the root s of f on $[a, b]$.*

Proof. For each $n \geq 1$, we have $b_n - a_n = \dfrac{1}{2^{n-1}}(b - a)$ and $s \in (a_n, b_n)$. Since $x_n = \frac{1}{2}(a_n + b_n)$, for all $n \geq 1$, it follows that

$$|x_n - s| \leq \frac{1}{2}(b_n - a_n) = \frac{b - a}{2^n}.$$

Hence,

$$\lim_{n\to\infty} |x_n - s| \leq (b - a) \lim_{n\to\infty} \frac{1}{2^n} = 0$$

□

Example 1.1. *Find an estimate for the root of the equation $x = 2^{-x}$ on $[0, 1]$ using 15 iterations of the bisection method.*

Solution. Let $f(x) = x - 2^{-x}$. Since $f(0) \cdot f(1) < 0$ then f has a root in $[0, 1]$. The following table lists the results of the bisection method using 15 iterations. Approximately the root of f is 0.6412048340.

a	b	x_n	$f(a)$	$f(b)$	$f(x_n)$
0	1	0.5	-1	0.5	-0.207106781
0.5	1	0.75	-0.207106781	0.5	0.155396442
0.5	0.75	0.625	-0.207106781	0.155396442	-0.023419777
0.625	0.75	0.6875	-0.023419777	0.155396442	0.066571094
0.625	0.6875	0.65625	-0.023419777	0.066571094	0.021724521
0.625	0.65625	0.640625	-0.023419777	0.021724521	-0.000810008
0.640625	0.65625	0.6484375	-0.000810008	0.021724521	0.010466611
0.640625	0.6484375	0.64453125	-0.000810008	0.010466611	0.004830646
0.640625	0.64453125	0.642578125	-0.000810008	0.004830646	0.002010906
0.640625	0.642578125	0.641601563	-0.000810008	0.002010906	0.000600596
0.640625	0.6416015625	0.6411132813	-0.000810008	0.000600596	-0.000104669
0.6411132813	0.6416015625	0.6413574219	-0.000104669	0.000600596	0.000247972
0.6411132813	0.6413574219	0.6412353516	-0.000104669	0.000247972	7.16538E-05
0.6411132813	0.6412353516	0.6411743164	-0.000104669	7.16538E-05	-1.65072E-05
0.6411743164	0.6412353516	0.6412048340	-1.65072E-05	7.16538E-05	2.75735E-05

1.2.2 Regula Falsi Method

This method is similar to the bisection method in the sense that intervals $[a_i, b_i]$ are generated bracketing a root. Assuming that the interval $[a_i, b_i]$ contains a root of $f(x) = 0$, compute the value of the x-intercept of the line joining the points $(a_i, f(a_i))$ and

$(b_i, f(b_i))$ and label this point x_i. If $f(x_i)f(a_i) < 0$ define $a_{i+1} = a_i$ and $b_{i+1} = x_i$ otherwise define $a_{i+1} = x_i$ and $b_{i+1} = b_i$.

The slope of the line joining the points $(a_i, f(a_i))$ and $(b_i, f(b_i))$ is given by $\dfrac{f(b_i) - f(a_i)}{b_i - a_i}$. Since the point $(x_i, 0)$ is on this line, we have

$$\begin{aligned} \frac{f(b_i) - f(a_i)}{b_i - a_i} &= \frac{f(b_i) - 0}{b_i - x_i} \\ \implies b_i - x_i &= \frac{(b_i - a_i)f(b_i)}{f(b_i) - f(a_i)} \\ \implies x_i &= b_i - \frac{(b_i - a_i)f(b_i)}{f(b_i) - f(a_i)} \\ &= \frac{a_i f(b_i) - b_i f(a_i)}{f(b_i) - f(a_i)} \end{aligned}$$

Example 1.2. *Solve $x = 2^{-x}$ on $[0, 1]$ using regula-falsi method.*

Solution. The following table lists the results of the regula-falsi method. Approximately the root of f is 0.641185744504986.

a	b	x_n	$f(a)$	$f(b)$	$f(x_n)$
0	1	0.666666666666667	-1	0.5	0.036706142
0	0.666666666666667	0.643062329659873	-1	0.036706142	0.002710065
0	0.643062329659873	0.641324299037687	-1	0.002710065	0.00020013
0	0.641324299037687	0.641195976351816	-1	0.00020013	1.47792E-05
0	0.641195976351816	0.641186500107318	-1	1.47792E-05	1.09142E-06
0	0.641186500107318	0.641185800304831	-1	1.09142E-06	8.05993E-08
0	0.641185800304831	0.641185748625702	-1	8.05993E-08	5.95211E-09
0	0.641185748625702	0.641185744809293	-1	5.95211E-09	4.39553E-10
0	0.641185744809293	0.641185744527458	-1	4.39553E-10	3.246E-11
0	0.641185744527458	0.641185744506646	-1	3.246E-11	2.39719E-12
0	0.641185744506646	0.641185744505109	-1	2.39719E-12	1.77081E-13
0	0.641185744505109	0.641185744504995	-1	1.77081E-13	1.29896E-14
0	0.641185744504995	0.641185744504987	-1	1.29896E-14	1.11022E-15
0	0.641185744504987	0.641185744504986	-1	1.11022E-15	0

1.3 Fixed Point Methods

1.3.1 Fixed-Point Iteration

Here, we consider methods for determining the solution to an equation that is expressed, for some function g in the form $g(x) = x$. A solution to such an equation is said to be a **fixed point** of the function g.

If a fixed point could be found for any given g, then every root-finding problem could also be solved. For example, the root-finding problem $f(x) = 0$ has solutions that

correspond precisely to the fixed points of $g(x) = x$ when $g(x) = x - f(x)$. The first task, then, is to decide when a function will have a fixed point and how the fixed points can be determined. Before we discuss this, we first recall the Mean Value Theorem.

Theorem 1.3. (Mean Value Theorem) *Let f be continuous over a closed interval $[a, b]$ and suppose f is differentiable (a, b). Then a number $c, a < c < b$ exists such that*

$$f'(c) = \frac{f(b) - f(a)}{b - a}$$

The following theorem gives sufficient conditions for the existence and uniqueness of a fixed point.

Theorem 1.4. *If $g(x)$ is a continuous function on $[a, b]$ and $g(x) \in [a, b]$ for all $x \in [a, b]$, then $g(x)$ has a fixed point in $[a, b]$. Further, suppose $g'(x)$ exists on (a, b) and $|g'(x)| \leq L < 1$ for all $x \in (a, b)$. Then $g(x)$ has a unique fixed point s in $[a, b]$.*

Proof. If $g(a) = a$ or $g(b) = b$, the existence of a fixed point is obvious. Suppose not, then $g(a) > a$ and $g(b) < b$. Define $h(x) = g(x) - x$; h is continuous on $[a, b]$, and, moreover,

$$h(a) = g(a) - a > 0, h(b) = g(b) - b < 0.$$

The Intermediate Value Theorem implies that there exists $s \in (a, b)$ for which $h(s) = 0$. Thus, $g(s) - s = 0$ and s is a fixed point of g.

Suppose in addition that $|g'(x)| \leq L < 1$ holds and that s and r are both fixed points in $[a, b]$ with $s \neq r$. By the Mean Value Theorem, a number c exists between s and r, and hence in $[a, b]$, with

$$|s - r| = |g(s) - g(r)| = |g'(c)||s - r| \leq L|s - r| < |s - r|,$$

which is a contradiction. Hence $s = r$ and the fixed point in $[a, b]$ is unique. □

We will recall the Extreme Value Theorem which will be used to illustrate the preceding theorem.

Theorem 1.5. (Extreme Value Theorem) *Let f be continuous over a closed interval $[a, b]$ then a number $c_1, c_2 \in [a, b]$ exist with $f(c_1) \leq f(x) \leq f(c_2)$ for each $x \in [a, b]$. If, in addition, f is differentiable on (a, b), then the numbers c_1 and c_2 occur either at endpoints of $[a, b]$ or where f' is zero.*

Example 1.3. Let $g(x) = \dfrac{x^2 - 1}{3}$ on $[-1, 1]$. Using the Extreme Value Theorem, it is easy to show that the absolute minimum of g occurs at $x = 0$ and is $g(0) = -\frac{1}{3}$. Similarly, the absolute maximum of g occurs at $x = \pm 1$ and has the value $g(\pm 1) = 0$. Moreover, g is continuous and

$$|g'(x)| = \left|\frac{2x}{3}\right| \leq \frac{2}{3} \forall x \in [-1, 1],$$

so g satisfies the hypotheses of theorem 1.4 and has a unique fixed point in $[-1, 1]$.

In this example, the unique fixed point s in the interval $[-1, 1]$ can be determined exactly. If

$$s = g(s) = \frac{s^2 - 1}{3}, \Rightarrow \ s^2 - 3s - 1 = 0,$$

which by quadratic formula, implies that

$$s = \frac{3 - \sqrt{13}}{2}.$$

Note that g also has a unique fixed point $s = \dfrac{3 + \sqrt{13}}{2}$ for the interval $[3, 4]$. However, $g(4) = 5$ and $g'(4) = \frac{8}{3} > 1$; so g does not satisfy the hypotheses of theorem 1.4. This shows that the hypotheses of theorem 1.4 are sufficient to guarantee a unique fixed point, but are not necessary.

Example 1.4. Let $g(x) = 3^{-x}$. Since $g'(x) = -3^{-x} \ln 3 < 0$ on $[0, 1]$, the function g is decreasing on $[0, 1]$. Hence $g(1) = \frac{1}{3} \leq g(x) \leq 1 = g(0)$. Thus, for $x \in [0, 1]$, $g(x) \in [0, 1]$. Therefore, g has a fixed point in $[0, 1]$. Since,

$$g'(0) = -\ln 3 = -1.098612289,$$

$|g'(x)| \not\leq 1$ on $[0, 1]$ and theorem 1.4 cannot be used to determine uniqueness. However, g is decreasing so it is clear that the fixed point must be unique.

To approximate the fixed point of a function g, we choose an initial approximation $s_0 = \dfrac{a + b}{2}$ and generate the sequence $\{s_n\}_{n=0}^{\infty}$ by letting $s_n = g(s_{n-1})$ for each $n \geq 1$. If the sequence converges to s and g is continuous, then

$$s = \lim_{n\to\infty} s_n = \lim_{n\to\infty} g(s_{n-1}) = g\left(\lim_{n\to\infty} s_{n-1}\right) = g(s),$$

and a solution to $x = g(x)$ is obtained. This method is called the **fixed-point iterative technique**.

Example 1.5. The equation $x^3 + 4x^2 - 10 = 0$ has a unique root in $[1, 2]$. There are many ways to change the equation to the form $x = g(x)$ by simple algebraic manipulation. For example, to obtain the function g described in (3) we could manipulate the equation $x^3 + 4x^2 - 10 = 0$ as follows:

$$\begin{aligned} 4x^2 &= 10 - x^3 \\ \Longrightarrow x^2 &= \frac{1}{4}(10 - x^3) \\ \Longrightarrow x &= \pm\frac{1}{2}\sqrt{10 - x^3} \end{aligned}$$

To obtain a positive solution $g_3(x)$ is chosen as shown. It is not important to derive these functions, but it should be verified that the fixed point of each is actually a solution to the general equation.

1. $x = g_1(x) = x - x^3 - 4x^2 + 10,$
2. $x = g_2(x) = \sqrt{\dfrac{10}{x} - 4x}$
3. $x = g_3(x) = \dfrac{1}{2}\sqrt{10 - x^3}$
4. $x = g_4(x) = \sqrt{\dfrac{10}{4 + x}}$
5. $x = g_5(x) = x - \dfrac{x^3 + 4x^2 - 10}{3x^2 + 8x}$

The table on the next page lists the results of the fixed-point iteration method for all five choices of g.

(1)	(2)	(3)	(4)	(5)
1.5	1.5	1.5	1.5	1.5
-0.875	0.816496581	1.28695376762338	1.34839972492648	1.37333333333333
6.732421875	2.996908806	1.40254080353958	1.36737637199128	1.36526201487463
-469.720012		1.34545837402329	1.36495701540249	1.36523001391615
102754555.2		1.37517025281604	1.36526474811344	1.36523001341410
-1.08493E+24		1.36009419276173	1.36522559416052	1.36523001341410
1.27706E+72		1.36784696759213	1.36523057567343	
-2.0827E+216		1.36388700388402	1.36522994187818	
		1.36591673339004	1.36523002251557	
		1.36487821719368	1.36523001225612	
		1.36541006116996	1.36523001356143	
		1.36513782066921	1.36523001339535	
		1.36527720852448	1.36523001341648	
		1.36520585029705	1.36523001341379	
		1.36524238371884	1.36523001341414	
		1.36522368022528	1.36523001341409	
		1.36523325574250	1.36523001341410	
		1.36522835346263	1.36523001341410	
		1.36523086324364		
		1.36522957833396		
		1.36523023615818		
		1.36522989937773		
		1.36523007179629		
		1.36522998352467		
		1.36523002871632		
		1.36523000557995		
		1.36523001742488		
		1.36523001136073		
		1.36523001446534		
		1.36523001287590		
		1.36523001368963		
		1.36523001327303		
		1.36523001348632		
		1.36523001337712		
		1.36523001343303		
		1.36523001340441		
		1.36523001341906		
		1.36523001341156		
		1.36523001341540		
		1.36523001341343		
		1.36523001341444		
		1.36523001341392		
		1.36523001341419		
		1.36523001341405		
		1.36523001341412		
		1.36523001341408		
		1.36523001341410		
		1.36523001341409		
		1.36523001341410		
		1.36523001341410		

1.3.2 Newton-Raphson Method

This method is one of the most powerful and well-known numerical methods for solving a root-finding problem $f(x) = 0$. Suppose that $f'(x)$ exists on $[a, b]$ and that $f'(x) \neq 0$ on $[a, b]$. Further, suppose there exists one $s \in [a, b]$ such that $f(s) = 0$. Let $x_0 \in [a, b]$ be arbitrary. Let x_1 be the point at which the tangent line to f at $(x_0, f(x_0))$ crosses the x-axis. For each $n \geq 1$, let x_n be the x-intercept of the line tangent to f at $(x_{n-1}, f(x_{n-1}))$. Hence, the slope of the tangent line at x_{n-1} is

$$\begin{aligned} f'(x_{n-1}) &= \frac{f(x_{n-1}) - 0}{x_{n-1} - x_n} \\ \implies x_{n-1} - x_n &= \frac{f(x_{n-1})}{f'(x_{n-1})} \\ \implies x_n &= x_{n-1} - \frac{f(x_{n-1})}{f'(x_{n-1})} \end{aligned}$$

Example 1.6. The equation $x^3 + 4x^2 - 10 = 0$ has a unique root in $[1, 2]$. Approximate the root using the Newton-Raphson method.

Solution. Given $f(x) = x^3 + 4x^2 - 10$ then $f'(x) = 3x^2 + 8x$, the following table lists the results with the initial value of $x_n = \frac{1}{2}(a + b)$. Approximately, the root of f is 1.365230013414100.

x_n	$f(x_n)$	$f'(x_n)$
1.5	2.375	18.75
1.373333333333330	0.1343454814814820	16.6448000000000000
1.365262014874630	0.0005284611795151	16.5139172267756000
1.365230013916150	0.0000000082905487	16.5133990840216000
1.365230013414100	0	

1.3.3 Secant Method

Newton's Method is an extremely powerful technique, but it has a major difficulty: the need to know the value of the derivative of f at each approximation. Frequently, $f'(x)$ is far more difficult and needs more arithmetic operations to calculate than $f(x)$. As a simple example, let $f(x) = x^2 3^x \cos 2x$, then $f'(x) = 2x3^x \cos 2x + x^2 3^x (\cos 2x) \ln 3 - 2x^2 3^x \sin 2x$.

To circumvent the problem of the derivative evaluation in Newton's Method, we derive a slight variation. By definition,

$$f'(x_{n-1}) = \lim_{x \to x_{n-1}} \frac{f(x) - f(x_{n-1})}{x - x_{n-1}}.$$

Letting $x = x_{n-2}$,

$$f'(x_{n-1}) \approx \frac{f(x_{n-2}) - f(x_{n-1})}{x_{n-2} - x_{n-1}} = \frac{f(x_{n-1}) - f(x_{n-2})}{x_{n-1} - x_{n-2}}.$$

Using this approximation for $f'(x_{n-1}$ in Newton's Formula gives

$$\begin{aligned} x_n &= x_{n-1} - \frac{f(x_{n-1})(x_{n-2} - x_{n-1})}{f(x_{n-1}) - f(x_{n-2})} \\ \Longrightarrow x_n &= \frac{x_{n-2}f(x_{n-1}) - x_{n-1}f(x_{n-2})}{f(x_{n-1}) - f(x_{n-2})} \end{aligned}$$

The technique using this formula is called the **Secant method**. We will use the initial values $x_1 = a$ and $x_2 = b$.

Example 1.7. Approximate the root of the preceding example using the Secant method.

Solution. The following table lists the results with the initial values given as $x_1 = 1$ and $x_2 = 2$. The root of f is the same as the previous one.

x_n	$f(x_n)$
1	-5
2	14
1.263157894736840	-1.602274384020990
1.338827838827840	-0.430364748004529
1.366616394719350	0.022909430775949
1.365211902631860	-0.000299067919329
1.365230001110860	-0.000000203168273
1.365230013414210	0.000000000001805
1.36523001341410	0

Exercises.

1. Show that $f(x) = x^3 - x - 1$ has exactly one root in the interval $[1, 2]$. Approximate the root using the bisection method.

2. Consider $f(x) = \tan(x)$ on the interval $[0, 3]$. Use the 20 iterations of the bisection method and see what happens. Explain the results that you obtained.

3. Use the bisection method to find solutions for the following problems:

   (a) $4^x + 5^x = 100 \quad$ for $1 \le x \le 3$

   (b) $e^x + 2^{-x} + 2\cos x - 6 = 0 \quad$ for $1 \le x \le 2$

   (c) $e^x - x^2 + 3x - 2 = 0 \quad$ for $0 \le x \le 1$

4. Repeat Exercise 2 using the Regula-Falsi method.

5. (a) Show that each of the following functions has a fixed point at s precisely when $f(s) = 0$, where $f(x) = x^4 + 2x^2 - x - 3$.

      i. $g_1(x) == \sqrt[4]{3 + x - 2x^2}$

      ii. $g_2(x) = \sqrt{\dfrac{x + 3 - x^4}{2}}$

      iii. $g_3(x) = \sqrt{\dfrac{x+3}{x^2+2}}$

      iv. $g_4(x) = \dfrac{3x^4 + 2x^2 + 3}{4x^3 + 4x - 1}$

   (b) Use the fixed-point iteration method on each of the functions g defined in (a). Let $x_0 = 1$.

   (c) Which function do you think gives the best approximation to the solution?

6. Solve $x^3 - x - 1 = 0$ for the root in $[1, 2]$, using fixed-point iteration.

7. Use Newton's method to approximate the solutions of the following equations in the given intervals.

   (a) $x^3 - 2x^2 - 5 = 0, \quad [1, 4]$

   (b) $x^3 + 3x^2 - 1 = 0, \quad [-4, 0]$

   (c) $x - \cos x = 0, \quad [0, \pi/2]$

   (d) $x - 0.8 - 0.2\sin x = 0, \quad [0, \pi/2]$

8. Repeat Exercise 6 using the Secant method.

Chapter 2

Systems of Linear Equations

2.1 Basic Concepts in Linear Algebra

Definition 2.1. *An n **by** m **matrix** $A = [a_{ij}]$ is a rectangular array of **elements** a_{ij} arranged in n rows and m columns. i is the row location of a_{ij} and j is the column location of a_{ij}.*

$$\mathbf{A} = [a_{ij}] = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix}$$

An n by $(n+1)$ matrix can be used to represent the linear system

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n &= b_2 \\ \vdots \qquad\quad \vdots \qquad\qquad \vdots \qquad\quad & \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n &= b_n, \end{aligned}$$

by first constructing

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} \qquad \text{and} \qquad \mathbf{B} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

and then combining these matrices to form the **augmented matrix**

$$[\mathbf{A}|\mathbf{B}] = \left[\begin{array}{cccc|c} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ \vdots & \vdots & & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} & b_n \end{array}\right]$$

where the line is used to separate the coefficients of the unknowns from the values on the right hand side of the equations.

To solve a linear system, three operations are permitted on the equations:

1. An equation can be multiplied by a nonzero scalar.
2. An equation can be multiplied by a nonzero scalar and add the result to another equation.
3. Any two equations can be interchanged in the system.

Considering the augmented matrix associated with the linear system, the following operations can be performed on $[\mathbf{A}|\mathbf{B}]$:

1. A row can be multiplied by a nonzero scalar.
2. A row can be multiplied by a nonzero scalar and add the result to another row.
3. Any two rows can be interchanged in the matrix.

2.2 Gauss-Jordan Method

This method used to solve the linear system

$$\begin{array}{rcl} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n & = & b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n & = & b_2 \\ \vdots \qquad\quad \vdots \qquad\qquad \vdots & & \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n & = & b_n, \end{array}$$

is handled by performing a sequence of operations on the augmented matrix $[\mathbf{A}|\mathbf{B}]$ reducing this to:

$$\left[\begin{array}{cccc|c} 1 & 0 & \dots & 0 & b_1^{(1)} \\ 0 & 1 & \dots & 0 & b_2^{(2)} \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \dots & 1 & b_n^{(n)} \end{array}\right],$$

the solution is obtained by setting $x_i = b_i^{(i)}$ for each $i = 1, 2, \ldots, n$.

Example 2.1. Use Gauss-Jordan Method to solve the linear system

$$\begin{aligned} x_1 - x_2 + 2x_3 - x_4 &= -8 \\ 2x_1 - 2x_2 + 3x_3 - 3x_4 &= -20 \\ x_1 + x_2 + x_3 &= -2 \\ x_1 - x_2 + 4x_3 + 3x_4 &= 4 \end{aligned}$$

Solution. The augmented matrix is:

$$\left[\begin{array}{cccc|c} 1 & -1 & 2 & -1 & -8 \\ 2 & -2 & 3 & -3 & -20 \\ 1 & 1 & 1 & 0 & -2 \\ 1 & -1 & 4 & 3 & 4 \end{array}\right],$$

and performing the operations

$$(-2R_1 + R_2) \to (R_2), (-R_1 + R_3) \to (R_3), \text{ and } (-R_1 + R_4) \to (R_4)$$

we write:

$$\left[\begin{array}{cccc|c} 1 & -1 & 2 & -1 & -8 \\ 0 & 0 & -1 & -1 & -4 \\ 0 & 2 & -1 & 1 & 6 \\ 0 & 0 & 2 & 4 & 12 \end{array}\right],$$

Interchanging row 2 and row 3, we have

$$\left[\begin{array}{cccc|c} 1 & -1 & 2 & -1 & -8 \\ 0 & 2 & -1 & 1 & 6 \\ 0 & 0 & -1 & -1 & -4 \\ 0 & 0 & 2 & 4 & 12 \end{array}\right],$$

Performing the operation $(\frac{1}{2}R_2) \to (R_2)$ gives

$$\left[\begin{array}{cccc|c} 1 & -1 & 2 & -1 & -8 \\ 0 & 1 & -\frac{1}{2} & \frac{1}{2} & 3 \\ 0 & 0 & -1 & -1 & -4 \\ 0 & 0 & 2 & 4 & 12 \end{array}\right],$$

and $(R_2 + R_1) \to (R_1)$ gives

$$\left[\begin{array}{cccc|c} 1 & 0 & \frac{3}{2} & -\frac{1}{2} & -5 \\ 0 & 1 & -\frac{1}{2} & \frac{1}{2} & 3 \\ 0 & 0 & -1 & -1 & -4 \\ 0 & 0 & 2 & 4 & 12 \end{array}\right].$$

Next, $(-R_3) \to (R_3)$ gives

$$\left[\begin{array}{cccc|c} 1 & 0 & \frac{3}{2} & -\frac{1}{2} & -5 \\ 0 & 1 & -\frac{1}{2} & \frac{1}{2} & 3 \\ 0 & 0 & 1 & 1 & 4 \\ 0 & 0 & 2 & 4 & 12 \end{array}\right],$$

and $(-\frac{3}{2}R_3 + R_1) \to (R_1), (\frac{1}{2}R_3 + R_2) \to (R_2),$ and $(-2R_3 + R_4) \to (R_4)$ gives

$$\left[\begin{array}{cccc|c} 1 & 0 & 0 & -2 & -11 \\ 0 & 1 & 0 & 1 & 5 \\ 0 & 0 & 1 & 1 & 4 \\ 0 & 0 & 0 & 2 & 4 \end{array}\right].$$

Finally, $(-\frac{1}{2}R_4) \to (R_4)$ gives

$$\left[\begin{array}{cccc|c} 1 & 0 & 0 & -2 & -11 \\ 0 & 1 & 0 & 1 & 5 \\ 0 & 0 & 1 & 1 & 4 \\ 0 & 0 & 0 & 1 & 4 \end{array}\right],$$

and $(2R_4 + R_1) \to (R_1), (-R_4 + R_2) \to (R_2),$ and $(-R_4 + R_3) \to (R_3)$ gives

$$\left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & -7 \\ 0 & 1 & 0 & 0 & 3 \\ 0 & 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 1 & 2 \end{array}\right].$$

The solution to the linear system is therefore $x_1 = -7, x_2 = 3, x_3 = 2$ and $x_4 = 2$.

