A Notebook on Numerical Methods

Book · December 2015

1 author:

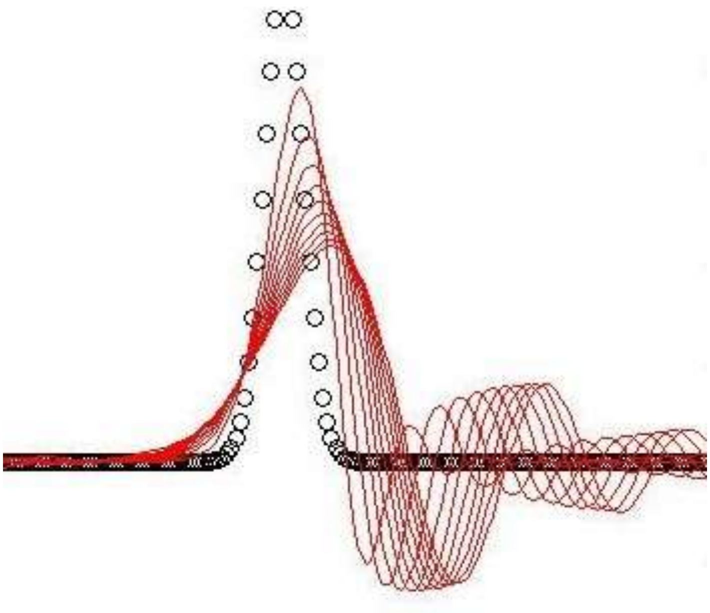
Shree Krishna Khadka
Tribhuvan University

A Note Book On

NUMERICAL METHODS

For BE-V Semester (Computer Engineering)

Prepared By
Er. SriKisna



Inside

- Numerical Methods: Introduction & Scope
- Errors in Numerical Calculation
- Solution of Algebraic & Transcendental Equation
- Interpolation
- Curve Fitting, B Splines & Approximation
- Numerical Differentiation & Integration
- Matrices & Linear System of Equations
- Solution of Ordinary Differential Equations
- Solution of Partial Differential Equations
- Solution of Integral Equations
- Programming Examples

CONTENT

WhiteHouse Institute of Science & Technology

Khumaltar, Lalitpur-Satdobato

Bachelor in E & C/Computer Engineering

Bachelor in Information Technology

<u>NUMRECIAL METHODS</u>

Lecture Handouts

Prepared by: Er Shree Krishna Khadka

2012

1. **Reference books:**

   a. Numerical Methods by E. Balagurusamy
   b. Introductory Methods of Numerical Analysis by S.S. Sastry

2. **Teaching Schedule:**

   a. Theory: 3hrs/week
   b. Practical: 1.5hrs/week

3. **Examination Scheme:**

   a. **Internal Assessment**
      i. Theory: 20 marks
      ii. Practical: 50 marks

   b. **Final Examination**
      i. Theory: 80 marks

   c. **Total:** 150 marks

4. **Internal marks evaluation:**

Theory (20)				Practical (50)		
A &CP	GD & P	TP & MT	PBE	A & CP	RS & P	PE & MCQs
25%	25%	25%	25%	20%	40%	40%

5. **Abbreviations:**
   a. **A & CP:** Attendance & Class Participation
   b. **GD & P:** Group Discussion & Presentation
   c. **TP & MT:** Test Papers & Mid Term
   d. **PBE :** Pre Board Examination
   e. **RS & P:** Report Submission & Presentation
   f. **PE & MCQs:** Practical Exam & Multiple Choice Questions

6. **Adjustments:**

   a. 80-100% accomplishment: Full marks (100%)
   b. 60-80% accomplishment: Semi marks (80%)
   c. 40-60% accomplishment: Fair marks (60%)

7. **Ratings:**

   a. *****:** Distinction with 100% marks.
   b. ****** **:** First Division with up to 80% marks.
   c. ***** **:** Fair with up to 60% marks.

Prepared By
Er. Shree Krishna Khadka

8. **Final Examination:**

a. **Board Exam:**

i. Full marks: 80
ii. Question Pattern: Generally consists of two groups

Pattern	BIT		BE C/C & E	
	Group A	**Group B**	**Group A**	**Group B**
Question type	Long	Short	Long	Long/Program
Total question	3	10	7	3
Attempt	2	8	6	2
Marks/each	12	7	10	10
Total	24	56	60	20

b. For BIT: Sometimes, one extra paper with 20 objectives questions can be asked each carrying 1 marks for twenty minutes as a Group A. In that case, long answer type questions weigh 10 marks each and short type questions weigh 5 marks for each.
For BE: Choices may vary from more than one questions in each groups.

c. **Lab Exam**

i. Of about 12 exercises on lab
- As per the syllabus, selected topics will be exercised.
ii. Programming
- Programming is done using C/C++ language in Visual Basic platform.
iii. Submission of lab report is compulsory.
iv. Practical Exam
  1. Programming test.
  2. Viva/MCQs test.

Introduction

1. **Why to study Numerical Methods?**
   a. Historical Review:
      - In early age of about 3000 BC, computing starts with a device called ABACUS by Chinese society, which was used for arithmetic operation only.
      - Slide rule by John Napier to compute a logarithmic problems in early 16th century.
      - An accounting machine by Blaise Pascal called Pascal Calculator.
      - Data storing facility at punched card by Loom Jacquard in 18th century.
      - 18th Century: Revolutionary invention of Difference Engine & Analytical Engine by Charles Babbage.
      - Later on till the date, all advanced/most recent computers are based on that principle.

   b. Present Context:

      Now-a-days, computers are great tools in numerical computation, which plays an indispensable role in solving real-life mathematical, physical and engineering problems.

      However, without fundamental understanding of engineering problems, they will be useless.

2. **How is the engineering system understood?**
   - By observation and experiment.
   - Theoretical analysis and generalization.

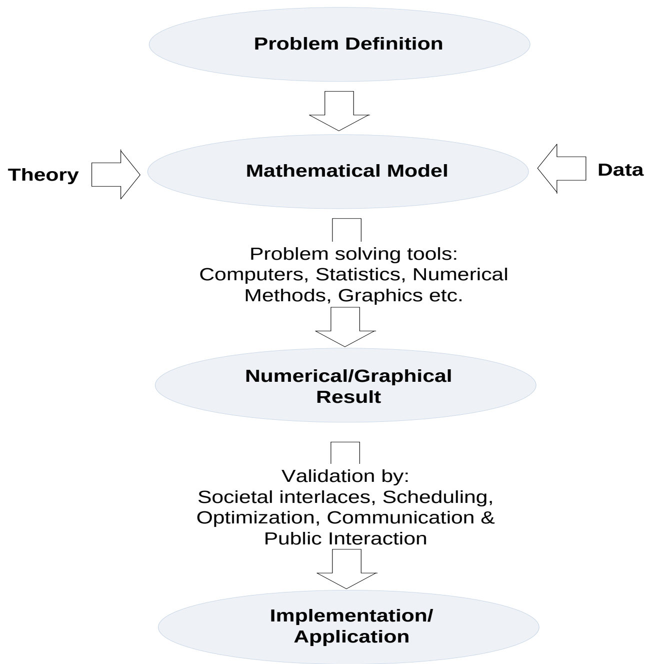


Fig: Computing Process

3. **Example:**

   Newton's Second Law of Motion

   1. **Theory:**

      The time rate of change of momentum of a body is equal to the resulting force acting on it.

   2. **Mathematical Model:**

$$\mathbf{F} = \mathbf{m}\frac{(\mathbf{v} - \mathbf{u})}{\mathbf{t}} = \mathbf{m}.\mathbf{a}$$

Where, F = Net force acting on the body.

m = Mass of the object
u = Initial velocity of the object
v = Final velocity of the object
t = time and a = acceleration

Data

3. **Analysis:**

   It characterizes the typical mathematical model of the physical world as:

   a. Natural process or system in mathematical term.
   b. Idealization and simplification of reality.
   c. Yields reproducible results.

4. **Solution:**

   Manipulation of mathematical model gives rise to the solution of the problem.

   Analytical Vs. Numerical Solution

   Analytical solution is like a 'Math: a great example', as we have been learning all the way. Numerical solution is computed via computer with some approximation but exact result.

   For example, the quadratic equation ax2+bx+c = 0 has:

   a) **Analytic solution:** $x = -\frac{b}{2a} \pm \frac{\sqrt{b^2-4ac}}{2a}$; which works for any set of values of a, b & c. The real solution exists only if $b^2 - 4ac \geq 0$, i.e. the properties of the solution are transparent.

   b) **Numerical solution:** Can only deal with a given set (a, b, c) at a given time. Solution is approximate. So, error estimation is needed.

   We want the computer to do repeated search for the solution, but not blindly. A clear set of rules (steps) has to be designed based on the sound mathematical reasoning, which guarantee a solution at a desirable level of accuracy.

5. **Validation & Implementation**

   Now, numerical method is a tool that deals with the mathematical model (formulated to describe the theory) to give a valid numerical or graphical result.

   This example is just a prototype; most of the real-world problems (99.9%) are complex in nature but can be approximately solved with numerical computation.

4. **Summary:**

   a. Science and engineering study convert physical phenomena into mathematical model often with complex-analysis and calculation.
   b. Numerical analysis is the study of procedure for solving the problem with computer.
   c. It is the study of algorithms and further problems of mathematics.
   d. Numerical analysis is always numerical.
   e. Goal of numerical analysis is the design and analysis of techniques to give approximate but accurate solution to hard problems.

5. **Applications:**

   a. Numerical weather forecasting.
   b. Computing transactory of space-craft (path-of-projectile).
   c. In super computers.
   d. In improvement of car safety to avoid accidents.

1

CHAPTER

Errors in Numerical Calculation

Contents:

- Numbers & Their Accuracy
- Different Types of Errors
- Mathematical Preliminaries
- General Error Formula
- Convergence
- Assignment 1

NUMBERS & THEIR ACCURACY

There are two kinds of numbers:

i) Exact numbers, e.g. 1, 2, 3 …1/2, 5/2 etc.
ii) Approximate numbers, e.g. PI, K … etc

- **Significant Digit:**

Digits that are used to express a number are called significant digits/figures. The following statements describe the notion of significant digits.

- All non-zero digits.
- All zeroes occurring between non- zero digits.
- Trailing zeroes following a decimal point, e.g. 3.50, 65.0, 0.230 have three significant digits.
- Zeroes between the decimal point are preceding a non-zero digits are not significant, e.g. 0.01234, 0.001234, 0.0001234 all have four significant digits.
- When the decimal point is not written, trailing zeroes are not considered to be significant, e.g. 4500 contain only two significant digits.

- **Accuracy & precision**

The concept of accuracy and precision are closely related to significant digits.

- Accuracy refers to the number of significant digits in a value. E.g. the number 57.345 is accurate to five significant digits.
- Precision refers to the number of decimal position, i.e. the order of magnitude of last digit in a value. E.g. 57.396 have a precision of 0.001.

In numerical computation, we come across numbers which have large numbers of digits and it will be necessary to cut them to usable numbers of figures. This process is known as rounding off. The error caused due to cut-off a large number into usable number of figure is called round-off error.

- **Rules for rounding off:**

To round off a number to n-significant digits, discard all digits to the right of the n^{th} digit if this discarded number is:

- Less than half a unit in the n^{th} place, leave the n^{th} digit unaltered.
- Greater than half a unit in n^{th} place, increase the n^{th} digit by unity.
- Exactly half a unit in the n^{th} place, increase the n^{th} digit by unity if it is odd; otherwise leave it unchanged.

Examples**: Following numbers are rounded off to four significant figures.**

1.6583(1.658), 30.0567(30.06), 0.859378(0.8594), 3.14159(3.142)

DIFFERENT TYPES OF ERROR:

a) **Absolute Error**: Numerical difference between true value of a quantity and its approximate value.

Mathematically;

$$\text{Absolute Error}(E_A) = \text{True Value}(X) - \text{Approximate Value}(X_1) = \Delta X$$

b) **Relative error**: Ratio of absolute error to a true value of that quantity being concerned.

Mathematically;

$$\text{Relative Error}(E_R) = \frac{\text{Absolute Error}(E_A)}{\text{True Value}(X)}$$

c) **Percentage Error**: The percentage value of relative error.

Mathematically;

$$\text{Percentage Error}(E_P) = \frac{\text{Absolute Error}(E_A)}{\text{True Value}(X)} \times 100\%$$

d) **Truncation Error**: Occurs due to truncation or terminating an infinite sequence of operation after a finite number have been performed.

Example: $\cos(x) = 1 - \frac{x^2}{2}, \quad e^x = 1 + x + \frac{x^2}{2}$

e) **Relative Accuracy**: Ratio of change in true value to the modulus of true value.

Mathematically;

$$\text{Relative Accuracy } (R_A) = \frac{\text{Absolute Error}(E_A)}{|\text{True Value}(X)|} \sim \frac{\text{Absolute Error}(E_A)}{|\text{Approximate Value}(X_1)|}$$

NUMERICAL EXAMPLES:

- An approximate value of **PI** is given as 3.1428571 and its true value is 3.1415926. Find E_A and E_R.

  Solution:

  $E_A = | X-X_1| = |3.1415926-3.1428571| = 1.265*10^{-3}$
  $E_R = E_A/X = 1.265*10^{-3}/3.1415926 = 4.025*10^{-4}$

- Three approximate values of the number 1/3 are given as 0.30, 0.33 and 0.34, which of these is the best approximation?

  Solution:

  $E_1 = |(1/3)-0.30| = 1/30$
  $E_2 = |(1/3)-0.33| = 1/300$
  $E_3 = |(1/3)-0.34| = 1/150$

  Since, $E_2<E_3<E_1$, 0.33 is the best approximation.

- Evaluate the sum S=_/3 + _/5 + _/7 to four significant digits and find its absolute and relative error.

  Solution:

  _/3 = 1.73205 = 1.732
  _/5 = 2.23606 = 2.236
  _/7 = 2.64574 = 2.646

  S = 1.732 + 2.236 + 2.646 = 6.614

  Absolute Error (E_A) = 0.0005 + 0.0005 + 0.0005 = 0.0015
  Relative Error (E_R) = E_A/S = 0.0015/6.614 = $2.268*10^{-4}$ = 0.0002

MATHEMATICAL PRELIMINARIES

Theorem 1: If f(x) is continuous in $a \leq x \leq b$, and if f(a) and f(b) are of opposite signs, then $f(\omega) = 0$ for at least one number ω, such that $a < \omega < b$.

Theorem 2 (Rolle's Theorem): If f(x) is continuous in $a \leq x \leq b$, $f'(x)$ exists in $a < x < b$ and f(a)=f(b)=0, then there exist at least one value of x say ω, such that $f'(\omega) = 0$.

Generalization: If f(x) be a function, which is n-times differentiable on [a, b]. If f(x) vanishes at the (n+1) distinct points $x_0, x_1, x_2, \ldots, x_n$ in (a,b), then there exist a number ω in (a,b) such that , $f^n(\omega) = 0$.

Theorem 3 (Intermediate Value Theorem): If f(x) be continuous in [a, b] and let K be any number between f(a) and f(b), then there exists a number ω in (a, b) such $f(\omega) = K$.

Theorem 4 (Mean Value Theorem for Derivative): If f(x) is continuous in [a, b] and f'(x) exists in (a, b), there exists at least one value of x, say ω between **a** and **b** such that:

$$f'(\omega) = \frac{f(b) - f(a)}{b - a}; \qquad a < \omega < b$$

Theorem 5 (Taylor's Series for a function of one variable): If f(x) is continuous and passes continuous derivates of order in an interval and includes x=a, then:

$$f(x) = f(a) + (x - a)f'(a) + \frac{(x - a)^2}{2!}f''(a) + \cdots + \frac{(x - a)^{n-1}}{(n - 1)!}f^{n-1}(a) + R_n(x)$$

Where; $R_n(x)$: Remainder term $= \frac{(x-a)^n}{n!}f^n(\omega)$; $a < \omega < b$.

Theorem 6 (McLaren's Expansion):

$$f(x) = f(0) + xf'(0) + \frac{x^2}{2!}f''(0) + \cdots + \frac{x^n}{n!}f^n(0)$$

Theorem 7 (Taylor's Series for 2-variables):

$$f\big((x_1 + \Delta x_1), (x_2 + \Delta x_2)\big) = f\big((x_1), (x_2)\big) + \frac{\partial f}{\partial x_1}\Delta x_1 + \frac{\partial f}{\partial x_2}\Delta x_2 + \cdots + \frac{\partial f}{\partial x_n}\Delta x_n + \frac{1}{2}\{\frac{\partial^2 f}{\partial {x_1}^2}(\Delta x_1)^2 + \frac{\partial^2 f}{\partial {x_2}^2}(\Delta x_2)^2 + \cdots + \frac{\partial^2 f}{\partial {x_n}^2}(\Delta x_n)^2\} + \cdots$$

Theorem 8 (Taylor's Series for several variables):

$$f\big((x_1+\Delta x_1),(x_2+\Delta x_2),(x_3+\Delta x_3)\ldots(x_n+\Delta x_n)\big)$$
$$= f\big((x_1),(x_2),(x_3)\ldots(x_n)\big)+\frac{\partial f}{\partial x_1}\Delta x_1+\frac{\partial f}{\partial x_2}\Delta x_2+\cdots+\frac{\partial f}{\partial x_n}\Delta x_n$$
$$+\frac{1}{2}\{\frac{\partial^2 f}{\partial {x_1}^2}(\Delta x_1)^2+\frac{\partial^2 f}{\partial {x_2}^2}(\Delta x_2)^2+\cdots+\frac{\partial^2 f}{\partial {x_n}^2}(\Delta x_n)^2\}$$
$$+2\left\{\frac{\partial^2 f}{\partial x_1 x_2}(\Delta x_1\Delta x_2)+\frac{\partial^2 f}{\partial x_2\,\partial x_3}(\Delta x_2\Delta x_3)\ldots+\frac{\partial^2 f}{\partial x_{n-1}\,\partial x_n}(\Delta x_{n-1}\Delta x_n)\right\}$$
$$+\cdots$$

GENERAL ERROR FORMULA

Formula for the error committed in using a certain formula or a functional relation.
Let; $u=f(x_1, x_2, x_3, x_4, x_5, \ldots x_n)$ be the function of several variable x_i (i=1,2…n) and let the error in each $\mathbf{x_i}$ be $\Delta\mathbf{x_i}$, then the error $\Delta\mathbf{u}$ in $\mathbf{u}$ is given by:

$$u+\Delta u = f((x_1+\Delta x_1),(x_2+\Delta x_2),(x_3+\Delta x_3)\ldots(x_n+\Delta x_n)) \;\ldots.(i)$$

On expansion of RHS using Taylor's Series, we get:

$$f\big((x_1+\Delta x_1),(x_2+\Delta x_2),(x_3+\Delta x_3)\ldots(x_n+\Delta x_n)\big)$$
$$= f\big((x_1),(x_2),(x_3)\ldots(x_n)\big)+\frac{\partial f}{\partial x_1}\Delta x_1+\frac{\partial f}{\partial x_2}\Delta x_2+\cdots+\frac{\partial f}{\partial x_n}\Delta x_n$$
$$+\frac{1}{2}\{\frac{\partial^2 f}{\partial {x_1}^2}(\Delta x_1)^2+\frac{\partial^2 f}{\partial {x_2}^2}(\Delta x_2)^2+\cdots+\frac{\partial^2 f}{\partial {x_n}^2}(\Delta x_n)^2\}$$
$$+2\left\{\frac{\partial^2 f}{\partial x_1 x_2}(\Delta x_1\Delta x_2)+\frac{\partial^2 f}{\partial x_2\,\partial x_3}(\Delta x_2\Delta x_3)\ldots+\frac{\partial^2 f}{\partial x_{n-1}\,\partial x_n}(\Delta x_{n-1}\Delta x_n)\right\}$$
$$+\cdots$$

i.e. $u+\Delta u = f\big((x_1),(x_2),(x_3)\ldots(x_n)\big)+\sum_{i=1}^{n}\frac{\partial f}{\partial x_i}\Delta x_i + \text{term involving } (\Delta x_i)^2$

So, ignoring powers of errors, we get:

$$u=\sum_{i=1}^{n}\frac{\partial f}{\partial x_i}\Delta x_i=\frac{\partial f}{\partial x_1}\Delta x_1+\frac{\partial f}{\partial x_2}\Delta x_2+\cdots+\frac{\partial f}{\partial x_n}\Delta x_n \;\ldots(ii)$$

Now, Relative Error: $E_R=\frac{\Delta u}{u}=\frac{\partial f}{\partial x_1}\frac{\Delta x_1}{u}+\frac{\partial f}{\partial x_2}\frac{\Delta x_2}{u}+\cdots+\frac{\partial f}{\partial x_n}\frac{\Delta x_n}{u}\ldots(iii)$

CONVERGENCE

- Numerical computing is based on the idea of iterative process.
- Iterative processes involve generation of sequence of approximation with the hope that, the process will end of the required solution.
- Certain methods convert faster than others.
- It is necessary to know that convergence rate of any method to get the required solution.
- Rapid convergent take less execution time.

ASSIGNMENT 1

Full Marks: 20
Pass Marks: 10

Que: 1 (3X2=6 Marks)

1.1) Round off the following numbers to two decimal places.
2.3742, 81.255, 52.275, 48.21416

1.2) Round off the following numbers to four significant digits.
0.70029, 0.00022218, 2.36425, 38. 46235

1.3) Calculate (_/102 - _/101) correct to four significant digits.

Que: 2 (2X2=4 Marks)

2.1) if u = $3v^7$-6v, find the percentage error in u at v=1, if the error in v is 0.05.

2.2) if y = (0.31x+2.73)/(x+0.35), where the coefficients are rounded off; find the absolute and relative error in y when x=0.5 $\pm$ 0.1.

Que: 3 (3X2=6 Marks)

If $a = 10.00 \pm 0.05, b = 0.0356 \pm 0.0002, c = 15300 \pm 100$ & $d = \pm 62000 \pm 500$, find the maximum absolute error in:

a) a+b+c+d
b) a+5c-d
c) c^3

Que: 4 (1X4=4 Marks)

If $u=5xy^2/z^3$,
$\Delta x = \Delta y = \Delta z = 1$ &
$x = y = z = 1$;

Find the maximum value of relative error.
(Hint: Use general error formula)

2

CHAPTER

Solution of Algebric & Transcendental Equations

Contents:

- Introduction
- Evaluation of Polynomials by Horner's Method
- Methods of solving non linear equations
  - Bracketing Methods
    - Bisection Method
    - Regular Falsi Method
  - Open End Methods
    - Newton Raphson Method
    - Secant Method
    - Fixed Point Iteration Method
- Horner's Method of Finding Root
- Assignment 2

INTRODUCTION

Wide varieties of problems in science and engineering can be formulated into equation of the form f(x) = 0. The solution process involves finding the value of x that would satisfy the equation f(x) = 0. These values are called roots of the equation.

- **Algebric Function**: Any equation of type y = f(x) is said to be algebraic if it can be expressed in the form: f(x, y) = 0, i.e. the function is dependent between the variables x and y.
  e.g. $3x+5y-21 = 0$; $2x+3xy-25=0$; $x^3-xy-3y^3 = 0$

- **Transcendental Equation**: A non-algebraic equation is called a transcendental equation, which includes trigonometric, exponential and logarithmic functions.
  e.g. $2\sin x-x = 0$; $e^x\cos x-0.5x = 0$; $\log x^2-1 = 0$
  These may have a finite or infinite number of real roots or may not have real root at all.

- **Polynomial Equation**: These are also a simple class of algebraic equation, represented by:
  $$a_n(x^n)+ a_{n-1}(x^{n-1})+ a_{n-2}(x^{n-2})+ \dots + a_1(x^1)+ a_0$$
  This is called nth degree polynomial and has 'n' number of roots. The roots may be:

  i) Real and different; e.g. $x^2 + 4x -5 =0$
  ii) Real and repeated; e.g. $x^2 - 2x +1 = 0$
  iii) Complex numbers; e.g. $x^2 - x + 1 = 0$

EVALUATION OF POLYNOMIALS BY HORNER'S METHOD

All iterative methods require the evaluation of function for which solution is sought. Since, it is a recurring task; the design of an efficient algorithm for evaluating the function assumes a greater importance.

$$\text{As: } f(x) = \sum_{i=0}^{n} a_i x^i = a_0 \sum_{i=1}^{n} a_i x^i$$

Now, let us consider the evaluation of a polynomial using Horner's Rule as:

$$f(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + \dots + a_{n-1}x^{n-1} + a_nx^n$$
$$= (a_0 + x(a_1 + x(a_2 + x(a_3 + \dots + x(a_{n-1} + xa_n) \dots)))$$

Where; innermost expression $a_{n-1} + xa_n$ is evaluated first. Hence, the resulting value constitutes a multiplicand for the expression at the next level. In this way, the number level will be n for n^{th}-order polynomial and consists of total n-addition and n-multiplication.

Algorithm:

$P_n = a_n$
$P_{n-1} = P_n(x) + a_{n-1}$
$P_{n-2} = P_{n-1}(x) + a_{n-2}$
……………………………………………
……………………………………………
$P_j = P_{j+1}(x) + a_j$
……………………………………………
……………………………………………
$P_1 = P_2(x) + a_1$
$P_0 = P_1(x) + a_0 = f(x)$

Example:

Evaluate the polynomial: $f(x) = x^3 - 4x^2 + x + 6$ using Horner's Rule at x = 2.

Solution:

$n = 3$, $a_3 = 1$, $a_2 = -4$, $a_1 = 1$, and $a_0 = 6$

Using algorithm:

$P_3 = a_3 = 1$
$P_2 = P_3(x) + a_2 = 1(2) + (-4) = -2$
$P_1 = P_2(x) + a1 = (-2)(2) +1 = -3$
$P_0 = P_1(x) + a0 = (-3)(2) +6 = 0 = f(x=0)$

- **Linear and Non Linear Equations**: If the dependent variable changes in exact proportion to the changes in independent variables, then the function is linear.

  e.g. $y = f(x) = x + 5$;

  Where, y – dependent parameter and x- independent parameter

  If the response of the dependent variable is not in direct or exact proportion to the changes in independent variable, then the function is non linear.

  e.g. $y = f(x) = x^2 + 1$;

METHOD OF SOLVING NON-LINEAR EQUATIONS

There are different methods to deal with non-linear equations:

i) Direct analytical method
ii) Graphical method
iii) Trial and error method
iv) Iterative method

We only deal with iterative method.

1. <u>Iterative Methods</u>

Iterative methods begin with one or more guesses at the solution being sought. It generates a sequence of estimate of the solution which is expected to converge at the solution. Based on the number of guess, we have two different methods.

A) Bracketing Method: It starts with two initial guesses that brackets the root and then, systematically reduced the width of bracket until the solution is reached. Example: Bisection Method, False Position Method etc.

B) Open End Method: It uses single starting values or two values that do not necessarily bracket the root. Example: Newton Raphson Method with one initial guess, Secant Method with two starting guesses, Fixed Point Iteration Method etc.

BISECTION METHOD

This is one of the bracketing methods and is also known as Bolzano Method, Binary Chopping or Half Interval Method.

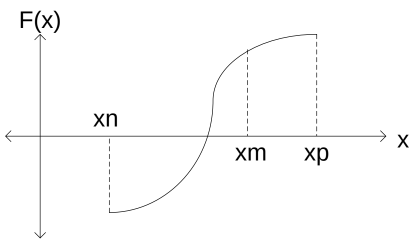


Fig: Bisection Method

Let **f(x)** be the function defined in the interval **[x_n, x_p]**, where **f(x_n)** and **f(x_p)** are of opposite sign i.e.**[f(x_n)*f(x_p)] < 0**, then the root of the function lies in between **x_n** and **x_p.**

Algorithm:

1. Read x_n and x_p , define stopping criteria: EPS
2. Calculate, $x_m = (x_n+x_p)/2$
3. If $|(x_p - x_n)/x_n| <$ EPS, go to step 6 (Optional)
4. Check for $f(x_m)$
   a. If $f(x_m) = 0$, goto step 6
   b. $F(x_m) < 0$, $x_n = x_m$
   c. $F(x_m) > 0$, $x_p = x_m$
5. Repeat steps 2, 3 and 4, until EPS is achieved.
6. Print x_m
7. Stop

Example:
Find the root of the equation x^3-2x-5, using bisection method.

Solution:

x	-5	-4	-3	-2	-1	0	1	2	3	4	5
f(x)	-120	-61	-26	-9	-6	-5	-6	-1	16	51	110

Here, the sign changes occur, when the value of x shifts from +2 to +3. So, $x_n = 2$ and $x_p = 3$

n	x_n	x_p	x_m	$F(x_m)$
1	2	3	2.5	Positive
2	2	2.5	2.25	Positive
3	2	2.25	2.125	Positive
4	2	2.125	2.0625	Negative
5	2.0625	2.125	2.09375	Negative
6	2.09375	2.125	2.109375	Positive
7	2.09375	2.109375	2.1015625	Positive
8	2.09375	2.1015625	2.09765625	Positive

Since, it has not mentioned the value of stopping criteria, so for our convenient stopping criteria, EPS = 0.005.
Since, $x_p = 2.1015625$, $x_n = 2.09375$, i.e. EPS = 0.003
Hence, the root of the equation is $x_m = 2.09765625$ and with rounded off to four significant digits, $x_m = 2.0976$

/Home Work/

Find the root of the equation $x(e^x) = 1$, correct to three decimal places using Bisection Method. **Ans: 0.567**

Convergence of Bisection Method:

In Bisection Method, we choose a mid point x_m in the interval between x_p and x_n. Depending upon the sign of $f(x_m)$, x_p or x_n is set equal to x_m such that the root lies In the interval. In other case the interval containing the root is reduced by a function of 2. The same process is repeated for the new interval. If the procedure is repeated 'n' times, then the interval containing the root is reduced to the size:

$$\frac{x_p - x_n}{2^n} = \frac{\Delta x}{2^n}$$

After iteration the root must lies within $\pm\frac{\Delta x}{2}$ of our estimate, this mean the error bound at n^{th} iteration is:

$$E_n = \frac{\Delta x}{2^n}.$$

Similarly,

$$E_{n+1} = \frac{\Delta x}{2^{n+1}} = \frac{E_n}{2^n}$$

Thus, the error decreases linearly with each step by a factor of 0.5.This method is therefore linearly convergent.

REGULAR FALSI METHOD/FALSE POSITION METHOD

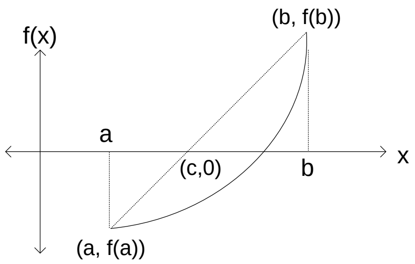


Fig: Regular Falsi Method

Suppose, we are given f(x) = 0. By the method of 'Falsi Position', the geometrical interpretation of is given by two point formula as follows:

i.e. $\quad y - f(a) = \frac{f(b)-f(a)}{b-a}(x - a)$

Or, $\quad 0 - f(a) = \frac{f(b)-f(a)}{b-a}(c - a)$

Or, $\quad c - a = -\frac{f(a)(b-a)}{f(b)-f(a)}$

Or, $\quad c = a - \frac{f(a)(b-a)}{f(b)-f(a)} = \frac{af(b)-af(a)-bf(a)+af(a)}{f(b)-f(a)} = \frac{af(b)-bf(a)}{f(b)-f(a)}$

Algorithm:

1. Find a and b such that f(a)*f(b) < 0.
2. Compute, $c = \frac{af(b) - bf(a)}{f(b) - f(a)}$
3. If f(c) = 0; c is the required root, stop. If f(c) $\neq$ 0, then go to next step.
4. If f(c) < 0; then root lies between (c, b) otherwise root lies between (a, c)
5. Repeat steps 2, 3 and 4 until the root is found to the desired degree of accuracy.

<u>**Example**</u>:

Find the root of the equation (x)log(x)=1; which lies in between 2 and 3 correct to 3 decimal places using Regular Falsi Method.

Solution:

We have: **f(x) = (x)log(x)-1**

As given: a = 2 and b = 3, i.e. the starting values. Now tabulating the data as below:

N	a	f(a)	b	f(b)	c	f(c)
1	2	-0.39794	3	0.43136	2.47985	-0.02188
2	2.47985	-0.02188	3	0.43136	2.50496	-0.00102
3	2.50496	-0.00102	3	0.43136	2.50612	-0.000053
4	2.50612	0.000053	3	0.43136	2.50618	-0.000003

Hence, the root of the equation is: 2.506 (correct to 3-decimal places).

Where; $c = \frac{af(b) - bf(a)}{f(b) - f(a)}$ and when; f(c) < 0, a=c otherwise, b = c;

Convergence of Regular Falsi Method:

In the false position iteration, one of the starting points is fixed while the other moves towards the solution. Assume that the initial points bracketing the solution are 'a' and 'b', where a moves towards the solution and 'b' is fixed as illustrated in following figure.

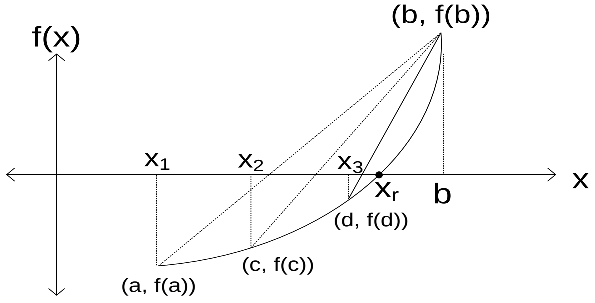


Fig: Convergence of Regular Falsi Method

Let $\mathbf{x_1}$ = '**a**' and '$\mathbf{x_r}$' be the solution. Then:

$\mathbf{e_1} \quad = \quad \mathbf{x_r - x_1}$

$\mathbf{e_2} \quad = \quad \mathbf{x_r - x_2}$

………………………………………

………………………………………

That is:

$\mathbf{e_i} \quad = \quad \mathbf{x_r - x_i}$

In general:

$$\mathbf{e_{i+1}} \quad = \quad \mathbf{e_r} \cdot \frac{(\mathbf{x_r} - \mathbf{b})\mathbf{f''(R)}}{\mathbf{f'(R)}}$$ (Ref: Go online via. Google Search)

NEWTON RAPHSON METHOD

Consider a graph f(x) as shown in the above figure. Let us assume that x_1 is an approximate root of f(x) = 0. Draw a tangent at the curve f(x) at x=x_1 as shown in the figure.

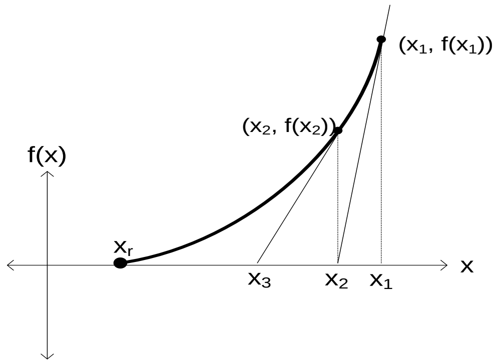


Fig: Newton Raphson Method

The point of intersection of this tangent with the x-axis gives the second approximation to the root. Let, the point of intersection be x_2. The slope of the tangent is given by:

$$\tan(\alpha) = \frac{f(x_1)}{x_1 - x_2} = f'(x_1)$$

Where; $f'(x_1)$ is the slope of f(x) at $x=x_1$.

Solving for x_2, we get:

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}$$

This is called Newton Raphson Formula for first iteration. Then the next approximation will be:

$$x_3 = x_2 - \frac{f(x_2)}{f'(x_2)}$$

In general;

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Algorithm:

1. Assign an initial value to x, say x_0 and read the stopping criteria (EPS).
2. Evaluate $f(x_0)$ and $f'(x_0)$. If $f(x_0) = 0$; then the root will be x_0.
3. Find the improved estimate of x_0 a:

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

4. Check for accuracy of estimate: if $\left|\left(\frac{x_1 - x_0}{x_1}\right)\right| < E$; then stop and go to step 6 otherwise continue.
5. Replace x_0 by x_1 and repeat steps 3 & 4.
6. Print x_1.
7. Stop.

Example:

Find the root of the equation $f(x) = x^2 - 3x + 2$ in the vicinity of x = 0 using Newton Raphson Method correct to 4-decimal places.

Solution:

Here, initial estimate of the root (x_0) = 0;
We have, **$f(x) = x^2 - 3x + 2$**, So, **$f'(x) = 2x - 3$**

At $x_0 = 0$; $f(x_0 = 0) = 2$ and $f'(x_0 = 0) = -3$

Now, for the better approximation, we have the tabular data as follow:

n	**x_0**	**$f(x_0)$**	**$f'(x_0)$**	**x_1**
0	0	2	-3	0.6667
1	0.6667	0.4444	-1.6667	0.9333
2	0.9333	0.0711	-1.1333	0.9961
3	0.9961	0.0032	-1.0078	0.9961

Hence, the root is: 0.9961 (correct to 4-decimal places)
Where: $x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$ and in each iteration, $x_0 = x_1$.

Derivation of Newton Raphson Formula by Taylor's Series Expansion:
Assume that: x_n is an estimate root of the function $f(x) = 0$ and consider a small interval 'h' such that:

$$h = x_{n+1} - x_n \quad \ldots\ldots \text{(i)}$$

Using Taylor's Series Expansion for **$f(x_{n+1})$**, we have:

$$f(x_{n+1}) = f(x_n) + f'(x_n)h + f''(x_n)\frac{h^2}{2} + \ldots$$

Neglecting the terms containing higher derivatives, we get:

$$f(x_{n+1}) = f(x_n) + f'(x_n)h \ \ldots\ldots \text{(ii)}$$

If x_{n+1} is the root of f(x), then: $f(x_{n+1}) = 0$. Putting this value in equation (ii) it gives:

$$0 = f(x_n) + f'(x_n)h$$

i.e. $$h = -\frac{f(x_n)}{f'(x_n)}$$

i.e. $$x_{n+1} - x_n = -\frac{f(x_n)}{f'(x_n)}$$

i.e. $$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Special cases of Newton Raphson Method:

Case 1: Find the solution to compute the sequence root of any positive integer 'N'.
Solution:

As per the case: $x = \sqrt{N}$

i.e. $x - \sqrt{N} = 0$

i.e. $x^2 - N = 0$

i.e. $f(x) = x^2 - N = 0$ and $f'(x) = x^2 - N$

If x_0 be the initial estimation of root, then:

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} = x_0 - \frac{x_0^2 - N}{2x_0} = \frac{x_0^2 + N}{2x_0}$$

i.e. $$x_1 = \frac{1}{2}\left\{x_0 + \frac{N}{x_0}\right\}$$

In general:

$$x_{n+1} = \frac{1}{2}\left\{x_n + \frac{N}{x_n}\right\}$$

Case 2: Show that the P^{th} root of any given positive integer 'N' is:

$$x_{n+1} = \frac{1}{P}\left\{(P-1)x_n + \frac{N}{x_n^{P-1}}\right\}$$

Solution:

As per the case: $x = \sqrt[P]{N} = N^{1/P}$

i.e. $x^P - N = 0$

i.e. $f(x) = x^P - N = 0$ and $f'(x) = Px^{P-1}$

If x_0 be the initial estimation of root, then:

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} = x_0 - \frac{x_0^P - N}{Px_0^{P-1}} = \frac{P x_0^P - x_0^P + N}{Px_0^{P-1}}$$

i.e. $$x_1 = \frac{1}{P}\left\{(P-1)x_0 + \frac{N}{x_0^{P-1}}\right\}$$

In general:

$$x_1 = \frac{1}{P}\left\{(P-1)x_n + \frac{N}{x_n^{P-1}}\right\}$$

Case 3: Show that the reciprocal of the Pth root of any positive number 'N' is:

$$x_{n+1} = \frac{1}{P}\{(P+1)x_n + N(x_n)^{P-1}\}$$

Solution:

As per the case: $x = \frac{1}{\sqrt[P]{N}} = N^{-1/P}$

i.e. $x^{-P} - N = 0$

i.e. $f(x) = x^{-P} - N = 0$ and $f'(x) = -Px^{-P-1}$

If x_0 be the initial estimation of root, then:

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} = x_0 - \frac{x_0^{-P} - N}{-Px_0^{-P-1}} = \frac{-P\,x_0^{-P} - x_0^{-P} - N}{-Px_0^{-P-1}}$$

i.e. $x_1 = \frac{1}{P}\{(P+1)x_0 - N(x_0)^{P+1}\}$

In general:

$$x_{n+1} = \frac{1}{P}\{(P+1)x_n - N(x_n)^{P+1}\}$$

Case 4: Derive the general equation to find the value of reciprocal of any positive integer 'N'.

Solution:

As per the case: $x = 1/N = N^{-1}$

i.e. $x^{-1} - N = 0$

i.e. $f(x) = x^{-1} - N = 0$ and $f'(x) = -x^{-2}$

If x_0 be the initial estimation of root, then:

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} = x_0 - \frac{x_0^{-1} - N}{-x_0^{-2}} = \frac{-x_0^{-1} - x_0^{-1} + N}{-x_0^{-2}}$$

i.e. $x_1 = 2x_0 + Nx_0^{\,2} = x_0(2 + Nx_0)$

In general:

$$x_{n+1} = x_n(2 + Nx_n)$$

Convergence of Newton Raphson Method:

Let x_n be the estimate to the root of the function f(x)=0. If x_n and x_{n+1} are close to each other, then using Taylor's Series Expansion:

$$f(x_{n+1}) = f(x_n) + f'(x_n)(x_{n+1} - x_n) + f''(x_n)\frac{(x_{n+1}-x_n)^2}{2} + \ \dots \text{(i)}$$

Let us assume that, the exact root of f(x) be x_r, then: $x_{n+1} = x_r$, then: $f(x_{n+1})=0$. So, equation (i) becomes:

$$0 = f(x_n) + f'(x_n)(x_r - x_n) + f''(x_n)\frac{(x_r-x_n)^2}{2} \dots \text{(ii)}$$

As we have:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \gg f(x_n) = (x_n - x_{n+1})f'(x_n)$$

Putting the value of $f(x_n)$ into equation (ii), we get:

$$0 = (x_n - x_{n+1})f'(x_n) + f'(x_n)(x_r - x_n) + f''(x_n)\frac{(x_r-x_n)^2}{2}$$

i.e. $$0 = (x_r - x_{n+1})f'(x_n) + f''(x_n)\frac{(x_r-x_n)^2}{2} \dots \text{(iii)}$$

We know that, the error in the estimate x_{n+1} is:

$$e_{n+1} = x_r - x_{n+1}$$

Similarly,

$$e_n = x_r - x_n$$

Now, neglecting the higher power terms and expressing equation (iii) in terms of error:

$$0 = e_{n+1}f'(x_n) + f''(x_n)\frac{(e_n)^2}{2}$$

Rearranging the term, we get:

$$e_{n+1} = -\frac{f''(x_n)}{2f'(x_n)}(e_n)^2$$

This shows that error is roughly proportional to the square of the error at previous iteration. Therefore, Newton Raphson method has quadratic convergence.

Limitation of Newton Raphson Method:

1. Division of zero may occur when f'(x) = 0.
2. If the initial guess is too far away from the required root, the process may converge to some other root.

SECANT METHOD

Secant method uses two initial estimates but does not required that they must bracket the root. It can use two points x_1 and x_2 as shown in the figure below as starting values although they do not bracket the root.

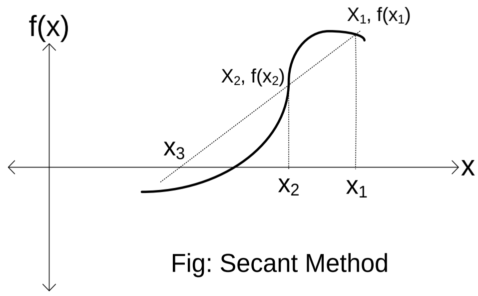


Fig: Secant Method

Slope of secant line passing through x_1 and x_2 is:

$$\frac{f(x_1)}{x_1 - x_3} = \frac{f(x_2)}{x_2 - x_3}$$

On rearranging the terms, we get:

$$x_3 = \frac{f(x_2)x_1 - f(x_1)x_2}{f(x_2) - f(x_1)} \quad \text{.... (i)}$$

On adding and subtracting $f(x_2)x_2$ at the numerator, equation (i) can be represented in the form:

$$x_3 = x_2 - \frac{f(x_2)\,(x_2 - x_1)}{f(x_2) - f(x_1)} \quad \text{... (ii)}$$

Hence, the approximate value of the root can be refined by repeating the process to the desired level of accuracy. So, in general:

$$x_{n+1} = x_n - \frac{f(x_n)\,(x_n - x_{n-1})}{f(x_n) - f(x_{n-1})} \quad \text{.... (iii)}$$

Algorithm:

1. Decide the initial points: x_1 and x_2 and EPS.
2. Compute: $f_1 = f(x_1)$ and $f2 = f(x_2)$
3. Compute: $x_3 = x_2 - \frac{f(x_2)\,(x_2 - x_1)}{f(x_2) - f(x_1)}$
4. Test for accuracy of x_3: if $\left|\frac{x_3 - x_2}{x_3}\right| > EPS$ then:

   Set ($x_1 = x_2$, $f_1 = f_2$) and set ($x_2 = x_3$, $f_2 = f(x_3)$); go to step 3, otherwise set root = x_3.
5. Print root
6. Stop

Example

Use Secant Method to estimate the root of the equation x^2 – 4x -10 = 0, with initial estimate x_1 = 4 and x_2 = 2.