2.3 LU Decomposition Method

Given a linear system $\mathbf{AX} = \mathbf{B}$, and suppose the coefficient matrix $\mathbf{A}$ can be written as a product $\mathbf{A} = \mathbf{LU}$ where $\mathbf{L}$ is a lower triangular matrix and $\mathbf{U}$ is an upper triangular matrix. Then the system can be written as $\mathbf{L}(\mathbf{UX}) = \mathbf{B}$. If we let $\mathbf{Y} = \mathbf{UX}$ then we can solve the original system $\mathbf{AX} = \mathbf{B}$ in two parts as follows:

1. Solve the system $\mathbf{LY} = \mathbf{B}$
2. Solve the system $\mathbf{UX} = \mathbf{Y}$

Thus, the problem is reduced to finding $\mathbf{L}$ and $\mathbf{U}$ such that $\mathbf{A} = \mathbf{LU}$.

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} = \begin{bmatrix} l_{11} & 0 & \dots & 0 \\ l_{21} & l_{22} & \dots & 0 \\ \vdots & \vdots & & \vdots \\ l_{n1} & l_{n2} & \dots & l_{nn} \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & \dots & u_{1n} \\ 0 & u_{22} & \dots & u_{2n} \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & u_{nn} \end{bmatrix}$$

There are n^2 entries in A, $\frac{n^2+n}{2}$ unknowns in $\mathbf{L}$ and $\frac{n^2+n}{2}$ unknowns in $\mathbf{U}$. Hence, there exists n free variables. According to the choice of the free variables, we have the following **LU Decomposition Methods**:

1. Doolittles Method where we set $l_{ii} = 1, i = 1, 2, \ldots, n$;
2. Crouts Method where we set $u_{ii} = 1, i = 1, 2, \ldots, n$; and
3. Choleski's Method where we set $u_{ii} = l_{ii}, i = 1, 2, \ldots, n$.

Example 2.2. Use Doolittes Method to solve the linear system

$$\begin{aligned} x_1 + x_2 + 3x_4 &= 4 \\ 2x_1 + x_2 - x_3 + x_4 &= 1 \\ 3x_1 - x_2 - x_3 + 2x_4 &= -3 \\ -x_1 + 2x_2 + 3x_3 - x_4 &= 4 \end{aligned}$$

Solution.

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 0 & 3 \\ 2 & 1 & -1 & 1 \\ 3 & -1 & -1 & 2 \\ -1 & 2 & 3 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ l_{21} & 1 & 0 & 0 \\ l_{31} & l_{32} & 1 & 0 \\ l_{41} & l_{42} & l_{43} & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ 0 & u_{22} & u_{23} & u_{24} \\ 0 & 0 & u_{33} & u_{34} \\ 0 & 0 & 0 & u_{44} \end{bmatrix}$$

where

$u_{11} = a_{11},\ u_{12} = a_{12},\ u_{13} = a_{13},\ u_{14} = a_{14},$
$l_{21}u_{11} = a_{21},\ l_{21}u_{12} + u_{22} = a_{22},\ l_{21}u_{13} + u_{23} = a_{23},\ l_{21}u_{14} + u_{24} = a_{24},$
$l_{31}u_{11} = a_{31},\ l_{31}u_{12}+l_{32}u_{22} = a_{32},\ l_{31}u_{13}+l_{32}u_{23}+u_{33} = a_{33},\ l_{31}u_{14}+l_{32}u_{24}+u_{34} = a_{34},$
$l_{41}u_{11} = a_{41},\ l_{41}u_{12} + l_{42}u_{22} = a_{42},\ l_{41}u_{13} + l_{42}u_{23} + l_{43}u_{33} = a_{43}$, and
$l_{41}u_{14} + l_{42}u_{24} + l_{43}u_{34} + u_{34} = a_{44}$.

Then we have the factorization

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 4 & 1 & 0 \\ -1 & -3 & 0 & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{U} = \begin{bmatrix} 1 & 1 & 0 & 3 \\ 0 & -1 & -1 & -5 \\ 0 & 0 & 3 & 13 \\ 0 & 0 & 0 & -13 \end{bmatrix}.$$

We now solve $\mathbf{LY} = \mathbf{B}$ for $\mathbf{Y}$ using the Gauss-Jordan method.

$$\left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 4 \\ 2 & 1 & 0 & 0 & 1 \\ 3 & 4 & 1 & 0 & -3 \\ -1 & -3 & 0 & 1 & 4 \end{array}\right] \quad \text{will be reduced to} \quad \left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 4 \\ 0 & 1 & 0 & 0 & -7 \\ 0 & 0 & 1 & 0 & 13 \\ 0 & 0 & 0 & 1 & -13 \end{array}\right].$$

Hence, $y_1 = 4, y_2 = -7, y_3 = 13$ and $y_4 = -13$. We then solve $\mathbf{UX} = \mathbf{Y}$ for $\mathbf{X}$ using the same method.

$$\left[\begin{array}{cccc|c} 1 & 1 & 0 & 3 & 4 \\ 0 & -1 & -1 & -5 & -7 \\ 0 & 0 & 3 & 13 & 13 \\ 0 & 0 & 0 & -13 & -13 \end{array}\right] \quad \text{will be reduced to} \quad \left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{array}\right].$$

We obtain $x_1 = -1, x_2 = 2, x_3 = 0$ and $x_4 = 1$.

2.4 Iterative Techniques for Solving Linear Systems

2.4.1 Gauss-Jacobi Method

Given a linear system $\mathbf{AX} = \mathbf{B}$ of n equations and n unknowns. We can solve for the ith variable x_i in the ith equation as follows:

$$\begin{aligned} x_1 &= -\frac{1}{a_{11}}(a_{12}x_2 + a_{13}x_3 + \ldots + a_{1n}x_n) + \frac{b_1}{a_{11}} \\ x_2 &= -\frac{1}{a_{22}}(a_{21}x_1 + a_{23}x_3 + \ldots + a_{2n}x_n) + \frac{b_2}{a_{22}} \\ &\vdots \\ x_n &= -\frac{1}{a_{nn}}(a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn-1}x_{n-1}) + \frac{b_n}{a_{nn}} \end{aligned}$$

This system can be written in the form $\mathbf{X} = \mathbf{CX} + \mathbf{D}$ where

$$\mathbf{C} = \begin{bmatrix} 0 & -\frac{a_{12}}{a_{11}} & -\frac{a_{13}}{a_{11}} & \ldots & -\frac{a_{1n}}{a_{11}} \\ -\frac{a_{21}}{a_{22}} & 0 & -\frac{a_{23}}{a_{22}} & \ldots & -\frac{a_{2n}}{a_{22}} \\ \vdots & \vdots & \vdots & & \vdots \\ -\frac{a_{n1}}{a_{nn}} & -\frac{a_{n2}}{a_{nn}} & -\frac{a_{n3}}{a_{nn}} & \ldots & 0 \end{bmatrix} \quad \text{and} \quad \mathbf{D} = \begin{bmatrix} \frac{b_1}{a_{11}} \\ \frac{b_2}{a_{22}} \\ \vdots \\ \frac{b_n}{a_{nn}} \end{bmatrix}$$

If we start with an arbitrary initial estimate $\mathbf{X}^{(0)} = \begin{bmatrix} x_1^{(0)} \\ x_2^{(0)} \\ \vdots \\ x_n^{(0)} \end{bmatrix}$ for the solution $\mathbf{X}$, we can use the iteration $\mathbf{X}^{(k)} = \mathbf{CX^{(k)}} + D$. In $\mathbf{X}^{(k)}$, the ith coordinate is given by

$$x_i^{(k)} = \frac{-\sum_{i \neq j} a_{ij} x_j^{(k-1)} + b_i}{a_{ii}}$$

Example 2.3. Using $\mathbf{X}^{(0)} = \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}$, use the Gauss-Jacobi Method to solve the linear system

$$\begin{aligned} 4x_1 - x_2 + x_3 &= 7 \\ 4x_1 - 8x_2 + x_3 &= -21 \\ -2x_1 + x_2 + 5x_3 &= 15 \end{aligned}$$

Solution. The table below lists the results of the Gauss-Jacobi method where $x_1 = 2, x_2 = 4$ and $x_3 = 3$.

$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$
1	1	2
1.5	3.375	3.2
1.79375	3.775	2.925
1.9625	3.8875	2.9625
1.98125	3.9765625	3.0075
1.992265625	3.9915625	2.9971875
1.99859375	3.99578125	2.99859375
1.999296875	3.99912109375	3.00028125
1.9997099609375	3.99968359375	2.99989453125
1.999947265625	3.999841796875	2.999947265625
1.99997363281250	3.999967041015620	3.000010546875
1.999989123535160	3.999988134765620	2.999996044921870
1.999998022460940	3.999994067382810	2.999998022460940
1.999999011230470	3.999998764038090	3.000000395507810
1.999999592132570	3.999999555053710	2.999999851684570
1.999999925842290	3.999999777526860	2.999999925842290
1.999999962921140	3.999999953651430	3.000000014831540
1.999999984704970	3.999999983314510	2.999999994438170
1.999999997219090	3.999999991657260	2.999999997219090
1.999999998609540	3.999999998261930	3.000000000556180
1.999999999426440	3.999999999374290	2.999999999791430
1.999999999895720	3.999999999687150	2.999999999895720
1.999999999947860	3.999999999934820	3.000000000020860
1.999999999978490	3.999999999976540	2.999999999992180
1.999999999996090	3.999999999988270	2.999999999996090
1.999999999998040	3.999999999997560	3.000000000000780
1.999999999999190	3.999999999999120	2.999999999999710
1.999999999999850	3.999999999999560	2.999999999999850
1.999999999999930	3.999999999999910	3.000000000000030
1.999999999999970	3.999999999999970	2.999999999999990
1.999999999999990	3.999999999999980	2.999999999999990
2	4	3
2	4	3

2.4.2 Gauss-Seidel Method

From the Gauss-Jacobi Method,

$$x_i^{(k)} = \frac{-\sum_{i \neq j} a_{ij} x_j^{(k-1)} + b_i}{a_{ii}}.$$

When computing $x_i^{(k)}$ we already know the values of $x_1^{(k)}, x_2^{(k)}, \ldots, x_{i-1}^{(k)}$. Since $\mathbf{X}^{(k)}$ is assumed to be closer to the true solution $\mathbf{X}$ than the preceding estimate $\mathbf{X}^{(k-1)}$, we can use the values $x_1^{(k)}, x_2^{(k)}, \ldots, x_{i-1}^{(k)}$ instead of $x_1^{(k-1)}, x_2^{(k-1)}, \ldots, x_{i-1}^{(k-1)}$ in the formula for $x_i^{(k)}$.

Thus, our formula for $x_i^{(k)}$ becomes

$$x_i^{(k)} = \frac{-\sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)} + b_i}{a_{ii}}.$$

which is the Gauss-Seidel iteration formula.

Example 2.4. Solve the linear system in the preceding example using the Gauss-Seidel iteration formula.

Solution. The table below lists the results of the Gauss-Seidel method where $x_1 = 2, x_2 = 4$ and $x_3 = 3$.

$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$
1	1	2
1.5	3.625	2.875
1.9375	3.953125	2.984375
1.99218750	3.9941406250	2.998046875
1.9990234375	3.999267578125	2.999755859375
1.99987792968750	3.999908447265620	2.999969482421870
1.999984741210930	3.999988555908200	2.999996185302730
1.999998092651360	3.999998569488520	2.999999523162840
1.999999761581420	3.999999821186060	2.999999940395350
1.999999970197670	3.999999977648250	2.999999992549410
1.999999996274700	3.999999997206030	2.999999999068670
1.999999999534330	3.999999999650750	2.999999999883580
1.999999999941790	3.999999999956340	2.999999999985450
1.999999999992720	3.999999999994540	2.999999999998180
1.999999999999090	3.999999999999320	2.999999999999770
1.999999999999890	3.999999999999910	2.999999999999970
1.999999999999990	3.999999999999990	3.000000000000000
2	4	3
2	4	3

The results of the previous two examples seem to imply that the Gauss-Seidel method is superior to the Gauss-Jacobi method. This is generally the case but is not always true. In fact, there are linear systems for which the Gauss-Jacobi method converges and the Gauss-Seidel method does not, and others for which the Gauss-Seidel method converges and the Gauss-Jacobi method does not.

In general, both the iterative techniques presented produces a sequence of estimates which converges to the solution of the linear system if the coefficient matrix is diagonally dominant. This means that the magnitude of each diagonal entry a_{ii} is greater than the sum of the magnitudes of the other entries on row i as shown on the following theorem:

Theorem 2.1. *Let* $\mathbf{AX} = \mathbf{B}$ *be a linear system of* n *equations in* n *unknowns. If* $\mathbf{A}$ *is diagonally dominant, then for any choice of the initial estimate* $\mathbf{X}^{(0)}$*, the sequence of estimates* $\{\mathbf{X}^{(k)}\}$ *generated by either the Gauss-Jacobi or the Gauss-Seidel methods will converge to the solution* $\mathbf{X}$ *of the linear system.*

If the coefficient matrix is **not** diagonally dominant, it can be made so by rearranging the equations in the system. For example, the coefficient matrix of the linear system

$$\begin{aligned} 2x + \ y - 5z &= \ 7 \\ x + 5y + 2z &= -1 \\ 3x - \ y + \ z &= \ 2 \end{aligned}$$

is not diagonally dominant, so we may rearrange the equations in the system into

$$\begin{aligned} 3x - \ y + \ z &= \ 2 \\ x + 5y + 2z &= -1 \\ 2x + \ y - 5z &= \ 7 \end{aligned}$$

which now has a diagonally dominant coefficient matrix.

If it is really impossible to find a rearrangement of the equations in the system in order o make the coefficient diagonally dominant, then we just try to look for the arrangement which would maximize the number of rows with dominant diagonal entries.

Exercises.

1. Solve the following linear systems using the Gauss-Jordan Method. Do not reorder the equations. (The exact solution to each system is $x_1 = 1, x_2 = -1, x_3 = 3$).

   a.) $\begin{aligned} 4x_1 + x_2 - x_3 &= 8 \\ 2x_1 + 5x_2 + 2x_3 &= 3 \\ x_1 + 2x_2 + 4x_3 &= 11 \end{aligned}$

   b.) $\begin{aligned} x_1 + 2x_2 + 4x_3 &= 11 \\ 4x_1 + x_2 - x_3 &= 8 \\ 2x_1 + 5x_2 + 2x_3 &= 3 \end{aligned}$

   c.) $\begin{aligned} 4x_1 + x_2 + 2x_3 &= 9 \\ 2x_1 + 4x_2 - x_3 &= -5 \\ x_1 + x_2 - 3x_3 &= -9 \end{aligned}$

   d.) $\begin{aligned} 2x_1 + 4x_2 - x_3 &= -5 \\ x_1 + x_2 - 3x_3 &= -9 \\ 4x_1 + x_2 + 2x_3 &= 9 \end{aligned}$

2. Repeat Exercise 1 using the LU Decomposition Method.

3. Repeat Exercise 1 using the Gauss-Jacobi Method.

4. Repeat Exercise 1 using the Gauss-Seidel Method.

5. Solve the following linear systems using the Gauss-Jordan Method.

   a.) $\begin{aligned} \tfrac{1}{4}x_1 + \tfrac{1}{5}x_2 + \tfrac{1}{6}x_3 &= 9 \\ \tfrac{1}{3}x_1 + \tfrac{1}{4}x_2 + \tfrac{1}{5}x_3 &= 8 \\ \tfrac{1}{2}x_1 + x_2 + 2x_3 &= 8 \end{aligned}$

   b.) $\begin{aligned} 3.333x_1 + 15920x_2 - 10.333x_3 &= 15913 \\ 2.222x_1 + 16.71x_2 + 9.612x_3 &= 28.544 \\ 1.5611x_1 + 5.1791x_2 + 1.6852x_3 &= 8.4254 \end{aligned}$

   c.) $\begin{aligned} 4.01x_1 + 1.23x_2 + 1.43x_3 - 0.73x_4 &= 5.94 \\ 1.23x_1 + 7.41x_2 + 2.41x_3 + 3.02x_4 &= 14.07 \\ 1.43x_1 + 2.41x_2 + 5.79x_3 - 1.11x_4 &= 8.52 \\ -0.73x_1 + 3.02x_2 - 1.11x_3 + 6.41x_4 &= 7.59 \end{aligned}$

   d.) $\begin{aligned} x_1 + \tfrac{1}{2}x_2 + \tfrac{1}{3}x_3 + \tfrac{1}{4}x_4 &= \tfrac{1}{6} \\ \tfrac{1}{2}x_1 + \tfrac{1}{3}x_2 + \tfrac{1}{4}x_3 + \tfrac{1}{5}x_4 &= \tfrac{1}{7} \\ \tfrac{1}{3}x_1 + \tfrac{1}{4}x_2 + \tfrac{1}{5}x_3 + \tfrac{1}{6}x_4 &= \tfrac{1}{8} \\ \tfrac{1}{4}x_1 + \tfrac{1}{5}x_2 + \tfrac{1}{6}x_3 + \tfrac{1}{7}x_4 &= \tfrac{1}{9} \end{aligned}$

6. Solve Exercise 5 using the Gauss-Jacobi and Gauss-Seidel methods whenever applicable.

Chapter 3

Interpolating Polynomials

One of the most useful and well-known classes of functions mapping the set of real numbers into itself is the class of **algebraic polynomials**, the set of functions of the form

$$P_n(x) = a_0 + a_1(x) + \ldots + a_n x^n,$$

where n is a nonnegative integer and $a_0, \ldots, a_n$ are real constants. One major reason for their importance is that they uniformly approximate continuous functions; that is, given any function, defined and continuous on a closed interval, there exists a polynomial that is as 'close' to the given function as desired. This result is expressed more precisely in the following theorem.

Theorem 3.1. (Weierstrass Approximation Theorem) *If f is defined and continuous on $[a, b]$ and $\epsilon > 0$ is given, then there exists a polynomial P, defined on $[a, b]$, with the property that*

$$|f(x) - P(x)| < \epsilon \quad \forall x \in [a, b].$$

Other important reasons for considering the class of polynomials in the approximation of functions are that the derivative and indefinite integral of any polynomial are easy to determine and the result is again a polynomial. For these reasons, the class of polynomials is often used for approximating other functions that are known or assumed to be continuous.

Definition 3.1. *Let f be a function defined on an interval $[a, b]$ and let $x_0, x_1, \ldots x_n$ be $n + 1$ points on the interval where $x_0 < x_1 < \ldots < x_n$. A polynomial $P(x)$ is an interpolating polynomial for f on $[a, b]$ if $f(x_j) = P(x_j)$, $j = 0, 1, \ldots, n$. The points x_j are called **interpolation points** or **nodes**.*

3.1 Lagrange Form of the Interpolating Polynomial

The following theorem defines this form of the interpolating polynomial.

Theorem 3.2. *Let $x_0 < x_1 < \ldots < x_n$ be $(n+1)$ distinct points on $[a, b]$ and let f be a function defined on $[a, b]$. There exists a unique polynomial P of degree at most n such that $f(x_j) = P(x_j)\ \forall\ j = 0, 1, \ldots, n$. This polynomial is given by*

$$P(x) = \sum_{j=0}^{n} f(x_j) l_j(x)$$

where

$$\begin{aligned} l_j(x) &= \frac{(x - x_0)(x - x_1)\ldots(x - x_{j-1})(x - x_{j+1})\ldots(x - x_n)}{(x_j - x_0)(x_j - x_1)\ldots(x_j - x_{j-1})(x_j - x_{j+1})\ldots(x_j - x_n)} \\ &= \prod_{k \neq j} \frac{(x - x_k)}{(x_j - x_k)} \end{aligned}$$

Proof.

$$\begin{aligned} P(x) &= \sum_{j=0}^{n} f(x_j) l_j(x) \\ &= \sum_{j \neq i} f(x_j) l_j(x) + f(x_i) l_i(x) \end{aligned}$$

Hence,

$$P(x_i) = \sum_{j \neq i} f(x_j) l_j(x_i) + f(x_i) l_i(x_i)$$

We will consider two cases: $j \neq i$ and when $j = i$.

Case 1. $j \neq i$

$$\begin{aligned} l_j(x_i) &= \frac{(x_i - x_0)(x_i - x_1)\ldots(x_i - x_{j-1})(x_i - x_{j+1})\ldots(x_i - x_n)}{(x_j - x_0)(x_j - x_1)\ldots(x_j - x_{j-1})(x_j - x_{j+1})\ldots(x_j - x_n)} \\ &= 0 \end{aligned}$$

Case 2. $j = i$

$$\begin{aligned} l_i(x_i) &= \frac{(x_i - x_0)(x_i - x_1)\ldots(x_i - x_{i-1})(x_i - x_{i+1})\ldots(x_i - x_n)}{(x_i - x_0)(x_i - x_1)\ldots(x_i - x_{i-1})(x_i - x_{i+1})\ldots(x_i - x_n)} \\ &= 1 \end{aligned}$$

It follows that

$$P(x_i) = f(x_i)\ \forall\ i.$$

Furthermore, let $P(x)$ and $Q(x)$ be both interpolating polynomials of f where $P(x) \neq Q(x)$. Then $P(x_i) = Q(x_i) = f(x_i)$ for all the interpolation points x_i. Define $R(x) = P(x) - Q(x)$. This gives $R(x_i) = P(x_i) - Q(x_i) = 0\ \forall\ x_i$. This implies that $R(x)$ has $(n + 1)$ roots which is a contradiction. Hence, $P(x) = Q(x)$. This proves that the interpolating polynomial is unique. □

Example 3.1. Using the interpolation points or nodes, $x_0 = 2, x_1 = 2.5$, and $x_2 = 4$ to find the second-degree interpolating polynomial for $f(x) = \frac{1}{x}$ requires that we first determine the coefficient polynomial l_0, l_1 and l_2:

$$\begin{aligned} l_0(x) &= \frac{(x-2.5)(x-4)}{(2-2.5)(2-4)} = x^2 - 6.5x + 10 \\ l_1(x) &= \frac{(x-2)(x-4)}{(2.5-2)(2.5-4)} = \frac{1}{3}(-4x^2 + 24x - 32) \\ l_2(x) &= \frac{(x-2)(x-2.5)}{(4-2)(4-2.5)} = \frac{1}{3}(x^2 - 4.5x + 5) \end{aligned}$$

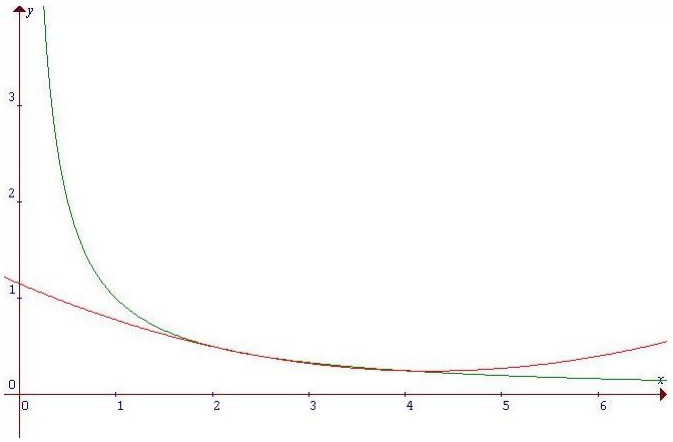
Since $f(x_0) = f(2) = 0.5, f(x_1) = f(2.5) = 0.4$, and $f(x_2) = f(4) = 0.25$,

$$\begin{aligned} P(x) &= \sum_{j=0}^{2} f(x_j) l_j(x) \\ &= 0.5(x^2 - 6.5x + 10) + \frac{0.4}{3}(-4x^2 + 24x - 32) + \frac{0.25}{3}(x^2 - 4.5x + 5) \\ &= 0.05x^2 - 0.425x + 1.15 \end{aligned}$$

An approximation to $f(3) = \frac{1}{3}$ is

$$f(3) \approx P(3) = 0.325.$$

The figure below shows that $P(x)$ could reasonably approximate $f(3)$ on [2,4].



3.2 Method of Undetermined Coefficients

Given a set of $(n+1)$ points. The interpolating polynomial takes the form:

$$P_n(x) = a_0 + a_1x + a_2x^2 + \ldots + a_nx^n.$$

Since $f(x_j) = P(x_j)\ \forall\ j = 0, 1, \ldots, n$ we have

$$\begin{aligned} a_0 + a_1x_0 + a_2x_0^2 + \ldots + a_nx_0^n &= f(x_0) \\ a_0 + a_1x_1 + a_2x_1^2 + \ldots + a_nx_1^n &= f(x_1) \\ &\vdots \\ a_0 + a_1x_n + a_2x_n^2 + \ldots + a_nx_n^n &= f(x_n) \end{aligned}$$

with the augmented matrix

$$\left[\begin{array}{ccccc|c} 1 & x_0 & x_0^2 & \ldots & x_0^n & f(x_0) \\ 1 & x_1 & x_1^2 & \ldots & x_1^n & f(x_1) \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 1 & x_n & x_n^2 & \ldots & x_n^n & f(x_n) \end{array}\right].$$

By the Gauss-Jordan method, $a_0, a_1, \ldots, a_n$ can be solved.

Example 3.2. Find the interpolating polynomial for the function f which passes through the points $(-1, 8), (0, 5), (1, 2)$ and $(2, 5)$

Solution. The augmented matrix is:

$$\left[\begin{array}{cccc|c} 1 & -1 & 1 & -1 & 8 \\ 1 & 0 & 0 & 0 & 5 \\ 1 & 1 & 1 & 1 & 2 \\ 1 & 2 & 4 & 8 & 5 \end{array}\right].$$

Interchanging row 1 and row 2,

$$\left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 5 \\ 1 & -1 & 1 & -1 & 8 \\ 1 & 1 & 1 & 1 & 2 \\ 1 & 2 & 4 & 8 & 5 \end{array}\right].$$

Performing the operations

$$(R_1 + R_2) \to (R_2), (-R_1 + R_3) \to (R_3), \text{ and } (-R_1 + R_4) \to (R_4)$$

we write:

$$\left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 5 \\ 0 & -1 & 1 & -1 & 3 \\ 0 & 1 & 1 & 1 & -3 \\ 0 & 2 & 4 & 8 & 0 \end{array}\right].$$

Multiplying row 2 by -1,

$$\left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 5 \\ 0 & 1 & -1 & 1 & -3 \\ 0 & 1 & 1 & 1 & -3 \\ 0 & 2 & 4 & 8 & 0 \end{array}\right].$$

Performing the operation $(R_2 + R_3) \rightarrow (R_3),$ and $(2R_1 + R_4) \rightarrow (R_4)$ gives

$$\left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 5 \\ 0 & 1 & -1 & 1 & -3 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 6 & 6 & 6 \end{array}\right].$$

Next, $(\frac{1}{2}R_3) \rightarrow (R_3)$ gives

$$\left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 5 \\ 0 & 1 & -1 & 1 & -3 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 6 & 6 & 6 \end{array}\right],$$

and $(R_3 + R_2) \rightarrow (R_2),$ and $(-6R_3 + R_4) \rightarrow (R_4)$ gives

$$\left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 5 \\ 0 & 1 & 0 & 1 & -3 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{array}\right].$$

Finally, $(-R_4 + R_2) \rightarrow (R_2)$ gives

$$\left[\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 5 \\ 0 & 1 & 0 & 0 & -4 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{array}\right].$$

The solution is therefore $a_0 = 5, a_1 = -4, a_2 = 0$ and $a_3 = 1$. Hence, the interpolating polynomial is $P(x) = 5 - 4x + x^3$.

3.3 Newton's Interpolatory Divided-Difference Formula

Suppose that P_n is the polynomial of degree at most n that agrees with the function f at the distinct numbers $x_0, x_1, \ldots, x_n$. The divided differences of f with respect to $x_0, x_1, \ldots, x_n$ can be derived by showing that P_n has the representation

$$P_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \ldots + a_n(x - x_0)(x - x_1)\ldots(x - x_{n-1})$$

for appropriate constants $a_0, a_1, \ldots, a_n$.

To determine the first of these constants, a_0, note that if $P_n(x)$ can be written in the form above, then evaluating P_n at x_0 leaves only the constant term a_0; that is $a_0 = P_n(x_0) = f(x_0)$.

Similarly, when P_n is evaluated at x_1, the only nonzero terms in the evaluation of $P_n(x_1)$ are the constant and linear terms,

$$f(x_0) + a_1(x_1 - x_0) = P_n(x_1) = f(x_1);$$

so

$$a_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0}.$$

At this stage we introduce what is known as the **divided difference notation**. The zeroth divided difference of the function f, with respect to x_i, is denoted by $f[x_i]$ and is simply the evaluation of f at x_i,

$$f[x_i] = f(x_i).$$

The remaining divided differences are defined intuitively; the first divided difference of f with respect to x_i and x_{i+1} is denoted by $f[x_i, x_{i+1}]$ and defined as

$$f[x_i, x_{i+1}] = \frac{f[x_{i+1}] - f[x_i]}{x_{i+1} - x_i}.$$

When the $(k-1)$st divided differences

$$f[x_i, x_{i+1}, x_{i+2}, \ldots, x_{i+k-1}] \quad \textit{and} \quad f[x_{i+1}, x_{i+2}, \ldots, x_{i+k-1}, x_{i+k}]$$

have been both determined, the kth divided difference relative to $x_i, x_{i+1}, x_{i+2}, \ldots, x_{i+k}$ is given by

$$f[x_i, x_{i+1}, \ldots, x_{i+k-1}, x_{i+k}] = \frac{f[x_{i+1}, x_{i+2}, \ldots, x_{i+k}] - f[x_i, x_{i+1}, \ldots, x_{i+k-1}]}{x_{i+k} - x_i}$$

With this notation, $a_1 = f[x_0, x_1]$ and the interpolating polynomial is

$$P_n(x) = f[x_0] + f[x_0, x_1](x - x_0) + a_2(x - x_0)(x - x_1) + \ldots + a_n(x - x_0)(x - x_1)\ldots(x - x_{n-1})$$

The constants $a_2, a_3, \ldots, a_n$ in P_n can be consecutively obtained in manner similar to the evaluation of a_0 and a_1. As might be expected, the required constants are:

$$a_k = f[x_0, x_1, x_2, \ldots, x_k],$$

for each $k = 0, 1, \ldots, n$ so P_n can be rewritten as

$$\begin{aligned} P_n(x) &= f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1) + \ldots \\ &+ f[x_0, x_1, \ldots, x_n](x - x_0)(x - x_1)\ldots(x - x_{n-1}), \end{aligned}$$

which is known as **Newton's interpolatory divided-difference formula**.

The determination of the divided differences from tabulated data points is outlined below.

x_i	$f[x_i]$	$f[x_i, x_{i+1}]$	$f[x_i, x_{i+1}, x_{i+2}]$	$\ldots$	$f[x_i, x_{i+1}, \ldots, x_{i+n}]$
x_0	$f[x_0]$				
x_1	$f[x_1]$	$f[x_0, x_1]$			
x_2	$f[x_2]$	$f[x_1, x_2]$	$f[x_0, x_1, x_2]$		
$\vdots$	$\vdots$	$\vdots$	$\vdots$		
x_n	$f[x_n]$	$f[x_{n-1}, x_n]$	$f[x_{n-2}, x_{n-1}, x_n]$	$\ldots$	$f[x_0, x_1, x_2, \ldots, x_n]$

Example 3.3. The following table lists values of a function at various points. Find the interpolating polynomial using the Newton's interpolatory divided-difference formula.

Solution. Given the divided difference table below, the coefficients of the interpolatory polynomial are along the diagonal in the table. The polynomial is

$$\begin{aligned} P_4(x) &= 0.7651977 - 0.4837057(x - 1.0) - 0.1087339(x - 1.0)(x - 1.3) \\ &+ 0.0658784(x - 1.0)(x - 1.3)(x - 1.6) \\ &+ 0.001825(x - 1.0)(x - 1.3)(x - 1.6)(x - 1.9). \end{aligned}$$

x_i	$f[x_i]$	$f[x_i, x_{i+1}]$	$f[x_i, x_{i+1}, x_{i+2}]$	$f[x_i, \ldots, x_{i+3}]$	$f[x_i, \ldots, x_{i+4}]$
1	0.7651977				
1.3	0.620086	-0.4837057			
1.6	0.4554022	-0.5489460	-0.1087339		
1.9	0.2818186	-0.5786120	-0.0494433	0.0658784	
2.2	0.1103623	-0.5715210	0.0118183	0.0680685	0.001825

It is easily verified that $f(1.5) \approx P_4(1.5) = 0.511820$

3.4 Interpolation at Equally Spaced Points

When $x_0, x_1, \ldots, x_n$ are arranged consecutively with equal spacing,

$$\begin{aligned} P_n(x) &= f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1) + \ldots \\ &+ f[x_0, x_1, \ldots, x_n](x - x_0)(x - x_1)\ldots(x - x_{n-1}), \end{aligned}$$

can be expressed in a simplified form.

Let h be the common difference between two consecutive interpolation points. Then $h = x_{i+1} - x_i$ for each $i = 0, 1, \ldots, n-1$.

$$\begin{aligned} h = x_1 - x_0 &\implies x_1 = x_0 + h \\ &\implies x_2 = x_0 + 2h \\ &\implies x_3 = x_0 + 3h \\ &\quad\vdots \\ &\implies x_n = x_0 + nh \end{aligned}$$

Let

$$\begin{aligned} r = \frac{x - x_0}{h} &\implies x - x_0 = rh \\ &\implies x - x_1 = (r-1)h \\ &\implies x - x_2 = (r-2)h \\ &\quad\vdots \\ &\implies x - x_n = (r-n)h \end{aligned}$$

Define the forward difference operator Δf_i by $\Delta f_i = f_{i+1} - f_i$ where $f_i = f(x_i)$. With this notation,

$$\begin{aligned} f[x_0, x_1] &= \frac{f(x_1) - f(x_0)}{x_1 - x_0} = \frac{1}{h}\Delta f(x_0), \\ f[x_0, x_1, x_2] &= \frac{1}{2h}\left[\frac{\Delta f(x_1) - \Delta f(x_0)}{h}\right] = \frac{1}{2h^2}\Delta^2 f(x_0) \end{aligned}$$

and, in general,

$$f[x_0, x_1, \ldots, x_k] = \frac{1}{k!h^k}\Delta^k f(x_0).$$

Consequently,

$$\begin{aligned}
P_n(x) &= f[x_0] + f[x_0,x_1](x-x_0) + f[x_0,x_1,x_2](x-x_0)(x-x_1) + \dots \\
&+ f[x_0,x_1,\dots,x_n](x-x_0)(x-x_1)\dots(x-x_{n-1}) \\
&= f_0 + \frac{\Delta f_0(rh)}{h} + \frac{\Delta^2 f_0(rh)(r-1)h}{2h^2} + \dots + \frac{\Delta^n f_0(rh)(r-1)h\dots(r-n+1)h}{n!h^n} \\
&= f_0 + r\Delta f_0 + \frac{(r)(r-1)\Delta^2 f_0}{2} + \dots + \frac{(r)(r-1)\dots(r-n+1)\Delta^n f_0}{n!} \\
&= \sum_{k=0}^{n} \binom{r}{k} \Delta^k f_0 \quad \text{where} \quad \Delta^0 f_0 = f_0.
\end{aligned}$$

This formula is called the **Newton's forward difference formula (NFDF)**.

If the interpolating points are reordered as $x_n, x_{n-1}, \dots, x_0$,

$$\begin{aligned}
P_n(x) &= f[x_n] + f[x_{n-1},x_n](x-x_n) + f[x_{n-2},x_{n-1},x_n](x-x_n)(x-x_{n-1}) + \dots \\
&+ f[x_0,x_1,\dots,x_n](x-x_n)(x-x_{n-1})\dots(x-x_1).
\end{aligned}$$

With

$$r = \frac{x-x_n}{h} \quad \Longrightarrow \quad x - x_i = (r+n-i)h.$$

The backward difference operator ∇f_i is defined by $\nabla f_i = f_i - f_{i-1}$. This implies that

$$\begin{aligned}
f[x_{n-1},x_n] &= \frac{1}{h}\nabla f(x_n), \\
f[x_{n-2},x_{n-1},x_n] &= \frac{1}{2h^2}\nabla^2 f(x_n)
\end{aligned}$$

and, in general,

$$f[x_{n-k},\dots,x_{n-1},x_n] = \frac{1}{k!h^k}\nabla^k f(x_n).$$

Consequently,

$$\begin{aligned}
P_n(x) &= f_n + \frac{\nabla f_n(rh)}{h} + \frac{\nabla^2 f_n(rh)(r+1)h}{2h^2} + \dots + \frac{\nabla^n f_n(rh)(r+1)h\dots(r+n-1)h}{n!h^n} \\
&= f_n + \binom{r}{1}\nabla f_n + \binom{r+1}{2}\nabla^2 f_n + \dots + \binom{r+n-1}{n}\nabla^n f_n \\
&= \sum_{k=0}^{n} \binom{r+k-1}{k} \nabla^k f_n \quad \text{where} \quad \nabla^0 f_n = f_n.
\end{aligned}$$

This formula is called the **Newton's backward difference formula (NBDF)**.

Example 3.4. Consider the interpolation points and the function values at these points in example 3.3. The difference table corresponding to this data is shown below.

x_i	f_j	Δf_j ∇f_j	$\Delta^2 f_j$ $\nabla^2 f_j$	$\Delta^3 f_j$ $\nabla^3 f_j$	$\Delta^4 f_j$ $\nabla^4 f_j$
1	0.7651977				
1.3	0.620086	-0.1451117			
1.6	0.4554022	-0.1646838	-0.0195721		
1.9	0.2818186	-0.1735836	-0.0088998	0.0106723	
2.2	0.1103623	-0.1714563	0.0021273	0.0110271	0.0003548

If an approximation to $f(1.1)$ is required, the reasonable choice for $x_0, x_1, \ldots, x_n$ would be $x_0 = 1.0, x_1 = 1.3, x_2 = 1.6, x_3 = 1.9$ and $x_4 = 2.2$, since this choice makes the greatest possible use of data points closest to $x = 1.1$ and also makes use of $\Delta^4 f_j$. This implies that $h = 0.3$ and $r = \frac{1}{3}$, so the Newton's forward difference formula is used with the differences that are underlined in the table.

$$\begin{aligned}
P_4(1.1) &= f_0 + r\Delta f_0 + \frac{(r)(r-1)\Delta^2 f_0}{2} + \frac{(r)(r-1)(r-2)\Delta^3 f_0}{3!} \\
&\quad + \frac{(r)(r-1)(r-2)(r-3)\Delta^4 f_0}{4!} \\
&= 0.7651977 + \frac{1}{3}(-0.1451117) + \left(\frac{1}{3}\right)\left(\frac{-2}{3}\right)\left(\frac{-0.0195721}{2}\right) \\
&\quad + \left(\frac{1}{3}\right)\left(\frac{-2}{3}\right)\left(\frac{-5}{3}\right)\left(\frac{0.0106723}{3!}\right) \\
&\quad + \left(\frac{1}{3}\right)\left(\frac{-2}{3}\right)\left(\frac{-5}{3}\right)\left(\frac{-8}{3}\right)\left(\frac{0.0003548}{4!}\right) \\
&= 0.719645994
\end{aligned}$$

To approximate a value when x is close to the end of the tabulated values, say, $x = 2$, we again would like to make maximum use of the data points closest to x. This requires using the Newton's backward difference formula with $s = \frac{-2}{3}$ and the differences in the table which are enclosed in a box:

$$\begin{aligned}
P_4(2) &= f_n + r\nabla f_n + \frac{(r)(r+1)\nabla^2 f_n}{2} + \frac{(r)(r+1)(r+2)\nabla^3 f_0}{3!} \\
&\quad + \frac{(r)(r+1)(r+2)(r+3)\nabla^4 f_n}{4!} \\
&= 0.1103623 + \frac{-2}{3}(-0.1714563) + \left(\frac{-2}{3}\right)\left(\frac{1}{3}\right)\left(\frac{0.0021273}{2}\right) \\
&\quad + \left(\frac{-2}{3}\right)\left(\frac{1}{3}\right)\left(\frac{4}{3}\right)\left(\frac{0.0110271}{3!}\right) + \left(\frac{-2}{3}\right)\left(\frac{1}{3}\right)\left(\frac{4}{3}\right)\left(\frac{7}{3}\right)\left(\frac{0.0003548}{4!}\right) \\
&= 0.223875
\end{aligned}$$

To compare this method with the divided-difference formula, we will find $P_4(1.5)$ using the Newton's forward difference formula,

$$\begin{aligned} P_4(1.5) &= 0.7651977 + \frac{5}{3}(-0.1451117) + \left(\frac{5}{3}\right)\left(\frac{2}{3}\right)\left(\frac{-0.0195721}{2}\right) \\ &\quad + \left(\frac{5}{3}\right)\left(\frac{2}{3}\right)\left(\frac{-1}{3}\right)\left(\frac{0.0106723}{3!}\right) + \left(\frac{5}{3}\right)\left(\frac{2}{3}\right)\left(\frac{-1}{3}\right)\left(\frac{-4}{3}\right)\left(\frac{0.0003548}{4!}\right) \\ &= 0.511820 \end{aligned}$$

This agrees with our result from example 3.3.

However, to approximate a value when x is located near the middle of the tabulated values, it would be better to use a central difference formula. In this case, the nodes are renamed so that the middle node is x_0. If the number of nodes is even, the choice of the middle node depends on which central difference formula to use. The first of the two middle values is chosen as x_0 in the **Gauss' Forward Formula** , while in the **Gauss' Backward Formula** ,the second middle value is used as x_0.

The Gauss' forward formula for the interpolating polynomial takes the form

$$P_n(x) = f_0 + \binom{r}{1}\Delta f_0 + \binom{r}{2}\Delta^2 f_{-1} + \binom{r+1}{3}\Delta^3 f_{-1} + \binom{r+1}{4}\Delta^4 f_{-2} + \dots$$

On the other hand, the Gauss' backward formula for the interpolating polynomial is given by

$$P_n(x) = f_0 + \binom{r}{1}\Delta f_{-1} + \binom{r+1}{2}\Delta^2 f_{-1} + \binom{r+1}{3}\Delta^3 f_{-2} + \binom{r+2}{4}\Delta^4 f_{-2} + \dots$$

Example 3.5. Consider the same problem in 3.4. Find $P_4(1.5)$ and $P_4(1.7)$ using the Gauss' Forward Formula and Gauss' Backward Fomrula, respectively.

x_i	f_j	Δf_j	$\Delta^2 f_j$	$\Delta^3 f_j$	$\Delta^4 f_j$
1	0.7651977				
1.3	0.620086	-0.1451117			
1.6	0.4554022	-0.1646838	-0.0195721		
1.9	0.2818186	-0.1735836	-0.0088998	0.0106723	
2.2	0.1103623	-0.1714563	0.0021273	0.0110271	0.0003548

Since there are odd number of nodes, $x_0 = 1.6$ for both Gauss Forward and Gauss Backward Formulas. For $P_4(1.5)$, we will use the underlined values in the table.

$$\begin{aligned}
P_4(1.5) &= f_0 + r\Delta f_0 + \frac{(r)(r-1)\Delta^2 f_{-1}}{2} + \frac{(r)(r-1)(r+1)\Delta^3 f_{-1}}{3!} \\
&\quad + \frac{(r)(r-1)(r+1)(r-2)\Delta^4 f_{-2}}{4!} \\
&= 0.4554022 + \frac{-1}{3}(-0.1735836) + \left(\frac{-1}{3}\right)\left(\frac{-4}{3}\right)\left(\frac{-0.0088998}{2}\right) \\
&\quad + \left(\frac{-1}{3}\right)\left(\frac{-4}{3}\right)\left(\frac{2}{3}\right)\left(\frac{0.0110271}{3!}\right) + \left(\frac{-1}{3}\right)\left(\frac{-4}{3}\right)\left(\frac{2}{3}\right)\left(\frac{-7}{3}\right)\left(\frac{0.0003548}{4!}\right) \\
&= 0.511820
\end{aligned}$$

The result is the same for the other two formulas used.

For $P_4(1.7)$, we will use the values which are enclosed in a box.

$$\begin{aligned}
P_4(1.7) &= f_0 + r\Delta f_0 + \frac{(r)(r+1)\Delta^2 f_{-1}}{2} + \frac{(r)(r+1)(r-1)\Delta^3 f_{-1}}{3!} \\
&\quad + \frac{(r)(r+1)(r-1)(r+2)\Delta^4 f_{-2}}{4!} \\
&= 0.4554022 + \frac{1}{3}(-0.1646838) + \left(\frac{1}{3}\right)\left(\frac{4}{3}\right)\left(\frac{-0.0088998}{2}\right) \\
&\quad + \left(\frac{1}{3}\right)\left(\frac{4}{3}\right)\left(\frac{-2}{3}\right)\left(\frac{0.0106723}{3!}\right) + \left(\frac{1}{3}\right)\left(\frac{4}{3}\right)\left(\frac{-2}{3}\right)\left(\frac{7}{3}\right)\left(\frac{0.0003548}{4!}\right) \\
&= 0.3979926
\end{aligned}$$

3.5 Error of Polynomial Interpolation

The next step is to calculate a remainder term or bound for the error involved in approximating a function by an interpolating polynomial. Before we state the theorem which establishes this formula, let us recall the Generalized Rolle's theorem which will be used for the proof.

Theorem 3.3. (Generalized Rolle's Theorem) *Let the continuous function f on $[a,b]$ be n times differentiable on (a,b). If f vanishes at the $n+1$ distinct numbers $x_0, x_1, \ldots x_n$ in $[a,b]$, then a number c in (a,b) exists with $f^n(c) = 0$.*

The following theorem will derive the error formula in approximating a function by an interpolating polynomial.

Theorem 3.4. *If $x_0, x_1, \ldots x_n$ are distinct numbers in the interval $[a,b]$ and if f has $(n+1)$ continuous derivatives on $[a,b]$, then, for each x in $[a,b]$, a number $\xi(x)$ in (a,b) exists with*

$$f(x) = P(x) + \frac{f^{n+1}(\xi(x))}{(n+1)!}(x-x_0)(x-x_1)\ldots(x-x_n),$$

where P is the Lagrange form of the interpolating polynomial.

Proof. Note first that, if $x = x_k$ for $k = 0, 1, \ldots n$, then $f(x_k) = P(x_k)$. If $x \neq x_k$ for any $k = 0, 1, \ldots n$, define the function g for t in $[a, b]$ by

$$\begin{aligned} g(t) &= f(t) - P(t) - [f(x) - P(x)]\frac{(t-x_0)(t-x_1)\ldots(t-x_n)}{(x-x_0)(x-x_1)\ldots(x-x_n)} \\ &= f(t) - P(t) - [f(x) - P(x)]\prod_{i=0}^{n}\frac{(t-x_i)}{(x-x_i)}. \end{aligned}$$

Since f has $(n+1)$ continuous derivatives on $[a, b]$, P has derivatives of all orders at each number in $[a, b]$, and $x \neq x_k$ for any k, it follows that g has $(n+1)$ continuous derivatives on $[a, b]$. For $t = x_k$

$$\begin{aligned} g(x_k) &= f(x_k) - P(x_k) - [f(x) - P(x)]\prod_{i=0}^{n}\frac{(x_k-x_i)}{(x-x_i)} \\ &= 0 - [f(x) - P(x)] \cdot 0 \\ &= 0. \end{aligned}$$

Moreover,

$$\begin{aligned} g(x) &= f(x) - P(x) - [f(x) - P(x)]\prod_{i=0}^{n}\frac{(x-x_i)}{(x-x_i)} \\ &= f(x) - P(x) - [f(x) - P(x)] \\ &= 0. \end{aligned}$$

Thus, g vanishes at the $(n+2)$ distinct numbers $x, x_0, x_1, \ldots x_n$. By the Generalized Rolle's Theorem, there exists $\xi \equiv \xi(x)$ in (a, b) for which $g^{n+1}(\xi) = 0$. Evaluating g^{n+1} at ξ gives

$$0 = g^{n+1}(\xi) = f^{n+1}(\xi) - P^{n+1}(\xi) - [f(x) - P(x)]\left[\frac{d^{n+1}}{dt^{n+1}}\left(\prod_{i=0}^{n}\frac{(t-x_i)}{(x-x_i)}\right)\right]_{t=\xi}.$$

Since P is a polynomial of degree at most n, the $(n+1)$st derivative P^{n+1}, must be identically zero. Also $\displaystyle\prod_{i=0}^{n}\frac{(t-x_i)}{(x-x_i)}$ is a polynomial of degree $(n+1)$, so

$$\prod_{i=0}^{n}\frac{(t-x_i)}{(x-x_i)} = \left(\frac{1}{\prod_{i=0}^{n}(x-x_i}\right)t^{n+1} + (\text{lower} - \text{degree terms in } t),$$

and

$$\frac{d^{n+1}}{dt^{n+1}}\prod_{i=0}^{n}\frac{(t-x_i)}{(x-x_i)} = \frac{(n+1)!}{\prod_{i=0}^{n}(x-x_i)}.$$

This gives us

$$0 = f^{n+1}(\xi) - 0 - [f(x) - P(x)]\frac{(n+1)!}{\prod_{i=0}^{n}(x - x_i)}$$

and upon solving for $f(x)$,

$$f(x) = P(x) + \frac{f^{n+1}(\xi(x))}{(n+1)!}\prod_{i=0}^{n}(x - x_i).$$

□

We will denote the error formula by $E(x)$ where

$$E(x) = \frac{f^{n+1}(\xi(x))}{(n+1)!}\prod_{i=0}^{n}(x - x_i).$$

Example 3.6. Find an error bound for the approximation of $f(3)$ in the example 3.1.