Solution:

$x_1 = 4$, $f(x_1) = f_1 = 4^2 - 4*4 - 10 = -10$

$x_2 = 2$, $f(x_1) = f_1 = 2^2 - 4*2 - 10 = -14$

Let EPS = 0.05

For the better estimation, we have the tabular data as below:

n	x_1	f_1	x_2	f_2	x_3	f_3	EPS
0	4	-10	2	-14	9	35	0.7778
1	2	-14	9	35	4	-10	1.2500
2	9	35	4	-10	5.1111	-4.321	0.2174
3	4	-10	5.1111	-4.321	5.9565	1.6539	0.1419
4	5.1111	-4.321	5.9565	1.6593	5.7225	-0.1429	0.0409

Hence, the root is: 5.7225, which is less than EPS = 0.05

Where EPS = $\left|\frac{x_3 - x_2}{x_3}\right|$ and in each iteration: $x_1 = x_2$, $f_1 = f_2$ and $x_2 = x_3$, $f_2 = f(x_3)$.

Comparison of Secant Iterative Formula with Newton Raphson Formula:

We have, Newton Raphson Formula:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad .... \text{(i)}$$

And Secant Iterative Formula:

$$x_{n+1} = x_n - \frac{f(x_n)\,(x_n - x_{n-1})}{f(x_n) - f(x_{n-1})} \quad ... \text{(ii)}$$

Comparing equation (i) and (ii):

$$f'(x_n) = \frac{f(x_n) - f(x_{n-1})}{(x_n - x_{n-1})}$$

Hence, the major advantage of Secant Method of iteration is no need to evaluate the derivative.

Convergence of Secant Method:

We have, Secant Iterative Formula:

$$x_{n+1} = x_n - \frac{f(x_n)\,(x_n - x_{n-1})}{f(x_n) - f(x_{n-1})} \text{ **Or** } x_{n+1} = \frac{f(x_n)x_{n-1} - f(x_{n-1})x_n}{f(x_n) - f(x_{n-1})} \dots \text{(i)}$$

If x_r be the actual root fo f(x) and e_i be the error in estimate of x_i, then:

$$x_{n+1} = e_{n+1} + x_r$$

$$x_n = e_n + x_r$$

$$x_{n-1} = e_{n-1} + x_r$$

Now, rewriting equation (i) in terms of error with these values, we get:

$$e_{n+1} = \frac{f(x_n)e_{n-1} - f(x_{n-1})e_n}{f(x_n) - f(x_{n-1})} \dots \text{(ii)}$$

According to Mean Value Theorem, if x = R_n lie in the interval x_n and x_r, then:

$$f'(R_n) = \frac{f(x_n) - f(x_r)}{x_n - x_r} \dots. \text{(iii)}$$

As obvious: $f(x_r) = 0$ and $x_n - x_r = e_n$. Therefore, equation (ii) becomes:

$$f'(R_i) = \frac{f(x_n)}{e_n} \gg f(x_n) = e_n.f'(R_n)$$

Similarly: $$f(x_{n-1}) = e_{n-1}.f'(R_{n-1})$$

Substituting these values in the numerator of equation (ii), we get:

$$e_{n+1} = e_n.e_{n-1}\frac{f'(R_n) - f'(R_{n-1})}{f(x_n) - f(x_{n-1})} \dots \text{(iv)}$$

i.e. $e_{n+1} \propto e_n.e_{n-1} \dots \text{(v)}$

As we also know that, if the order of convergence of an iterative process is P, then:

$$e_n \propto {e_{n-1}}^P \quad \text{i.e.} \quad e_{n+1} \propto {e_n}^P \dots \text{(vi)}$$

Now, from equation (iv) and (v):

$$e_n^{\,P} \propto {e_{n-1}}^P.e_{n-1} \gg e_n \propto {e_{n-1}}^{(P+1)/P} \dots \text{(vii)}$$

Finally, from equation (vi) and (vii):

$$P = \frac{(P+1)}{P} \gg P^2 - P - 1 = 0 \gg P = \frac{1 \pm \sqrt{5}}{2} = +1.618$$

So, the order of convergence of Secant Method is 1.618, which is a slow rate (<2). It also requires previous two iterative to estimate the new one. These are recognized as drawbacks of Secant Method.

ITERATION METHOD:

It is also known as, direct substitution method or method of fixed point iteration or method of successive approximation. It is applicable if the equation f(x) = 0 can be expressed in terms of x = g(x). If x_0 is the initial approximation to the root, then the next approximation will be: $x_1 = g(x_0)$ and the next again will be: $x_2 = g(x_1)$ and so on.

In general: $x_{n+1} = g(x_n)$

Where, x = g(x) is known as fixed point equation.

Algorithm:

1. Read an initial guess: x_0 and EPS
2. Evaluate: $x_1 = g(x_0)$
3. Check for error: if $\left|\frac{x_1-x_0}{x_1}\right|$ <EPS, x_1 is root, else continue.
4. Assign the value of x_1 to x_0, i.e. $x_0 = x_1$ and repeat steps 2 & 3.
5. Print x_1.
6. Stop

Example:

Evaluate the square root of 5 using Fixed Point Iterating method.

Solution:

As, $x = \sqrt{5} \gg x^2 - 5 = 0 \gg x = \frac{5}{x} = g(x)$

Let, $x_0 = 1$, then:

$g(x_0) = 5/1 = 5 = x_1$

$g(x_1) = 5/5 = 1 = x_2$

$g(x_2) = 5/1 = 5 = x_3$

$g(x_3) = 5/5 = 1 = x_4$ and so on.

Here, the solution does not converge to the solution, this type of divergence is known as oscillatory divergence.

Let's try another form:

Adding x on both sides of $x^2 - 5 = 0$, we get: $x^2 + x - 5 = x$ =g(x)

Let $x_0 = 0$, then:

$g(x_0) = -5 = x_1$

$g(x_1) = (-5)^2+(-5)-5=15 = x_2$

$g(x_3) = 15^2+15-5=235 = x_4$

$g(x_4) = 235^2+235-5=55455= x_5$ and so on.

This is also divergent i.e. does not converge to the solution. This type of divergence is known as monotonic divergence.

Again Let's try another form.

As x = 5/x can also be written in the form: $2x = \left(\frac{5}{x}\right) + x \gg x = \frac{x+\frac{5}{x}}{2} = g(x)$

For $x_0 = 1$,

$g(x_0) = 3 = x_1$

$g(x_1) = 2.3333 = x_2$

$g(x_2) = 2.2381 = x_3$

$g(x_3) = 2.2361 = x_4$

$g(x_4) = 2.2361 = x_5$

Now, the process converges rapidly to the solution. Hence, the root of 5 is 2.2361.

Convergence of Iteration Method:

The iteration formula is:

$$x_{i+1} = g(x_i) \text{ ..... (i)}$$

If x_f be the real root of the equation, then:

$$x_f = g(x_f) \text{ ..... (ii)}$$

Substituting equation (i) from equation (ii), we get:

$$x_f - x_{i+1} = g(x_f) - g(x_i) \text{.... (iii)}$$

According to mean value theorem: if x = R lies in the interval between x_f and x_i , then:

$$g'(R) = \frac{g(x_f)-g(x_i)}{x_f-x_i} \gg g(x_f) - g(x_i) = g'(R)(x_f - x_i) \text{ ... (iv)}$$

Now, from equation (iii) and (iv)

$$x_f - x_{i+1} = g'(R)(x_f - x_i) \text{ .... (v)}$$

If e_i is the error in i^{th} iteration then equation (v) in terms of error will be:

$$e_{i+1} = g'(R)e_i \text{ .... (vi)}$$

This shows that, the error will decrease with each iteration iff g'(R) < 1. Hence, the Iteration Method converges only when |g'(x)|<1.

HORNER'S METHOD FOR FINDING ROOTS

- Historic name for synthetic division is Horner's Method.
- Let positive root of f(x) = 0 lies in between $\boldsymbol{\alpha}$ and $\boldsymbol{\alpha+1}$, where $\boldsymbol{\alpha}$ is an integer. Then the value of the root will be $\boldsymbol{\alpha.d_1d_2d_3}$ ... , where d_1, d_2 ... are digits in their decimal parts.
- To find the value of d_1, first diminish the roots of f(x) = 0 by $\boldsymbol{\alpha}$. For example, if root lies in between 0 and 1 then root will be $\mathbf{0.\ d_1d_2d_3.}$ And make the transformed equation.
- Multiply the root of the equation by the order of 10 and diminish other decimal parts and so on.
- Continue the process to obtain the root of the equation at any desired degree of accuracy by digit.

Example:

Find the root of the equation $x^3 + 9x^2 - 18 = 0$ to two decimal digit using Horner's Method.

Solution:

As: $x^3 + 9x^2 - 18 = 0 = f_1(x)$ ... (i)

So, $f(x_1)*f(x_2) < 0$ is analysed using following table.

x	**-5**	**-4**	**-3**	**-2**	**-1**	**0**	**1**	**2**	**3**	**4**	**5**
f(x)	82	62	36	10	-10	-18	-8	26	90	190	332

So, root lies in between 1 and 2 as f(x) shifted from negative to positive at these values. Hence, the integer part of the root is: 1. d_1d_2 up to two decimal places.

Now, we are going to diminish the root of the equation by 1 using synthetic division as follow.

Coeff.	x^3	x^2	x^1	x^0
1)	1	9	0	-18
	0	1	10	10
1)	1	10	10	(-8)
	0	1	11	
1)	1	11	(21)	
	0	1		
	(1)	(12)		

Now, the transformed equation is: $x^3 + 12x^2 + 21x - 8 = 0$. Multiplying the coefficients by the order of 10, we get: $x^3 + 120x^2 + 2100x - 8000 = 0 = f_2(x)$... (ii)

Again finding d_1 by following table:

x	-4	-3	-2	-1	0	1	2	3	4
f(x)	-14544	-13247	-11728	-9981	-8000	-5779	-3312	-593	2384

Here, $f_2(x=3) * f_2(x=4) < 0$. So, d_1 must be 3. Hence, the integer part updated to $1.3d_2$. Now again, diminishing the root by 3 using synthetic division.

Coeff.	x^3	x^2	x^1	x^0
3)	1	120	2100	-8000
	0	3	369	10
3)	1	123	2469	(-593)
	0	3	378	
3)	1	126	(2847)	
	0	3		
	(1)	(129)		

Again the transformed equation is: $x^3 + 129x^2 + 2847x - 593 = 0$. Again, multiplying the coefficients by the order of 10, we get: $x^3 + 1290x^2 + 284700x - 593000 = 0 = f_3(x)$... (iii)

Finally, for d_2, we have the following table:

X	-3	-2	-1	0	1	2	3
f(x)	-1435517	-1157248	-876411	-593000	-307009	-18432	272737

Here $f_3(x=2) * f_3(x=3) < 0$. So, d_3 must be 2.
Finally, the root of the equation $f(x) = 0$ is: 1.32

Assignment: 2

Full Marks: 50
Pass Marks: 25

1. (a) Evaluate the polynomial $f(x) = 5x^5 + 4x^4 + 3x^3 + 2x^2 + x + 12$ by Horner's Method at x=1.5 [5]

   (b) Use Fixed Point Iteration Method to evaluate: $x^3 + 2x^2 + x = 1$, correct to four significant figures. [5]

2. Use Bisection Method to evaluate $f(x) = x^3 + x^2 + x + 7$ for :

   (a) EPS = 0.05, [5]

   (b) EPS = 0.001 and compare the result. [5]

3. Use False Position Method to evaluate $f(x) = x^3 - x^2 - 1$ correct to:

   (a) 2 decimal places [5]

   (b) 3 decimal places and compare the result. [5]

4. Use Newton Raphson Method to evaluate following function correct to 3 decimal places:

   (a) $\mathrm{f(x)} = \mathrm{x} + \mathrm{logx} - 2$ [5]

   (b) $\mathrm{f(x)} = \mathrm{x} - \frac{1}{\sqrt[p]{\mathrm{N}}}$ & find the value of $\frac{1}{\sqrt[4]{7}}$ [5]

5. Use Secant Method to evaluate correct to 3 decimal places:

   (a) $\mathrm{xe^x} - 1 = 0$ & compare the output with the true value of x as 0.567143. [5]

   (b) $\mathrm{x}^{2.2} = 69$ [5]

3

CHAPTER

Interpolation

Contents:

- Introduction
- Errors in Polynomial Interpolation
- Interpolation with Equally Spaced Values
  - Finite Difference Method
    - Forward Difference Method
    - Backward Difference Method
    - Central Difference Method
  - Difference of Polynomial
- Interpolation with Unequally Spaced Values
  - Linear Interpolation Formula
  - Lagrange Interpolation Formula
  - Newton Interpolation Polynomial
- Inverse Interpolation
- Divided Differences
- Assignment 3

INTERPOLATION

If a function, say f(x) is constructed, such that it passes through all the set of data points and then evaluating f(x) for the specified value of x is known as interpolation. This method of constructing a function gives the estimation of values at non tabular print. The functions are known as interpolation polynomials.

Most common form of n^{th} order polynomials are:

1. $P(x) = a_0 + a_1x + a_2x^2 + \cdots + a_{n-1}x^{n-1} + a_nx^n$ ... (i), this is known as n^{th} order polynomial in **Power Form**.

2. $P(x) = a_0 + a_1(x-c) + a_2(x-c)^2 + \cdots + a_{n-1}(x-c)^{n-1} + a_n(x-c)^n$...(ii), this is known as n^{th} order polynomial in **Shifted Power Form**, and

3. $P(x) = a_0 + a_1(x-c_1) + a_2(x-c_1)(x-c_2) + \cdots + ((x-c_1)(x-c_2)\ldots(x-c_n))$ ...(iii), this is known as n^{th} order polynomial in **Newton Form**, which reduces to shifted power form, when $c_1 = c_2 = c_3 = \ldots = c_n$ and reduces to simple power form, when $c_i = 0$ for all i.

ERRORS IN POLYNOMIAL INTERPOLATION

Let the function y(x), defined by the (n+1) points (x_i, y_i), i = 0, 1, 2 ... n, be the continuous and differentiable (n+1) times and let y(x) be approximated by a polynomial $\emptyset(x)$ of degree not exceeding 'n' such that:

$$\emptyset_n(x_i) = y_i,\ i = 0, 1 \ldots n$$

Then error is given by: $e = y(x) - \emptyset_n(x) = \frac{\pi_{n+1}(x)}{(n+1)!}y^{n+1}(\gamma), \qquad x_0 < \gamma < x_n$.

Where, $\pi_{n+1}(x) = (x-x_0)(x-x_1)\ldots.(x-x_n)$ and if $L = \frac{y^{n+1}(\gamma)}{(n+1)!}$ then:

$$e = y(x) - \emptyset_n(x) = L.\pi_{n+1}(x) \ldots (i)$$

Since, y(x) is generally unknown and hence, we do not have any information concerning $y^{(n+1)}(x)$. It is almost useless in practical computation. But, on the other hand, it is extremely useful in theoretical work in different branches of numerical analysis. It is useful in determining errors in Newton Interpolating Formula.

FINITE DIFFERENCES: INTERPOLATION WITH EQUALLY SPACED VALUES

Assume that, we have a table of values: (x_i, y_i), $(i = 0, 1, 2, \ldots n)$ of any function $y = f(x)$, the values of x being equally spaced, i.e. $x_i = x_0 + ih$, $(i = 0, 1, 2, \ldots n)$. The method of finding the solution of f(x) for some 'x' in the range $x_0 \le x \le x_n$ is based on the concept of such finite differences.

Forward Difference:

If $y_0, y_1, \ldots Y_n$ denote a set of values of y at $x_0, x_1, \ldots x_n$ respectively, then $(y_1 - y_0)$, $(y_2 - y_1)$, ... $(y_n - y_{n-1})$ are called difference of y; i.e. $\Delta y_0 = y_1 - y_0$, $\Delta y_1 = y_2 - y_1$, ... $\Delta y_{n-1} = y_n - y_{n-1}$, where Δ is the forward difference operator. Usually, the functions are tabulated at equal intervals, i.e. $x_1 - x_0 = x_2 - x_1 = x_3 - x_2 \ldots x_n - x_{n-1} = na$. Here, with tabulation at equal intervals a difference table for 'n' points may be expressed as shown below:

x	f(x)	Δf	$\Delta^2 f$	$\Delta^3 f$	$\Delta^4 f$	....
x_0	$\mathbf{f_0}$					
		$\mathbf{\Delta f_0}$				
$x_0 + h$	f_1		$\mathbf{\Delta^2 f_0}$			
		Δf_1		$\mathbf{\Delta^3 f_0}$		
$x_0 + 2h$	f_2		$\Delta^2 f_1$		$\mathbf{\Delta^4 f_0}$	....
		Δf_2		$\Delta^3 f_1$		
$x_0 + 3h$	f_3		$\Delta^2 f_2$	....	....	
		Δf_3	....	....	....	
$x_0 + 4h$	f_4	....	....	....	....	
....	....	....	....	....	....	

Here 'h' is the uniform difference in the value of 'x'.

$\Delta f_0 = f_1 - f_0, \Delta f_1 = f_2 - f_1, \ldots$

In general: $(\Delta f_i = f_{i+1} - f_i)$, which is called first forward difference.

Similarly:

$\Delta^2 f_0 = \Delta(\Delta f_0) = \Delta f_1 - \Delta f_0 = f_2 - f_1 - (f_1 - f_0) = f_2 - 2f_1 + f_0$

In general: $(\Delta^2 f_i = f_{i+2} - 2f_{i+1} + f_i)$, which is called second forward difference and so on.

For the polynomial that passes through a group of equidistant point i.e. Newton Gregory Forward Polynomial, we write in terms of the index 's', such that $s = (x-x_0)/h$.

$$\therefore P_n(x) = f_0 + s\frac{\Delta f_0}{1!} + s(s-1)\frac{\Delta^2 f_0}{2!} + s(s-1)(s-2)\frac{\Delta^3 f_0}{3!} + \ldots$$

This equation for polynomial is called Newton Gregory Forward Interpolation Formula. It is applied when the required point is close to the start of the table.

Example:

For the given table, find the value of f(0.16).

x	0.1	0.2	0.3	0.4
f(x)	1.005	1.020	1.045	1.081

Solution:

$x_0 = 0.1$, $x_1 = 0.2$ … i.e. $h = x_1 - x_0 = 0.2 - 0.1 = 0.1$

We have given, x = 0.16. So: $s = (x - x_0)/h = (0.16 - 0.1)/0.1 = 0.6$

Now, generating the Forward Difference Table:

X	**f(x)**	Δ**f**	Δ^2**f**	Δ^3**f**
0.1	**1.005**			
		0.015		
0.2	1.020		**0.01**	
		0.025		**0.001**
0.3	1.045		0.011	
		0.036		
0.4	1.081			

Now:

$$P_3(x) = f_0 + s\frac{\Delta f_0}{1!} + s(s-1)\frac{\Delta^2 f_0}{2!} + s(s-1)(s-2)\frac{\Delta^3 f_0}{3!}$$

$$P_3(x) = 1.005 + 0.6 \times 0.015 + 0.6(0.6-1)\frac{0.01}{2!} + 0.6(0.6-1)(0.6-2)\frac{0.001}{3!}$$

$$P_3(x) = 1.0128$$

Backward Difference:

If the required point is close to the end of the table, we can use another formula known as Newton Gregory Backward Difference Formula. Here, the reference point is: 'x_n' instead of x_0. Therefore, $s = (x - x_n)/h$.

So, the Newton Gregory Formula is given by:

$$\therefore P_n(x) = f_n + s\frac{\nabla f_n}{1!} + s(s+1)\frac{\nabla^2 f_n}{2!} + s(s+1)(s+2)\frac{\nabla^3 f_n}{3!} + \ \ldots$$

The table for the backward difference will be identical to the forward difference table. However, the reference point will be below the point for which the estimate is required. So, the value of 's' will be negative for backward interpolation. The backward difference table is shown below.

X	f(x)	∇f	$\nabla^2 f$	$\nabla^3 f$	$\nabla^4 f$	....
x_0	f_0					
		∇f_1				
$x_0 + h$	f_1		$\nabla^2 f_2$			
		∇f_2		$\nabla^3 f_3$		
$x_0 + 2h$	f_2		$\nabla^2 f_3$		$\nabla^4 f_4$	....
		∇f_3		$\nabla^3 f_4$		
$x_0 + 3h$	f_3		$\nabla^2 f_4$	....	....	
		∇f_4	....	....	....	
$x_0 + 4h$	f_4	....	....	....	....	
....	....	....	....	....	....	

Where, ∇ is a backward difference operator, and

Here 'h' is the uniform difference in the value of 'x'.
$\nabla f_1 = f_1 - f_0, \nabla f_2 = f_2 - f_1, \ldots.$
In general: $(\Delta f_i = f_i - f_{i-1})$, which is called first backward difference.
Similarly:
$\nabla^2 f_1 = \nabla(\nabla f_1) = \nabla f_1 - \nabla f_0 = f_2 - f_1 - (f_1 - f_0) = f_2 - 2f_1 + f_0$
In general: $(\nabla^2 f_i = f_{i+1} - 2f_i + f_{i-1})$, which is called second backward difference and so on.

Example:
Estimate the value of sin(45°) using backward difference method with the following set of data.

x	10	20	30	40	50
f(x)=sin(x)	0.1736	0.342	0.5	0.6428	0.768

Solution:
$x_0 = 10, x_1 = 20$ ... i.e. $h = x_1 - x_0 = 20 - 10 = 10$
We have given, x = 45. So: $s = (x - x_n)/h = (45 - 50)/10 = -0.5$

$$\therefore P_4(x) = f_4 + s\frac{\nabla f_4}{1!} + s(s+1)\frac{\nabla^2 f_4}{2!} + s(s+1)(s+2)\frac{\nabla^3 f_4}{3!} + s(s+1)(s+2)(s-4)\frac{\nabla^4 f_4}{4!}$$

Where, the necessary data are fetched from the following Backward Difference Table.

Backward Difference Table:

x	**f(x)**	**∇f**	**∇^2f**	**∇^3f**	**∇^4f**
10	0.1736				
		0.1684			
20	0.342		−0.0104		
		0.158		0.0048	
30	0.5		−0.0152		0.0024
		0.1428		0.0024	
40	0.6428		−0.0176		
		0.1252			
50	0.768				

$$\therefore P_4(x) = 0.768 - 0.5 \times \frac{0.1252}{1!} - 0.5 \times (-0.5+1)\frac{(-0.0176)}{2!} - 0.5 \times (-0.5+1)(-0.5+2)\frac{0.0024}{3!} - 0.5 \times (-0.5+1)(-0.5+2)(-0.5+3)\frac{0.0024}{4!}$$

$$\therefore P_4(x) = 0.768 + 0.0022 - 0.00015 - 0.00009375 = 0.70735625$$

Difference of a polynomial:

As, we have discussed difference table by forward and backward difference method. Let us analyse, what actually differs in the first, second and ... difference in their degrees.