Solution.

We have $E(x) = \dfrac{f^{n+1}(\xi(x))}{(n+1)!}\prod_{i=0}^{n}(x - x_i)$. Since $f^3(x) = \dfrac{-6}{x^4}$ then

$$E(x) = \frac{-6}{\xi^4 3!}(x-2)(x-2.5)(x-4).$$

Since $f^3(x) = \frac{-6}{x^4} < 0$ for all x in $[2, 4]$ we can take $\xi = 2$ such that $|\frac{-6}{2^4}|$ is an upper bound for $\frac{-6}{\xi^4}$. Thus,

$$\begin{aligned} |E(3)| &= \left|\frac{-6}{\xi^4 3!}(3-2)(3-2.5)(3-4)\right| \\ &\leq \left|\frac{-6}{2^4 3!}(1)(0.5)(-1)\right| \\ &= 0.03125 \end{aligned}$$

This shows that the estimate is accurate to at least one decimal place. The actual error is 0.008333333 which is less then the upper bound for the error obtained above.

Exercises.

1. Use the interpolation points $x_0 = 0.3, x_1 = 0.32, x_2 = 0.35$ given $f(x) = \sin x$ to construct a Lagrange polynomial of degree two or less. Find an approximation to $\sin 0.34$ and find the error bound for the approximation.

2. Add the interpolation point 0.33 to the data in Exercise 1 and construct a Lagrange polynomial of degree three or less. Approximate $\sin 0.34$ and find the error bound for the approximation.

3. Repeat Exercises 1 and 2 using the method of undetermined coefficients.

4. Use the interpolation points $x_0 = 1, x_1 = 1.05, x_2 = 1.1, x_3 = 1.15$ to construct a third-degree Lagrange polynomial approximation to $f(1.09)$. The function being approximated is $f(x) = \log_{10} \tan x$. Use this knowledge to find an error bound for the approximation.

5. Repeat Exercise 4 using the method of undetermined coefficients.

6. Use the method of divided differences to construct the interpolating polynomial of degree four for the unequally spaced points given in the following table:

x	0.0	0.1	0.3	0.6	1.0
$f(x)$	-6.00000	-5.89483	-5.65014	-5.17788	-4.28172

7. Suppose the data $f(1.1) = -3.99583$ is added to Exercise 6. Construct the interpolating polynomial of degree five.

8. Approximate the given values using the following data and the appropriate difference formula:

x	0.0	0.2	0.4	0.6	0.8
$f(x)$	1.00000	1.22140	1.49182	1.82212	2.22554

   (a) $f(0.05)$
   (b) $f(0.35)$
   (c) $f(0.5)$
   (d) $f(0.65)$

9. Suppose $f(x) = \arccos(x)$ with $h = 0.1, x_0 = -1$ and $n = 5$, create the difference table and approximate the given values using the appropriate difference formula:

   (a) $f(-0.98)$
   (b) $f(-0.83)$
   (c) $f(-0.68)$
   (d) $f(-0.52)$

Chapter 4

Numerical Differentiation

One reason for using the class of algebraic polynomials to approximate an arbitrary set of data is that, given any continuous function defined on a closed interval, there exists a polynomial which is arbitrarily close to the function at every point in the interval. Another property that this class possesses is that the derivatives and integrals of polynomials are quite easily obtained and evaluated. It should not be surprising, then, that most procedures for approximating integrals and derivatives commence with algebraic polynomials approximating the function.

Approximating derivatives of a function $f(x)$ may be found from a polynomial approximation $P(x)$ simply by accepting $P'(x), P''(x), \ldots$ in place of $f'(x), f''(x), \ldots$. Hence, given a Lagrange form of the interpolating polynomial or an interpolating polynomial obtained by the method of divided differences, an approximation of the derivatives of the given function could easily be obtained.

If a function f is described only in terms of tabulated data, it would be impossible to use an analytical method to compute the derivative. If this derivative is required, the derivative of the interpolating polynomial at the tabulated points may be used.

4.1 Numerical Differentiation Using Newton's Formulas

Consider Newton's Forward Difference Formula,

$$f(x) \approx P_n(x) = f_0 + r\Delta f_0 + \frac{(r)(r-1)\Delta^2 f_0}{2} + \ldots + \binom{r}{n}\Delta^n f_0$$

Differentiating this with respect to x, we obtain

$$f'(x) \approx P_n'(x) = \frac{d}{dr}(P_n(x))\ \frac{dr}{dx}\ .$$

Since $\dfrac{x - x_0}{h}$ then $\dfrac{dr}{dx} = \dfrac{1}{h}$, and hence,

$$\begin{aligned} f'(x) &\approx \frac{d}{dr}(P_n(x))\ \frac{1}{h} \\ &= \frac{1}{h}\left(\Delta f_0 + \frac{(2r-1)\Delta^2 f_0}{2} + \frac{(3r^2-6r+2)\Delta^3 f_0}{3!} + \ldots + \frac{d}{dr}\binom{r}{n}\Delta^n f_0\right) \end{aligned}$$

From the formula above for $f'(x)$, we have the following modification if $x = x_0$ since $r = 0$.

$$f'(x) \approx \frac{1}{h}\left(\Delta f_0 - \frac{1}{2}\Delta^2 f_0 + \frac{1}{3}\Delta^3 f_0 - \ldots + \frac{(-1)^{n+1}}{n}\Delta^n f_0\right).$$

If $x = x_j$ is a node other than x_0, we may choose to enter the forward difference table at x_j instead of at x_0. In this case,

$$f'(x) \approx \frac{1}{h}\left(\Delta f_j - \frac{1}{2}\Delta^2 f_j + \frac{1}{3}\Delta^3 f_j - \ldots + \ frac(-1)^{n-j+1}n - j\Delta^{n-j} f_j\right).$$

Similarly, considering the Newton's backward difference formula,

$$\begin{aligned} f'(x) &\approx \frac{d}{dr}(P_n(x))\ \frac{dr}{dx} \\ &= \frac{d}{dr}(P_n(x))\ \frac{1}{h} \\ &= \frac{1}{h}\left(\nabla f_n + \frac{(2r+1)\nabla^2 f_n}{2} + \frac{(3r^2+6r+2)\nabla^3 f_n}{3!} + \ldots + \frac{d}{dr}\binom{r+n-1}{n}\nabla^n f_n\right) \end{aligned}$$

since $\dfrac{x - x_n}{h}$ which implies that $\dfrac{dr}{dx} = \dfrac{1}{h}$.

If $x = x_n$ then $r = 0$ and

$$f'(x) \approx \frac{1}{h}\left(\nabla f_n + \frac{1}{2}\nabla^2 f_n + \frac{1}{3}\nabla^3 f_n - \ldots + \frac{1}{n}\nabla^n f_n\right).$$

Likewise, if $x = x_j$ is a node other than x_n, we may choose to enter the backward difference table at x_j instead of at x_n. In this case,

$$f'(x) \approx \frac{1}{h}\left(\nabla f_j + \frac{1}{2}\nabla^2 f_j + \frac{1}{3}\nabla^3 f_j - \ldots + \frac{1}{j}\nabla^j f_j\right).$$

Example 4.1. Given $x_0 = 0.2, n = 4$ and $h = 0.2$, the function values at these points and the difference table is shown below.

x_j	f_j	Δf_j ∇f_j	$\Delta^2 f_j$ $\nabla^2 f_j$	$\Delta^3 f_j$ $\nabla^3 f_j$	$\Delta^4 f_j$ $\nabla^4 f_j$
0.2	1.221402758				
0.4	1.491824698	0.270421939			
0.6	1.8221188	0.330294103	0.059872163		
0.8	2.225540928	0.403422128	0.073128025	0.013255862	
1.0	2.718281828	0.4927409	0.089318772	0.016190747	0.002934884

An approximation to $f'(0.3)$ will be

$$\begin{aligned} f'(0.3) &\approx \frac{1}{h}\left(\Delta f_0 + \frac{(2r-1)\Delta^2 f_0}{2} + \frac{(3r^2-6r+2)\Delta^3 f_0}{3!} + \frac{(4r^3-18r^2+22r-6)\Delta^4 f_0}{4!}\right) \\ &= \frac{1}{0.2}\left(0.270421939 + \frac{(2(\frac{1}{2})-1)0.059872163}{2} + \frac{(3(\frac{1}{2})^2-6(\frac{1}{2})+2)0.013255862}{3!}\right. \\ &\quad \left. + \frac{(4(\frac{1}{2})^3-18(\frac{1}{2})^2+22(\frac{1}{2})-6)0.002934884}{4!}\right) \\ &= 1.349959494 \end{aligned}$$

Similarly, an approximation to $f'(0.9)$ will be

$$\begin{aligned} f'(0.9) &\approx \frac{1}{h}\left(\nabla f_n + \frac{(2r+1)\nabla^2 f_n}{2} + \frac{(3r^2+6r+2)\nabla^3 f_n}{3!} + \ldots + \frac{(4r^3+18r^2+22r+6)\nabla^4 f_0}{4!}\right) \\ &= \frac{1}{0.2}\left(0.4927409 + \frac{(2(\frac{-1}{2})-1)0.089318772}{2} + \frac{(3(\frac{-1}{2})^2+6(\frac{-1}{2})+2)0.016190747}{3!}\right. \\ &\quad \left. + \frac{(4(\frac{-1}{2})^3+18(\frac{-1}{2})^2+22(\frac{-1}{2})+6)0.002934884}{4!}\right) \\ &= 2.459719993 \end{aligned}$$

Let us consider approximating the derivatives at interpolation points.

$$\begin{aligned} f'(0.2) &\approx \frac{1}{h}\left(\Delta f_0 - \frac{1}{2}\Delta^2 f_0 + \frac{1}{3}\Delta^3 f_0 - \frac{1}{4}\Delta^4 f_0\right) \\ &= \frac{1}{0.2}\left(0.270421939 - \frac{1}{2}(0.059872163) + \frac{1}{3}(0.013255862) - \frac{1}{4}(0.002934884)\right) \\ &= 1.220853787 \end{aligned}$$

$$\begin{aligned} f'(0.4) &\approx \frac{1}{h}\left(\Delta f_1 - \frac{1}{2}\Delta^2 f_1 + \frac{1}{3}\Delta^3 f_1\right) \\ &= \frac{1}{0.2}\left(0.330294103 - \frac{1}{2}(0.073128025) + \frac{1}{3}(0.016190747)\right) \\ &= 1.495635028 \end{aligned}$$

$$\begin{aligned} f'(1.0) &\approx \frac{1}{h}\left(\nabla f_n + \frac{1}{2}\nabla^2 f_n + \frac{1}{3}\nabla^3 f_n + \frac{1}{4}\nabla^4 f_n\right) \\ &= \frac{1}{0.2}\left(0.4927409 + \frac{1}{2}(0.089318772) + \frac{1}{3}(0.016190747) + \frac{1}{4}(0.002934884)\right) \\ &= 2.717654613 \end{aligned}$$

$$\begin{aligned} f'(0.8) &\approx \frac{1}{h}\left(\nabla f_3 + \frac{1}{2}\nabla^2 f_3 + \frac{1}{3}\nabla^3 f_3\right) \\ &= \frac{1}{0.2}\left(0.403422128 + \frac{1}{2}(0.073128025) + \frac{1}{3}(0.013255862)\right) \\ &= 2.222023807 \end{aligned}$$

4.2 Error of Numerical Differentiation

The error of polynomial interpolation for equally spaced points is given by

$$\begin{aligned} E(x) &= \frac{f^{n+1}(\xi(x))}{(n+1)!}\prod_{i=0}^{n}(x - x_i) \\ &= \frac{r(r-1)(r-2)\ldots(r-n)h^{n+1}f^{n+1}(\xi(x))}{(n+1)!} \\ &= \binom{r}{n+1}h^{n+1}f^{n+1}(\xi(x)) \end{aligned}$$

so the error of the numerical differentiation formula is

$$\begin{aligned} E'(x) &= \frac{d}{dx}\left[\binom{r}{n+1}h^{n+1}f^{n+1}(\xi(x))\right] \\ &= \frac{1}{h}\frac{d}{dr}\left[\binom{r}{n+1}h^{n+1}f^{n+1}(\xi(x))\right] \end{aligned}$$

If $x = x_j$ is a node, the corresponding error of the approximation is given by

$$E'(x_j) = \frac{(-1)^{n-j+1}}{n-j+1}f^{n-j+1}(\xi(x))h^{n-j}$$

Example 4.2. Consider the data in example 4.1. These data were generated using the function $f(x) = e^x$, so that $f'(x) = e^x$. Find an error bound for the approximations in this example.

$$\begin{aligned}
|E'(0.3)| &\leq \frac{d}{dr}\left[\binom{r}{5}h^4 f^5(\xi)\right] \\
&= \frac{d}{dr}\left[\frac{r(r-1)(r-2)(r-3)(r-4)}{5!}0.2^4 e\right] \\
&= \frac{d}{dr}\left[\frac{r^5-10r^4+35r^3-50r^2+24r}{5!}0.2^4 e\right] \\
&= \left[\frac{5r^4-40r^3+105r^2-100r+24}{5!}0.2^4 e\right] \\
&= 0.000160832
\end{aligned}$$

$$\begin{aligned}
|E'(0.9)| &\leq \frac{d}{dr}\left[\binom{r}{5}h^4 f^5(\xi)\right] \\
&= \frac{d}{dr}\left[\frac{r(r+1)(r+2)(r+3)(r+4)}{5!}0.2^4 e\right] \\
&= \frac{d}{dr}\left[\frac{r^5+10r^4+35r^3+50r^2+24r}{5!}0.2^4 e\right] \\
&= \left[\frac{5r^4+40r^3+105r^2+100r+24}{5!}0.2^4 e\right] \\
&= 0.000160832
\end{aligned}$$

$$\begin{aligned}
|E'(0.2)| &\leq \frac{(-1)^{n-j+1}}{n-j+1} f^{n-j+1}(\xi(x))h^{n-j} \\
&= \frac{(-1)^{n+1}}{n+1} f^{n+1}(\xi)h^n \\
&= \frac{(-1)^5}{5} f^5(\xi)h^4 \\
&= 0.00086985
\end{aligned}$$

$$\begin{aligned}
|E'(0.4)| &\leq \frac{(-1)^{n-j+1}}{n-j+1} f^{n-j+1}(\xi(x))h^{n-j} \\
&= \frac{(-1)^n}{n} f^n(\xi)h^{n-1} \\
&= \frac{(-1)^4}{4} f^4(\xi)h^3 \\
&= 0.005436564
\end{aligned}$$

$$\begin{aligned} |E'(0.8)| &\leq \frac{(-1)^{n-j+1}}{n-j+1} f^{n-j+1}(\xi(x)) h^{n-j} \\ &= \frac{(-1)^{n-2}}{n-2} f^{n-2}(\xi) h^{n-3} \\ &= \frac{(-1)^2}{2} f^2(\xi) h \\ &= 0.271828183 \end{aligned}$$

$$\begin{aligned} |E'(1.0)| &\leq \frac{(-1)^{n-j+1}}{n-j+1} f^{n-j+1}(\xi(x)) h^{n-j} \\ &= f(\xi) \\ &= 2.718281828 \end{aligned}$$

The absolute error for the estimate at 0.3 is 0.000100686, at 0.9 is 0.000116882, at 0.2 is 0.000548971, at 0.4 is 0.00381033, at 0.8 is 0.000627216, and at 1.0 is 0.003517121.

Exercises.

1. Use the interpolating polynomial obtained from Exercise 1 of the preceding chapter to approximate $f'(x)$. Find an approximation to $f'(0.34)$ afterwards and compare your answer with the actual value.

2. Use the interpolating polynomial obtained from Exercise 2 of the preceding chapter to approximate $f'(x)$. Find an approximation to $f'(0.34)$ afterwards. Compare the absolute error obtained with your estimate in no. 1

3. Use the interpolating polynomial obtained from Exercise 4 of the preceding chapter to approximate $f'(x)$. Find an approximation to $f'(1.09)$ afterwards.

4. Use the interpolating polynomial obtained from Exercise 6 of the preceding chapter to approximate $f'(x)$.

5. Use the interpolating polynomial obtained from Exercise 7 of the preceding chapter to approximate $f'(x)$.

6. Create the difference table for the following data. Then find an estimate for $f'(0.01)$ and $f'(0.65)$ using the appropriate difference formula.

x	0.0	0.2	0.4	0.6	0.8
$f(x)$	1.00000	1.22140	1.49182	1.82212	2.22554

7. Given $x_0 = 1, n = 5, h = 0.5$ and $f(x) = \cos x$, find an approximation for $f'(1.25)$, $f'(3.25)$, $f'(1.5)$, and $f'(3)$. Find the corresponding error bounds.

8. Find the approximate second derivatives of Exercise 7.

Chapter 5

Numerical Integration

The need often arises for evaluating the definite integral of a function that has no explicit antiderivative or whose antiderivatives has values that are not easily obtained. The basic method involved in approximating $\int_a^b f(x)dx$ is called numerical quadrature and uses a sum of the type

$$\sum_{i=0}^{n} A_j f(x_j)$$

to approximate $\int_a^b f(x)dx$.

5.1 Lagrange Formula

The method of quadrature we discuss in this section are based on the Lagrange form of the interpolating polynomial

$$P(x) = \sum_{j=0}^{n} f(x_j) l_j(x) \quad \text{where} \quad l_j(x) = \prod_{k \neq j} \frac{(x - x_k)}{(x_j - x_k)}$$

If we integrate this interpolating polynomial,

$$\begin{aligned} \int_a^b f(x)dx &\approx \int_a^b P(x)dx \\ &= \int_a^b \sum_{j=0}^{n} f(x_j) l_j(x)dx \\ &= \sum_{j=0}^{n} f(x_j) \int_a^b l_j(x)dx \\ &= \sum_{i=0}^{n} A_j f(x_j) \end{aligned}$$

where $A_j = \int_a^b l_j(x)dx$, for each $j = 0, 1, \ldots n$. The A_j's are called **weights** of the quadrature while the interpolation points are called **nodes** of the quadrature.

Example 5.1. Find an estimate for the integral

$$\int_{-1}^{2} f(x)dx$$

given the following values:

x_j	-1	0	1	2
$f(x_j)$	-4	2	4	8

Solution. We have

$$\begin{aligned} l_0(x) &= -\frac{x^3 - 3x^2 + 2x}{6} \\ l_1(x) &= \frac{x^3 - 2x^2 - x + 2}{2} \\ l_2(x) &= -\frac{x^3 - x^2 - 2x}{2} \\ l_3(x) &= \frac{x^3 - x}{6} \end{aligned}$$

which will give us

$$\begin{aligned}
A_0 &= \int_{-1}^{2} -\frac{x^3 - 3x^2 + 2x}{6}\, dx \\
&= -\frac{1}{6}\left[\frac{x^4}{4} - x^3 + x^2\right]_{-1}^{2} \\
&= \frac{3}{8} \\
A_1 &= \int_{-1}^{2} \frac{x^3 - 2x^2 - x + 2}{2}\, dx \\
&= \frac{1}{2}\left[\frac{x^4}{4} - \frac{2x^3}{3} - \frac{x^2}{2} + 2x\right]_{-1}^{2} \\
&= \frac{9}{8} \\
A_2 &= \int_{-1}^{2} -\frac{x^3 - x^2 - 2x}{2}\, dx \\
&= -\frac{1}{2}\left[\frac{x^4}{4} - \frac{x^3}{3} - x^2\right]_{-1}^{2} \\
&= \frac{9}{8} \\
A_3 &= \int_{-1}^{2} \frac{x^3 - x}{6}\, dx \\
&= \frac{1}{6}\left[\frac{x^4}{4} - \frac{x^2}{2}\right]_{-1}^{2} \\
&= \frac{3}{8}
\end{aligned}$$

Thus,

$$\begin{aligned}
\int_{-1}^{2} f(x)dx &\approx \sum_{i=0}^{3} A_j f(x_j) \\
&= \frac{3}{8}(-4) + \frac{9}{8}(2) + \frac{9}{8}(4) + \frac{3}{8}(8) \\
&= \frac{33}{4}
\end{aligned}$$

Suppose we have the same interpolation points as in the above example, but the function values are different.

x_j	-1	0	1	2
$g(x_j)$	-3	1	2	5

Then

$$\begin{aligned}\int_{-1}^{2} g(x)dx &\approx \sum_{i=0}^{3} A_j g(x_j) \\ &= \frac{3}{8}(-3) + \frac{9}{8}(1) + \frac{9}{8}(2) + \frac{3}{8}(5) \\ &= \frac{33}{8}\end{aligned}$$

5.2 The Method of Undetermined Coefficients

In the preceding section, we showed that

$$\int_a^b f(x)dx \approx \sum_{j=0}^{n} A_j f(x_j).$$

Since the A_j's depend only on the nodes and the interval $[a, b]$, we can compute their values using functions other than the function f.

For example, if we consider the function

$$f_k = x^k, k = 0, 1, 2 \ldots, n.$$

We will have the system of equations

$$\int_a^b x^k dx = \sum_{j=0}^{n} A_j (x_j)^k$$

with $(n + 1)$ equations in $(n + 1)$ unknowns, $A_0, A_1, \ldots, A_n$. These will then be substituted into the quadrature.

Example 5.2. For the data given in example 5.1, use the method of undetermined coefficients to find an estimate for $\int_{-1}^{2} f(x)\ dx$.

Solution. Solving $\int_a^b x^k dx$ for $k = 0, 1, 2, 3$ we have

$$\begin{aligned}
k=0: \int_{-1}^{2} x^0\, dx &= x\,]_{-1}^{2} \\
&= 3 \\
k=1: \int_{-1}^{2} x^1\, dx &= \frac{x^2}{2}\,\Big]_{-1}^{2} \\
&= \frac{3}{2} \\
k=2: \int_{-1}^{2} x^2\, dx &= \frac{x^3}{3}\,\Big]_{-1}^{2} \\
&= 3 \\
k=3: \int_{-1}^{2} x^3\, dx &= \frac{x^4}{4}\,\Big]_{-1}^{2} \\
&= \frac{15}{4}
\end{aligned}$$

Now solving for $\sum_{i=0}^{n} A_j(x_j)^k$ will give us

$$\begin{aligned}
k=0: \sum_{i=0}^{3} A_j(x_j)^0 &= A_0 + A_1 + A_2 + A_3 \\
k=1: \sum_{i=0}^{3} A_j(x_j)^1 &= A_0(-1) + A_1(0) + A_2(1) + A_3(2) \\
k=2: \sum_{i=0}^{3} A_j(x_j)^2 &= A_0(1) + A_1(0) + A_2(1) + A_3(4) \\
k=3: \sum_{i=0}^{3} A_j(x_j)^3 &= A_0(-1) + A_1(0) + A_2(1) + A_3(8)
\end{aligned}$$

Since $\int_a^b x^k dx = \sum_{i=0}^{n} A_j(x_j)^k$ we have the following system of equations:

$$\begin{aligned}
A_0 + A_1 + A_2 + A_3 &= 3 \\
A_0(-1) + A_1(0) + A_2(1) + A_3(2) &= \frac{3}{2} \\
A_0(1) + A_1(0) + A_2(1) + A_3(4) &= 3 \\
A_0(-1) + A_1(0) + A_2(1) + A_3(8) &= \frac{15}{4}
\end{aligned}$$

Solving this linear system, we obtain, as before $A_0 = \frac{3}{8}, A_1 = \frac{9}{8}, A_2 = \frac{9}{8}, A_3 = \frac{3}{8}$ and $\int_{-1}^{2} f(x)dx \approx \frac{33}{4}$.

5.3 The Newton-Cotes Integration Formulas

For equally spaced points along the interval $[a, b]$, a convenient formula to interpolate a function f is Newton's forward difference formula. In using this method, if $x_0 = a$ and $x_n = b$, then the resulting formulas are known as **Newton-Cotes formulas** .

We know that the error depends to a certain extent on the stepsize h. For a fixed interval $[a, b]$, we woould want to minimize the error by making h smaller. However, the computations become increasingly difficult as the degree of the interpolating polynomial increases. For this reason, only the lower degree Newton-Cotes formulas are often used. The first three Newton-Cotes formulas are shown below.

1. If $n = 1$ we have

$$\begin{aligned}
\int_a^b f(x)dx &= \int_{x_0}^{x_1} f(x)dx \\
&\approx \int_{x_0}^{x_1} P(x)dx \\
&= \int_{x_0}^{x_1} (f_0 + r\Delta f_0)dx \\
&= h\int_0^1 (f_0 + r\Delta f_0)dr \\
&= h\left[rf_0 + \frac{r^2}{2}\Delta f_0\right]_0^1 \\
&= h\left[f_0 + \frac{1}{2}\Delta f_0\right] \\
&= h\left[f_0 + \frac{1}{2}(f_1 - f_0)\right] \\
&= \frac{h}{2}(f_0 + f_1)
\end{aligned}$$

2. If $n = 2$ we have

$$\begin{aligned}
\int_a^b f(x)dx &= \int_{x_0}^{x_2} f(x) \\
&\approx \int_{x_0}^{x_1} P(x)dx \\
&= \int_{x_0}^{x_2} \left(f_0 + r\Delta f_0 + \frac{r(r-1)}{2}\Delta^2 f_0 \right) dx \\
&= h\int_0^2 \left(f_0 + r\Delta f_0 + \frac{r^2-r}{2}\Delta^2 f_0 \right) dr \\
&= h\left[rf_0 + \frac{r^2}{2}\Delta f_0 + \left(\frac{r^3}{6} - \frac{r^2}{4} \right)\Delta^2 f_0 \right]_0^2 \\
&= h\left[2f_0 + \frac{4}{2}\Delta f_0 + \left(\frac{8}{6} - \frac{4}{4} \right)\Delta^2 f_0 \right] \\
&= h\left[2f_0 + 2(f_1 - f_0) + \frac{1}{3}(f_2 - 2f_1 + f_0) \right] \\
&= \frac{h}{3}(f_0 + 4f_1 + f_2)
\end{aligned}$$

3. If $n = 3$ we have

$$\begin{aligned}
\int_a^b f(x)dx &= \int_{x_0}^{x_3} f(x) \\
&\approx \int_{x_0}^{x_3} P(x)dx \\
&= \int_{x_0}^{x_3} \left(f_0 + r\Delta f_0 + \frac{r(r-1)}{2}\Delta^2 f_0 + \frac{r(r-1)(r-2)}{6}\Delta^3 f_0 \right) dx \\
&= h\int_0^3 \left(f_0 + r\Delta f_0 + \frac{r^2-r}{2}\Delta^2 f_0 + \frac{r^3-3r^2+2r}{6}\Delta^3 f_0 \right) dr \\
&= h\left[rf_0 + \frac{r^2}{2}\Delta f_0 + \left(\frac{r^3}{6} - \frac{r^2}{4} \right)\Delta^2 f_0 + \left(\frac{r^4}{24} - \frac{r^3}{6} + \frac{r^2}{6} \right)\Delta^3 f_0 \right]_0^3 \\
&= \frac{3h}{8}(f_0 + 3f_1 + 3f_2 + f_3)
\end{aligned}$$

To find the error in using the Newton-Cotes formulas to approximate the value of a definite integral, recall that the error of polynomial interpolation is given by $E(x) = f(x) - P(x)$. The error of the Newton-Cotes formulas is obtained by integrating both sides of the equation.

The errors of the first three Newton-Cotes formulas are shown below:

1. n=1

$$\text{Error} = -\frac{h^3}{12} f''(\xi) \qquad \text{where } x_0 \le \xi \le x_1$$

2. n=2

$$\text{Error} = -\frac{h^5}{90} f^{(4)}(\xi) \qquad \text{where } x_0 \le \xi \le x_2$$

3. n=3

$$\text{Error} = -\frac{3h^5}{80} f^{(4)}(\xi) \qquad \text{where } x_0 \le \xi \le x_3$$

Example 5.3. Find the estimate for

$$\int_1^2 \ln x \; dx$$

using the Newton-Cotes formula of order 2. Compare this with the actual value of the integral and find an error bound for the approximation.

Solution. We have $h = \dfrac{b-a}{n} = \dfrac{1}{2}$. Thus,

$$\begin{aligned} \int_1^2 \ln x \; dx &\approx \frac{h}{3}(f_0 + 4f_1 + f_2) \\ &= \frac{1}{6}(\ln 1 + 4 \ln 1.5 + \ln 2) \\ &= 0.3858346. \end{aligned}$$

Since $f^{(4)}(x) = \dfrac{-6}{x^4}$, an upper bound for $|f^{(4)}(x)|$ is 6. Hence,

$$\begin{aligned} \text{Error} &= -\frac{h^5}{90} f^{(4)}(\xi) \\ &\le \left| -\frac{6}{2^5 90} \right| \\ &= 2.1 \times 10^{-3} \end{aligned}$$

The actual value is

$$x \ln x - x \,]_1^2 = 0.386294$$

which shows that the result is actually accurate to two decimal places.

5.4 Composite Numerical Integration

The Newton-Cotes formulas are generally unsuitable for use over large integration intervals. High-degree formulas would be required for use over such intervals and the values of the coefficients in these formulas are difficult to obtain. In this section, we discuss a piecewise approach to numerical integration using the Newton-Cotes formulas of order one, two, and three.

5.4.1 Trapezoidal Rule

Subdivide $[a, b]$ into n subintervals of equal width h. Applying Newton-Cotes formula of order one to each of the subinterval, we have

$$\begin{aligned}
\int_a^b f(x)dx &= \int_{x_0}^{x_n} f(x)\ dx \\
&\approx \int_{x_0}^{x_1} f(x)\ dx + \int_{x_1}^{x_2} f(x)\ dx + \ldots + \int_{x_{n-1}}^{x_n} f(x)\ dx \\
&= \frac{h}{2}(f_0 + f_1) + \frac{h}{2}(f_1 + f_2) + \ldots + \frac{h}{2}(f_{n-1} + f_n) \\
&= \frac{h}{2}\left[f_0 + f_n + 2\sum_{i=1}^{n-1} f_i\right]
\end{aligned}$$

This is known as the **trapezoidal rule** of order n denoted by $T_n(f)$. Hence,

$$T_n(f) = \frac{h}{2}\left[f_0 + f_n + 2\sum_{i=1}^{n-1} f_i\right]$$

Since the local error for each subinterval is $-\frac{h^3}{12}f''(\xi)$where $x_{i-1} \leq \xi \leq x_i$ and $i = 1, 2, \ldots n$, the total error is

$$-\frac{h^3}{12}\sum_{i=1}^{n} f''(\xi_i).$$

If f'' is continuous on $[a, b]$, then there exists a number ξ in (a, b) such that the total error is $-\frac{h^3}{12}f''(\xi)(n)$. Since $h = \frac{b-a}{n}$, we have $b - a = hn$ and hence,

$$\text{Error} = -\frac{(b-a)h^2}{12}f''(\xi).$$

Example 5.4. Find an estimate for $\int_0^1 \frac{dx}{1+x}$ using the trapezoidal rule of order 12.

Solution. We have $h = \dfrac{b-a}{n} = \dfrac{1}{12}$. The values that will be used are given in the table below.

j	x_j	$f(x_j)$
0	0	1
1	1/12	12/13
2	1/6	6/7
3	1/4	4/5
4	1/3	3/4
5	5/12	12/17
6	1/2	2/3
7	7/12	12/19
8	2/3	3/5
9	3/4	4/7
10	5/6	6/11
11	11/12	12/23
12	1	1/2

$$\begin{aligned} T_n(f) &= \frac{h}{2}\left[f_0 + f_n + 2\sum_{i=1}^{n-1} f_i\right] \\ &= \frac{1}{24}\left[1 + \frac{1}{2} + 2\left(\frac{12}{13} + \frac{6}{7} + \frac{4}{5} + \frac{3}{4} + \frac{12}{17} + \frac{2}{3} + \frac{12}{19} + \frac{3}{5} + \frac{4}{7} + \frac{6}{11} + \frac{12}{23}\right)\right] \\ &\approx 0.69358083 \end{aligned}$$

The actual value is

$$\ln(1+x)\ ]_0^1 = 0.693147181$$

which shows that the result is accurate to three decimal places.

5.4.2 Simpson's 1/3 Rule

Subdivide $[a, b]$ into an even number $n = 2k$ of subintervals, each of width h. Applying Newton-Cotes formula of order two to each of the subinterval, we have

$$\begin{aligned} \int_a^b f(x)dx &= \int_{x_0}^{x_n} f(x)\ dx \\ &\approx \int_{x_0}^{x_2} f(x)\ dx + \int_{x_2}^{x_4} f(x)\ dx + \ldots + \int_{x_{n-2}}^{x_n} f(x)\ dx \\ &= \frac{h}{3}(f_0 + 4f_1 + f_2) + \frac{h}{3}(f_2 + 4f_3 + f_4) + \ldots + \frac{h}{3}(f_{n-2} + 4f_{n-1} + f_n) \\ &= \frac{h}{3}\left[f_0 + f_n + 2\sum_{\substack{i=2 \\ i \text{ even}}}^{n-2} f_i + 4\sum_{\substack{i=1 \\ i \text{ odd}}}^{n-1} f_i\right] \end{aligned}$$

This is known as the **Simpson's 1/3 rule** of order n denoted by $S_{1/3}(f)$. Hence,

$$S_{1/3}(f) = \frac{h}{3}\left[f_0 + f_n + 2\sum_{\substack{i=2 \\ i \text{ even}}}^{n-2} f_i + 4\sum_{\substack{i=1 \\ i \text{ odd}}}^{n-1} f_i\right]$$

Since the local error for each subinterval is $-\frac{h^5}{90}f^{(4)}(\xi)$where $x_{i-1} \leq \xi \leq x_i$ and $i = 1, 2, \ldots n$, the total error is

$$-\frac{h^5}{90}\sum_{i=1}^{n} f^{(4)}(\xi_i).$$

If $f^{(4)}$ is continuous on $[a, b]$, the total error is $-\frac{h^5}{90}f^{(4)}(\xi)(\frac{n}{2})$. Since $h = \frac{b-a}{n}$, we have $b - a = hn$ and hence,

$$\text{Error} = -\frac{(b-a)h^4}{180}f^{(4)}(\xi).$$

Example 5.5. Find an estimate for $\int_0^1 \frac{dx}{1+x}$ using Simpson's 1/3 rule of order 12.

Solution. We have $h = \frac{b-a}{n} = \frac{1}{12}$. The values that will be used are given in the table in example 5.4.

$$\begin{aligned} S_{1/3}(f) &= \frac{h}{3}\left[f_0 + f_n + 2\sum_{\substack{i=2 \\ i \text{ even}}}^{n-2} f_i + 4\sum_{\substack{i=1 \\ i \text{ odd}}}^{n-1} f_i\right] \\ &= \frac{1}{36}\left[1 + \frac{1}{2} + 2\left(+\frac{6}{7} + \frac{3}{4} + \frac{2}{3} + \frac{3}{5} + \frac{6}{11}\right) + 4\left(\frac{12}{13} + \frac{4}{5} + \frac{12}{17} + \frac{12}{19} + \frac{4}{7} + \frac{12}{23}\right)\right] \\ &\approx 0.69314866 \end{aligned}$$

The result is accurate to four decimal places which is significantly better compared to the approximation given by the trapezoidal rule.

5.4.3 Simpson's 3/8 Rule

Subdivide $[a, b]$ into $n = 3k$ subintervals of width h. Applying Newton-Cotes formula of order three to each of the subinterval, we have

$$\begin{aligned}
\int_a^b f(x)dx &= \int_{x_0}^{x_n} f(x)\ dx \\
&\approx \int_{x_0}^{x_3} f(x)\ dx + \int_{x_3}^{x_6} f(x)\ dx + \ldots + \int_{x_{n-3}}^{x_n} f(x)\ dx \\
&= 3\frac{h}{8}(f_0 + 3f_1 + 3f_2 + f_3) + 3\frac{h}{8}(f_3 + 3f_4 + 3f_5 + f_6) + \\
&\quad \ldots + 3\frac{h}{8}(f_{n-3} + 3f_{n-2} + 3f_{n-1}f_n) \\
&= 3\frac{h}{8}\left[f_0 + f_n + 2\sum_{\substack{i=3 \\ i \text{ div. by } 3}}^{n-3} f_i \quad + 3\sum_{\substack{i=1 \\ i \text{ not div. by } 3}}^{n-1} f_i\right]
\end{aligned}$$

This is known as the **Simpson's 3/8 rule** of order n denoted by $S_{3/8}(f)$. Hence,

$$S_{3/8}(f) = 3\frac{h}{8}\left[f_0 + f_n + 2\sum_{\substack{i=3 \\ i \text{ div. by } 3}}^{n-3} f_i \quad + 3\sum_{\substack{i=1 \\ i \text{ not div. by } 3}}^{n-1} f_i\right]$$

Since the local error for each subinterval is $-\frac{3h^5}{80}f^{(4)}(\xi)$where $x_{i-1} \leq \xi \leq x_i$ and $i = 1, 2, \ldots n$, the total error is

$$-\frac{3h^5}{80}\sum_{i=1}^{n} f^{(4)}(\xi_i).$$

If $f^{(4)}$ is continuous on $[a, b]$, the total error is $-\frac{3h^5}{80}f^{(4)}(\xi)(\frac{n}{3})$. Since $h = \frac{b-a}{n}$, we have $b - a = hn$ and hence,

$$\text{Error} = -\frac{(b-a)h^4}{80}f^{(4)}(\xi).$$

Example 5.6. Find an estimate for $\int_0^1 \frac{dx}{1+x}$ using Simpson's 3/8 rule of order 12.

Solution. We have $h = \frac{b-a}{n} = \frac{1}{12}$. The values that will be used are given in the table in example 5.4.

$$\begin{aligned} S_{3/8}(f) &= 3\frac{h}{8}\left[f_0 + f_n + 2\sum_{\substack{i=3 \\ i \text{ div. by } 3}}^{n-3} f_i + 3\sum_{\substack{i=1 \\ i \text{ not div. by } 3}}^{n-1} f_i\right] \\ &= \frac{1}{32}\left[1 + \frac{1}{2} + 2\left(\frac{4}{5} + \frac{2}{3} + \frac{4}{7}\right) + 4\left(\frac{12}{13} + \frac{6}{7} + \frac{3}{4} + \frac{12}{17} + \frac{12}{19} + \frac{3}{5} + \frac{6}{11} + \frac{12}{23}\right)\right] \\ &\approx 0.69315046080 \end{aligned}$$

If the subinterval is not divisible by 2 or 3, the common practice is to use a combination of the Simpson's 1/3 rule and Simpson's 3/8 rule. Since the truncation error of the Simpson's 1/3 rule is smaller, it is advisable to use Simpson's 1/3 rule for as many subintervals as possible, then use Simpson's 3/8 rule for the three remaining subintervals where $f^{(4)}(x)$ is at a minimum.

Example 5.7. Find an estimate for $\displaystyle\int_0^1 \frac{dx}{1+x}$ using Simpson's 3/8 rule of order 11.

Solution. Use $S_{1/3}$ for the first 8 subintervals and $S_{3/8}$ for the remaining 3 since $f^{(4)}(x) = \dfrac{24}{(1+x)^5}$ is a decreasing function.

$$\begin{aligned} \int_0^1 f(x)\,dx &\approx \frac{1}{33}\left[1 + \frac{11}{19} + 2\left(\frac{11}{13} + \frac{11}{15} + \frac{11}{17}\right) + 4\left(\frac{11}{12} + \frac{11}{14} + \frac{11}{16} + \frac{11}{18}\right)\right] + \\ &\quad \frac{3}{88}\left[\frac{11}{19} + \frac{1}{2} + 3\left(\frac{11}{20} + \frac{11}{21}\right)\right] \\ &\approx 0.69314941 \end{aligned}$$

5.5 Gaussian Integration

All the Newton-Cotes formulas require that the values of the function whose integral is to approximated be known at evenly spaced points, which might be the expected situation if tabulated data for the functionwas being used. If the function is given explicitly, however, the points for evaluating the function could be chosen in another manner, which leads to increased accuracy of approximation. **Gaussian quadrature** is concerned with choosing the points for evaluation in an optimal manner. It presents a procedure for choosing values $x_0, x_1, \ldots, x_n$ in the interval $[a, b]$ and weights $A_0, A_1, \ldots A_n$, that are expected to minimize the error obtained in performing the approximation

$$\int_a^b f(x)\,dx \approx \sum_{j=0}^{n} A_j f(x_j)$$

for an arbitrary function f. In order to measure this accuracy, it is generally assumed that the best choice of these values will be the choice that maximizes the degree of precision for the formula.

Since the values of $A_0, A_1, \ldots A_n$ are arbitrary and those of $x_0, x_1, \ldots, x_n$ are restricted only in the sense that the function whose integral is being approximated must be defined at these points, there are $2n+2$ parameters involved, $(n+1)$ given by the weights $A_0, A_1, \ldots A_n$ and $(n+1)$ given by $x_0, x_1, \ldots, x_n$.

If the coefficients of a polynomial are also considered as parameters, the class of polynomials of degree at most $(2n+1)$ contains $2n+2$ parameters and is the largest class of polynomials for which it is reasonable to expect $\int_a^b f(x)\ dx \approx \sum_{j=0}^{n} A_j f(x_j)$ to be exact.

Example 5.8. Use Gaussian integration with two nodes to find an estimate for $\int_{-1}^{1} e^{-x^2}\ dx$.

Solution. Since there are four unknowns, we want a formula which is exact for all polynomials of degree ≤ 3.

Let $f_k = x^k$ where $k = 0, 1, 2, 3$ and if we set

$$\int_{-1}^{1} f_k\ dx = \sum_{j=0}^{n} A_j f(x_j)$$

then we obtain the following system of equations:

$$\begin{aligned}
\int_{-1}^{1} x^0\ dx = 2 &= A_0(1) + A_1(1) \\
\int_{-1}^{1} x\ dx = 0 &= A_0 x_0 + A_1 x_1 \\
\int_{-1}^{1} x^2\ dx = \frac{2}{3} &= A_0 x_0^2 + A_1 x_1^2 \\
\int_{-1}^{1} x^3\ dx = 0 &= A_0 x_0^3 + A_1 x_1^3
\end{aligned}$$

The system yields the unique solution $A_0 = A_1 = 1, x_0 = -\frac{\sqrt{3}}{3}$ and $x_1 = \frac{\sqrt{3}}{3}$. Thus,

$$\begin{aligned}
\int_{-1}^{1} e^{-x^2}\ dx &\approx A_0 f(x_0) + A_1 f(x_1) \\
&= exp\left(-\left(-\frac{\sqrt{3}}{3}\right)^2\right) + exp\left(-\left(\frac{\sqrt{3}}{3}\right)^2\right) \\
&\approx 1.4330626
\end{aligned}$$

From the preceding example, we can see that the process of solving for the weights and the nodes directly is long, and it becomes increasingly difficult as the value of n increases. For this reason, we need an alternative way of solving these unknowns. This would involve a class of polynomials known as **orthogonal polynomials**.

A class of polynomials $\{P_m(x)\}$ which satisfy the condition

$$\int_a^b P_i(x)P_j(x)\ dx = 0, \qquad \text{whenever } i \neq j$$

where the subscripts represent the degrees of the polynomials are said to be **orthogonal**. An important property of of orthogonal polynomials is as follows:

If $P(x)$ is any polynomial of degree less than m, then

$$\int_a^b P_m(x)P(x)\ dx = 0$$

where $P_m(x)$ is an element of the class of orthogonal polynomials. We wish to show that the roots of some orhogonal polynomials are exactly the required nodes of the Gaussian quadrature.

Let $P_{n+1}(x)$ be a polynomial of degree $(n+1)$ where $P_{n+1}(x)$ belongs to a class of orthogonal polynomials. Let $x_0, x_1, \ldots x_n$ be the $(n+1)$ roots of this polynomial. To show that these are the required nodes of the Gaussian quadratures, we must show that the resulting formula is exact for all polynomials of degree $\leq 2n+1$. Equivalently, the formula must be exact for all integrals of the form

$$\int_a^b x^k dx, \qquad k = 0, 1, 2, \ldots, 2n+1$$

Recall that

$$\int_a^b x^k dx = \sum_{j=0}^{n} A_j x_j^k + \int_a^b E(x)dx.$$

We then have

$$\int_a^b x^k dx = \sum_{j=0}^{n} A_j x_j^k + \int_a^b \frac{f_k^{(n+1)}(\xi)(x-x_0)(x-x_1)\cdots(x-x_n)\ dx}{(n+1)!}.$$

Since the degree of f_k is at most $2n+1$, the degree of $f_k^{(n+1)}$ is at most n. Since

$$\int_a^b P_m(x)P(x)\ dx = 0$$

where $P(x)$ is a polynomial of degree less than m, then

$$\int_a^b P_{n+1}(x)f_k^{(n+1)}(x)\ dx = 0$$

Since $x_0, x_1, \ldots x_n$ are roots of $P_{n+1}(x) = 0$, we can write

$$P_{n+1}(x) = c(x - x_0)(x - x_1)\cdots(x - x_n).$$

Thus,

$$c\int_a^b \frac{f_k^{(n+1)}(\xi)(x - x_0)(x - x_1)\cdots(x - x_n)\ dx}{(n+1)!} = 0$$

Since c is the leading coefficient of $P_{n+1}(x)$, it follows that $c \neq 0$, and hence

$$\int_a^b \frac{f_k^{(n+1)}(\xi)(x - x_0)(x - x_1)\cdots(x - x_n)\ dx}{(n+1)!} = 0 \qquad a < \xi < b.$$

This means that the error is zero, and hence the formula

$$\int_a^b f_k(x)dx = \sum_{j=0}^{n} A_j f_k(x_j)$$

is exact. Therefore, the nodes required by the Gaussian quadratures are the roots of the polynomial $P_{n+1}(x)$.

5.5.1 The Gauss-Legendre Formulas

The orthogonal polynomials $\{L_n(x)\}$ where

$$L_n(x) = \frac{1}{2^n n!}\frac{d^n(x^2 - 1)^n}{dx^n}$$

are called the **Legendre polynomials**. These polynomials are orthogonal on the interval $[-1, 1]$. If the $(n+1)$th degree Legendre polynomial is used in Gaussian integration, the resulting formula is known as the **Gauss-Legendre formula**.

The values of the weights A_j for the Gauss-Legendre formula may be computed using the formula

$$A_j = \frac{2(1 - x_j^2)}{(n+1)^2[L_n(x_j)]^2}.$$

Tables for the values of the weights and nodes for different values of n have been made to facilitate the implementation of Gaussian integration in the computer.

Since the Legendre polynomials are orthogonal in the interval $[-1, 1]$, the Gauss-Legendre formula is applied directly to integrals of the form $\int_{-1}^{1} f(x)\ dx$. For an arbitrary interval of integration $[a, b]$, the formula may be applied by first transforming the original integral $\int_a^b f(t)\ dt$ into an equivalent integral of the form $\int_{-1}^{1} g(x)\ dx$.

Using a linear function from $[-1, 1]$ to $[a, b]$ such that -1 and 1 correspond to a and b, respectively. We have

$$\begin{aligned} t &= \left(\frac{b-a}{2}\right)x + \frac{b+a}{2} \\ t &= \alpha x + \beta \end{aligned}$$

where $\alpha = \dfrac{b-a}{2}$ and $\beta = \dfrac{b+a}{2}$. Thus,

$$\begin{aligned} \int_a^b f(t)\ dt &= \alpha \int_{-1}^{1} f(\alpha x + \beta) dx \\ &\approx \alpha \sum_{j=0}^{n} A_j f(\alpha x_j + \beta) \\ &= \alpha \sum_{j=0}^{n} A_j f(t_j) \end{aligned}$$

where the x_j's and the A_j's are given in the Excel file named Weights and Nodes.

Example 5.9. Use the six-point Gauss-Legendre formula to find an estimate for $\displaystyle\int_1^2 e^{x^2}\ dx$.

Solution. Since $a = 1, b = 2$ then $\alpha = \dfrac{b-a}{2} = \dfrac{1}{2}$ and $\beta = \dfrac{b+a}{2} = \dfrac{3}{2}$. We obtain the following values:

x_j	t_j	A_j	$f(t_j)$	$A_j f(t_j)$
-0.932469514203	1.033765243	0.171324492379	2.911506304	0.49881234
-0.661209386466	1.169395307	0.360761573048	3.925467229	1.416157733
-0.238619186083	1.380690407	0.467913934573	6.728188906	3.148213344
0.238619186083	1.619309593	0.467913934573	13.76547379	6.441057004
0.661209386466	1.830604693	0.360761573048	28.53449034	10.29414762
0.932469514203	1.966234757	0.171324492379	47.75477777	8.181563061
			$\displaystyle\sum_{j=0}^{5} A_j f(t_j) =$	29.9799511

Hence, $\displaystyle\int_1^2 e^{x^2}\ dx \approx 14.98997555$.