If y(x) be a polynomial of n^{th} degree:

i.e; $y(x) = a_0x^n + a_1x^{n-1} + a_2x^{n-2} + \cdots a_{n-1}x + a_n,$

Then we obtain:

$$y(x+h) - y(x) = a_0[(x+h)^n - x^n] + a_1[(x+h)^{n-1} - x^{n-1}] + \cdots$$
$$= a_0[(nh)x^{n-1}] + a_1^{\prime[x^{n-1}]} + \cdots + a_n'$$

Where, a_1', a_2' ... are new coefficients.

Then:

$$\Delta y(x) = a_0[(nh)x^{n-1}] + a_1'[x^{n-1}] + \cdots + a_n'$$

This shows that the first difference of polynomial of the nth degree is a polynomial of degree (n-1) and so on.

Where;

Coefficient of	x^{n-1}	is	$a_0(nh)$	in first difference.
Coefficient of	x^{n-2}	is	$a_0 n(n-1)h^2$	in second difference.
………………	…..	…	……………	……………………….
………………	…..	….	……………	…………………………
Coefficient of	x^0	is	$a_0 n! h^n$	in n^{th} difference.

Newton's Formula for Interpolation

We had used Newton Gregory Formula for two different methods:

a) Newton's Gregory Formula for Forward Difference, and

$$P_n(x) = f_0 + s\frac{\Delta f_0}{1!} + s(s-1)\frac{\Delta^2 f_0}{2!} + s(s-1)(s-2)\frac{\Delta^3 f_0}{3!} + \ \ldots$$

Where, $s = (x-x_0)/h$; 'x' is referred near to the start of the table and 'h' is a interval constant

b) Newton's Gregory Formula for Backward Difference

$$P_n(x) = f_n + s\frac{\nabla f_n}{1!} + s(s+1)\frac{\nabla^2 f_n}{2!} + s(s+1)(s+2)\frac{\nabla^3 f_n}{3!} + \ \ldots$$

Where, $s = (x-x_n)/h$; 'x' is referred near to the end of the table and 'h' is a interval constant

Central Difference Method

If interpolation is desired near the beginning or end of the table, there is no alternative to Newton's Forward and Backward Difference Formula respectively, simply because higher order differences do not exist at the beginning or end of the table values.

For interpolation near the middle of the table, Stirling's Formula gives the most accurate result for $-\frac{1}{4} \le P \le \frac{1}{4}$ and Bessel's Formula is most efficient near $P = \frac{1}{2}$, say $\frac{1}{4} \le P \le \frac{3}{4}$. But in the case, where a series of calculations have to made, it would be inconvenient to use both of these formulae and a choice must be made between them. Everett's Formula may be gainfully employed for the aforesaid reason.

LINEAR INTERPOLATION

The simplex form of interpolation is to approximate two data points by straight line. Suppose, we are given two points, $[x_1, f(x_1)]$ and $[x_2, f(x_2)]$, these two points can be connected linearly as shown in the figure below.

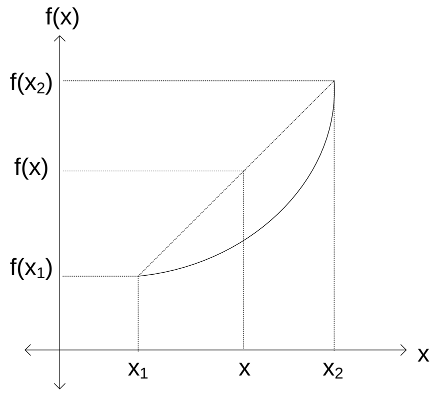


Fig: Representation of Linear Interpolation Method

Using the concept of similar triangle, we have:

$$\frac{f(x)-f(x_1)}{(x-x_1)} = \frac{f(x_2)-f(x_1)}{(x_2-x_1)}$$

Solving for f(x), we get:

$$f(x) = f(x_1) + (x - x_1)\left\{\frac{f(x_2)-f(x_1)}{(x_2-x_1)}\right\} \dots \text{(i)}$$

This equation is known as Linear Interpolation Formula, where, $\frac{f(x_2)-f(x_1)}{(x_2-x_1)}$ represents the slope of the line. Comparing equation (i) with Newton form of polynomial of first order given by: $P_n(x) = a_0 + a_1(x-c_1)$, we get:

$C_1 = x_1$, $\quad a_0 = f(x_1) \quad$ and $\quad a_1 = \frac{f(x_2)-f(x_1)}{(x_2-x_1)}$

Hence, the co-efficient 'a_1' represent the first derivative of the given function.

Example:
Given the table of data:

x	1	2	3	4	5
f(x)	1	1.4142	1.7321	2	2.2361

Determine the square root of 2.5.

Solution:
Since, the given value 2.5 lie in between the point 2 and 3. So, $x_1 = 2$ and $x_2 = 3$ whereas x=2.5.
Now:

$$f(x) = f(x_1) + (x - x_1)\left\{\frac{f(x_2)-f(x_1)}{(x_2-x_1)}\right\}$$

i.e. $$f(2.5) = f(2) + (2.5 - 2)\left\{\frac{f(3)-f(2)}{(3-2)}\right\}$$

i.e. $$f(2.5) = 1.4142 + 0.5\left\{\frac{1.7321-1.4142}{1}\right\} = 1.5732$$

Again, if we choose, $x_1 = 2$ and $x_2 = 4$, then we get: f(2.5) = 1.5607
Since, the true value of square root of 2.5 from calculator is 1.5811 so, error produced on each cases will be:

Error_1 = |1.5811-1.5732| = 0.0079
Error_2 = |1.5811-1.5607| = 0.0204

Here, Error_2 > Error_1. Hence, it can be concluded that: smaller the interval between the interpolation data points, the better will be the approximation.

LAGRANGE'S INTERPOLATION POLYNOMIAL FORMULA:

Let (x_0, f_0), (x_1, f_1), (x_2, f_2) … (x_n, f_n) be the (n+1) points. To derive the formula for the polynomial of degree 'n', let us consider a second order polynomial of the form:

$$P_2(x) = b_1(x - x_0)(x - x_1) + b_2(x - x_1)(x - x_2) + b_3(x - x_2)(x - x_0) \quad \text{... (i)}$$

If (x_0, f_0), (x_1, f_1) and (x_2, f_2) are three interpolations then:

$$P_2(x_0) = b_2(x_0 - x_1)(x_0 - x_2) = f_0$$
$$P_2(x_1) = b_3(x_1 - x_2)(x_1 - x_0) = f_1$$
$$P_2(x_2) = b_1(x_2 - x_0)(x_2 - x_1) = f_2 \quad \text{..... (ii)}$$

Substituting the values of b_1, b_2 and b_3 from equation (iii) to equation (i), we get:

$$P_2(x) = \frac{f_2}{(x_2-x_0)(x_2-x_1)}(x-x_0)(x-x_1) + \frac{f_0}{(x_0-x_1)(x_0-x_2)}(x-x_1)(x-x_2) + \frac{f_1}{(x_1-x_2)(x_1-x_0)}(x-x_2)(x-x_0)$$

i.e. $$P_2(x) = f_0\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + f_1\frac{(x-x_2)(x-x_0)}{(x_1-x_2)(x_1-x_0)} + f_2\frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} \ \ldots \text{(iii)}$$

Now, equation (iii) may be represented as:

$$P_2(x) = f_0 l_0(x) + f_1 l_1(x) + f_2 l_2(x) \ \ldots \text{(iv)}$$

Where:

$$l_0(x) = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}, l_1(x) = \frac{(x-x_2)(x-x_0)}{(x_1-x_2)(x_1-x_0)}, l_2(x) = \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}$$

Equation (iv) second degree Lagrange's Polynomial. For Lagrange's Polynomial of n^{th} degree, we should have given (n+1) interpolating points. Then the polynomial will be:

$$P_n(x) = f_0 l_0(x) + f_1 l_1(x) + f_2 l_2(x) + \ \ldots. + f_{n-1} l_{n-1}(x) + f_n l_n(x) \ \ldots \text{(iv)}$$

This can be generally expressed as:

$$P_n(x) = \sum_{i=0}^{n} f_i l_i(x) \quad \ldots \text{(v)}$$

Where;

$$l_i(x) = \prod_{j=0, j\neq i}^{n} \left(\frac{x-x_j}{x_i-x_j}\right) \ \ldots \text{(vi)}$$

Here, equation (v) is called Lagrange's Interpolation Polynomial of degree 'n' and then equation (vi) is called Lagrange's Basic Polynomial.

Example:

Find the Lagrange's Interpolation Polynomial to fit the following data. Use polynomial to estimate the value of $e^{1.5}$.

i	0	1	2	3
x_i	0	1	2	3
$e^{x_i} - 1$	0	1.7183	6.3891	19.0855

Solution:

$$l_0(x) = \frac{(x-1)(x-2)(x-3)}{(0-1)(0-2)(0-3)} = \frac{x^3-6x^2+11x-6}{-6}$$

$$l_1(x) = \frac{(x-2)(x-3)(x-0)}{(1-2)(1-3)(1-0)} = \frac{x^3-5x^2+6x}{2}$$

$$l_2(x) = \frac{(x-3)(x-0)(x-1)}{(2-3)(2-0)(2-1)} = \frac{x^3-4x^2+3x}{-2}$$

$$l_3(x) = \frac{(x-0)(x-1)(x-2)}{(3-0)(3-1)(3-2)} = \frac{x^3-3x^2+2x}{6}$$

$$P_3(x) = f_0 l_0(x) + f_1 l_1(x) + f_2 l_2(x) + f_2 l_2(x)$$

$$= 0 + 1.7183\left\{\frac{x^3-5x^2+6x}{2}\right\} + 6.3891\left\{\frac{x^3-4x^2+3x}{-2}\right\} + 19.0855\left\{\frac{x^3-3x^2+2x}{6}\right\}$$

$P_3(x) = 0.84455x^3 - 1.0604x^2 + 1.9331x$; This is the required polynomial.

$$P_3(1.5) = 0.84455(1.5)^3 - 1.0604(1.5)^2 + 1.9331(1.5) = 3.3677$$

Now:

$e^{1.5} = P(1.5) + 1 = 3.3677 + 1 = 4.3677$

Concept of Inverse Interpolation

For a given set of values of x and y, the process of finding the value of x for a certain value of y is called inverse interpolation. When the values of x are at unequal intervals, the way of performing inverse interpolation is by interchanging x and y in Lagrange's method.

Inverse interpolation is meaningful only if the function is single valued in the interval. When the values of x are equally spaced, the method of successive approximation is used.

Divided Differences

The Lagrange's Interpolation Formula has the disadvantage that if another interpolation point were added, then the interpolation coefficients $l_i(x)$ will have to be recomputed; we therefore seek an interpolation polynomial which has the property that a polynomial of higher degree may be derived from it by simply adding new terms.

Newton general interpolation formula is one such formula, which employs what are called divided differences. It is our principle purpose to define sch differences and discuss certain of their properties to obtain the basic formula.

We have Newton's form of polynomial as:

$$f(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \cdots . + a_{n-1}(x - x_0)(x - x_1) \ldots (x - x_{n-1})$$

In this polynomial, coefficients can be calculated using divided difference table as given below.

i	**x_i**	**$f(x_i)$**	**IDD**	**IIDD**	**IIIDD**	**IVDD**
0	x_0	$f(x_0)$				
			$f(x_0, x_1)$			
1	x_1	$f(x_1)$		$f(x_0, x_1, x_2)$		
			$f(x_1, x_2)$		$f(x_0, x_1, x_2, x_3)$	
2	x_2	$f(x_2)$		$f(x_1, x_2, x_3)$		$f(x_0, x_1, x_2, x_3, x_4)$…
			$f(x_2, x_3)$		$f(x_1, x_2, x_3, x_4)$	
3	x_3	$f(x_3)$		$f(x_2, x_3, x_4)$		
			$f(x_3, x_4)$			
4	x_4	$f(x_4)$				
…						

If (x_0,y_0), (x_1,y_1)…. (x_n,y_n) be the given (n+1) points, then the divided difference of order 1, 2, … n are defined by the relation>

$f(x_0) = a_0$

$$f(x_0, x_1) = \frac{f(x_1)-f(x_0)}{x_1-x_0} = a_1 \qquad f(x_1, x_2) = \frac{f(x_2)-f(x_1)}{x_2-x_1} \ldots..$$

$$f(x_0, x_1, x_2) = \frac{\frac{f(x_2)-f(x_1)}{x_2-x_1} - \frac{f(x_1)-f(x_0)}{x_1-x_0}}{x_2-x_0} = a_2 \qquad f(x_0, x_1, x_2) = \frac{\frac{f(x_3)-f(x_2)}{x_3-x_2} - \frac{f(x_2)-f(x_1)}{x_2-x_1}}{x_3-x_1} \ldots.$$

Example:

A table of polynomial is given below. Fit the polynomial and find the value of f(x) at x = 2.5. Use divided difference table.

x	-3	-1	0	3	5
f(x)	-30	-22	-12	330	3458

Solution:

Let's build up the divided difference table as follow:

i	x_i	$f(x_i)$	IDD	IIDD	IIIDD	IVDD
0	-3	-30				
			4			
1	-1	-22		2		
			10		4	
2	0	-12		26		5
			114		44	
3	3	330		290		
			1564			
4	5	3458				
...						

From table, a_0 = -30, a_1 = 4, a_2 = 2, a_3 = 4 and a_4 = 5

Therefore,

f(x) = -30+4(x+3)+2(x+3)(x+1)+4(x+3)(x+1)(x-0)+5(x+3)(x+1)(x-0)(x-3)

i.e. f(2.5)= -30+4(2.5+3)+2(2.5+3)(2.5+1)+4(2.5+3)(2.5+1)(2.5-0)+5(2.5+3)(2.5+1)(2.5-0)(2.5-3)

Example:2 (Home Work)

Given the following set of data points, obtain the table of divided differences. Use the table to estimate the value of f(1.5)

i	**0**	**1**	**2**	**3**	**4**
x_i	**1**	**2**	**3**	**4**	**5**
$f(x_i)$	**0**	**7**	**26**	**63**	**124**

Solution:

a_0 = 0, a_1 = 7, a_2 = 6, a_3 = 1 and a_4 = 0

f(1.5) = 2.375

Prepared By
Er. Shree Krishna Khadka

Assignment: 3

Full Marks: 40
Pass Marks: 20

1. From the following table of values of x and f(x), determine

   (a) f(0.23) [5]
   (b) f(0.29) [5]

x	0.20	0.22	0.24	0.26	0.28	0.30
f(x)	1.6596	1.6698	1.6804	1.6912	1.7024	1.7139

2. Use linear interpolation method to calculate the square root of 4.5 from following table.

   (a) Take two initial values as 4 and 5 [5]
   (b) Take two initial values as 3 and 6
   Compare the result of (a) and (b).

x	1	2	3	4	5	6
f(x)	1	1.4142	1.7321	2	2.2361	2.4494

3. Applying Lagrange's Interpolation Formula, find a cubic polynomial which approximates the following data. [10]

x	-2	-1	2	3
f(x)	-12	-8	3	5

   Also find the polynomial values at -2.5 and 2.5.

4. Given the table of values

x	50	52	54	56
f(x)=$\sqrt[3]{x}$	3.684	3.732	3.779	3.825

   Use Lagrange Interpolation to find x when $\sqrt[3]{x} = 3.756$ [10]
   (Example: Inverse Interpolation)

5. Given the following set of data points. Obtain the table of divided difference and use that table to estimate the value of f(1.5), f(3.45) & $(4.2)^3$. [5]

x	1	2	3	4	5
f(x)=x^3-1	0	7	26	63	124

Prepared By
Er. Shree Krishna Khadka

4
CHAPTER

Curve Fittings, B-Splines & Approximations

Contents:

- Curve Fitting & Regression: Introduction
- Least Square Regression Technique
    - Fitting Linear Equations
    - Fitting Transcendental Equations
    - Fitting Polynomial Equations
- Multiple Linear Equations
- Spline Interpolation: Introduction
    - Cubic B-Splines
- Approximation of Functions
- Assignment 4

CURVE FITTING & REGRESSION: INTRODUCTION

Previously, we have discussed the methods of curve fitting for the data points of well defined functions. In this topic, we will discuss on methods of curve fitting for experimental data. In many applications it often becomes necessary to establish a mathematical relationship between experimental values. The mathematical equation can be used to predict values of the dependent variables. For example: we would like to know the maintenance cost of equipment as a function of age or mileage.

The process of establishing such a relationship in the form of mathematical model is known as regression analysis or curve fitting. In this topic, we shall discuss a technique known as least square regression to fit the data under the following situation.

a) Relation is linear
b) Relation is transcendental
c) Relation is polynomial

LEAST SQUARE REGRESSION

Let us consider the mathematical equation for a straight line: $y = f(x) = a + bx$, to describe the data. We know that 'a' is the intercept of the line and 'b' is its slope.

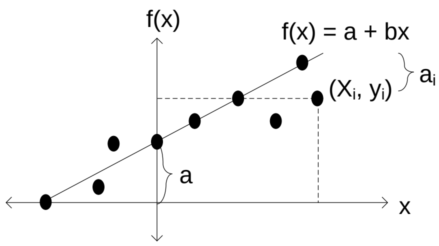


Fig: Least Square Regression

Consider a point (x_i, y_i) as shown in figure. The vertical distance of this point from the line $f(x) = a + bx$ is the error: a_i. Then $a_i = y_i - f(x_i) = y_i - a - bx_i$ ….. (i)

The best approach that could be tried for fitting a best line through the data is to minimize the sum of squares of errors.

$$\text{i.e.} \quad \sum a_i^2 = \sum(y_i - a - bx_i)^2$$

The technique of minimizing the sum of square of errors is known as least square regression.

FITTING LINEAR EQUATIONS

Let, the sum of square of individual errors be as:

$$Q = \sum_{i=1}^{n} a_i^2 = \sum_{i=1}^{n} (y_i - a - bx_i)^2 \quad \ldots \text{(i)}$$

In this method of least square, we choose 'a' and 'b' such that Q is minimum. Since, Q depends on 'a' and 'b', a necessary condition for Q to be minimum is given by:

$$\frac{\partial Q}{\partial a} = 0 \qquad \& \qquad \frac{\partial Q}{\partial b} = 0$$

Then:

$$\frac{\partial Q}{\partial a} = -2\sum_{i=1}^{n}(y_i - a - bx_i) = 0 \qquad \& \qquad \frac{\partial Q}{\partial b} = -2x_i\sum_{i=1}^{n}(y_i - a - bx_i) = 0$$

i.e. $\sum y_i = a + b\sum x_i$ .... (ii) & $\sum x_i y_i = a\sum x_i + b\sum x_i^2$ ... (iii)

Equations, (ii) and (iii) are called normal equations, solving for 'a' and 'b', we get:

$$b = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{n\sum x_i^2 - (\sum x_i)^2} \quad \ldots. \text{(iv)} \qquad \& \qquad a = \frac{\sum y_i}{n} - b\frac{\sum x_i}{n} = \bar{y} - b\bar{x} \ldots \text{(v)}$$

Example

Fit a straight line to the following set of data.

x	**1**	**2**	**3**	**4**	**5**
y	**3**	**4**	**5**	**6**	**8**

Solution:

The various summations are calculated as below:

x_i	y_i	x_i^2	$x_i y_i$
1	3	1	3
2	4	4	8
3	5	9	15
4	6	16	24
5	8	25	40
$\sum x_i = 15$	$\sum y_i = 26$	$\sum x_i^2 = 55$	$\sum x_i y_i = 90$

Here, n = 5

On calculation using above formula, we get: b = 1.2 and a = 1.6

Therefore, the required linear equation is: y = a + bx = 1.6 + 1.2x

FITTING TRANSCENDENTAL EQUATIONS

The relationship between dependent and independent variables is not always linear. The non-linear relationship may exist in the form of transcendental equation. For example, the familiar equation of population growth is given by:

$$\mathbf{P = P_0 e^{Kt}} \quad \text{... (i)}$$

Where, 'P_0' is the initial population, 'K' is the rate of growth and 't' is time.

Another example is the non-linear model in which relation is established between pressure and volume.

$$\mathbf{P = av^b} \quad \text{... (ii)}$$

Where, 'P' is pressure and 'v' is volume. If we observe the values of 'P' for the various values of 'v', we can then determine the parameter 'a' and 'b'.

The problem of finding the values of 'a' and 'b' can be solved by using the algorithm given for linear equation. Let us rewrite the equation using the conversion of variable as 'x' and 'y'. So, the equation becomes:

$$\mathbf{y = ax^b} \quad \text{... (iii)}$$

Taking log on both sides, we get:

$$\ln(y) = \ln(a) + b\ln(x) \qquad \text{... (iv)}$$

Now, comparing equation (iv) with standard linear equation, y = a + bx, we found:

$$y \leftarrow \ln(y)\,, b \leftarrow b, a \leftarrow \ln(a)\ \&\ x \leftarrow ln(x) \ \text{.... (v)}$$

Then, obviously we will have the following results:

$$b = \frac{n\sum \ln(x_i)\ln(y_i) - \sum \ln(x_i)\sum \ln(y_i)}{n\sum\{\ln(x_i)\}^2 - \{\sum \ln(x_i)\}^2} \ \text{.... (vi) \&}$$

$$\ln(a) = R = \frac{\sum \ln(y_i)}{n} - b\frac{\sum \ln(x_i)}{n}$$

i.e. $$a = e^R = \exp\left\{\frac{\sum \ln(y_i)}{n} - b\frac{\sum \ln(x_i)}{n}\right\} \ \text{.... (vi)}$$

Example:
Given the data table:

x	1	2	3	4	5
y	0.5	2	4.5	8	12.5

Fit a power function model of the form $y = ax^b$.