5.5.2 The Gauss-Laguerre Formulas

The **Laguerre polynomials** are of the form

$$P_n(x) = e^x \frac{d^n(x^n e^{-x})}{dx^n}$$

are orthogonal in the interval $[0, +\infty)$. These polynomials are used to find estimate for the integrals of the form

$$\int_0^{+\infty} e^{-x} f(x)\ dx.$$

The nodes used in forming the Gaussian quadrature are the roots of $P_{n+1}(x)$, while the weights are computed using the formula

$$A_j = \frac{[(n+1)!]^2}{x_j[P'_{n+1}(x_j)]^2}.$$

The values for A_j and x_j for different values of n are given in the Excel file named Weights and Nodes.

Example 5.10. Use the ten-point Gauss-Laguerre formula to find an estimate for $\int_0^{\infty} e^{-x}\sqrt{x}\ dx$.

Solution. We have $f(x) = \sqrt{x}$. The following table is obtained.

x_j	A_j	$f(x_j)$	$A_j f(x_j)$
0.137793470540	0.308441115765	0.371205429	0.114495017
0.729454549503	0.401119929155	0.854081114	0.342588956
1.808342901740	0.218068287612	1.344746408	0.293246546
3.401433697850	0.062087456099	1.844297616	0.114507747
5.552496140060	0.009501516975	2.356373515	0.022389123
8.330152746760	0.000753008389	2.8862004	0.002173333
11.843785837900	0.000028259233	3.441480181	9.72536E-05
16.279257831400	0.000000424931	4.034756229	1.71449E-06
21.996585812000	0.000000001840	4.690051792	8.62765E-09
29.920697012300	0.000000000001	5.469981445	5.42175E-12
		$\sum_{j=0}^{9} A_j f(x_j) =$	0.889499699

Thus, $\int_0^{\infty} e^{-x}\sqrt{x}\ dx \approx 0.889499699$.

5.5.3 The Gauss-Hermite Formulas

The $(n+1)$th degree Hermite polynomial is defined as

$$H_{n+1}(x) = (-1)^{n+1} e^{x^2} \frac{d^{n+1}(e^{-x^2})}{dx^{n+1}}$$

and its roots are the nodes used in the Gauss-Hermite formulas. The weights are computed using the formula

$$A_j = \frac{2^{n+2}(n+1)!\sqrt{\pi}}{[H'_{n+1}(x_j)]^2}.$$

The Gauss-Hermite formulas are applied to integrals of the form

$$\int_{-\infty}^{\infty} e^{-x^2} f(x)\ dx.$$

The Hermite polynomials are orthogonal over the interval $(-\infty, +\infty)$

Example 5.11. Use the ten-point Gauss-Hermite formula to find an estimate for $\int_{-\infty}^{\infty} e^{-x^2} \cos x\ dx$.

Solution. We have $f(x) = \cos x$. The following table is obtained.

x_j	A_j	$f(x_j)$	$A_j f(x_j)$
-3.436159118840	0.000007640433	-0.956928098	-7.31134E-06
-2.532731674230	0.001343645747	-0.820299994	-0.001102193
-1.756683649300	0.033874394456	-0.184818642	-0.00626062
-1.036610829790	0.240138611082	0.509140179	0.122264215
-0.342901327224	0.610862633735	0.941783144	0.575300132
0.342901327224	0.610862633735	0.941783144	0.575300132
1.036610829790	0.240138611082	0.509140179	0.122264215
1.756683649300	0.033874394456	-0.184818642	-0.00626062
2.532731674230	0.001343645747	-0.820299994	-0.001102193
3.436159118840	0.000007640433	-0.956928098	-7.31134E-06
		$\sum_{j=0}^{9} A_j f(x_j) =$	1.380388447

Thus, $\int_{-\infty}^{\infty} e^{-x^2} \cos x\ dx \approx 1.380388447$.

5.5.4 The Gauss-Chebyshev Formulas

The $(n+1)$th degree Chebyshev polynomial is defined as

$$T_{n+1}(x) = \cos[(n+1)\cos^{-1} x].$$

These polynomials are orthogonal over the interval $[-1, 1]$. If we apply the addition formula for trigonometric function,

$$\begin{aligned}
T_{n+1}(x) &= \cos[(n+1)\cos^{-1} x] \\
&= \cos(n\cos^{-1} x + \cos^{-1} x)) \\
&= \cos(n\cos^{-1} x)\cos(\cos^{-1} x) - \sin(n\cos^{-1} x)\sin(\cos^{-1} x) \\
&= x\cos(n\cos^{-1} x) - \frac{1}{2}\cos((n-1)\cos^{-1} x) + \frac{1}{2}\cos((n+1)\cos^{-1} x) \\
&= xT_n(x) - \frac{1}{2}T_{n-1}(x) + \frac{1}{2}T_{n+1}(x)
\end{aligned}$$

and we obtain the recursive formula

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x), \qquad n = 1, 2, \ldots$$

with $T_0(x) = 1, T_1(x) = x$. The roots of the $(n+1)$th degree Chebyshev polynomial are used as nodes to generate the **Gauss-Chebyshev quadrature**. This formula is used to obtain approximations for integrals of the form

$$\int_{-1}^{1} \frac{f(x)\ dx}{\sqrt{1-x^2}}.$$

The weights of this quadrature are identically $\dfrac{\pi}{n+1}$.

Example 5.12. Use the three-point Gauss-Chebyshev formula to find an estimate for $\displaystyle\int_{-1}^{1} \frac{x\ dx}{\sqrt{1-x^2}}$.

Solution. We have $f(x) = x$.

$$\begin{aligned} T_3(x) &= 2xT_2(x) - T_1(x) \\ &= 2x[2xT_1(x) - T_0(x)] - x \\ &= 2x[2x(x) - 1] - x \\ &= 4x^3 - 3x \\ &= x(4x^2 - 3) \end{aligned}$$

The roots of $T_3(x)$ are 0 and $\pm\dfrac{\sqrt{3}}{2}$, and $\dfrac{\pi}{3}$ is the uniform weight. Thus,

$$\int_{-1}^{1} \frac{x\ dx}{\sqrt{1-x^2}} = \frac{\pi}{3}(x_0 + x_1 + x_2) = 0.$$

5.6 Romberg Integration

Romberg Integration is a method that has wide application because it uses the trapezoidal rule repeatedly to give preliminary approximations. With each repeated application, the number of subintervals is doubled which reduces the error to approximately one-furth the original balue. Thus, better approximations are obtained. An extrapolation technique is then used to obtain improvements of the approximations.

Let T denote the quadrature obtained by applying the trapezoidal rule to $n = 2^i$ subintervals. Thus we have $h = \dfrac{b-a}{2^i}$ and

$$T_{i,0} = \frac{b-a}{2^i}\left[\frac{1}{2}(f_0 + f_n) + \sum_{k=1}^{n-1} f_k\right].$$

If we let

$$I = \int_a^b f(x)\ dx \text{ and } E_i = I - T_{i,0}$$

then $E_i \approx \frac{1}{4}E_{i-1}$. We have $I = T_{i,0} + E_i \approx T_{i,0} + \frac{1}{4}E_{i-1} = T_{i,0} + \frac{1}{4}(I - T_{i-1,0})$. Thus,

$$I \approx \frac{4T_{i,0} - T_{i-1,0}}{3}.$$

This means that if we have $(m+1)$ trapezoidal quadratures $T_{0,0}, T_{1,0}, T_{2,0}, \ldots, T_{m,0}$, then we can use extrapolation to obtain a new and improved set of approximations for the definite integral by using the above approximation for I. We define this set of new approximations as follows:

$$T_{i,1} = \frac{4T_{i,0} - T_{i-1,0}}{3}, \qquad i = 1, 2 \ldots m.$$

Since the error of this quadrature is proportional to h^4, bisecting the subintervals once more will reduce the error approximately 1/16 of the previous error; i.e. $E_i \approx \frac{1}{16}E_{i-1}$. This gives us $I = T_{i,1} + E_i \approx T_{i,1} + \frac{1}{16}I - T_{i-1,1}$. Thus,

$$I \approx \frac{16T_{i,1} - T_{i-1,1}}{15}.$$

which is a third set of approximations for the integral. We define the elements of this set as follows:

$$T_{i,2} = \frac{16T_{i,1} - T_{i-1,1}}{15} = \frac{4^2T_{i,1} - T_{i-1,1}}{4^2 - 1}, \qquad i = 1, 2 \ldots m.$$

If the process of extrapolatin is carried out m times, then only one approximation will remain at the end, namely, $T_{m,m}$ which is the final value returned by Romberg integration. Its value is given by

$$T_{m,m} = \frac{4^mT_{m,m-1} - T_{m-1,m-1}}{4^m - 1}, \qquad i = 1, 2 \ldots m.$$

The following table summarizes the values that are used in the computation of the Romberg integral.

i	$T_{i,0}$	$T_{i,1}$	$T_{i,2}$	$\ldots$	$T_{i,m}$
0	$T_{0,0}$				
1	$T_{1,0}$	$T_{1,1}$			
2	$T_{2,0}$	$T_{2,1}$	$T_{2,2}$		
		$\vdots$			
m	$T_{m,0}$	$T_{m,1}$	$T_{m,2}$	$\ldots$	$T_{m,m}$

where $T_{i,j} = \frac{4^jT_{j,j-1} - T_{j-1,j-1}}{4^j - 1}$.

Example 5.13. Find an estimate for $\displaystyle\int_0^1 \frac{dx}{1+x^2}$ using Romberg integral of order three.

Solution. We have $f(x) = \dfrac{1}{1+x^2}$. First we compute the values of $T_{i,0}$.

The function values at $T_{0,0}$ where $h = 2^0$ is given by the following table:

x_j	f_j
0	1
1	1/2

Hence,

$$\begin{aligned} T_{0,0} &= \frac{1}{2}\left(1+\frac{1}{2}\right) \\ &= 0.75 \end{aligned}$$

The function values at $T_{1,0}$ where $h = 2^1$ is given by the following table:

x_j	f_j
0	1
1/2	4/5
1	1/2

Hence,

$$\begin{aligned} T_{1,0} &= \frac{1}{2}\left[\frac{1}{2}\left(1+\frac{1}{2}\right)+\frac{4}{5}\right] \\ &= 0.775 \end{aligned}$$

The function values at $T_{2,0}$ where $h = 2^2$ is given by the following table:

x_j	f_j
0	1
1/4	16/17
1/2	4/5
3/4	16/25
1	1/2

Hence,

$$\begin{aligned} T_{2,0} &= \frac{1}{2^2}\left[\frac{1}{2}\left(1+\frac{1}{2}\right)+\frac{4}{5}+\frac{16}{17}+\frac{16}{25}\right] \\ &= 0.782794118 \end{aligned}$$

The function values at $T_{3,0}$ where $h = 2^3$ is given by the following table:

x_j	f_j
0	1
1/8	64/65
1/4	16/17
3/8	64/73
1/2	4/5
5/8	64/89
3/4	16/25
7/8	64/113
1	1/2

Hence,

$$\begin{aligned} T_{3,0} &= \frac{1}{2^2}\left[\frac{1}{2}\left(1+\frac{1}{2}\right)+\frac{4}{5}+\frac{16}{17}+\frac{16}{25}+\frac{64}{65}+\frac{64}{73}+\frac{64}{89}+\frac{64}{113}\right] \\ &= 0.784747124 \end{aligned}$$

The following table summarizes the successive approximations.

i	$T_{i,0}$	$T_{i,1}$	$T_{i,2}$	$T_{i,3}$
0	0.75			
1	0.775	0.783333333		
2	0.782794118	0.785392157	0.785529412	
3	0.784747124	0.785398126	0.785398524	0.785396446

Thus, the Romberg integral of order three has a value of 0.785396446. The actual value is 0.785398163.

Exercises.

1. Find an estimate for $\int_{-1}^{2} f(x)\ dx$ using the Lagrange form of the interpolating polynomial, given the following data:

x_j	-2	-1	0	1
$f(x_j)$	-17	-8	-5	-2

2. Repeat Exercise 1 using the method of undetermined coefficients.

3. Use the Newton-Cotes formula of order three to approximate $\int_{1}^{3} e^{-x/2}\ dx$. Find a bound for the error.

4. Use the trapezoidal and Simpson's rules with $n = 12$ to approximate the following integrals. Compare the approximations to the actual value and find a bound for the error in each case.

   (a) $\int_{1}^{2} \ln x\ dx$

   (b) $\int_{0}^{0.1} x^{1/3}\ dx$

   (c) $\int_{0}^{\pi/3} (\sin x)^2\ dx$

   (d) $\int_{0.2}^{0.4} e^{3x} \cos 2x\ dx$

   (e) $\int_{0}^{\pi/4} \tan x\ dx$

   (f) $\int_{\pi/2}^{\pi/4} \cot x\ dx$

5. Use the eight-point Gauss-Legendre formula to find an estimate for $\int_{1}^{2} x^2 e^x\ dx$.

6. Approximate $\int_{0}^{\infty} \sin x e^{-x}\ dx$ using the most number of points available for the Gauss-Laguerre formula.

7. Find an approximation for the integral $\int_{-1}^{1} \frac{x\ dx}{\sqrt{1-x^2}}$ using the four-point Gauss-Chebyshev formula.

8. Find an estimate for $\int_{-\infty}^{\infty} xe^{-x^2}\ dx$ using the six-point Gauss Hermite formula.

9. Use the Romberg integral of order four to obtain an approximation for the integral
$$\int_1^3 e^x \sin x \; dx$$

Chapter 6

Ordinary Differential Equations

An ordinary differential equation contains at least one derivative of an unknown function with one variable. Together with an initial condition, we have an initial value problem.

Any textbook on the subject of ordinary differential equations details a number of methods for explicitly finding solutions to first-order initial-value problems, in practice few of the problems originating from the study of physical phenomena can be solved exactly.

This chapter is concerned with approximating the solution $y(x)$ to **initial value problems of order one** of the form

$$\frac{dy}{dx} = f(x, y) \qquad a \leq x \leq b,$$

subject to an initial condition

$$y(x_0) = y_0.$$

The methods include both **single-step** and **multi-step** methods.

For initial value problems of order one, we assume that the function f is continuous in some domain D which contains the initial point (x_0, y_0). Moreover, f is assumed to satisfy the **Lipschitz condition**, which requires that for every pair of points $(x, y_1), (x, y_2) \in D$ there is a nonnegative number k such that

$$|f(x, y_1) - f(x, y_2)| \leq k|y_1 - y_2| \text{ for every } x \in [a, b].$$

In this case, it is known that for any number y_0, the initial value problem

$$\frac{dy}{dx} = f(x, y), y(a) = y_0$$

has a unique solution $y = y(x)$ defined on the interval $[a, b]$.

Example 6.1. Let $f(x, y) = 2xy$. We have

$$|2xy_1 - 2xy_2| = |2x||y_1 - y_2| \leq |y_1 - y_2|$$

for all $x \in [-1, 1]$. this shows that f satisfies the Lipschitz condition on the interval $[-1, 1]$, with $k = 2$. Thus the initial value problem

$$\frac{dy}{dx} = 2xy, y(-1) = y_0$$

has a unique solution $y = y(x)$ for every real number y_0. It can be shown that the function defined by $y(x) = y_0 e^{x^2-1}$ is a solution to the above differential equation on the interval $[-1, 1]$.

6.1 One-Step Methods

6.1.1 Euler's Method

Although Euler's method is seldom used in practice, the simplicity of its derivation can be used to illustrate the techniques involved in the construction of some of the more advanced techniques, without the cumbersome algebra that accompanies these constructions.

As most of the numerical methods for solving initial value problems, this method consist of $(n+1)$ numbers $y_0, y_1, \ldots, y_n$ representing values of the approximate solution at the numers $x_0, x_1, \ldots, x_n$ where $x_0 < x_1 < \ldots < x_n$.

In this method, $n+1$ equispaced points $x_0, x_1, \ldots, x_n$ are chosen along the real line, with h as the uniform distance between consecutive points. The tangent line to the curve $y = y(x)$ at the point where $x = x_0$ is used to approximate $y(x)$ on the interval $[x_0, x_1]$. We then have

$$y'(x_0) = f(x_0, y_0) = \frac{y_1 - y_0}{x_1 - x_0} = \frac{y_1 - y_0}{h} \approx \frac{y(x_1) - y(x_0)}{h}.$$

Thus,

$$y(x_1) \approx y(x_0) + hf(x_0, y_0) = y_1.$$

From this, we obtain the algorithm for Euler's method, which is given by the formula

$$y_{i+1} = y_i + hf(x_i, y_i), \qquad i = 0, 1, \ldots n-1$$

We then apply the algorithm to each of the subintervals $[x_0, x_1], [x_1, x_2], \ldots, [x_{n-1}, x_n]$ to find the estimates $y_1, y_2, \ldots, y_n$, respectively.

Example 6.2. Use Euler's method to obtain an estimate for $y(0.3)$ given the initial value problem

$$y' = 2xy, y(0) = 1$$

Use a stepsize of $h = 0.1$. Compare the result with the actual value.

Solution. We generate the following table of values:

x_j	y_j	$f(x_j, y_j)$
0	1	0
0.1	1	0.2
0.2	1.02	0.408
0.3	1.0608	0.63648

The true solution to given initial value problem is $y = e^{x^2}$ so $y(0.3) = 1.094174284$.

6.1.2 The Taylor Series Method

Let $y(x)$ represent the true solution to the initial value problem

$$\frac{dy}{dx} = f(x, y), \qquad y(x_0) = y_0.$$

This solution may be expanded into a Taylor series about the point $x = x_0$, giving us

$$y(x) = y(x_0)+y'(x_0)(x-x_0)+\frac{y''(x_0)(x-x_0)^2}{2!}+\ldots+\frac{y^{(k)}(x_0)(x-x_0)^k}{k!}+\frac{y^{(k+1)}(\xi)(x-x_0)^{(k+1)}}{(k+1)!}$$

where $\xi \in (x_0, x)$.

Since $y' = f(x, y)$, the derivatives of $y(x)$ can be computed using the total derivatives of f. The first few derivatives are shown below:

$$\begin{aligned}
y' &= f \\
y'' &= f_x + f_y y' = f_x + f_y f \\
y''' &= f_{xx} + f_{xy}y' + f_{yx}y' + f_{yy}y'f + f_y f_x + f_y f_y y' \\
&= f_{xx} + f_{xy}f + f_{yx}f + f_{yy}f^2 + f_y f_x + (f_y)^2 f \\
&= f_{xx} + 2f_{xy}f + f_{yy}f^2 + f_y f_x + (f_y)^2 f
\end{aligned}$$

Observe that the derivatives become increasingly complicated as the order increases. For this reason, the series expansion above is usually truncated after a finite number of terms. If we truncate after the term containing the kth derivative of y, we obtain

$$y(x) \approx y(x_0) + y'(x_0)(x - x_0) + \frac{y''(x_0)(x - x_0)^2}{2!} + \ldots + \frac{y^{(k)}(x_0)(x - x_0)^k}{k!}$$

with the truncation error given by

$$\frac{y^{(k+1)}(\xi)(x - x_0)^{(k+1)}}{(k+1)!}$$

where $\xi \in (x_0, x)$. If $x = x_1$ we have

$$y(x_1) \approx y(x_0) + hy'(x_0) + \frac{h^2 y''(x_0)}{2!} + \ldots + \frac{h^k y^{(k)}(x_0)}{k!}.$$

This process may be repeated with a Taylor series expansion at $x = x_1$ which will generate an estimate for $y(x_2)$; a Taylor series expansion at $x = x_2$ which will generate an estimate for $y(x_3)$, and so on. From this process we come up with with the iteration equation

$$y_{j+1} \approx y_j + hy'_j + \frac{h^2 y''_j}{2!} + \ldots + \frac{h^k y_j^{(k)}}{k!}.$$

The method which makes use of this iteration equation to generate approximate solutions to an initial value problem is known as the **Taylor series method of order k.** If $k = 1$, we obtain the equation for Euler's method. Thus the local truncation error for Euler's method can be computed from

$$\frac{y^{(k+1)}(\xi)(x - x_0)^{(k+1)}}{(k+1)!}$$

and is found to be

$$\frac{h^2 y''(\xi)}{2!}, \text{ where } \xi \in (x_j, x_{j+1}).$$

This gives the error when the estimate y_{j+1} is computed, assuming that y_j is exact.

Example 6.3. Use the Taylor series method of order 2 to find an estimate for $y(1.3)$ using $h = 0.05$, given the initial value problem

$$y' = y, \qquad y(1) = e.$$

Solution. We have $y' = y, y'' = y'$. Our iteration equation for the given initial value problem is

$$y_{j+1} \approx y_j + hy_j + \frac{h^2 y'_j}{2!}.$$

Using this equation, the following values are generated:

x_j	y_j
1	2.718281828
1.05	2.857593772
1.1	3.004045453
1.15	3.158002782
1.2	3.319850425
1.25	3.489992759
1.3	3.668854888

The approximate value is 3.668854888 while the actual value is $e^{1.3} = 3.669296668$.

6.1.3 The Runge-Kutta Methods

The simplest numerical method for solving initial value problems is Euler's method but it requires a very small stepsize h in order to obtain an accurate approximation. On the other hand, the Taylor series method of high order k yield very accurate estimates but are difficult to implement on the computer because of the need to compute the derivatives of f.

A class of methods which yield estimates whose accuracy is comparable to those of the estimates from the Taylor series method but does not require the computation and evaluation of derivatives is known as the **Runge-Kutta methods**.

Recall that in Euler's method where

$$y_{j+1} = y_j + hf(x_j, y_j)$$

the value of f at (x_j, y_j) is used to generate the estimate y_{j+1}. In the Runge-Kutta methods, several points near (x_j, y_j), possibly including (x_j, y_j) itself, are chosen, and the function f is evaluated at these points. A weighted average of these function values is then used to compute the value of y_{j+1}. The order of the Runge-Kutta thus obtained represents the number of points that are chosen. The weights assigned to the function values are chosen in such a way that the resulting Runge-Kutta formula will agree as much as possible with the corresponding Taylor series equation of the same order.

As an example, we will derive the Runge-Kutta formula of order two. Let the points be denoted by (x_j, y_j) and $(x_j + \alpha h, y_j + \alpha h f(x_j, y_j))$, where h is the stepsize and α is still to be determined. The formula for y_{j+1} then takes the form

$$y_{j+1} = y_j + h[A_1 f(x_j, y_j) + [A_2 f(x_j + \alpha h, y_j + \alpha h f(x_j, y_j))]$$

A_1 and A_2 represent the weights. The values of A_1, A_2 and α will be chosen so that the above equation will agree as much as possible with the Taylor series formula of order two. Using Taylor series expansion where $f(x, y)$ is expanded about the point (a, b)

$$f(x, y) = f(a, b) + f_x(a, b)(x - a) + f_y(a, b)(y - b) + E,$$

we have

$$f(x_j + \alpha h, y_j + \alpha h f(x_j, y_j)) = f(x_j, y_j) + \alpha h f_x(x_j, y_j) + \alpha h f(x_j, y_j) f_y(x_j, y_j) + E$$

where E represents the remainder of the expansion. By substitution, we have

$$\begin{aligned} y_{j+1} &= y_j + h[A_1 f(x_j, y_j) + A_2 f(x_j + \alpha h, y_j + \alpha h f(x_j, y_j)] \\ &= y_j + h\{A_1 f(x_j, y_j) + A_2[f(x_j, y_j) + \alpha h f_x(x_j, y_j) + \alpha h f(x_j, y_j) f_y(x_j, y_j) + E] \\ &= y_j + h(A_1 + A_2) f(x_j, y_j) + A_2 h^2[\alpha f_x + \alpha f_y f(x_j, y_j) + E] \end{aligned}$$

The formula for the Taylor series method of order two is as follows:

$$y_{j+1} = y_j + hf(x_j, y_j) + \frac{h^2}{2}[f_x + f_y f(x_j, y_j)]$$

If we equate the coefficients of similar terms, we then obtain the following:

$$A_1 + A_2 = 1, A_2\alpha = \frac{1}{2}$$

Since we have three unknowns and only two equations, we have one free variable and there are infinitely many solutions. If we set α as a free variable we obtain

$$A_2 = \frac{1}{2}\alpha \text{ and } A_1 = 1 - \frac{1}{2}\alpha.$$

If we choose $\alpha = \frac{1}{2}$, we get $A_1 = 0, A_2 = 1$ and we obtain the equation

$$y_{j+1} = y_j + hf\left(x_j + \frac{h}{2}, y_j + \frac{h}{2}f(x_j, y_j)\right)$$

which is known as the **modified Euler's method**.