Solution:

x_i	y_i	$\ln(x_i)$	$\ln(y_i)$	$\{\ln(x_i)\}^2$	$\ln(x_i).\ln(y_i)$
1	0.5	0	-0.6931	0	0
2	2	0.6931	0.6931	0.4804	0.4804
3	4.5	1.0986	1.5641	1.2069	1.6524
4	8	1.3863	2.0794	1.9218	2.8827
5	12.5	1.6094	2.5257	2.5901	4.0649
$\sum = 15$	$\sum = 27.5$	$\sum = 4.7869$	$\sum = 6.1092$	$\sum = 6.1992$	$\sum = 9.0804$

Now:

$$b = \frac{n\sum \ln(x_i)\ln(y_i) - \sum \ln(x_i)\sum \ln(y_i)}{n\sum\{\ln(x_i)\}^2 - \{\sum \ln(x_i)\}^2} = b = \frac{5\times 9.0804 - 4.7869\times 6.1092}{5\times 6.1992 - 4.7869^2} = 2.2832$$

$$\ln(a) = \frac{\sum \ln(y_i)}{n} - b\frac{\sum \ln(x_i)}{n} = \frac{27.5}{5} - 2.2832\frac{15}{5} = -0.9641$$

i.e. $a = e^{-0.9641} = 0.381$

Finally, the power form equation is: $y = 0.381.x^{2.2832}$

Home Work
The temperature of a metal strip was measured at various time intervals during heating and the values are given in the table below:

Time 't(min)'	12	2	3	4
Temperature 'T(°C)	70	83	100	124

If the relation between the time 't' and temperature 'T' is of the form: $T = b.e^{t/4} + a$, estimate the temperature at t = 6 minute.

Solution:
Hint: As $T = b.e^{t/4} + a$ ... (i), we can write this equation in the form: $y = b.f(x) + a$ .... (ii), where, $y = T$ and $f(x) = e^{t/4}$. Now, use linear formula and adjust the values at final. **Ans**: $37.56e^{t/4}+21.31$

FITTING A POLYNOMIAL EQUATION

When a given series of data does not appear to satisfy a linear equation, we can try a suitable polynomial as a regression to fit the data. The least square technique can be used to fit the data to a polynomial

Consider the polynomial of degree (m-1) as below:

$$y = a_1 + a_2x + a_3x^2 + a_4x^3 + \cdots . + a_m x^{m-1} \; \ldots \text{(i)}$$

If the data contains 'n' sets of 'x' and 'y', then the sum of squares of the error is given by:

$$Q = \sum_{i=1}^{n} \{y_i - f(x_i)\}^2 \; \ldots \text{(ii)}$$

Hence; f(x) is a polynomial and contain coefficients: $a_1, a_2, \ldots, a_m$. We have to estimate all 'm' coefficients. As before, we have the following 'm' equations that can be solved for these coefficients.

$$\text{i.e. } \frac{\partial Q}{\partial a_1} = 0; \quad \frac{\partial Q}{\partial a_2} = 0; \quad \ldots . \frac{\partial Q}{\partial a_m} = 0;$$

Consider a general term:

$$\frac{\partial Q}{\partial a_j} = -2\sum_{i=1}^{n} \{y_i - f(x_i)\} \frac{\partial f(x_i)}{\partial a_j} = 0 \; ; \qquad \text{Where: } \frac{\partial f(x_i)}{\partial a_j} = {x_i}^{j-1}$$

Thus we have:

$$\sum_{i=1}^{n} \{y_i - f(x_i)\} {x_i}^{j-1} = 0 \; ; \qquad \text{Where: } j = 1, 2, \ldots . m$$

Or

$$\sum_{i=1}^{n} \{y_i . {x_i}^{j-1} - {x_i}^{j-1} . f(x_i)\} = 0; \qquad \ldots \text{(iii)}$$

Now, Substituting for $f(x_i)$:

$$\sum_{i=1}^{n} {x_i}^{j-1} (a_1 + a_2x + a_3x^2 + a_4x^3 + \cdots . + a_m x^{m-1}) = \sum_{i=1}^{n} y_i . {x_i}^{j-1}$$

So, there are 'm' equations: (j=1, 2, 3, …. ,n) and each summation is for i =1 to i =n.

$$a_1 n + a_2 \sum x_i + a_3 \sum {x_i}^2 + \cdots + a_m \sum {x_i}^{m-1} = \sum y_i$$
$$a_1 \sum x_i + a_2 \sum {x_i}^2 + a_3 \sum {x_i}^3 + \ldots + a_m \sum {x_i}^{m-1} = \sum x_i y_i$$

……………………………… …….. ……..

……………………………… …….. ……..

$$a_1 \sum {x_i}^{m-1} + a_2 \sum {x_i}^{m} + a_3 \sum {x_i}^{m+1} + \cdots + a_m \sum {x_i}^{2m-2} = a_3 \sum {x_i}^{m-1} y_i$$

#Note: For a second order polynomial, we will have the following simultaneous equations:

$$a_1 n + a_2 \sum x_i + a_3 \sum x_i^2 = \sum y_i$$
$$a_1 \sum x_i + a_2 \sum x_i^2 + a_3 \sum x_i^3 = \sum x_i y_i$$
$$a_1 \sum x_i^2 + a_2 \sum x_i^3 + a_3 \sum x_i^4 = \sum x_i^2 y_i$$

Example:
Fit a second order polynomial to the data in the table below:

x	**1**	**2**	**3**	**4**
y	**6**	**11**	**18**	**27**

Solution:

Below is the tabulation the values as second order polynomial components.

x	**y**	x^2	x^3	x^4	**xy**	x^2y
1	6	1	1	1	6	6
2	11	4	8	16	22	44
3	18	9	27	81	54	162
4	27	16	64	256	108	432
$\sum = 10$	$\sum = 62$	$\sum = 30$	$\sum = 100$	$\sum = 354$	$\sum = 190$	$\sum = 644$

So, we have the following simultaneous equations:

$4a_1 + 10a_2 + 30a_3 = 62$ ... (i)
$10a_1 + 30a_2 + 100a_3 = 190$ ... (ii)
$30a_1 + 100a_2 + 226a_3 = 644$ ... (iii)

On calculation: $a_1 = 3$, $a_2 = 2$ and $a_3 = 1$
Hence, the least square quadratic polynomial is: $y = 3 + 2x + x^2$.

Home Work:
From the given data table, fit a second order polynomial.

x	**1**	**2.1**	**3.2**	**4**
y	2	2.5	3.0	4

Equations:
$4a_1 + 10.3a_2 + 31.65a_3 = 11.5$ ... (i)
$10.3a_1 + 31.65a_2 + 107.029a_3 = 32.85$ ... (ii)
$31.65a_1 + 107.029a_2 + 381.3057a_3 = 107.745$ ... (iii)

The polynomial is: $y = 2.074 - 0.205x + 0.168x^2$

MULTIPLE LINEAR REGRESSIONS:

There are number of situations, where the dependent variable is a function of two or more variables. For example, the salary of a sales person may be expressed as: y = 500 + $5x_1$ + $8x_2$, where x_1 and x_2 are the number of unit sold of product 1 and 2 respectively. We shall discuss an approach to fit the experimental data, where the variable under consideration is a linear function of two independent variables.

Let us consider a two variables linear function as follows:

$$y = f(x, z) = a_1 + a_2x_i + a_3z_i \text{ ... (i)}$$

Then, the sum of square of errors is given by:

$$Q = \sum_{i=1}^{n}\{y_i - f(x, z)\}^2 = \sum_{i=1}^{n}\{y_i - (a_1 + a_2x_i + a_3z_i)\}^2 \text{ ... (ii)}$$

Differentiation equation ... (ii) w. r. t. a_1, a_2, a_3 and equating them to zero, we will get the condition for minimum error.

$$\frac{\partial Q}{\partial a_1} = -2\sum\{y_i - (a_1 + a_2x_i + a_3z_i)\} = 0$$

$$\frac{\partial Q}{\partial a_2} = -2\sum\{y_i - (a_1 + a_2x_i + a_3z_i)x_i\} = 0$$

$$\frac{\partial Q}{\partial a_3} = -2\sum\{y_i - (a_1 + a_2x_i + a_3z_i)z_i\} = 0$$

With these conditions, we will get the following simultaneous equation:

$$a_1n + a_2\sum x_i + a_3\sum z_i = \sum y_i$$
$$a_1\sum x_i + a_2\sum {x_i}^2 + a_3\sum z_ix_i = \sum x_iy_i$$
$$a_1\sum z_i + a_2\sum x_iz_i + a_3\sum {z_i}^2 = \sum y_iz_i$$

Example:
Given the table of data:

x	**1**	**2**	**3**	**4**
z	**0**	**1**	**2**	**3**
y	**12**	**18**	**24**	**30**

Solution:

Constructing the table as a component of multiple linear regression equation.

x	z	Y	x^2	z^2	z.x	x.y	y.z
1	0	12	1	0	0	12	0
2	1	18	4	1	2	36	18
3	2	24	9	4	6	72	48
4	3	30	16	9	12	120	90
$\sum = 10$	$\sum = 62$	$\sum = 84$	$\sum = 30$	$\sum = 14$	$\sum = 20$	$\sum = 240$	$\sum = 156$

So, three simultaneous equations will be:

$4a_1 + 10a_2 + 6a_3 = 84$ … (i)
$10a_1 + 30a_2 + 20a_3 = 240$ … (ii)
$6a_1 + 20a_2 + 14a_3 = 156$ … (iii)

On solving we can get the values of unknowns.

Home Work
Given the data points

X	5	4	3	2	1
Z	3	-2	-1	4	0
Y	15	8	-1	26	8

Fit the multiple linear regressions.

Solution:

Ans: $y = 10 - 2x + 5z$

SPLINE INTERPOLATION:

Data fitting by means of polynomials has been considered in the previous exercise. From the viewpoint of cubic splines, the resulting approximation called the cubic spline approximation, suffers from the disadvantage of being a global approximation, which means that a change in one point affects the entire approximation curve. So, a method based on the basis splines, which posses a local character, viz; a change in one point introduces a change only in the immediate neighbourhood of that point, called B-spline method that finds important application in computer graphics and smoothing of data.

Cubic B-Splines:
It resembles the ordinary cubic spline in which a separate cubic is derived for each interval. Specially, a cubic B-spline (or a B-spline of order-4) denoted by $B_{4i}(x)$, is a cubic spline with knots K_{i-4}, K_{i-3}, K_{i-2}, K_{i-1} and K_i, which is zero everywhere except in the

range $K_{i-4} < x < K_i$. In such case, $B_{4i}(x)$ is said to have a support $[K_{i-1}, K_i]$. It may be noted that a B-spline need not necessarily pass through any or all of the data points. Cubic splines are popular because of their ability to interpolate data with smooth curves. If we consider the construction of a cubic spline function which would interpolate the points (x_0, f_0), (x_1, f_1) ... (x_n, f_n). The cubic spline S(x) consists of (n-1) cubic corresponding to (n-1) sub intervals. If we denote such cubic by $S_i(x)$, then:

$$S(x) = S_i(x); \text{ for } I = 1, 2, \dots n.$$

These cubic must satisfy the following conditions:

a) S(x) must interpolate the function at all the points: x_0 , x_1, ... x_n; i.e. for each 'i', $S(x_i) = f_i$.
b) The function values must be equal at all the interior knots, i.e. $S_i(x_i) = S_{i+1}(x_i)$
c) The first derivative at the interior knots must be equal; $S_i'(x) = S_{i+1}'(x)$
d) The second derivative at the interior knots must be equal; $S_i''(x_i) = S_{i+1}''(x_i)$
e) The second derivative at the end points are zero; $S''(x_0) = S''(x_n) = 0$

Let us consider (n+1) sample points with 'n' number of different section as shown in the figure below. Suppose, $g_i(x)$ represents a cubic polynomial between $[x_i, f(x_i)]$ and $[x_{i+1}, f(x_{i+1})]$.

Then: $$g_i(x) = a_i(x-x_i)^3 + b_i(x-x_i)^2 + c_i(x-x_i) + d_i \quad \text{.... (i)}$$

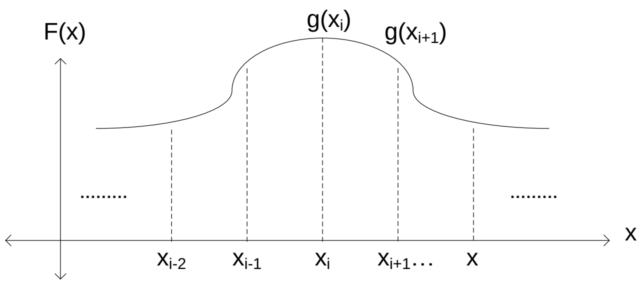


Fig: Spline Interpolation

1.) Since, function must pass through all the points; i.e; $g_i(x_i) = d_i = f(x_i)$ ....(i)

2.) Differentiating equation (i) w. r. t. 'x' we get:

$g_i'(x) = 3a_i(x-x_i)^2 + 2b_i(x-x_i) + C_i$ ... (ii)
$g_i''(x) = 6a_i(x-x_i) + 2b_i$ ... (iii)

at $x = x_i$, equation (iii) becomes: $g_i''(x) = 2b_i = S_i$ say; i.e. $b_i = S_i/2$ ...(iv)

Similarly:

$$g_i''(x_{i+1}) = S_{i+1} = 6a_i(x_{i+1} - x_i) + 2b_i = 6a_ih_i + 2b_i$$

Where; $x_{i+1} - x_i = h_i$

Then; $a_i = (S_{i+1} - S_i)/6h_i$ .... (v)

3.) Again, $g_i(x_{i+1}) = f(x_{i+1})$;

But, $f(x_{i+1}) = a_ih_i^3 + b_ih_i^2 + C_ih_i + d_i = a_ih_i^3 + b_ih_i^2 + C_ih_i + f(x_i)$

i.e. $C_i = [\{f(x_{i+1}) - f(x_i)\}/h] - a_ih_i^2 - b_ih_i$ ....(vi)

4.) Also from equation (ii) we have: $g_{i-1}'(x) = 3a_{i-1}(x-x_{i-1})^2 + 2b_{i-1}(x-x_{i-1}) + C_{i-1}$ ... (vii)

And we also have the condition that, at $x = x_i$, $g_i'(x_i) = g_{i-1}'(x_i)$, then:

$C_i = 3a_{i-1}(h_{i-1})^2 + 2b_{i-1}(h_{i-1}) + C_{i-1}$ .... (viii)

5.) Finally on manipulating all these values, we get a final equation as:

$$h_{i-1}S_{i-1} + 2(h_{i-1} + h_i)S + h_iS_{i+1} = 6\left\{\frac{f_{i+1}-f_i}{h_i} - \frac{f_i-f_{i-1}}{h_{i-1}}\right\}$$

6.) Expressing above equation in matrix form:

$$\begin{bmatrix} 2(h_0 + h_1) & h_1 & 0 \ldots. \\ h_1 & 2(h_1 + h_2) & h_2 \ldots \\ 0 \ldots & h_2 \ldots & 2(h_2 + h_3) \ldots \end{bmatrix} \begin{bmatrix} S_1 \\ S_2 \\ S_3 \ldots \end{bmatrix} = 6 \begin{bmatrix} f(x_1, x_2) - f(x_0, x_1) \\ f(x_2, x_3) - f(x_1, x_2) \\ f(x_3, x_4) - f(x_2, x_3) \ldots \end{bmatrix}$$

Where, $f(x_1, x_2)$, $f(x_0, x_1)$ .... are first divided difference values calculated from the tabulated data and h_0, h_1 ... are the successive intervals between two data points that may or may not be equal. For natural spline, we have: $S_0 = S_n = 0$ (Second order derivative at end points is zero)

Example:

Fit the data of table with a natural cubic spline curve and evaluate the spline values, g(0.66) and g(1.75), with $h_0 = 1$, $h_1 = 0.5$ and $h_2 = 0.75$

X	**0**	**1**	**1.5**	**2.25**
f(x)	**2**	**4.4366**	**6.7134**	**13.9130**

(Hint: The relation is described as: $f(x) = 2e^x - x^2$)

Solution:

1.) From the given table, we have:

$h_0 = x_1 - x_0 = 1 - 0 = 1$

$h_1 = x_2 - x_1 = 1.5 - 1 = 0.5$

$h_2 = x_3 - x_2 = 2.25 - 1.5 = 0.75$

2.) For natural spline, $S_0 = 0$ and $S_3 = 0$

3.) Now, Finding $S_i = S_1$ and S_2. Generating Matrix Model as follow:

$$\begin{bmatrix} 2(h_0 + h_1) & h_1 \\ h_1 & 2(h_1 + h_2) \end{bmatrix} \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} = 6 \begin{bmatrix} f(x_1, x_2) - f(x_0, x_1) \\ f(x_2, x_3) - f(x_1, x_2) \end{bmatrix}$$

i.e. $$\begin{bmatrix} 2(1 + 0.5) & 0.5 \\ 0.5 & 2(0.5 + 0.75) \end{bmatrix} \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} = 6 \begin{bmatrix} 12.702 \\ 30.2748 \end{bmatrix}$$

i.e. $3S_1 + 0.5S_2 = 12.702$ … (i) and $0.5S_1 + 2.5S_2 = 30.2742$ …(ii)

On solving: $S_1 = 2.292$ and $S_2 = 11.65$

4.) Now, finding a_i, b_i, c_i and d_i

i	0	1	2
$a_i = (S_{i+1} - S_i)/6h_i$	0.381	3.12	-2.598
$b_i = S_i/2$	0	1.146	5.825
$C_i = [\{f(x_{i+1}) - f(x_i)\}/h] - a_ih_i^2 - b_ih_i$	2.0556	3.200	6.680
$d_i = f(x_i)$	2	4.4366	6.7134

5.) Finally, $g(0.66) = a_0(x-x_0)^3 + b_0(x-x_0)^2 + c_0(x-x_0) + d_0 = 3.466$
And, $g(1.75) = a_2(x-x_2)^3 + b_2(x-x_2)^2 + c_2(x-x_2) + d_0 = 8.70$

Home Work:
Given the data points:

i	**0**	**1**	**2**
$\mathbf{x_i}$	**4**	**9**	**16**
$\mathbf{f(x_i)}$	**2**	**3**	**4**

Estimate the value of f(7)

Solution:
$h_0 = 5$ and $h_1 = 7$
$S_0 = 0 = S_2$, but $S_1 = -0.01428$
$a_0 = -0.000476$, $a_1 = 0.00034$
$b_0 = 0$, $b_1 = -0.00714$
$c_0 = 0.2119$, $c_1 = 0.6259$
$d_0 = 2$, $d_1 = 3$
Finally, $f(7) = g_0(x) = 2.6228$

APPROXIMATION OF FUNCTIONS:

The problem of approximating a function is a central problem in numerical analysis due to its importance in the development of software for digital computers. Functions evaluation through interpolation techniques over stored table of values has been found to be quite costlier, when compared to the use of efficient function approximations.

If f_1, f_2 … f_n be the values of the given function and g_1, g_2, … g_n be the corresponding values of the approximating function; then the error vector '**e**', where the components of '**e**' is given by: $\mathbf{e_i = f_i - g_i}$ **.....**(i).

The approximation may be chosen in a number of ways. For example, we may find the approximation such that the quantity $\sqrt{e_1{}^2 + e_1{}^2 + e_1{}^2 + \cdots}$ is minimum. This leads us to the least square approximation, which we have already studied. On the other hand, we may choose the approximation such that the maximum component of '**e**' is minimized. This leads us to the 'Celebrated Chebyshev Polynomials', which have found important application in the approximation of function in digital computers.

Assignment 4

Full Marks: 25
Pass Marks: 15

6. The table below gives the temperatures T(^{0}C) and lengths(mm) of a heated rod. If $l = a_0 + a_1T$, find the values of a_0 and a_1 using linear least squares. Find the length of the rod when T = 72.5^{o}C. [5]

T	40	50	60	70	80
l	600.5	600.6	600.8	600.9	601.0

7. The temperature of a metal strip was measured at various time intervals during heating and the values are given in the table below.

Time('t' min)	1	2	3	4
Temp('T' ^{0}C)	70	83	100	124

If the relation between the time 't' and temperature 'T' is of the form: $T = be^{t/4} + a$. Estimate the temperature at t = 6 minute. [5]

8. Fit a second order polynomial to the data given in the table below. [5]

x	1	2.1	3.2	4
f(x)	2	2.5	3.0	4

Find the polynomial value at 1.5 and 3.8.

9. Given the table of values

x	5	4	3	2	1
z	3	-2	-1	4	0
y	15	8	-1	26	8

Use multiple linear regression formula to fit the data of the form:
$y = a_1 + a_{2}(x) + a_{3}(z)$ & Find f(5.3) and f(3.7). [5]

10. Fit a natural cubic B-Spline, S to the data given below in the table. [5]

x	1.2	2.8	3.5	4.1	5.9
$f(x)=x^3-1$	-2	-1	0	1	2

Find the functional value at 1.5 and 6

Prepared By
Er. Shree Krishna Khadka

5
CHAPTER

Numerical Differentiation & Integration

Contents:

- Numerical Differentiation: Introduction
  - Differentiating Continuous Functions
  - Differentiating Tabulated Functions
  - Higher Order Derivatives
- Numerical Integration: Introduction
  - Newton Cotes General Formula
  - Trapezoidal & Composite Trapezoidal Rule
  - Simpson's & Composite Simpson's Rule
  - Gaussian Integration & Changing the limit of integration
  - Romberg Integration
  - Numerical Double Integration

NUMERICAL DIFFERENTIATION: INTRODUCTION

The method of obtaining the derivative of a function using a numerical technique is known as numerical differentiation. The general method for deriving the numerical differentiation formulae is to differentiate the interpolating polynomial. There are essentially two situations where numerical differentiation is required. They are:

1. The function values are known but the function is unknown. Such functions are called tabulated function.

2. The function to be differentiated is continuous and therefore complicated and difficult to differentiate.

Since, analytical methods give exact answers; the numerical techniques provide only approximations to derivatives. Numerical differentiation methods are very sensitive to round off errors, in addition to the truncation error introduced by the methods of themselves. So, it is necessary to discuss the errors and ways to minimize them.

DIFFERENTIATING CONTINUOUS FUNCTION

Here, the numerical process of approximating the derivative f'(x) of a function f(x) is carried out when the function is continuous and itself available.

Forward Difference & Backward Difference Quotient

Consider a small increment, $\Delta x = h$ in x. According to Taylor's Theorem, we have

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2} f''(\theta) \text{ for } x \leq \theta \geq (x+h)$$

Rearranging the terms, we get: $f'(x) = \frac{f(x+h)-f(x)}{h}$ .... (a) with the truncation error: $E_t(h) = -\frac{h}{2} f''(\theta)$. Equation (a) is called the first order forward different quotient. This is also called as two point formula. The truncation error is in the order of 'h' and can be decreased by decreasing 'h'.

Similarly, we can also show that the first order backward difference quotient is:

$$f'(x) = \frac{f(x)-f(x-h)}{h} \qquad \text{.... (a)}$$

Example:

Estimate approximate derivative of $f(x) = x^2$ at $x = 1$, for $h = 0.2, 0.2, 0.05, 0.01$ using first order forward difference formula.

Solution:

We have: $f(x) = x^2$

a) Analytical Method: $f'(x) = 2x$, i.e. $f'(x=1) = 2 \times 1 = 2$ (true value)

b) Numerical Method:

$f'(x) = \frac{f(x+h)-f(x)}{h}$, which will give the approximation of derivative.

Now, derivative approximations are tabulated below:

h	f'(1)	Error
0.2	2.2	0.2
0.1	2.1	0.1
0.05	2.05	0.05
0.01	2.01	0.01

Conclusively, it is found that the derivative approximation approaches the exact value as ‘h’ decreases. The truncation error decreases proportionally with decrease in ‘h’. There is no round off error.

Central Difference Quotient

Equation (a) was obtained using the linear approximation to f(x). This would give large truncation error if the functions were of higher order. In such cases, we can reduce truncation errors for a given ‘h’ by using a quadratic approximation rather than a linear one. This can be achieved by the following form of Taylor’s Expansion.

i.e. $f(x+h) = f(x) + hf'(x) + \frac{h^2}{2!}f''(x) + \frac{h^3}{3!}f'''(\theta_1)$ ... (a) and similarly,

$f(x-h) = f(x) - hf'(x) + \frac{h^2}{2!}f''(x) - \frac{h^3}{3!}f'''(\theta_2)$ ... (b)

Subtracting equation (b) from equation (a), we get:

$f(x+h) - f(x-h) = 2hf'(x) + \frac{h^3}{3!}\{f'''(\theta_1) + f'''(\theta_2)\}$ ... (c)

Thus we have: $f'(x) = \frac{f(x+h)-f(x-h)}{2h}$ ... (d) with truncation error of: $E_t(h) = -\frac{h^2}{3!}\{f'''(\theta_1) + f'''(\theta_2)\} = -\frac{h^2}{3}\{f'''(\theta)\}$; which is order of ‘h^2’. Equation © is called second order central difference quotient. This is also known as three point formula.

Example:
Repeat the previous example for three point formula.

Solution:
We have: $f(x) = x^2$, i.e. $f(x=1) = 2$ (true value)
Now, for different values of 'h' the derivative approximations are tabulated below:

h	f'(1)	Error
0.2	2	0
0.1	2	0
0.05	2	0
0.01	2	0

Where: $f'(x) = \frac{f(x+h)-f(x-h)}{2h}$

Error Analysis
Numerical differentiation is very sensitive to round off errors. If $E_r(h)$ is the round off error introduced in an approximation of derivative, then the total error is given by:

$$E(h) = E_r(h) + E_t(h) \text{ .... (i)}$$

Let us consider two point formulae for the purpose of analysis.

$$\text{i.e. } f'(x) = \frac{f(x+h)-f(x)}{h} = \frac{f_1-f_0}{h}$$

If we assume the round off error in f_1 and f_0 as e_1 and e_0 respectively, then:

$$f'(x) = \frac{(f_1+e_1)-(f_0+e_0)}{h} = \left(\frac{f_1-f_0}{h}\right) + \left(\frac{e_1-e_0}{h}\right)$$

If the errors e_1 and e_0 are of the magnitude 'e' and of opposite signs, the we get bound for round off error as:

$$|E_r(h)| \leq \frac{2e}{h}$$

We know, the truncation error for two point formula is:

$$|E_t(h)| \leq -\frac{h}{2} f''(\theta) \leq \frac{M_2 h}{2}$$

Where, M_2 is the bound given by $M_2 = \max|f''(\theta)|$ $\{\therefore x \leq \theta \leq (x+h)\}$

Thus the boundary for total error in the derivative is:

$$|E(h)| \leq \frac{M_2 h}{2} + \frac{2e}{h} \text{ ...(a)}$$

We can obtain a rough estimation of ‘h’ that gives the minimum error. By differentiating equation (a) with respect to ‘h’, we get:

$$|E'(h)| = \frac{M_2}{2} - \frac{2e}{h^2}$$

Now, we know that E(h) is minimum when E’(h) = 0. So, solving for ‘h’, we obtain

$$h_{opt} = 2\sqrt{\frac{e}{M_2}}$$

Where, h_{opt} is optimum step size. Substituting this value in equation (a), we get:

$$E(h_{opt}) = 2\sqrt{eM_2}$$

Example:

Compute the approximate derivative of f(x) = sin(x) at x = 0.45 radian at increasing value of ‘h’ from 0.01 to 0.04 with a step size of 0.005. Analyse the total error. What is the optimum step size?

Solution:

We have given: y = f(x) = **sin**(x) and f’(x) = **cos**(x)

So, f(x=0.45rad) = **sin**(0.45rad) = 0.4350(rounded to four digits)

And f’(x=0.45rad) = **cos**(0.45rad) = 0.9004

Using two point formula, table below gives the approximate derivatives of sin(x) at x = 0.45 radian using different values of ‘h’.

h	f(x+h)	f’(x)	Error
0.010	0.4439	0.8900	0.0104
0.015	0.4484	0.8933	0.0071
0.020	0.4529	0.8950	0.0054
0.025	0.4573	0.8935	0.0069
0.030	0.4618	0.8933	0.0071
0.035	0.4662	0.8914	0.0090
0.040	0.4706	0.8900	0.0104

The total error decreases from 0.0104 (at h = 0.01) till h = 0.02 and again increases when 'h' is increased.

Since, we have used four significant digits; the bound for round off error 'e' is 0.5×10^{-4}. For the two point formula, the bound M_2 is given by:

$$M_2 = \max|f''(\theta)| \qquad \text{for: } 0.41 \leq \theta \leq 0.49$$

$$\text{i.e. } M_2 = |-\sin(0.49)| = 0.4706$$

Therefore, the optimum step size is:

$$h_{opt} = 2\sqrt{\frac{e}{M_2}} = 2\sqrt{\frac{0.5\times10^{-4}}{0.4706}} = 0.0206$$

Finally, the error at optimum step size is:

$$E(h_{opt}) = 2\sqrt{0.5 \times 10^{-4} \times 0.4706} = 0.009701$$

Higher Order Derivatives:
We can also obtain approximations to higher order derivatives using Taylor's Series Expansion.

As we know that:

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2!}f''(x) + \frac{h^3}{3!}f'''(x) + R_1 \text{ ... (a) and similarly,}$$
$$f(x-h) = f(x) - hf'(x) + \frac{h^2}{2!}f''(x) - \frac{h^3}{3!}f'''(x) + R_2 \text{ ... (b)}$$

Where: $R_1 = \frac{h^4}{4!}f^{iv}(\theta_1)$ and $R_2 = = \frac{h^4}{4!}f^{iv}(\theta_2)$

So, the addition of two expansions gives:

$$f(x+h) + f(x-h) = 2f(x) + h^2 f''(x) + R_1 + R_2 \text{ .... (c)}$$

Therefore:

$$f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$ with truncation error of: $E_t(h) = -(R_1 + R_2)/h^2$

i.e. $E_t(h) = -\frac{1}{h^2}\frac{2h^4}{4!}\{f^4(\theta_1) + f^4(\theta_2)\} = -\frac{h^2}{12}f^4(\theta)$

So, the error is of the order h^2.

Example:
Find the approximation to second derivative of cos(x) at x = 0.75 with = 0.01. Compare with the true value.

Solution:

We have given: y = f(x) = cos(x)
Then: y' = f'(x) = -sin(x) and y'' = f''(x) = -cos(x)
Therefore: f''(x=0.75) = -cos(0.75) = -0.7316888

Using the formula,
i.e. $f''(x) = \frac{f(x+h)-2f(x)+f(x-h)}{h^2} = \frac{f(0.76)-2f(0.75)+f(0.74)}{0.0001}$ (at h = 0.01)

Therefore,
f''(0.75) = $\frac{0.7248360-2\times0.7316888+0.7384685}{0.0001} = -0.7300000$

Finally, Error = |-0.7316888-(-0.7300000)| = 0.0016888

DIFFERENTIATING TABULATED FUNCTION

If we are given a set of data points (x_i, f_i), for i = 0, 1, 2 …. n, which corresponds to the values of an unknown function f(x) and we wish to estimate the derivative at these points. Assume that the points are equally spaced with a step size of 'h'.

When function values are available in tabulated from, we may approximate this function by an interpolation polynomial P(x) as discussed in earlier chapter and then differentiate P(x).

If we consider the quadratic approximation, we need to use three point as referenced by Newton Polynomial Form:

i.e. $P_2(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1)$ …. (i)
Then: $P_2{}'(x) = a_1 + a_2\{(x - x_0) + (x - x_1)\}$ …. (ii)

If $x_0 = x$, $x_1 = x + h$, $x_2 = x + 2h$, then:

$a_1 = f[x_0, x_1] = \frac{f(x+h)-f(x)}{h}$ … (a), which is well known as two point formula for the approximation fo first derivative, and on calculation:

$a_2 = f[x_0, x_1, x_2] = \frac{f(x+2h)-2f(x+h)+f(x)}{2h^2}$ … (b)

Now, substituting the values of a_1 and a_2 in equation (ii), we will get the three-point forward difference formula for the approximation of first derivative as below:

$$P_2'(x) = f'(x) = \frac{-3f(x)+4f(x+h)-f(x+2h)}{2h} \quad \text{... (iii)}$$

Hence, the backward form of equation (iii) will be:

$$P_2'(x) = f'(x) = \frac{3f(x)-4f(x-h)-f(x-2h)}{2h} \quad \text{... (iv)}$$

Similarly, we can obtain the three-point central difference formula by letting $x_0 = x$, $x_1 = x - h$ and $x_2 = x + h$ in equation (ii), Then:

$a_1 = f[x_0, x_1] = \frac{f(x)-f(x-h)}{h}$ ... (c) and

$a_2 = f[x_0, x_1, x_2] = \frac{f(x+h)-2f(x)+f(x-h)}{2h^2}$

Finally, by substituting these values in equation (ii), we get:

$f'(x) = \frac{f(x+h)-f(x-h)}{2h}$ ...(v) which is the same result as derived in the earlier section for the approximation of derivative of continuous function.

Example:

The table below gives the values of distance travelled by a car at various time intervals during the initial running.

Time, t(s)	5	6	7	8	9
Distance Travelled, s(t) (km)	10.0	14.5	19.5	25.5	32.0

Estimate the velocity at time t = 5s, t = 7s and t = 9s. Also find the acceleration at t = 7s.

Solution:

The velocity is given by the first derivative of s(t).

Case I:
At time, t = 5, we use the three-point forward difference formula.

i.e. $v(t{=}5) = \frac{-3s(t)+4s(t+h)-s(t+2h)}{2h}$

$= \frac{-3s(5)+4s(6)-s(7)}{2h}$

$= \frac{-3\times10+4\times14.5-19.5}{2\times1} = 4.25\text{km/s}$

Case II:
At time, t = 7, we use the central difference formula.

i.e. $v(t{=}7) = \frac{s(t+h)-s(t-h)}{2h}$

$= \frac{s(8)-s(6)}{2h}$

$= \frac{25.5-14.5}{2} = 5.5 \text{ km/s}$

Case III:
At time, t = 9, we use the three-point backward difference formula.

i.e. $v(t{=}9) = \frac{3s(t)-4s(t-h)-s(t-2h)}{2h}$

$= \frac{3s(9)-4s(8)-s(7)}{2h}$

$= \frac{3\times32-4\times25.5-19.5}{2} = 6.75 \text{ km/s}$

Case IV:
Acceleration is given by the second derivative of s(t). Therefore:

$$a(t) = s''(t) = \frac{s(t+h) - 2s(t) + s(t-h)}{h^2}$$

i.e. $a(t = 7) = s''(t) = \frac{s(8)-2s(7)+s(6)}{h^2} = \frac{25.5-2(19.5)+14.5}{1} = 1 \text{ km/s}^2$

NUMERICAL INTEGRATION: INTRODUCTION

The general objective of numerical integration is to compute the value of the definite integral given below for a given set of data points (x_0, y_0), (x_1, y_1), (x_2, y_2), (x_3, y_3), … ,(x_n, y_n) of a function y = f(x), where f(x) is not know explicitly. So, f(x) is determined using the interpolation technique and then integration is carried out.

$$\text{i.e. } I = \int_a^b y dx \text{ ...(i)}$$

Like in numerical differentiation, we need to evaluate the numerical integration in the following cases.

1. Functions do not possess closed form solutions.
   Example: $f(x) = C \int_0^x e^{-t^2} dt$
2. Closed form solutions exist but these solutions are complex and difficult to use for calculations.
3. Data for variables are available in the form of table, but no mathematical relationship between them is known, as is often the case with the experimental data.

A definite integral of the form: $I = \int_a^b y dx$ can be treated as the area under curve y = f(x), enclosed between the limits x = a and x = b. Then the problem of integration is then simply reduced to the problem of finding the shaded area.

NEWTON COTES GENERAL INTEGRATION FORMULA:

It is the most popular and widely used numerical integration formula, which provides the basis for a number of numerical integration methods known as Newton Cotes Method.

It is based on the polynomial interpolation. An n^{th} degree polynomial $P_n(x)$ that interpolates the values of f(x) at (n+1) evenly spaced points can be used to replace the integrand f(x) of the integral: $I = \int_a^b f(x) dx$ and the resultant formula is called (n+1) Newton Cotes Formula.

If the limits of integration 'a' and 'b' are in the set of interpolating points x_i = 0, 1, 2, …, n, then the formula is referred to as closed form. Whereas if the points 'a' and 'b' lie beyond the set of interpolating points, then the formula is termed as open form. Since, open form formula is not used for definite integration; we consider here only the closed form methods.

Let us consider a General Newton's Interpolation Formula, given by:

$$f(x) = f_0 + s\Delta f_0 + \frac{s(s-1)}{2!}\Delta^2 f_0 + \frac{s(s-1)(s-2)}{3!}\Delta^3 f_0 + \cdots \text{(ii)}$$

Where, $x = a + sh$, then $dx = hds$
When, $x = a$ then, $s = 0$ and When $x = b = a + nh$ then, $s = n$

Now, substituting all these values in equation (i), we get:

$$I = \int_a^b y\{= f(x)\}dx = \int_0^n \{f_0 + s\Delta f_0 + \frac{s(s-1)}{2!}\Delta^2 f_0 + \frac{s(s-1)(s-2)}{3!}\Delta^3 f_0 + \cdots\}hds$$

$$= h\int_0^n \{f_0 + s\Delta f_0 + \frac{s(s-1)}{2!}\Delta^2 f_0 + \frac{s(s-1)(s-2)}{3!}\Delta^3 f_0 + \cdots\}ds$$

$$= h\left\{sf_0 + \frac{s^2}{2}\Delta f_0 + \left(\frac{s^3}{3} - \frac{s^2}{2}\right)\frac{\Delta^2 f_0}{2!} + \left(\frac{s^4}{4} - s^3 + s^2\right)\frac{\Delta^3 f_0}{3!} + \cdots\right\}\Big|_0^n$$

$$I = h\left\{nf_0 + \frac{n^2}{2}\Delta f_0 + \left(\frac{n^3}{3} - \frac{n^2}{2}\right)\frac{\Delta^2 f_0}{2!} + \left(\frac{n^4}{4} - n^3 + n^2\right)\frac{\Delta^3 f_0}{3!} + \cdots\right\} \text{.... (iii)}$$

Where: $h = (x_n - x_0)/n$ and equation (iii) is the general equation for numerical integration and is called **General Newton Cotes Formula.**

TRAPEZOIDAL RULE (TWO POINT FORMULA)

Setting, n = 1 in General Newton Cotes Formula and neglecting second and higher order differences gives a two point formula by which numerical integration is carried out known as Trapezoidal Rule.

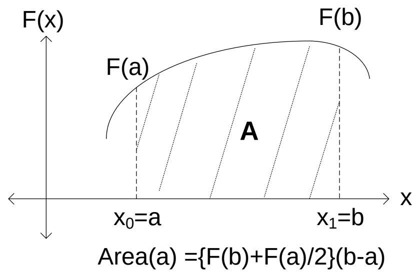


Fig: Trapezoidal Rule

i.e. $I = \int_{x_0}^{x_1} f(x)dx = h(f_0 + \frac{\Delta f_0}{2}) = h(f_0 + \frac{f_1 - f_0}{2}) = \frac{h(f_1 + f_0)}{2} =$ Area of trapezoid.

Example:

Evaluate the integral: I = $\int_a^b (x^3 + 1)dx$ for (i) $(1, 2)$ and (ii) $(1, 1.5)$

Solution:

(a,b)	H	f(a)=a³ + 1	f(b)= b³ + 1	I=(f(a)+f(b))(h/2)
(1, 2)	1	2	9	5.5
(1, 1.5)	0.5	2	4.375	1.59375

COMPOSITE TRAPEZOIDAL RULE:

If the range is to be integrated and the function is non linear, the trapezoidal rule can be improved by dividing the range (a, b) into number of small intervals and applying the rule discussed above for all sub intervals. The sum of areas of all the sub intervals is the integral of the interval (a, b) as shown in the figure below.

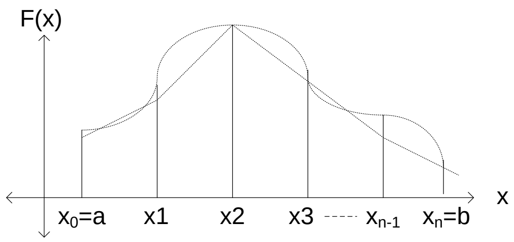


Fig: Composite Trapezoidal Rule

If the data range (a, b) is divided into ‘n’ stripes of equal width ‘h’, then:

$$I = \int_{x_0}^{x_n} f(x)dx = \int_{x_0}^{x_1} f(x)dx + \int_{x_1}^{x_2} f(x)dx + \cdots + \int_{x_{n-1}}^{x_n} f(x)dx$$

$$= \frac{h(f_0+f_1)}{2} + \frac{h(f_1+f_2)}{2} + \cdots + \frac{h(f_{n-1}+f_n)}{2}$$

i.e. $\quad I = \frac{h}{2}\{f_0 + 2(f_1 + f_2 + f_3 + \ldots + f_{n-1}) + f_n\} \qquad \ldots$ (i)

Here, equation (i) is the general formula for composite trapezoidal rule.

Example:
Evaluate: $I = \int_0^{\pi} \sin(x)dx$ by dividing into four stripes.

Solution:
We have given: f(x) = sin(x)
Number of stripes (n) = 4
Limits (a, b) = (0, π)
Interval (h) = (b-a)/n = (π − 0)/4 = π/4

Now the approximation of the integration is tabulated below.

x	0	π/4	π/2	3 π/4	π
f(x)	0	0.7071	1	0.7071	0

As we have the formula:

$$I = \frac{h}{2}\{f_0 + 2(f_1 + f_2 + f_3) + f_4\} = \frac{\pi}{8}\{0 + 2(0.7071 + 1 + 0.7071) + 0\} = 1.8961$$

Compute the integral, : $I = \int_{-1}^{1} e^x dx$ using composite trapezoidal rule for (a) n = 2 and (b) n = 4.

Solution:

I Case:
For n = 2, we have: h = (1-(-1))/2=1

x	-1	0	1
f(x)	0.368	1	2.718

$$\therefore I = \frac{h}{2}\{f_0 + 2(f_1) + f_2\} = \frac{1}{2}\{0.368 + 2 \times 1 + 2.718\} = 2.543$$

II case:
For n = 4, we have: h = (1-(-1))/4 = 0.5

x	-1	-0.5	0	0.5	1
f(x)	0.368	0.607	1	1.649	2.718

$$I = \frac{h}{2}\{f_0 + 2(f_1 + f_2 + f_3) + f_4\} = \frac{0.5}{2}\{0.368 + 2(0.607 + 1 + 1.649) + 2.718\} = 2.3995$$

SIMPSON (1/3) RULE: (THREE POINT FORMULA)

Here, the function f(x) is approximated by a second order polynomial, which passes through three sampling points. So, taking n = 2 in General Newton Cotes Formula and neglecting third and higher order differences, we get:

$$\mathbf{I} \quad = \int_{x_0}^{x_2} f(x)dx = h\{2f_0 + \frac{4}{2}\Delta f_0 + (\frac{8}{3} - \frac{4}{2})\frac{\Delta^2 f_0}{2!}\}$$

$$= h(2f_0 + 2(f_1 - f_0) + \frac{1}{3}(f_0 - 2f_1 + f_0)$$

$$\mathrm{I} \quad = \frac{h}{3}(f_0 + 4f_1 + f_2)$$

Since, it uses three points to evaluate the integral; it is also known as three-point formula. The three points are: x_0 = a, x_2 = b and x_1 = (a + b)/2. Where, the interval 'h' is given by: h = (b – a)/2.

Example:

Evaluate the integral, I $= \int_{-1}^{1} e^x dx$ using Simpson's 1/3 Rule.

Solution:

We have given: f(x) = e^x

Limit Interval (a, b) = (-1, 1) = (x_0, x_2)

Then, x_1 = (-1+1)/2 = 0 and h = (1+1)/2 = 1

Now, for h = 1, the range (a, b), the function will generate the following table:

x	-1	0	1
f(x) = e^x	0.3698	1	2.7182

Hence, by formula we have:

$$I = \frac{h}{3}(f_0 + 4f_1 + f_2)$$

$$= \frac{1}{3}(0.3698 + 4 \times 1 + 2.7182)$$

$$= 2.362$$

COMPOSITE SIMPSON'S 1/3 RULE:

As in the case of composite trapezoidal rule, the range over which the integral have to be carried out is divided into 'n' number of stripes in order to improve the accuracy of the estimate of the area. Each stripe is then integrate by Simpson's 1/3 Rule and finally tied together to give the Composite Simpson's Output. Hence, the integration will be:

$$I = \int_{x_0}^{x_n} f(x)dx = \int_a^b f(x)dx$$

$$= \int_{x_0}^{x_2} f(x)dx + \int_{x_2}^{x_4} f(x)dx + \int_{x_4}^{x_6} f(x)dx + \cdots + \int_{x_{n-2}}^{x_n} f(x)dx$$

$$= \left\{\frac{h}{3}(f_0 + 4f_1 + f_2)\right\} + \left\{\frac{h}{3}(f_2 + 4f_3 + f_4)\right\} + \left\{\frac{h}{3}(f_4 + 4f_5 + f_6)\right\} + \cdots$$

$$\mathbf{I} = \frac{h}{3}\{f_0 + 4(f_1 + f_3 + f_5 + \cdots + f_{n-1}) + 2(f_2 + f_4 + f_6 + \cdots + f_{n-2}) + f_n\}$$

$$\mathbf{I} = \frac{h}{3}\{f_0 + 4\sum_{i=1}^{n/2} f(x_{2i-1}) + 2\sum_{i=1}^{\frac{n}{2}-1} f(x_{2i}) + f_n\} \qquad (\therefore h = \frac{b-a}{n})$$

Example:

Compute the integral: $I = \int_0^{\frac{\pi}{2}} \sqrt{\sin(x)}\, dx$ applying Composite Simpson's 1/3 Rule for n = 4 and n = 6 with accuracy to five decimal places.