If $\alpha = 1$, we get $A_1 = A_2 = \frac{1}{2}$ and we obtain the equation

$$y_{j+1} = y_j + \frac{h}{2}\left[f(x_j, y_j) + f(x_{j+1}, y_j + hf(x_j, y_j))\right]$$

which is known as the **improved Euler's method** or **Heun's method**.

Formulas for the Runge-Kutta methods of other orders may be determined in a similar way. The most popular and perhaps the most stable among the Runge-Kutta formula of order four defined by

$$y_{j+1} = y_j + \frac{h}{6}\left[k_1 + 2k_2 + 2k_3 + k_4\right]$$

where

$$\begin{aligned} k_1 &= f(x_j, y_j) \\ k_2 &= f\left(x_j + \frac{h}{2}, y_j + \frac{h}{2}k_1\right) \\ k_3 &= f\left(x_j + \frac{h}{2}, y_j + \frac{h}{2}k_2\right) \\ k_4 &= f(x_j + h, y_j + hk_3) \end{aligned}$$

Example 6.4. Use the the modified Euler's method, improved Euler's method, and the Runge-Kutta method of order four to find an estimate for $y(1)$ using $h = 0.1$, given the initial value problem

$$y' = 2xy, \qquad y(0) = 1.$$

Solution.

1. Modified Euler's Method

x_j	y_j
0	1
0.1	1.01
0.2	1.040603
0.3	1.093673753
0.4	1.172527631
0.5	1.282276217
0.6	1.43037912
0.7	1.627485363
0.8	1.888696763
0.9	2.235461489
1	2.698425563

Using this method, $y(1) \approx 2.698425563$.

2. Improved Euler's Method

x_j	y_j
0	1
0.1	1.01
0.2	1.040704
0.3	1.093988045
0.4	1.173192779
0.5	1.2834729
0.6	1.432355757
0.7	1.630593794
0.8	1.893445513
0.9	2.242596866
1	2.709057014

Using this method, $y(1) \approx 2.709057014$.

3. Runge-Kutta Method of Order Four

x_j	y_j	k_1	k_2	k_3	k_4
0	1	0	0.1	0.1005	0.20201
0.1	1.010050167	0.202010033	0.306045201	0.307605728	0.416324296
0.2	1.04081077	0.416324308	0.530813493	0.533675722	0.656507005
0.3	1.094174265	0.656504559	0.788899645	0.793533473	0.93882209
0.4	1.173510814	0.938808651	1.098406122	1.105588008	1.284069614
0.5	1.284025256	1.284025256	1.48304917	1.493995486	1.720109765
0.6	1.433328995	1.719994793	1.975127354	1.991710971	2.285500128
0.7	1.632315187	2.285241262	2.619865876	2.644962722	3.034898335
0.8	1.896478467	3.034365548	3.481934466	3.519977824	4.047257249
0.9	2.24790259	4.046224662	4.655406264	4.713278517	5.438460884
1	2.718270175	5.436540351	6.279204105	6.367683799	7.381084822

Using this method, $y(1) \approx 2.718270175$.

The actual value is $e = 2.718281828$. Hence, the Runge-Kutta method is the most accurate estimate among the three methods.

6.2 Multistep Methods

In each of the methods we have discussed so far, the algorithm enables us to compute the value of each new estimate y_{j+1} using only the current estimate y_j. In this section, we will describe several **multistep methods**, which compute y_{j+1} using k back values $y_j, y_{j-1}, \ldots, y_{j-k+1}$. The number of "steps" refers to the number of back values used in the formulas.

Unlike one-step methods, multistep methods are not "self-starting", since they require the use of a one-step method to generate the initial set of back values needed to start the implementation of these methods. Inspite of this drawback, multistep methods continue to be popular because they yield highly accurate results and use relatively simple formulas which are well suited for computer implementation.

6.2.1 The Adams Method

The **Adams method** is one of the most popular multistep method. It uses two formulas which are used in pairs. The first formula, known as the **Adams-Bashforth predictor formula**, is used to generate a preliminary estimate for y_{j+1} which is denoted by $y_{j+1}^{(p)}$ and is reffered to as the **predicted value** of y_{j+1} . the second formula which is known as the **Adams-Moulton corrector formula** then uses this preliminary estimate to obtain an improved estimate denoted by $y_{j+1}^{(c)}$, called the **corrected value** of y_{j+1} .

The Adams formuals re derived by making use of the fact that

$$\begin{aligned} y(x_{j+1}) - y(x_j) &= \int_{x_j}^{x_{j+1}} y'(x)\, dx \\ &= \int_{x_j}^{x_{j+1}} f(x, y(x))\, dx \end{aligned}$$

Since the function $y(x)$ is unknown, the integral on the right-hand side of the above equation can not be directly evaluated. To obtain an estimate for its value, f is replaced by its interpolating polynomial. For a k-step Adams-Bashforth formula, the nodes used are equispaced points $x_j, x_{j-1}, \ldots, x_{j-k+1}$, and Newton's backward difference formula is used for the interpolating polynomial. This gives us

$$y_{j+1}^{(p)} - y_j = h \int_0^1 \left[f_j + r\nabla f_j + \binom{r+1}{2} \nabla^2 f_j + \ldots + \binom{r+k-2}{k-1} \nabla^{k-1} f_j \right] dr$$

where $r = \dfrac{(x - x_j)}{h}$ and $f_j = f(x_j, y_j^{(c)})$. This is equivalent to the k-step Adams-Bashforth predictor formula, given by

$$y_{j+1}^{(p)} = y_j + h \int_0^1 \left[f_j + r\nabla f_j + \binom{r+1}{2} \nabla^2 f_j + \ldots + \binom{r+k-2}{k-1} \nabla^{k-1} f_j \right] dr$$

After obtaining the value of $y_{j+1}^{(p)}$ from the above equation, we may now compute $f_{j+1} = f(x_{j+1}, y_{j+1}^{(p)})$. This will allow us to advance one node forward and use the nodes $x_{j+1}, x_j, \ldots, x_{j-k+2}$ for Newton's backward difference formula. We obtain the k-step Adams-Moulton formula

$$y_{j+1}^{(c)} - y_j = h \int_{-1}^{0} \left[f_{j+1} + r\nabla f_{j+1} + \binom{r+1}{2} \nabla^2 f_{j+1} + \ldots + \binom{r+k-2}{k-1} \nabla^{k-1} f_{j+1} \right] dr$$

or equivalently,

$$y_{j+1}^{(c)} = y_j + h \int_{-1}^{0} \left[f_{j+1} + r\nabla f_{j+1} + \binom{r+1}{2} \nabla^2 f_{j+1} + \ldots + \binom{r+k-2}{k-1} \nabla^{k-1} f_{j+1} \right] dr$$

where $r = \dfrac{(x - x_{j+1})}{h}$

Example 6.5. Derive the 4-step Adams formulas.

Solution.

Adam's Bashforth Formula:

$$\begin{aligned}
y_{j+1}^{(p)} &= y_j + h \int_0^1 \left[f_j + r(f_j - f_{j-1}) + \frac{r(r+1)}{2!}(f_j - 2f_{j-1} + f_{j-2}) + \right. \\
&\quad \left. \frac{r(r+1)(r+3)}{3!}(f_j - 3f_{j-1} + 3f_{j-2} - f_{j-3}) \right] dr \\
&= y_j + h \left[rf_j + \frac{r^2}{2}(f_j - f_{j-1}) + \left(\frac{r^3}{6} + \frac{r^2}{4} \right)(f_j - 2f_{j-1} + f_{j-2}) + \right. \\
&\quad \left. \left(\frac{r^4}{24} + \frac{r^3}{6} + \frac{r^2}{6} \right)(f_j - 3f_{j-1} + 3f_{j-2} - f_{j-3}) \right]_0^1 \\
&= y_j + h \left[f_j + \frac{1}{2}(f_j - f_{j-1}) + \left(\frac{5}{12} \right)(f_j - 2f_{j-1} + f_{j-2}) + \right. \\
&\quad \left. \left(\frac{3}{8} \right)(f_j - 3f_{j-1} + 3f_{j-2} - f_{j-3}) \right] \\
&= y_j + h \left[f_j \left(1 + \frac{1}{2} + \frac{5}{12} + \frac{3}{8} \right) f_{j-1} \left(-\frac{1}{2} - \frac{5}{6} - \frac{9}{8} \right) \right. \\
&\quad \left. + f_{j-2} \left(\frac{5}{12} + \frac{9}{8} \right) + f_{j-3} \left(-\frac{3}{8} \right) \right] \\
&= y_j + h \left[\frac{55}{24} f_j - \frac{59}{24} f_{j-1} + \frac{37}{24} f_{j-2} - \frac{3}{8} f_{j-3} \right] \\
&= y_j + \frac{h}{24} \left[55f_j - 59f_{j-1} + 37f_{j-2} - 9f_{j-3} \right]
\end{aligned}$$

Similarly, we can show the 4-step Adams-Moulton formula:

$$y_{j+1}^{(c)} = y_j + \frac{h}{24}\left[9f_{j+1} + 19f_j - 5f_{j-1} + f_{j-2}\right]$$

The Adams formulas for $k = 1, 2, 3, 4$ are shown below:

1. The first four Adams-Bashforth formulas are shown below:

   (a) one step: $y_{j+1}^{(p)} = y_j + hf_j$

   (b) two step: $y_{j+1}^{(p)} = y_j + \frac{h}{2}(3f_j - f_{j-1})$

   (c) three step: $y_{j+1}^{(p)} = y_j + \frac{h}{12}(23f_j - 16f_{j-1} + 5f_{j-2})$

   (d) four step: $y_{j+1}^{(p)} = y_j + \frac{h}{24}(55f_j - 59f_{j-1} + 37f_{j-2} - 9f_{j-3})$

2. The first four Adams-Moulton formulas are shown below:

   (a) one step: $y_{j+1}^{(c)} = y_j + hf_{j+1}$

   (b) two step: $y_{j+1}^{(c)} = y_j + \frac{h}{2}(f_{j+1} - f_j)$

   (c) three step: $y_{j+1}^{(c)} = y_j + \frac{h}{12}(5f_{j+1} + 8f_j - f_{j-1})$

   (d) four step: $y_{j+1}^{(c)} = y_j + \frac{h}{24}(9f_{j+1} + 19f_j - 5f_{j-1} + f_{j-2})$

Example 6.6. Use the 3-step Adams formulas to estimate $y(1)$ with $h = 0.1$ if $y' = -2y, y(0) = 1$. Use the Taylor series method of order 3 to generate the required starting values.

Solution. The formula for the Taylor series method of order 3 is given by

$$y_{j+1} = y_j + y_j'h + \frac{y_j''h^2}{2!} + \frac{y_j'''h^3}{3!}.$$

Since $y' = -2y$ then $y'' = -2y' = 4y$ and $y''' = 4y' = -8y$. We generate the following table:

x_j	$y_j^{(c)}$	$y_j^{(p)}$	$f_j^{(c)}$	$f_j^{(p)}$
0	1		-2	
0.1	0.818666667		-1.637333333	
0.2	0.670215111		-1.340430222	
0.3	0.548807783	0.548277096	-1.097615565	-1.096554193
0.4	0.449392557	0.448933273	-0.898785115	-0.897866547
0.5	0.367985105	0.367622893	-0.73597021	-0.735245786
0.6	0.301324619	0.301028182	-0.602649237	-0.602056363
0.7	0.246739685	0.24649683	-0.493479371	-0.492993659
0.8	0.202042809	0.201843945	-0.404085618	-0.403687891
0.9	0.165442768	0.16527993	-0.330885536	-0.33055986
1	0.135472822	0.135339482	-0.270945645	-0.270678965

Hence, $y(1) \approx 0.135472822$ while the actual value is $e^{-2} = 0.135335283$.

6.2.2 Milne's Method

Like the Adams method, Milne's method uses a pair of **predictor-corrector equations** to obtain approximate solutions to an initial value problem. The derivation of these formulas is also done by replacing $f(x, y)$ by its interpolating polynomial and evaluating the integrals on both sides of the resulting integral equation.

The main difference between Milne's method and the Adams method is the choice of nodes and the use of Newton's forward difference formula instead of the backward difference formula.

From the initial value problem

$$y' = f(x, y), y(x_0) = y_0$$

we form the integral equation

$$\int_{x_{j-3}}^{x_{j+1}} y' \, dx = \int_{x_{j-3}}^{x_{j+1}} f(x, y) \, dx$$

which is equivalent to

$$y(x_{j+1}) = y(x_{j-3}) + \int_{x_{j-3}}^{x_{j+1}} f(x, y) \, dx$$

If we replace f by its interpolating polynomial over the nodes x_{j-2}, x_{j-1} and x_j, we

will obtain

$$\begin{aligned}
y(x_{j+1}) &\approx y(x_{j-3}) + h\int_{-1}^{3}\left(f_{j-2} + r\Delta f_{j-2} + \frac{r^2-r}{2}\Delta^2 f_{j-2}\right)\,dr \\
&= y(x_{j-3}) + h\left[rf_{j-2} + \frac{r^2}{2}(f_{j-1}-f_{j-2}) + \left(\frac{r^3}{6}-\frac{r^2}{4}\right)(f_j - 2f_{j-1}+f_{j-2})\right]_{-1}^{3} \\
&= y(x_{j-3}) + h\left\{\left[3f_{j-2} + \frac{9}{2}(f_{j-1}-f_{j-2}) + \frac{9}{4}(f_j-2f_{j-1}+f_{j-2})\right]\right. \\
&\quad \left. -\left[-f_{j-2} + \frac{1}{2}(f_{j-1}-f_{j-2}) - \frac{5}{12}(f_j-2f_{j-1}+f_{j-2})\right]\right\} \\
&= y(x_{j-3}) + h\left\{\left(\frac{9}{4}+\frac{5}{12}\right)f_j + \left(\frac{9}{2}-\frac{9}{2}-\frac{1}{2}-\frac{5}{6}\right)f_{j-1}\right. \\
&\quad \left. + \left(3-\frac{9}{2}+\frac{9}{4}+1+\frac{1}{2}\frac{5}{12}\right)f_{j-2}\right\} \\
&= y(x_{j-3}) + h\left(\frac{8}{3}f_j - \frac{4}{3}f_{j-1} + \frac{8}{3}f_{j-2}\right)
\end{aligned}$$

from which we get the predictor formula

$$y_{j+1}^{(p)} = y_{j-3} + \frac{4h}{3}(2f_j - f_{j-1} + 2f_{j-2})$$

For the corrector formula, we form the integral equation

$$\int_{x_{j-1}}^{x_{j+1}} y'\,dx = \int_{x_{j-1}}^{x_{j+1}} f(x,y)\,dx$$

which is equivalent to

$$y(x_{j+1}) = y(x_{j-1}) + \int_{x_{j-1}}^{x_{j+1}} f(x,y)\,dx$$

This time f is replaced by its interpolating polynomial over the nodes x_{j-1}, x_j and x_{j+1}. From this we have

$$\begin{aligned}
y(x_{j+1}) &\approx y(x_{j-1}) + h\int_{0}^{2}\left(f_{j-1} + r\Delta f_{j-1} + \frac{r^2-r}{2}\Delta^2 f_{j-1}\right)\,dr \\
&= y(x_{j-1}) + h\left[rf_{j-1} + \frac{r^2}{2}(f_j - f_{j-1}) + \left(\frac{r^3}{6}-\frac{r^2}{4}\right)(f_{j+1}-2f_j+f_{j-1})\right]_{0}^{2} \\
&= y(x_{j-1}) + h\left[2f_{j-1} + 2(f_j - f_{j-1}) + \frac{1}{3}(f_{j+1}-2f_j+f_{j-1})\right] \\
&= y(x_{j-1}) + h\left(\frac{1}{3}f_{j+1} + \frac{4}{3}f_j + \frac{1}{3}f_{j-1}\right)
\end{aligned}$$

from which we get the corrector formula

$$y_{j+1}^{(c)} = y_{j-1} + \frac{h}{3}(f_{j+1} + 4f_j + f_{j-1})$$

Example 6.7. Do example 6.6 using Milne's method. Use the Runge-Kutta method of order 4 for the initial values.

Solution. We generate the following table.

x_j	$y_j^{(c)}$	$y_j^{(p)}$	$f_j^{(c)}$	$f_j^{(p)}$
0	1		-2	
0.1	0.818733333		-1.637466667	
0.2	0.670324271		-1.340648542	
0.3	0.548816825		-1.09763365	
0.4	0.449325296	0.449393055	-0.898650592	-0.898786109
0.5	0.367879754	0.367938051	-0.735759508	-0.735876102
0.6	0.301186375	0.301239508	-0.602372749	-0.602479016
0.7	0.246595057	0.246645202	-0.493190115	-0.493290403
0.8	0.20188715	0.201921763	-0.4037743	-0.403843526
0.9	0.165296643	0.165332556	-0.330593286	-0.330665112
1	0.135325744	0.135347375	-0.270651488	-0.270694749

Hence, $y(1) \approx 0.135325744$ while the actual value is $e^{-2} = 0.135335283$.

Exercises.

1. Use Euler's method to approximate the solution for each of the following initial-value problems.

   (a) $y' = \left(\frac{y}{x}\right)^2 + \left(\frac{y}{x}\right), 1 \le x \le 1.2, y(1) = 1$ with $h = 0.01$

   (b) $y' = \sin x + e^{-x}, 0 \le x \le 1, y(0) = 0$ with $h = 0.01$

   (c) $y' = \frac{1}{x}(y^2 + y), 1 \le x \le 3, y(1) = -2$ with $h = 0.05$

   (d) $y' = -xy + \frac{4x}{y}, 0 \le x \le 1, y(0) = 1$ with $h = 0.02$

2. Repeat Exercise 1 using Taylor's method of order two.

3. Repeat Exercise 1 using modified Euler's method.

4. Repeat Exercise 1 using improved Euler's method.

5. Repeat Exercise 1 using Runge-Kutta method of order 4.

6. Derive the Adams-Bashforth and the Adams-Moulton formulas of order 1, 2 and 3.

7. Use the Adams formulas of order 4 to approximate the solutions to Exercise 1. Use Runge-Kutta method of order 4 for starting values.

8. Use the Milnes method to approximate the solutions to Exercise 1. Use the improved Euler's method for starting values.

Chapter 7

Eigenvalues and Eigenvectors

In your course in Linear Algebra, you were introduced to the concept of eigenvalues and eigenvectors. These concepts have many important applications. In certain cases, it is very difficult to determine the eigenvalues of a matrix. In this chapter, we will discuss numerical methods for approximating the eigenvalues of a matrix.

Before we do this, let us review some concepts from linear algebra that we will need in our discussion.

7.1 Review of Some Concepts From Linear Algebra

Let V be a real vector space, and let $S = \{\ v_1,\ v_2,\ \ldots,\ v)k\ \}$ be a subset of V. If there exist scalars (i.e. real numbers) $c_1,\ c_2,\ \ldots,\ c_k$, *at least one of which is not zero*, such that we can write

$$c_1v_1 + c_2v_2 + \cdots + c_kv_k = 0$$

then we say that the given set of vectors is *linearly dependent*. If, on the other hand, the only way the above equation can be satisfied is for $c_1 = c_2 = \cdots = c_k = 0$, then S is said to be *linearly independent*.

Example 7.1. Let $V = \mathbb{R}^3$ and let $S = \{\ (1,1,0),\ (1,0,1), (0,1,1)\ \} = \{\ v_1,\ V_2,\ V_3\ \}$. Suppose

$$\begin{aligned}(0,0,0) &= c_1v_1 + c_2v_2 + c_3v_3\\ &= c_1(1,1,0) + c_2(1,0,1) + c_3(0,1,1)\\ &= (c_1 + c_2,\ c_1 + c_3,\ c_2 + c_3)\end{aligned}$$

This gives us the linear system

$$c_1 + c_2 = 0 \qquad c_1 + c_3 = 0 \qquad c_2 + c_3 = 0$$

which has the unique solution

$$c_1 = c_2 = c_3 = 0$$

This shows that S is linearly independent.

If $S = \{ v_1,\ v_2,\ \ldots,\ v_n \}$ is a linearly independent set of n vectors in $\mathbb{R}^n$, then S is a *basis* for $\mathbb{R}^n$. This means that for every $v \in \mathbb{R}^n$, there exists a unique set of scalars $c_1,\ c_2,\ \ldots,\ c_n$ such that

$$v = c_1 v_1 + c_2 v_2 + \cdots + c_n v_n$$

Example 7.2. Consider the set S and the vector space $V = \mathbb{R}^3$ in Example 7.1. Since $n = 3$ and S is a linearly independent set of $n = 3$ vectors in V, it follows that S is a basis for $\mathbb{R}^3$. Moreover, if $v = (x, y, z)$ is any vector in $\mathbb{R}^3$, we can write it in the form

$$(x, y, z) = \frac{x + y - z}{2}(1, 1, 0) + \frac{x - y + z}{2}(1, 0, 1) + \frac{-x + y + z}{2}(0, 1, 1)$$

If A is a square matrix of order n, and λ is a real number for which there exists a *nonzero vector* $X \in \mathbb{R}^n$ such that

$$AX = \lambda X$$

then λ is called an *eigenvalue* of A, while X is called an *eigenvector* of A associated with the eigenvalue λ. The ordered pair (λ, X) is called an *eigenpair* of A.

Example 7.3. Let $A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$, $\lambda = 3$ and $X = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$.Then we have

$$AX = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 \\ 3 \end{bmatrix} = 3 \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \lambda X$$

which shows that $\lambda = 3$ is an eigenvalue of A and that X is an associated eigenvector for this eigenvalue. The ordered pair $(\lambda, X) = (3, (1\ \ 1)^T)$ is an eigenpair for the matrix A.

If A is a square matrix of order n, λ is an eigenvalue of A, and X is an eigenvector of A associated with λ, then we have

$$AX = \lambda X \Rightarrow (\lambda I_n - A)X = 0$$

which shows that X is a nontrivial solution of the homogeneous system $(\lambda I_n - A)X = 0$. This means that the coefficient matrix $\lambda I_n - A$ must be nonsingular, and its determinant is zero. Thus, if we define the *characteristic polynomial* of A to be

$$f_A(\lambda) = |\lambda I_n - A|$$

it follows that the eigenvalues of A are the zeros of its characteristic polynomial.

Example 7.4. Consider the matrix $A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$ from Example 7.3. The characteristic polynomial of this matrix is

$$\begin{aligned} f_A(\lambda) &= |\lambda I_2 - A| = \begin{vmatrix} \lambda - 2 & -1 \\ -1\lambda - 2 & \end{vmatrix} \\ &= (\lambda - 2)^2 - (-1)(-1) = \lambda^2 - 4\lambda + 3 \\ &= (\lambda - 3)(\lambda - 1) \end{aligned}$$

Thus the roots of the polynomial, 1 and 3, are the eigenvalues of A.

The following well known result from Linear Algebra will be used to prove the convergence of the estimates generated by one of our numerical methods for approximating eigenvalues and eigenvectors.

Theorem 7.1. *Let A be an $n \times n$ matrix. Eigenvectors associated with distinct eigenvalues of A form a linearly independent set.*

As a consequence of this theorem, suppose that the eigenvalues $\lambda_1, \ \ldots, \ \lambda_n$ of a matrix A are all real and distinct. Then there exists a set $S = \{ \ X_1, \ X_2, \ \ldots, \ X_n \ \}$ of linearly independent eigenvectors of A, where X_j is an eigenvector associated with the eigenvalue λ_j. This set then forms a basis of eigenvectors for $\mathbb{R}^n$.

When the order n of A is large, its characteristic polynomial will be of high degree. Since formulas for computing the roots of a polynomial exist only for $n \leq 4$, it will be difficult to determine the exact eigenvalues of such matrices. For this reason, we can use numerical methods to obtain reasonably accurate estimates for the eigenvalues.

7.2 The Power Method

In this section, we will discuss a numerical method that yields estimates for the *dominant eigenvalue* of a square matrix A. This is the eigenvalue with the largest magnitude or absolute value. If the eigenvalues of an $n \times n$ matrix are arranged according to magnitude, so that

$$|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n|$$

then the *power method* will yield an estimate for the dominant eigenvalue λ_1. An eigenvector associated with λ_1 is called a *dominant eigenvector*. An estimate for a dominant eigenvector X_1 corresponding to λ_1 will also be generated by the power method.

Definition 7.1. *An eigenvector $X \in \mathbb{R}^n$ is said to be* normalized *if the component of X with the largest magnitude is equal to 1.*

If $X = (x_1, \ x_2, \ \ldots, \ x_n)^T \in \mathbb{R}^n$ is a nonzero vector, then X can be normalized to the vector $\overline{X}$ by taking

$$\overline{X} = \frac{1}{c} X, \quad c = x_j, \quad |x_j| = \max\{ \ |x_1|, \ |x_2|, \ \ldots, \ |x_n| \ \}$$

Example 7.5. Let $A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$ be the matrix from Example 7.3. The eigenvalues of this matrix are 1 and 3, so $\lambda_1 = 3$ is the dominant eigenvalue. The eigenvectors associated with λ_1 are of the form $X_1 = (r, \ r)$ where r is a nonzero real number. In particular, $X_1 = (-2, -2)$ is an eigenvector associated with λ_1,and it is normalized to the vector $\overline{X_1} = (1, 1)$.