Solution:

I Case: For n = 4

We have given: $f(x) = \sqrt{\sin(x)}$, Limit Range (a, b) = $(0, \frac{\pi}{2})$

Interval (h) = (b – a)/n = $(\frac{\pi}{2} - 0)/4$ = $\boldsymbol{\pi/8}$

So, the functional table will be:

x	0	**π/8**	**π/4**	**3π/8**	**π/2**
$f(x)=\sqrt{\sin(x)}$	0	0.61861	0.84090	0.96119	1

Now, by formula we have:

$$\mathbf{I} = \frac{h}{3}\{f_0 + 4(f_1 + f_3) + 2(f_2) + f_4\} = \frac{\pi}{8\times 3}\{0 + 4(0.61869 + 0.96119) + 2(0.84090) + 1\} = 1.17823$$

II Case: For n = 6, h = $\boldsymbol{\pi/12}$ **,** and the integration will be: I = 1.18728 (same way)

SIMPSON'S 3/8 RULE

Simpson's 1/3 rule was derived using three sampling points that fit a quadratic equation. We can extent this approach to incorporate four sampling points so that the rule can be exact for f(x) of degree 3.

By using the first four terms from General Newton Cotes Formula and applying the same procedure followed in the previous case, we will get the following result:

$$I = \frac{3h}{8}(f_0 + 3f_1 + 3f_2 + f_3) \qquad \text{Where, } h = (b - a)/3$$

Example:

Use Simpson's 3/8 Rule to evaluate, $I = \int_1^2 (x^3 + 3)dx$ for n = 3.

Solution:

Here, a = 1, b = 2; So, h = (b – a)/3 = 1/3
Now, for different values of 'x' f(x) will generate the following values.

x	1	1.33	1.66	2
$f(x) = x^3 + 3$	4	5.353	7.574	11

Finally, using Simpson's 3.8 Rule, we have:

$$I = \frac{3h}{8}(f_0 + 3f_1 + 3f_2 + f_3) = I = \frac{3\times1}{8\times3}(4 + 5.353 + 3 \times 7.574 + 11) = 6.732$$

COMPOSITE SIMPSON'S 3/8 RULE:

In this case, the total range of the integral is divided into 'n' number of stripes and the integral is estimated taking three stripes at a time (4-points) using Simpson's 3/8 Rule. Finally, the total integral is obtained by adding all the sub integrals.

So, following the same procedure as we did in earlier section, we will get the formulae for the Composite Simpson's 3/8 Rule as below:

$$I = \frac{3h}{8}(f_0 + 3(f_1 + f_2 + f_4 + \cdots) + 2(f_3 + f_6 + \cdots) + f_n)$$

Where, h = (b – a)/n

Example:

Evaluate $I = \int_{0.1}^{0.7} f(x)dx$ for h = 0.1 using Composite Simpson's 3/8 Rule.

/Home Work/

NUMERICAL DOUBLE INTEGRATION

Formula for the evaluation of a double integral can be obtained by repeatedly applying the trapezoidal and Simpson's rule derived in earlier section. Let us consider, as an example, the double integral defined by:

$$I = \int_{y=c}^{d} \int_{x=a}^{b} f(x, y) dx\, dy \quad \ldots (i)$$

Dividing the range [a, b] into 'n' equal sub-intervals of length 'h' and dividing the range [c, d] into 'm' equal sub intervals of equal length 'k'.

Here we are using composite trapezoidal rule repeatedly.

$$I = \int_{y=c}^{d} \int_{x=a}^{b} f(x, y) dx\, dy$$

$$= \int_{y=c}^{d} \frac{h}{2} \{f(x_0, y) + 2f(x_1, y) + 2f(x_2, y) + \cdots . +2f(x_{n-1}, y) + f(x_n, y)\} dy$$

$$= \frac{h}{2} \left[\int_c^d \{f(x_0, y) dy + 2 \int_c^d \{f(x_1, y) dy + 2 \int_c^d \{f(x_2, y) dy + \cdots + \int_c^d \{f(x_{n-1}, y) dy + \int_c^d \{f(x_n, y) dy \right]$$

$$= \frac{h}{2} \left[\frac{k}{2} \{f(x_0, y_0) + 2f(x_0, y_1) + 2f(x_0, y_2) + \cdots + f(x_0, y_m)\} + 2\{\frac{k}{2} \{f(x_1, y_0) + 2f(x_1, y_1) + 2f(x_1, y_2) + \cdots + f(x_1, y_m)\}\} + 2\{\frac{k}{2} \{f(x_2, y_0) + 2f(x_2, y_1) + 2f(x_2, y_2) + \cdots + f(x_2, y_m)\}\} + \frac{k}{2} \{f(x_n, y_0) + 2f(x_n, y_1) + 2f(x_n, y_2) + \cdots + f(x_n, y_m)\} \right]$$

$$I = \frac{hk}{4} [\{f(x_0, y_0) + 2f(x_0, y_1) + 2f(x_0, y_2) + \cdots + f(x_0, y_m)\} + 2\{\{f(x_1, y_0) + 2f(x_1, y_1) + 2f(x_1, y_2) + \cdots + f(x_1, y_m)\}\} + 2\{\{f(x_2, y_0) + 2f(x_2, y_1) + 2f(x_2, y_2) + \cdots + f(x_2, y_m)\}\} + \cdots + \{f(x_n, y_0) + 2f(x_n, y_1) + \quad 2f(x_n, y_2) + \cdots + f(x_n, y_m)\}]$$

Example:

Evaluate the double integral, $I = \int_0^1 \int_0^1 e^{x+y} dxdy$ using composite trapezoidal rule. Take h = k = 0.5.

Solution:

We have given that: $f(x) = e^{x+y}$
Range for integration over 'x' is: [a, b] = [0, 1]
Range for integration over 'y' is: [c, d] = [0, 1]
Interval in 'x' is: (h) = 0.5
Interval in 'y' is: (k) = 0.5

Now, for different combination of (x, y) characterized by the interval (h, k), we can generate the following table:

x		x_0	x_1	x_2
y		0	0.5	1
y_0	0	1	1.648	2.718
y_1	0.5	1.648	2.718	4.482
y_2	1	2.718	4.482	7.389

Now, the double integral is evaluated using the composite trapezoidal formula given by:

$$I = \frac{hk}{4}[\{f(x_0, y_0) + 2f(x_0, y_1) + 2f(x_0, y_2)\} + 2\{f(x_1, y_0) + 2f(x_1, y_1) + 2f(x_1, y_2)\} + 2\{f(x_2, y_0) + 2f(x_2, y_1) + 2f(x_2, y_2)\}]$$

$$= \frac{0.5 \times 0.5}{4}[\{1 + 21.648 + 2.718\} + 2\{1.648 + 22.718 + 4.482\} + \{2.718 + 4.482 + 7.389\}]$$

$= 0.0625(7.014+23.132+19.067)$

$= 3.0758$

GAUSSIAN INTEGRATION (GAUSS LEGENDRE'S INTEGRATION)

We have discussed so far a set of rules based on the Newton Cotes Formula, which was derived by integrating Newton Gregory Forward Difference Interpolating Polynomial. Consequently, all the rules were based on evenly faced sampling points (functional values) within the range of integral.

Gauss Integration is based on the concept that the accuracy of numerical integration can be improved by choosing the sampling points wisely, rather than on the basis of equal spacing.

So, Gauss Integration assumes an approximation of the form:

$$I_g = \int_{-1}^{1} f(x)dx = \sum_{i=1}^{n} \omega_i f(x_i) \;\; .... \text{(i)}$$

Here, the method of implementing the strategy of finding appropriate values of x_i and w_i and obtaining the integral f(x) over the range [-1, 1] is called Gaussian Integration or Quadrature. Where w_i are known as weights of that function associated with and x_i are the nodes (both are unknowns).

Therefore, equation (i) contains 2n unknowns to be determined, which can be determined using the condition given in the integration formula. So, it should give the exact value of the integral for polynomials of as high a degree as possible.

Rewriting the equation (i) in the following form of representation:

$$I_g = \int_{-1}^{1} f(x)dx = \omega_1 f(x_1) + \omega_2 f(x_2) \;\; ... \text{(ii)}$$

Let us find the Gaussian Quadrature Formula for n = 2. In this case, we need to find out the values of four unknowns, namely: ω_1, ω_2 and x_1 and x_2. Let us assume that the integral will be exact up to cubic polynomials. This implies that the function 1, x, x^2, x^3 can be numerically integrated to obtain the exact results. Then from equation (i) we will get:

$\omega_1 + \omega_2 = \int_{-1}^{1} dx = 2$ ... (a) (for f(x) = 1)

$\omega_1 x_1 + \omega_2 x_2 = \int_{-1}^{1} x dx = 0$ ... (b) (for f(x) = x)

$\omega_1 {x_1}^2 + \omega_2 {x_2}^2 = \int_{-1}^{1} x^2 dx = \frac{2}{3}$ ... (c) (for f(x) = x^2)

$\omega_1 {x_1}^3 + \omega_2 {x_2}^3 = \int_{-1}^{1} x^3 dx = 0$ ... (d) (for f(x) = x^3)

Solving these four simultaneous equations we will get:

$\omega_1 = \omega_2 = 1$ and ($x_1 = -\frac{1}{\sqrt{3}} = -0.5113502$, $x_2 = \frac{1}{\sqrt{3}} = 0.5113502$)

Thus we have the Gaussian Quadrature Formula for n = 2 as:

$$I_g = \int_{-1}^{1} f(x)dx = \sum_{i=1}^{n} \omega_i f(x_i) = f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right)$$

Example:

Compute the integral, $I = \int_{-1}^{1} e^x dx$ using two-point Gauss Legendre Formula.

Solution:

$$I = \int_{-1}^{1} e^x dx = f(x_1) + f(x_2) = f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right) = e^{-\frac{1}{\sqrt{3}}} + e^{\frac{1}{\sqrt{3}}} = 2.3426961$$

CHANGING LIMITS OF INTEGRATION

Since, the Gaussian Formula imposes a restriction on the limits of integration to be the form -1 to 1. This restriction can be overcome by using the technique of "interval transformation". Let:

$$\int_a^b f(x)dx = C\int_{-1}^{1} g(z)dz \quad .... \text{(i)}$$

Assume the following transformation between 'x' and the new variable 'z'.

$$x = Az + B \quad .... \text{(a)}$$

This must satisfy the following conditions:

At x = a, z = -1 and at x = b, z = 1, which gives: B – A = a & A + B = b.

Solving for A and B, we get: A = (b – a)/2 and B = (a + b)/2

Therefore, Equation (a) becomes:

$$x = \frac{b-a}{2}z + \frac{a+b}{2} \quad ... \text{(b)}$$

Then: $$dx = \frac{b-a}{2}dz \quad ... \text{(c)}$$

Now, the equation (i) will be:

$$\int_a^b f(x)dx = C\int_{-1}^{1} g(z)dz = \frac{b-a}{2}\int_{-1}^{1} g(z)dz \quad ... \text{(ii)}$$

Hence, the Gaussian formula for this integration is:

$$\frac{b-a}{2}\int_{-1}^{1} g(z)dz = \frac{b-a}{2}\sum_{i=1}^{n} \omega_i g(z_i) \quad ... \text{(iii)}$$

Where ω_i and z_i are the weights and Quadrature points for the integration domain (-1, 1).

Example:

Compute the integral, $I = \int_{-2}^{2} e^{-\frac{x}{2}} dx$ using Gaussian two point formula.

Solution:

For n = 2 (Gaussian two point formula)

$$I_g = \frac{b-a}{2}[\omega_1 g(z_1) + \omega_2 g(z_2)]$$

Where, $x = \frac{b-a}{2} z + \frac{b+a}{2} = 2z$ and therefore, $g(z) = e^{-\frac{2z}{2}} = e^{-z}$

Also, for two point formula: $\omega_i = \omega_2 = 1, z_1 = -\frac{1}{\sqrt{3}}$ and $z_2 = \frac{1}{\sqrt{3}}$

Now, upon substitution of these values, we get:

$$I_g = \frac{2-(-2)}{2}\left[1 \times g\left(-\frac{1}{\sqrt{3}}\right) + 1 \times g\left(\frac{1}{\sqrt{3}}\right)\right] = 2\left[e^{\frac{1}{\sqrt{3}}} + e^{-\frac{1}{\sqrt{3}}}\right] = 4.6853922$$

/Home Work/

Compute the integral, $I = \int_{0}^{\frac{\pi}{2}} \sin(x) dx$ using GL-2-Point Method. [Ans: 0.9984]

GAUSS LEGENDRE THREE-POINT FORMULA:

By using a procedure similar to one applied in deriving two-point formula, we can obtain the parameters ω_i and z_i for higher order versions of Gaussian Quadrature.

So, for Gauss Legendre Three-Point Formula, we will have:

$$\omega_1 = 0.55556, \omega_2 = 0.88889, \omega_3 = 0.55556$$
And
$$z_1 = -0.77460, z_2 = 0, 0.77460$$

The integration is again given by:

$$I = \frac{b-a}{2}\int_{-1}^{1} g(z)dz = \frac{b-a}{2}\sum_{i=1}^{n} \omega_i g(z_i)$$

Example:

Use Gauss Legendre three-point formula to evaluate the following integral:

$$\mathbf{I = \int_2^4 (x^4 + 1)\, dx\,.}$$

Solution:

For 3-point formula, n= 3

We have given, [a, b] = [2, 4] and

The function: f(x) = (x^4 +1)

Then:

$I_g = \frac{b-a}{2}\int_{-1}^{1} g(z)dz = \frac{b-a}{2}\sum_{i=1}^{3} \omega_i g(z_i) = \omega_1 g(z_1) + \omega_2 g(z_2) + \omega_3 g(z_3)$

$x = \frac{b-a}{2}z + \frac{b+a}{2} = z + 3$, then: $g(z) = (z+3)^4 + 1$

For Gauss Legendre 3-point formula, we have:

$\omega_1 = 0.55556, \omega_2 = 0.88889, \omega_3 = 0.55556$ and

$z_1 = -0.77460, z_2 = 0, 0.77460$

So,

I_g = 0.55556[(-0.77460+3)^4+1] + 0.88889[(0+3)^4+1] + 0.55556[(0.77460+3)^4+1]

= 14.18140 + 72.88898 + 113.33105

= 200.40143

ROMBERG INTEGRATION

In conclusion, what we have found that, the accuracy of a numerical integration process can be improved in two ways:

1. By increasing the number of subintervals (i.e. by decreasing 'h'): this decreases the magnitude of error terms. Here, the order of the method is fixed.
2. By using higher order methods: this eliminates the lower order error terms. Here, the order of the method is varied and, therefore, this method is known as variable order approach.

The variable order method involves combining two estimates of a given order to obtain a third estimate of higher order. The method that incorporates this process (i.e. Richardson's Extrapolation) to the trapezoidal rule is called Romberg Integration.

In order to achieve the Romberg Integration, we have the following steps:

1. Compute the integration with given value of 'h' by trapezoidal rule.
2. Each time, halve the value of 'h' and again compute the integral using composite trapezoidal rule.
3. Refine the above computed values using the relation given below:

$$I = I_2 + \frac{1}{3}(I_2 - I_1)$$

Example:

Compute the integral: $\mathbf{I = \int_0^1 \frac{dx}{1+x^2}}$ **using Romberg Integration, take h = 0.5, 0.25, and 0.125**

Solution:

Case I: For h = 0. 5

$$I_1 = \frac{h}{2}(f_0 + 2f_1 + f_2)$$

$$= \frac{0.5}{2}(1 + 2 \times 0.8 + 0.5)$$

$$= 0.775$$

Case II: For h = 0.25

$$I_2 = \frac{h}{2}(f_0 + 2(f_1 + f_2 + f_3) + f_4)$$

$$= \frac{0.25}{2}(1 + 2 \times (0.64 + 0.8 + 0.9412) + 0.5)$$

$$= 0.782775$$

Case III: For h = 0.125

$$I_3 = \frac{h}{2}(f_0 + 2(f_1 + f_2 + f_3 + f_4 + f_5 + f_6 + f_7) + f_8)$$

$$= \frac{0.125}{2}(1 + 2(0.9846 + 0.9411 + 0.8767 + 0.8 + 0.7191 + 0.64 + 0.5663) + 0.5)$$

$$= 0.784725$$

Now, from I_1 and I_2, we have to implement the Romberg Integration.

Let, $I' = I_2 + \frac{1}{3}(I_2 - I_1) = 0.782775 + \frac{1}{3}(0.782775 - 0.775) = 0.7853667$

Again for I_2 and I_3, we have:

$I'' = I_3 + \frac{1}{3}(I_3 - I_2) = 0.784725 + \frac{1}{3}(0.784725 - 0.782775) = 0.785375$

Finally, we have to refine the integration using Romberg Integration for I' and I'' as:

$I = I'' + \frac{1}{3}(I'' - I') = 0.785375 + \frac{1}{3}(0.785375 - 0.7853667) = 0.785277$

Assignment 5

Full Marks: 45
Pass Marks: 25
Grace Mark: 5

11. Estimate the approximate derivative of $f(x) = \sin^2(x)$ at $x = 0.45$ radian, for $h = 0.05$ and 0.1 using:

    a) First order Forward Difference Quotient Formula and [5]
    b) First order Central Difference Quotient Formula [5]

    Compare the result of a) and b).

    (Ref: Differentiation of continuous function)

12. The table below gives the values of distance travelled by a car at various time intervals during the initial running.

Time('t' sec)	5	6	7	8	9
Temp('T' ^{0}C)	10.0	14.5	19.5	25.5	32.0

    a) Estimate the velocity at time t = 5sec, 7sec and 9sec. [5]
    b) Also estimate the acceleration at t = 7sec. [5]

    (Ref: Differentiation of tabulated function)

13. Use trapezoidal rule with n =4 to estimate, $I = \int_0^1 \frac{dx}{1+x^2}$ correct to five decimal places. Repeat the same question using composite trapezoidal rule and analyse the results obtained by two methods. [5]

14. Estimate the integral, $I = \int_1^2 \frac{e^x}{x} dx$ with n = 4, using Simpson's (3/8) Rule. Repeat the same question using Composite Simpson's (3/8) Rule and analyse the two results. [5]

15. Evaluate the double integral, $I = \int_{y=0}^{\frac{\pi}{2}} \int_{x=0}^{\pi} \sin^2(x).\cos(y)\, dxdy$ for $h = \frac{\pi}{2}$ and $k = \frac{\pi}{4}$. [5]

16. Evaluate by Gauss Integration, the integral, $I = \int_0^{10} e^{\left(-\frac{1}{1+x^2}\right)} dx$ for n = 2. [5]

17. Use Romberg Integration to evaluate the integral, $I = \int_1^e \frac{\sqrt{\ln(x)}}{x} dx$ for h = e/2 and e/4. [5]

Prepared By
Er. Shree Krishna Khadka

6
CHAPTER

Matrices & Linear System of Equations

Contents:

- Introduction
- Method of Solving System of Linear Equations
    - Elimination/Direct Method
        - Gauss Elimination Method
        - Gauss Elimination with Pivoting
        - Gauss Jordan Method
        - Triangular Factorization Method
        - Singular Value Decomposition
    - Iterative Method
        - Jacobi Iteration Method
        - Gauss Seidel Method

Prepared By
Er. Shree Krishna Khadka

INTRODUCTION: MATRICES & SYSTEM OF LINEAR EQUATIONS

Matrices occur in a variety of problems of interest; e.g. in the solution of linear algebraic system, solution of partial and ordinary differential equations and Eigen value problems. In this chapter, we introduce the matrices, through the theory o linear transformations.

A linear equation involving two variables 'x' and 'y' has the standard form, ax + by = c, where a, b and c are real numbers such that (a and b) $\neq$ 0. This equation becomes non linear if any of the variables (x or y) have the exponent other than one.
e.g. 4x + 4y = 15 and 3u – 2v = -0.5 are the examples of linear equations, whereas 2x – xy + y = 2 and x + $\sqrt{x}$ = 6 are the examples of non linear equations.

A linear equation with 'n' variables has the form:

$$a_1x_1 + a_2x_2 + a_3x_3 + \cdots + a_nx_n = b_n \;\ldots\; (i)$$

If we need a unique solution of an equation with 'n' variables or unknown then we need a set of 'n' such independent equations known as system of simultaneous equations. Such as:

$$\begin{gathered} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2n}x_n = b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \cdots + a_{3n}x_n = b_3 \\ \text{---} \\ \text{---} \\ a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \cdots + a_{nn}x_n = b_n \end{gathered}$$

The above equation can be expressed in matrix form as: AX = B
Where; A is 'nxn' matrix, b is an 'n' vector and 'x' is a vector of 'n' unknowns.

Existence of Solution

In solving systems of equations, we are interested in identifying values of the variables that satisfy all equations in the system simultaneously. If we have given an arbitrary system of equations, it is difficult to say whether the system has solution or not. There may be four possibilities that the system has:

i) Unique Solution ii) No Solution
iii) Solution but not unique iv) Ill Conditioned

i) **Systems with Unique Solution**

For example: x + 2y = 9 and 2x – 3y = 4 has a solution (x, y) = (5, 2). Since no other pair of values of x and y would satisfy the equation, the solution is said to be unique.

ii) **Systems with No Solution**

For example: 2x – y = 5 and 3x – 3/2y = 4 has no solution. These two lines are parallel and, therefore, they never meet. Such equations are called inconsistent equations.

iii) **System has Solutions but not Unique (i.e. infinite solution)**

For example: -2x + 3y = 6 and 4x – 6y = -12 has many different solutions. We can see that these are two different forms of the same equation and, therefore, they represent the same line. Such equations are called dependent equations.

iv) **System is Ill-Conditioned**

Ill conditioned systems are very sensitive to round off errors. These errors during computing process may induce small changes in the coefficients which, in turn, may result in a large error in the solution.

Graphically, if two lines appear almost parallel, then we can say the system is ill-conditioned, since it is hard to decide just at which point they intersect.

The problem of ill-condition can be mathematically described for following two equation systems.

$$a_{11}x_1 + a_{12}x_2 = b_1$$

$$a_{21}x_1 + a_{22}x_2 = b_2$$

If these two lines are almost parallel, their slopes must be nearly equal.

i.e. $\dfrac{a_{11}}{a_{12}} \approx \dfrac{a_{21}}{a_{22}}$

Alternatively: $a_{11}a_{22} \approx a_{12}a_{21}$ i.e. $a_{11}a_{22} - a_{12}a_{21} = 0$ ... (i)

Where, equation (i) is the determinant of the coefficient matrix A. This shows that the determinant of all ill-conditioned system is very small ore nearly equal to zero.

Example:
Solve the following equations:

$$\mathbf{2x + y = 25}$$
$$\mathbf{2.001x + y = 25.01}$$

And thereby discuss the effect of ill conditioning.