In order for the power method to work, it is assumed that the eigenvalues of the matrix A can be associated with corresponding eigenvectors $X_1, \ X_2, \ \ldots, \ X_n$ such that the set $S = \{ \ X_1, \ X_2, \ \ldots, \ X_n \ \}$ is a linearly independent set of vectors in $\mathbb{R}^n$. Moreover, the dominant eigenvalue λ_1 must be unique. This means that λ_1 is a root of multiplicity one of the characteristic polynomial.

7.2.1 The Power Method for Estimating the Dominant Eigenvalue

Let $X^{(0)}$ be a nonzero vector in $\mathbb{R}^n$. We generate the sequence $\{ X^{(k)} \}$ using the equations

$$\begin{aligned} Y^{(k)} &= AX^{(k)} \\ X^{(k+1)} &= \frac{1}{c_{k+1}} Y^{(k)} \end{aligned}$$

where c_{k+1} is the component of $Y^{(k)}$ with the largest magnitude. If there are two or more components with the largest magnitude, take c_{k+1} to be the component that comes first in $Y^{(k)}$. We will show that under certain conditions, the sequence $\{ X^{(k)} \}$ will converge to $\overline{X_1}$ and that the sequence $\{ c_k \}$ will converge to λ_1.

Example 7.6. Consider the matrix $\begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & 0 \\ 2 & 1 & 2 \end{bmatrix}$. We will use five iterations of the power method to obtain estimates for the dominant eigenvalue and a dominant eigenvector using $X^{(0)} = (1, 1, 1)^T$ as the initial estimate.

Solution.

$$\begin{aligned} Y^{(0)} &= AX^{(0)} = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & 0 \\ 2 & 1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 6 \\ 1 \\ 5 \end{bmatrix} = 6 \begin{bmatrix} 1 \\ \frac{1}{6} \\ \frac{5}{6} \end{bmatrix} = c_1 X^{(1)} \\ Y^{(1)} &= AX^{(1)} = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & 0 \\ 2 & 1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ \frac{1}{6} \\ \frac{5}{6} \end{bmatrix} = \begin{bmatrix} \frac{23}{6} \\ \frac{1}{6} \\ \frac{23}{6} \end{bmatrix} = \frac{23}{6} \begin{bmatrix} 1 \\ \frac{1}{23} \\ 1 \end{bmatrix} = c_2 X^{(2)} \\ Y^{(2)} &= AX^{(2)} = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & 0 \\ 2 & 1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ \frac{1}{23} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{94}{23} \\ \frac{1}{23} \\ \frac{93}{23} \end{bmatrix} = \frac{94}{23} \begin{bmatrix} 1 \\ \frac{1}{94} \\ \frac{93}{94} \end{bmatrix} = c_3 X^{(3)} \\ Y^{(3)} &= AX^{(3)} = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & 0 \\ 2 & 1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ \frac{1}{94} \\ \frac{93}{94} \end{bmatrix} = \begin{bmatrix} \frac{375}{94} \\ \frac{1}{94} \\ \frac{375}{94} \end{bmatrix} = \frac{375}{94} \begin{bmatrix} 1 \\ \frac{1}{375} \\ 1 \end{bmatrix} \\ &= c_4 X^{(4)} \\ Y^{(4)} &= AX^{(4)} = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & 0 \\ 2 & 1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ \frac{1}{375} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1502}{375} \\ \frac{1}{375} \\ \frac{1501}{375} \end{bmatrix} = \frac{1502}{375} \begin{bmatrix} 1 \\ \frac{1}{1502} \\ \frac{1501}{1502} \end{bmatrix} \\ &= c_5 X^{(5)} \end{aligned}$$

Our estimates are

$$\lambda_1 \approx c_5 = \frac{1502}{375} \qquad \overline{X_1} \approx X^{(5)} = \begin{pmatrix} 1 & \frac{1}{1502} & \frac{1501}{1502} \end{pmatrix}^T$$

It can be verified that $\lambda_1 = 4$ and $\overline{X_1} = (1 \;\; 0 \;\; 1)^T$, which shows that the estimates given by the power method compare well with the actual values.

The following theorem describes conditions under which the sequences $\{ c_k \}$ and $\{ X^{(k)} \}$ generated by the power method will converge to λ_1 and $\overline{X_1}$, respectively.

Theorem 7.2. *Let A be an $n \times n$ matrix with n distinct eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$ such that*

$$|\lambda_1| > |\lambda_2| > \cdots > |\lambda_n|$$

If $X^{(0)}$ is a nonzero vector in $\mathbb{R}^n$ such that $X^{(0)} \neq \overline{X_1}$, then the sequences $\{ c_k \}$ and $\{ X^{(k)=(x_1^{(k)} \;\; x_2^{(k)} \;\; \cdots \;\; x_n^{(k)})^T} \}$ generated recursively by the equations

$$\begin{aligned} Y^{(k)} &= AX^{(k)} \\ X^{(k+1)} &= \frac{1}{c_{k+1}} Y^{(k)} \end{aligned}$$

where

$$y_j^{(k)} = \max \{ |y_1^{(k)}|, |y_2^{(k)}|, \ldots, |y_n^{(k)}| \}, \qquad c_{k+1} = y_j^{(k)}$$

will converge to the dominant eigenvalue λ_1 and the associated normalized eigenvector $\overline{X_1}$.

Proof. Since A has distinct eigenvalues, there exists a set of n associated eigenvectors $\overline{X_1}, \overline{X_2}, \ldots, \overline{X_n}$ that are normalized and form a basis for $\mathbb{R}^n$. If $X^{(0)}$ is the initial vector used to start the power method, then we can write

$$X^{(0)} = a_1\overline{X_1} + a_2\overline{X_2} + \cdots + a_n\overline{X_n}$$

Without loss of generality, assume that $X^{(0)}$ was chosen such that $X^{(0)}$ is normalized and $a_1 \neq 0$. Thus,

$$\begin{aligned} Y^{(0)} &= AX^{(0)} = A(a_1\overline{X_1} + a_2\overline{X_2} + \cdots + a_n\overline{X_n}) \\ &= a_1A\overline{X_1} + a_2A\overline{X_2} + \cdots + a_nA\overline{X_n} \\ &= a_1\lambda_1\overline{X_1} + a_2\lambda_2\overline{X_2} + \cdots + a_n\lambda_n\overline{X_n} \\ &= \lambda_1 \left(a_1\overline{X_1} + a_2 \left(\frac{\lambda_2}{\lambda_1}\right)\overline{X_2} + \cdots + a_n \left(\frac{\lambda_n}{\lambda_1}\right)\overline{X_n} \right) \\ X^{(1)} &= \frac{1}{c_1} Y^{(0)} = \frac{\lambda_1}{c_1} \left(a_1\overline{X_1} + a_2 \left(\frac{\lambda_2}{\lambda_1}\right)\overline{X_2} + \cdots + a_n \left(\frac{\lambda_n}{\lambda_1}\right)\overline{X_n} \right) \end{aligned}$$

After performing k iterations, we obtain

$$\begin{aligned}
Y^{k-1} &= AX^{(k-1)} = A\frac{\lambda_1^{k-1}}{c_1c_2\cdots c_{k-1}}\left(a_1\overline{X_1} + a_2\left(\frac{\lambda_2}{\lambda_1}\right)^{k-1}\overline{X_2} + \cdots + a_n\left(\frac{\lambda_n}{\lambda_1}\right)^{k-1}\overline{X_n}\right) \\
&= \frac{\lambda_1^{k-1}}{c_1c_2\cdots c_{k-1}}\left(a_1A\overline{X_1} + a_2\left(\frac{\lambda_2}{\lambda_1}\right)^{k-1}A\overline{X_2} + \cdots + a_n\left(\frac{\lambda_n}{\lambda_1}\right)^{k-1}A\overline{X_n}\right) \\
&= \frac{\lambda_1^{k-1}}{c_1c_2\cdots c_{k-1}}\left(a_1\lambda_1\overline{X_1} + a_2\left(\frac{\lambda_2}{\lambda_1}\right)^{k-1}\lambda_2\overline{X_2} + \cdots + a_n\left(\frac{\lambda_n}{\lambda_1}\right)^{k-1}\lambda_n\overline{X_n}\right) \\
&= \frac{\lambda_1^{k}}{c_1c_2\cdots c_{k-1}}\left(a_1\overline{X_1} + a_2\left(\frac{\lambda_2}{\lambda_1}\right)^{k}\overline{X_2} + \cdots + a_n\left(\frac{\lambda_n}{\lambda_1}\right)^{k}\overline{X_n}\right) \\
X^{(k)} &= \frac{\lambda_1^{k}}{c_1c_2\cdots c_{k-1}c_k}\left(a_1\overline{X_1} + a_2\left(\frac{\lambda_2}{\lambda_1}\right)^{k}\overline{X_2} + \cdots + a_n\left(\frac{\lambda_n}{\lambda_1}\right)^{k}\overline{X_n}\right)
\end{aligned}$$

Since $\left|\dfrac{\lambda_j}{\lambda_1}\right| < 1$ for $j = 2,\ 3,\ \ldots,\ n$, we get

$$\lim_{k\to\infty}\ a_j\left(\frac{\lambda_j}{\lambda_1}\right)^{k}\overline{X_j} = 0, \quad j = 2,\ 3,\ \ldots,\ n \tag{7.1}$$

Thus, we have

$$\begin{aligned}
\lim_{k\to\infty} X^{(k)} &= \lim_{k\to\infty}\ \frac{\lambda_1^{k}}{c_1c_2\cdots c_{k-1}c_k}\left(a_1\overline{X_1} + a_2\left(\frac{\lambda_2}{\lambda_1}\right)^{k}\overline{X_2} + \cdots + a_n\left(\frac{\lambda_n}{\lambda_1}\right)^{k}\overline{X_n}\right) \\
&= \lim_{k\to\infty}\ \frac{a_1\lambda_1^{k}}{c_1c_2\cdots c_{k}}\overline{X_1}
\end{aligned}$$

Now both $X^{(k)}$ and $\overline{X_1}$ are normalized so that the largest component of each is 1. Thus, the limit of the scalar multiple of $\overline{X_1}$ on the right hand side of the last equation above must exist and must be equal to 1, that is,

$$\lim_{k\to\infty}\ \frac{a_1\lambda_1^{k}}{c_1c_2\cdots c_{k}} = 1 \tag{7.2}$$

This gives us

$$\lim_{k\to\infty}\ X^{(k)} = \overline{X_1} \tag{7.3}$$

which shows the convergence of the sequence $\{\ X^{(k)}\ \}$ to the normalized dominant eigenvector $\overline{X_1}$.

On the other hand, as k increases to ∞, so does $k-1$, which shows that we can replace k with $k-1$ in Equation 7.2. This gives us

$$\lim_{k\to\infty}\ \frac{a_1\lambda_1^{k-1}}{c_1c_2\cdots c_{k-1}} = 1 \tag{7.4}$$

Combining Equations 7.2 and 7.4, we obtain

$$\lim_{k\to\infty}\frac{\lambda_1}{c_k} = \lim_{k\to\infty}\left(\frac{\lambda_1}{c_k}\right)\frac{\dfrac{a_1\lambda_1^{k-1}}{c_1c_2\cdots c_{k-1}}}{\dfrac{a_1\lambda_1^{k-1}}{c_1c_2\cdots c_{k-1}}}$$

$$= \lim_{k\to\infty}\frac{\dfrac{a_1\lambda_1^{k}}{c_1c_2\cdots c_{k}}}{\dfrac{a_1\lambda_1^{k-1}}{c_1c_2\cdots c_{k-1}}} = \frac{1}{1} = 1$$

This shows that

$$\lim_{k\to\infty} c_k = \lambda_1$$

which means that the sequence $\{\ c_k\ \}$ converges to the dominant eigenvalue λ_1. □

7.2.2 The Power Method to Obtain the Least Eigenvalue

In the preceding section, we showed how the power method can be used to obtain estimates for the dominant eigenpair $(\lambda_1, \overline{X_1})$. We now show how this method can be used to obtain the eigenvalue of the least magnitude, λ_n.

First note that if A is a nonsingular $n \times n$ matrix and λ is an eigenvalue of A, then there exists a nonzero $X \in \mathbb{R}^n$ such that $AX = \lambda$. Since A is nonsingular, we can multiply both members of this equation by A^{-1} to obtain

$$X = \lambda A^{-1}X \ \text{ which means } \ \frac{1}{\lambda}X = A^{-1}X$$

This shows that $\dfrac{1}{\lambda}$ is an eigenvalue of A^{-1}, with the same associated eigenvector X. Hence, the nonzero eigenvalues of A are the reciprocals of the nonzero eigenvalues of A^{-1}.

Now if λ_n, which is the eigenvalue of least magnitude of A, is nonzero, then $\mu_1 = \dfrac{1}{\lambda_n}$ is the dominant eigenvalue of A^{-1}. Thus, we can obtain an estimate for the value of λ_n by first applying the power method to A^{-1} to obtain an estimate for μ_1, then taking its reciprocal to obtain an estimate for λ_n.

Example 7.7. Consider the matrix $A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$ from Example 7.3. Since

$$|A| = \begin{vmatrix} 2 & 1 \\ 1 & 2 \end{vmatrix} = 2(2) - 1(1) = 3 \neq 0$$

Thus, A is nonsingular. Moreover, it can be verified that

$$A^{-1} = \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} \\ -\frac{1}{3} & \frac{2}{3} \end{bmatrix}$$

If we apply the power method on A^{-1} with $X^{(0)} = (1 \;\; 0)^T$, then five iterations of the power method yield the following results:

k	c_k	$X^{(k)}$
1	$\frac{2}{3}$	$\begin{bmatrix} 1 & -\frac{1}{2} \end{bmatrix}^T$
2	$\frac{5}{6}$	$\begin{bmatrix} 1 & -\frac{4}{5} \end{bmatrix}^T$
3	$\frac{14}{15}$	$\begin{bmatrix} 1 & -\frac{13}{14} \end{bmatrix}^T$
4	$\frac{41}{42}$	$\begin{bmatrix} 1 & -\frac{40}{41} \end{bmatrix}^T$
5	$\frac{122}{123}$	$\begin{bmatrix} 1 & -\frac{121}{122} \end{bmatrix}^T$

We have $c_5 = \dfrac{122}{123}$ and $X^{(5)} = \begin{bmatrix} 1 & -\dfrac{121}{122} \end{bmatrix}^T$. Thus, $\lambda_2 \approx \dfrac{1}{c_5} = \dfrac{123}{122}$. The actual value of λ_2 is 1, and the corresponding normalized eigenvector is $\overline{X_2} = [\, 1 \;\; -1]^T$, which compare well with our estimates.

7.3 The Shifted Inverse Power Method

We have shown how to use the power method to obtain both the dominant and the least eigenvalues of a matrix A, together with their associated normalized eigenvectors. We now discuss a method to obtain an arbitrary eigenvalue λ of a square matrix A. The method is based on the following theorem:

Theorem 7.3. *Let (λ, X) be an eigenpair of A, and let α be some real number such that $\alpha \neq \lambda$. Then $\left(\dfrac{1}{\lambda - \alpha},\ X\right)$ is an eigenpair of $(A - \alpha I_n)^{-1}$.*

Proof. From the discussion in the preceding section, it suffices to show that $\lambda - \alpha$ is an eigenvalue of $(A - \alpha I_n)$. To this end, let X be an eigenvector of A associated with λ, so that $AX = \lambda X$. We have

$$\begin{aligned} (A - \alpha I_n)X &= AX - \alpha I_n X = AX - \alpha X \\ &= \lambda X - \alpha X = (\lambda - \alpha)X \end{aligned}$$

This shows that $\lambda - \alpha$ is an eigenvalue of $A - \alpha I_n$. Moreover, since $\alpha \neq \lambda$, we know that $\dfrac{1}{\lambda - \alpha}$ is an eigenvalue of $(A - \alpha I_n)^{-1}$. □

The following theorem describes the *shifted inverse power method*, which gives an algorithm for computing estimates for any eigenvalue of a matrix A, when A satisfies a given set of conditions.

Theorem 7.4. *Let A be an $n \times n$ matrix with distinct eigenvalues $\lambda_1,\ \lambda_2,\ \ldots,\ \lambda_n$. For each eigenvalue λ_j, a real number α_j can be chosen such that*

$$\lambda_j \neq \alpha_j \quad \textit{and} \quad |\lambda_j - \alpha_j| < |\lambda_i - \alpha_j|, \quad i \neq j$$

and hence

$$\mu_j = \frac{1}{\lambda_j - \alpha_j}$$

is the dominant eigenvalue of the matrix $(A - \alpha_j I_n)^{-1}$. If an appropriate initial vector $X^{(0)}$ is chosen, then the sequences of estimates $\{\ c_k\ \}$ and $\{\ X^{(k)}\ \}$ generated by the equation

$$Y^{(k)} = (A - \alpha_j I_n)^{-1} X^{(k)} = c_{k+1} X^{(k+1)}$$

where

$$X^{(k)} = [\ x_1^{(k)} \quad x_2^{(k)} \quad \cdots \quad x_n^{(k)}\ ]^T$$

converge to the eigenpair $(\mu_j,\ \overline{X_j})$ of the matrix $(A - \alpha I_n)^{-1}$. The eigenvalue λ_j of A is given by the equation

$$\lambda_j = \frac{1}{\mu_j} + \alpha_j \tag{7.5}$$

Proof. According to Theorem 7.3, the pair $\left(\dfrac{1}{\lambda_j - \alpha_j}, \overline{X_j}\right)$ is an eigenpair of the matrix $(A - \alpha_j I_n)^{-1}$. Since

$$|\lambda_j - \alpha_j| < |\lambda_i - \alpha_j|, \quad i \neq j$$

we know that

$$\mu_j = \frac{1}{\lambda_j - \alpha_j}$$

is a dominant eigenvalue of $(A - \alpha_j I_n)^{-1}$. Moreover, $\overline{X_j}$ is a normalized associated eigenvector. By Theorem 7.2, applying the power method generates the sequences $\{\ c_k\ \}$ and $\{\ X^{(k)}\ \}$ which converge to the eigenpair $(\mu_j,\ \overline{X_j})$ of the matrix $(A - \alpha I_n)^{-1}$. We then have

$$\begin{aligned} \mu_j \overline{X_j} &= (A - \alpha I_n)^{-1} \overline{X_j} \\ (A - \alpha I_n)\overline{X_j} &= \frac{1}{\mu_j} \overline{X_j} \end{aligned}$$

The last equation shows that $\dfrac{1}{\mu_j}$ is an eigenvalue of $(A - \alpha I_n)$, and hence $\dfrac{1}{\mu_j} + \alpha_j$ is an eigenvalue of A. Since the sequence $\{\ c_k\ \}$ converges to $\mu_j = \dfrac{1}{\lambda_j - \alpha_j}$, it follows that the sequence $\left\{ \frac{1}{c_k} + \alpha_j \right\}$ converges to λ_j. □

From the above theorem, we obtain the following algorithm for estimating an eigenvalue λ_j of a square matrix A:

Algorithm: Shifted Inverse Power Method

Let A be an $n \times n$ matrix with n distinct eigenvalues $\lambda_1,\ \lambda_2,\ \ldots,\ \lambda_n$. Choose a real number α_j such that $|\lambda_j - \alpha_j| < |\lambda_i - \alpha_j|,\ \ i \neq j$. Let $X^{(0)} \in \mathbb{R}^n$ be a normalized nonzero vector.

- Use $X^{(0)}$ and the power method on the matrix $(A-\alpha_j I_n)^{-1}$ to obtain the sequences $\{\ c_k\ \}$ and $\{\ X^{(k)}\ \}$ that converge to the dominant eigenpair $(\mu_j, \overline{X_j})$ of $(A - \alpha_j I_n)^{-1}$.
- The sequence $\left\{ \frac{1}{c_k} + \alpha_j \right\}$ converges to the eigenvalue λ_j of A, with the same associated eigenvector $\overline{X_j}$.

Example 7.8. Use four iterations of the shifted inverse power method to obtain estimates for the eigenpairs of the matrix $A = \begin{bmatrix} 1 & 2 \\ 5 & 4 \end{bmatrix}$. Compare your answers with the actual values.

Solution.

$$\begin{aligned} |\lambda I_2 - A| &= \begin{vmatrix} \lambda - 1 & -2 \\ -5 & \lambda - 4 \end{vmatrix} = (\lambda - 1)(\lambda - 4) - (-2)(-5) \\ \lambda^2 - 5\lambda - 6 &= (\lambda - 6)(\lambda + 1) = 0 \Rightarrow \lambda_1 = 6,\ \lambda_2 = -1 \end{aligned}$$

The corresponding normalized eigenvectors are $\overline{X_1} = [2/5 \quad 1]^T$ and $\overline{X_2} = [1 \quad -1]^T$

1. $\lambda_1 = 6,\ \alpha_1 = 5$

$$\begin{aligned} A - \alpha_1 I_2 &= A - 5I_2 = \begin{bmatrix} -4 & 2 \\ 5 & -1 \end{bmatrix} \\ (A - \alpha_1 I_2)^{-1} &= \begin{bmatrix} 1/6 & 1/3 \\ 5/6 & 2/3 \end{bmatrix} \end{aligned}$$

Four iterations of the power method with $X^{(0)} = [\ 1 \ \ 1\ ]^T$ give the following results:

k	c_k	$X^{(k)}$
1	11/6	$[3/11 \quad 1]^T$
2	59/66	$[25/59 \quad 1]^T$
3	401/414	$[163/401 \quad 1]^T$
4	2419/2406	$[965/2419 \quad 1]^T$

Thus,

$$\begin{aligned} \mu_1 &\approx c_4 = \frac{2419}{2406} \Rightarrow \lambda_1 \approx \frac{1}{c_k} + \alpha_1 = \frac{2406}{2419} + 5 = 6.00166251 \\ \overline{X_1} &\approx X^{(4)} = \begin{bmatrix} \frac{965}{2419} \\ 1 \end{bmatrix} \end{aligned}$$

2. $\lambda_2 = -1,\ \alpha_2 = 0.5$

$$\begin{aligned} A - \alpha_2 I_2 &= A - 0.5 I_2 = \begin{bmatrix} 1/2 & 2 \\ 5 & 7/2 \end{bmatrix} \\ (A - \alpha_2 I_2)^{-1} &= \begin{bmatrix} -14/33 & 8/33 \\ 20/33 & -2/33 \end{bmatrix} \end{aligned}$$

Four iterations of the power method with $X^{(0)} = [1\ \ 0]^T$ gives the following results:

k	c_k	$X^{(k)}$
1	20/33	$[-7/10\ \ 1]^T$
2	89/165	$[1\ \ -80/89]^T$
3	1940/2937	$[-943/970\ \ 1]^T$
4	20962/32010	$[1\ \ -10400/10481]^T$

This gives us

$$\begin{aligned} \mu_2 &\approx c_4 = \frac{20962}{32010} \Rightarrow \lambda_2 \approx \frac{1}{c_4} + \alpha_2 = \frac{32010}{20962} + 0.5 = 2.02704895 \\ \overline{X_2} &\approx X^{(4)} = \begin{bmatrix} 1 \\ -\frac{10400}{10481} \end{bmatrix} \end{aligned}$$

The values obtained compare well with the actual eigenpairs.

Exercises.

Use ten iterations of the power method and the shifted inverse power method to find estimates for the eigenvalues of the following matrices.

1. $\begin{bmatrix} 1 & -1 & 0 \\ -2 & 4 & -2 \\ 0 & -1 & 1 \end{bmatrix}$. Use $X^{(0)} = [1\ \ 0\ \ 0]^T$.
2. $\begin{bmatrix} 1 & -1 & 0 \\ -2 & 4 & -2 \\ 0 & -1 & 2 \end{bmatrix}$. Use $X^{(0)} = [1\ \ 0\ \ 0]^T$.
3. $\begin{bmatrix} 4 & 1 & 1 & 1 \\ 1 & 3 & -1 & 1 \\ 1 & -1 & 2 & 0 \\ 1 & 1 & 0 & 2 \end{bmatrix}$. Use $X^{(0)} = [1\ \ 1\ \ 1\ \ 1]^T$.
4. $\begin{bmatrix} 5 & -2 & -0.5 & 1.5 \\ -2 & 5 & 1.5 & -0.5 \\ -0.5 & 1.5 & 5 & -2 \\ 1.5 & -0.5 & -2 & 5 \end{bmatrix}$. Use $X^{(0)} = [1\ \ 1\ \ 1\ \ 1]^T$