Solution:

Solving above equation, we will get: $x = 10$ and $y = 5$.
If we change the coefficient of x in the second equation to 2.0005 then values of x and y will be 20 and -15 respectively.

On conclusion, a small change in one of the coefficients has resulted in a large change in the result. If we substituted these values back into the equations, we get the residuals as: $R_1 = 0$ and $R_2 = 0.01$. This illustrates the effect of round off errors on ill conditioned systems.

METHODS OF SOLVING SYSTEM OF LINEAR EQUATIONS

The techniques and methods for solving system of linear algebraic equations belong to two fundamentally different approaches.

1. **Elimination/Direct Method**
   It reduces the given system of equations to a form from which the solution can be obtained by simple substitution. We will discuss the following elimination method.

   a) Basic Gauss Elimination Method
   b) Gauss Elimination with Pivoting
   c) Gauss Jordan Method
   d) Triangular Factorization Method
   e) Singular Value Decomposition

2. **Iterative Method**
   Iterative approach, as usual, involves assumption of some initial values which are then refined repeatedly till they reach some accepted level of accuracy. We will discuss following iterative methods in this chapter.

   a) Jacobi Iterative Method
   b) Gauss Seidel Iterative Method

SOLUTION BY ELIMINATION

Elimination is a method of solving simultaneous linear equations. It involves elimination of a term containing one of the unknowns in all but one equation. One such step reduces the order of equation by one. Repeated elimination leads finally to one equation with one unknown.

Rule:

a) An equation can be multiplied or divided by a constant.
b) One equation can be added or subtracted from another equation.
c) Equations can be written in any order.

Basic Gauss Elimination Method

Gauss Elimination Method is a computer base technique for solving large systems. It proposes a systematic strategy for reducing the system of equations to the upper triangular form using the forward elimination approach and then for obtaining values of unknowns using the back substitution process.

Example:

Solve the following 3X3 system using Gauss Elimination Method.

$$3x + 2y + z = 10$$
$$2x + 3y + 2z = 11$$
$$x + 2y + 3z = 14$$

Solution:

Representing the above system of three simultaneous equation in Standard Matrix Form: i.e AX = B

$$\begin{bmatrix} 3 & 2 & 1 \\ 2 & 3 & 2 \\ 1 & 2 & 3 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 10 \\ 11 \\ 14 \end{bmatrix} = \begin{bmatrix} 3 & 2 & 1:10 \\ 2 & 3 & 2:11 \\ 1 & 2 & 3:14 \end{bmatrix}$$

$$\begin{matrix} R_2 \leftarrow R_2 - \frac{2}{3}R_1 \\ R_3 \leftarrow R_3 - \frac{1}{3}R_1 \end{matrix} \quad \begin{bmatrix} 3 & 2 & 1:10 \\ 0 & \frac{5}{3} & \frac{4}{3}:\frac{22}{3} \\ 0 & 4 & 8:32 \end{bmatrix} = \begin{bmatrix} 3 & 2 & 1:10 \\ 0 & 5 & 4:22 \\ 0 & 1 & 2:\ 8 \end{bmatrix}$$

$$R_3 \leftarrow R_3 - \frac{1}{5}R_2 \quad \begin{bmatrix} 3 & 2 & 1:10 \\ 0 & 5 & 4:22 \\ 0 & 0 & 6:18 \end{bmatrix}$$

The job up to this level is a forward elimination process. Finally, on backward substitution, we will get: z = 3, y = 2 and x = 1.

Limitation of Basic GEM and its Overcome:

Let us try another example, to solve the following system of 3-simultaneous equations, given by:

$$3x + 6y + z = 16$$
$$2x + 4y + 3z = 13$$
$$x + 3y + 2z = 9$$

$$\text{i.e.} \begin{bmatrix} 3 & 6 & 1 \\ 2 & 4 & 3 \\ 1 & 3 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 16 \\ 13 \\ 9 \end{bmatrix} = \begin{bmatrix} 3 & 6 & 1:16 \\ 2 & 4 & 3:13 \\ 1 & 3 & 2:9 \end{bmatrix}$$

$$\begin{matrix} R_2 \leftarrow R_2 - \frac{2}{3}R_1 \\ R_3 \leftarrow R_3 - \frac{1}{3}R_1 \end{matrix} \qquad \begin{bmatrix} 3 & 6 & 1:16 \\ 0 & 0 & 7:7 \\ 0 & 3 & 5:11 \end{bmatrix}$$

Here, the elimination procedure breaks down since, $A_{22} = 0$. So, second row cannot be normalized. Therefore, the procedure fails. One way to overcome this problem is to interchange this row with another row below it which does not have a zero element in that position. Then, we will have:

$$\begin{bmatrix} 3 & 6 & 1:16 \\ 0 & 3 & 5:11 \\ 0 & 0 & 7:7 \end{bmatrix}$$

Now, on backward substitution: z = 1, y = 2 and x = 1.

Gauss Elimination with Pivoting:

Here, a_{ij}, when i = j, is known as a pivot element. Each row is normalized by dividing the coefficient of that row by its pivot element.

$$\text{i.e. } a_{kj} = \frac{a_{kj}}{a_{kk}}, \text{ for } j = 1, 2, \dots, n$$

If $a_{kk} = 0$; k^{th} row cannot be normalized. Therefore, the procedure fails as in the above case. And of course, to overcome this problem is to interchange this row with another row below it which does not have a zero element in that position. But, there may be more than one non-zero values in the k^{th} column below the element a_{kk}. So, the question is: which one of them is to be selected? It can be proved that round off error would be reduced if the absolute value of the pivot element is large. Therefore, it is suggested that the row with zero pivot element should be interchanged with the row having the largest (absolute) coefficient in that position.

In general, the reordering of equations is done to improve accuracy, even if the pivot element is not zero.

Example:

Solve the following system of linear equation by Gauss Elimination with Pivoting:

$$2x + 2y + z = 6$$
$$4x + 2y + 3z = 4$$
$$x - y + z = 0$$

Solution:

$$\text{i.e.} \begin{bmatrix} 2 & 2 & 1 \\ 4 & 2 & 3 \\ 1 & -1 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 6 \\ 4 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 & 2 & 1 : 6 \\ 4 & 2 & 3 : 4 \\ 1 & -1 & 1 : 0 \end{bmatrix}$$

1. Original System, 4 in the first column below 2 is the largest absolute value among all. So, by interchanging first and second row, we will get the following modified original system with first row as a PIVOT after interchanged.

$$\begin{bmatrix} 4 & 2 & 3 : 4 \\ 2 & 2 & 1 : 6 \\ 1 & -1 & 1 : 0 \end{bmatrix}$$

2. Now, applying the Basic GEM, we will get the first derived system as below:

$$\begin{bmatrix} 4 & 2 & 3 : 4 \\ 0 & 1 & -\frac{1}{2} : 4 \\ 0 & -\frac{3}{2} & \frac{1}{4} : -1 \end{bmatrix}$$

3. Here, the largest absolute value is: 3/2 below 1. So, interchanging second and third row to have a modified first derived system as below with second row as a PIVOT after interchanged.

$$\begin{bmatrix} 4 & 2 & 3 : 4 \\ 0 & -\frac{3}{2} & \frac{1}{4} : -1 \\ 0 & 1 & -\frac{1}{2} : 4 \end{bmatrix}$$

4. Again applying Basic GEM, we will get the second and final derived system as:

$$\begin{bmatrix} 4 & 2 & 3 : 4 \\ 0 & -\frac{3}{2} & \frac{1}{4} : -1 \\ 0 & 0 & -\frac{1}{3} : \frac{10}{3} \end{bmatrix}$$

5. Finally, on backward substitution: z = -10, y = -1 and x = 9.

Gauss Jordan Method:

This method also uses the process of elimination of variables. But, the variable is eliminated from all other rows (both below and above). This process thus eliminates all the off-diagonal terms producing a diagonal matrix rather than a triangular matrix. Further, all rows are normalized by dividing them by their pivot elements.

Example:

Solve by Gauss Jordan Method:

$$2x + 4y - 6z = -8$$
$$x + 3y + z = 10$$
$$2x - 4y - 2z = -12$$

Solution:

$$\text{i.e.} \begin{bmatrix} 2 & 4 & -6 \\ 1 & 3 & 1 \\ 2 & -4 & -2 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -8 \\ 10 \\ -12 \end{bmatrix} = \begin{bmatrix} 2 & 4 & -6:-8 \\ 1 & 3 & 1:10 \\ 2 & -4 & -2:-12 \end{bmatrix}$$

$$R_1 \leftarrow \tfrac{1}{2}R_1 \qquad \begin{bmatrix} 1 & 2 & -3:-4 \\ 1 & 3 & 1:10 \\ 2 & -4 & -2:-12 \end{bmatrix}$$

$$\begin{matrix} R_2 \leftarrow R_2 - R_1 \\ R_3 \leftarrow R_3 - 2R_1 \end{matrix} \qquad \begin{bmatrix} 1 & 2 & -3:-4 \\ 0 & 1 & 4:14 \\ 0 & -8 & 4:-4 \end{bmatrix}$$

$$\begin{matrix} R_1 \leftarrow R_1 - 2R_2 \\ R_3 \leftarrow R_3 + 8R_2 \end{matrix} \qquad \begin{bmatrix} 1 & 0 & -11:-32 \\ 0 & 1 & 4:14 \\ 0 & 0 & 36:108 \end{bmatrix}$$

$$\begin{matrix} R_3 \leftarrow \tfrac{1}{36}R_3 \\ R_3 \leftarrow R_3 - \dfrac{1}{3}R_1 \end{matrix} \qquad \begin{bmatrix} 1 & 0 & -11:-32 \\ 0 & 1 & 4:14 \\ 0 & 0 & 1:3 \end{bmatrix}$$

$$\begin{matrix} R_1 \leftarrow R_1 + 11R_3 \\ R_2 \leftarrow R_2 - 4R_3 \end{matrix} \qquad \begin{bmatrix} 1 & 0 & 0:1 \\ 0 & 1 & 0:2 \\ 0 & 0 & 1:3 \end{bmatrix}$$

Triangular Factorization Method

Since, the system of linear equation can be expressed in the matrix form as: AX = B. So, here in the triangular factorization method, the coefficient matrix A of a system of linear equations can be factorized or decomposed into two triangular matrices L and U such that: A = LU …. (i).

$$\text{Where, } L = \begin{bmatrix} l_{11} & 0 & 0 & 0 & 0 & 0 \\ l_{21} & l_{22} & 0 & 0 & 0 & 0 \\ l_{31} & l_{32} & l_{33} & 0 & 0 & 0 \\ \ldots & \ldots & \ldots & l_{44} & 0 & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & 0 \\ l_{n1} & l_{n2} & \ldots & \ldots & l_{n(n-1)} & l_{nn} \end{bmatrix}, \text{ known as lower triangular matrix.}$$

$$\& \; U = \begin{bmatrix} u_{11} & u_{12} & u_{13} & \ldots & \ldots & u_{1n} \\ 0 & u_{22} & u_{23} & u_{24} & \ldots & u_{2n} \\ 0 & 0 & u_{33} & u_{34} & \ldots & u_{3n} \\ \ldots & \ldots & \ldots & u_{44} & u_{45} & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & \ldots & u_{n(n-1)} & u_{nn} \end{bmatrix}, \text{ known as upper triangular matrix.}$$

Once, A is factorized into L and U, the system of equations AX = B can be expressed by: (LU)X = B, i.e. L(UX) = B … (ii)

If we assume, UX = Y, where Y is an unknown vector. Then:

$$LY = B \;…. \text{(iii)}$$

Now, we can solve AX = B in two stages:

a) Solving the equation: LY = B for Y by forward substitution and
b) Solving the equation UX = Y for X using Y by backward substitution.

The elements of L and U can be determined by comparing the elements of the product o L and U with those of A. This is done by assuming the diagonal elements of L or U to be unity.

- The decomposition with L having unit diagonal values is called the Dolittle LU Decomposition.

- The decomposition with U having unit diagonal elements is called the Crout LU Decomposition.

Example:

Solve the system of three simultaneous linear equations by using Dolittle LU Decomposition Method.

$$\mathbf{3x + 2y + z = 10}$$
$$\mathbf{2x + 3y + 2z = 14}$$
$$\mathbf{x + 2y + 3z = 14}$$

Solution:

Using Dolittle LU Decomposition, we have:

$$[A] = \begin{bmatrix} 3 & 2 & 1 \\ 2 & 3 & 2 \\ 1 & 2 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix}$$

$$= \begin{bmatrix} 3 & 2 & 1 \\ 2 & 3 & 2 \\ 1 & 2 & 3 \end{bmatrix} = \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ l_{21}u_{11} & l_{21}u_{12} + u_{22} & l_{21}u_{12} + u_{23} \\ l_{31}u_{11} & l_{31}u_{12} + l_{32}u_{22} & l_{31}u_{13} + l_{32}u_{23} + u_{33} \end{bmatrix}$$

On comparisons, we will have the following relations:

$u_{11} = 3$, $u_{12} = 2$, $u_{13} = 1$, $l_{21}u_{11} = 2 : l_{21} = 2/3$,

$l_{21}u_{12} + u_{22} = 3$: $u_{22} = 3 - (2/3)2 = 5/3$,

$l_{21}u_{12} + u_{23} = 2$: $u_{23} = 2 - (2/3)1 = 4/3$,

$l_{31}u_{11} = 1$: $l_{31} = 1/3$, $l_{31}u12 + l_{32}u_{22} = 2$: $l_{32} = (2 - (1/3)2)/(5/3) = 4/5$

$l_{31}u_{13} + l_{32}u_{23} + u_{33} = 3$: $u_{33} = (3 - (4/5)(4/3) - (1/3)(1) = 24/15$

Thus we have: $L = \begin{bmatrix} 1 & 0 & 0 \\ \frac{2}{3} & 1 & 0 \\ \frac{1}{3} & \frac{4}{5} & 1 \end{bmatrix}$ & $U = \begin{bmatrix} 3 & 2 & 1 \\ 0 & \frac{5}{3} & \frac{4}{3} \\ 0 & 0 & \frac{24}{15} \end{bmatrix}$

(i) Forward Substitution: Solving: $\begin{bmatrix} 1 & 0 & 0 \\ \frac{2}{3} & 1 & 0 \\ \frac{1}{3} & \frac{4}{5} & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 10 \\ 14 \\ 14 \end{bmatrix}$, we will get:

$y_1 = 10$, $y_2 = 22/3$ and $y_3 = 72/15$.

(ii) Backward Substitution: Solving: $\begin{bmatrix} 3 & 2 & 1 \\ 0 & \frac{5}{3} & \frac{4}{3} \\ 0 & 0 & \frac{24}{15} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 10 \\ \frac{22}{3} \\ \frac{72}{15} \end{bmatrix}$, we will get:

$z = 3$, $y = 2$ and $x = 1$.

SINGULAR VALUE DECOMPOSITION

Since, we used LU Decomposition only for a square matrix A. But a similar decomposition is also possible for a rectangular matrix, which is given by Singular Value Decomposition. This method is of great importance in matrix theory since it is useful in finding the generalized inverse of a singular matrix and has several image processing applications.

Let A be an (mxn) real matrix with m $\geq$n. Then, the matrices A^TA and AA^T are non-negative, symmetric and have identical Eigen values, say λ_n. We can then obtain the 'n' ortho-normalized Eigen vectors, say X_n of A^TA such that:

$$(A^TA)X_n = \lambda_n X_n \ \dots \text{(i)}$$

If we assume Y_n to be the 'n' ortho-normalized Eigen vectors of AA^T, then we have:

$$(AA^T)Y_n = \lambda_n Y_n \ \dots \text{(ii)}$$

Then, A can be decomposed into the form: $A = UDV^T$ … (iii)

$$\text{Where:} \qquad U = [\overrightarrow{y_1}, \overrightarrow{y_2}, \overrightarrow{y_3} \dots \overrightarrow{y_n}], V = [\overrightarrow{x_1}, \overrightarrow{x_2}, \overrightarrow{x_3} \dots \overrightarrow{x_n}]^T$$

And D is a diagonal matrix, given by: Diag $(\sqrt{\lambda_1}, \sqrt{\lambda_2}, \sqrt{\lambda_3} \dots . \sqrt{\lambda_n})$

It follows that the Eigen vectors of AA^T, i.e. Y_n are given by:

$$Y_n = \frac{1}{\sqrt{\lambda_n}} AX_n \ \dots \text{(iv)}$$

Example:

Obtain the Singular Value Decomposition of: $\mathbf{A} = \begin{bmatrix} \mathbf{1} & \mathbf{2} \\ \mathbf{1} & \mathbf{1} \\ \mathbf{1} & \mathbf{3} \end{bmatrix}$.

Solution:

We have:

1. $A = \begin{bmatrix} 1 & 2 \\ 1 & 1 \\ 1 & 3 \end{bmatrix}$ So, $A^T = \begin{bmatrix} 1 & 1 & 1 \\ 2 & 1 & 3 \end{bmatrix}$, Then: $A^TA = \begin{bmatrix} 3 & 6 \\ 6 & 14 \end{bmatrix}$

2. Now, to find Eigen Vectors of A^TA, we have the characteristics equation:
   (Eigen Value Problem: An Example)

   $|A^TA - \lambda I| = 0$

$$\left|\begin{bmatrix} 3 & 6 \\ 6 & 14 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right| = 0$$

$$\left|\begin{bmatrix} 3-\lambda & 6 \\ 6 & 14-\lambda \end{bmatrix}\right| = 0$$

$$(3-\lambda)(14-\lambda) - 36 = 0$$

$$42 - 3\lambda - 14\lambda + \lambda^2 = 0$$

On solving for λ, we will get: λ_1 = 16.64 and λ_2 = 0.36
Then: D = Diag.($\sqrt{16.64}$, $\sqrt{0.36}$) = Diag.(4.080, 0.60)

3. Now, for the corresponding Eigen Vectors of A^TA, we have:

4. $\begin{bmatrix} 3 & 6 \\ 6 & 14 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 16.64\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ & $\begin{bmatrix} 3 & 6 \\ 6 & 14 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0.36\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ ($\therefore$ From equation (i))

This gives the systems:

$$3x_1 + 6x_2 = 16.64x_1 \qquad\qquad 3x_1 + 6x_2 = 0.36x_1$$
$$6x_1 + 14x_2 = 16.64x_1 \qquad\qquad 6x_1 + 14x_2 = 0.36x_1$$

The solution is then given by:

$$x_1 = \begin{bmatrix} 0.4033 \\ 0.9166 \end{bmatrix} \qquad\qquad x_2 = \begin{bmatrix} 0.9166 \\ -0.4033 \end{bmatrix}$$

Then: $X_n = [\overrightarrow{x_1}, \overrightarrow{x_2}]^T = \begin{bmatrix} 0.4033 & 0.9166 \\ 0.9166 & -0.4033 \end{bmatrix}$

5. Finally, the Eigen Vectors of AA^T, we have:

$$Y_1 = \frac{1}{\sqrt{\lambda_1}} AX_1 \qquad\qquad Y_2 = \frac{1}{\sqrt{\lambda_2}} AX_2$$

$$= \frac{1}{4.080}\begin{bmatrix} 1 & 2 \\ 1 & 1 \\ 1 & 3 \end{bmatrix}\begin{bmatrix} 0.4033 \\ 0.9166 \end{bmatrix} = \begin{bmatrix} 0.5480 \\ 0.3235 \\ 0.7727 \end{bmatrix} \qquad = \frac{1}{0.6}\begin{bmatrix} 1 & 2 \\ 1 & 1 \\ 1 & 3 \end{bmatrix}\begin{bmatrix} 0.9166 \\ -0.4033 \end{bmatrix} = \begin{bmatrix} 0.1833 \\ 0.8555 \\ -0.4889 \end{bmatrix}$$

So, the singular value decomposition of A is:

$$[A] = \begin{bmatrix} 1 & 2 \\ 1 & 1 \\ 1 & 3 \end{bmatrix} = \begin{bmatrix} 0.5480 & 0.1833 \\ 0.3235 & 0.8555 \\ 0.7727 & -0.4889 \end{bmatrix}\begin{bmatrix} 4.080 & 0 \\ 0 & 0.60 \end{bmatrix}\begin{bmatrix} 0.4033 & 0.9166 \\ 0.9166 & -0.4033 \end{bmatrix}$$

SOLUTION BY ITERATION

Solution by direct method poses some problems when the systems grow larger or when most of the coefficients are zero. They require prohibitively large number of floating point operation and therefore, not only time consuming but also severely affect the accuracy of the solution due to round off errors. Iterative methods provide an alternative for this and also ill conditioned systems can be solved by this method without facing the problem of round off errors.

Jacobi Iteration Method

Jacobi method is base on the idea as that for the fixed point iteration method discussed in chapter two. Recalling that equation of the form: f(x) = 0 can be rearranged into a form: x = g(x). The function g(x) can be evaluated iteratively using an initial approximation x as: $x_{i+1} = g(x_i)$ … (i)

Jacobi method extends this idea to a system of equations. It is direct substitution method where the values of unknowns are improved by substituting directly the previous values.

Let us consider a system of 'n' equations in 'n' unknowns:

$$\begin{gathered}
a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n = b_1 \\
a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2n}x_n = b_2 \\
a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \cdots + a_{3n}x_n = b_3 \\
\text{---} \\
\text{---} \\
a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \cdots + a_{nn}x_n = b_n
\end{gathered}$$

We can rewrite these original systems as:

$$\begin{gathered}
x_1 = \{b_1 - (a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n)\}/a_{11} \\
x_2 = \{b_2 - (a_{21}x_1 + a_{23}x_3 + \cdots + a_{2n}x_n)\}/a_{22} \\
\text{---} \\
\text{---} \\
x_n = \{b_n - (a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + a_{n(n-1)}x_{n-1})\}/a_{nn}
\end{gathered}$$

Now, we can compute x_1, x_2… x_n by using initial guesses for these values. These new values are again used to compute the next set of x values. The process can continue till we obtain a desired level of accuracy in the x values.

Example:

Obtain the solution of the following system using the Jacobi Iteration Method.

$$2x + y + z = 5$$
$$3x + 5y + 2z = 15$$
$$2x + y + 4z = 8$$

Solution:

Rewriting the above equation in the form:

$$x = (5 - y - z)/2$$
$$y = (15 - 3x - 2z)/5$$
$$z = (8 - 2x - y)/4$$

If we assume initial values of x, y and z to be zero, then going by iteration we will get the following results:

n	x	y	Z
0	0	0	0
1	2.5	3	2
2	0	0.7	0
3	2.15	3	1.825
4	0.0875	1.225	0.175
5	…	…	…

The process can be continued till the value of x reach a desired level of accuracy.

Gauss Seidel Iteration Method

It is an improved version of Jacobi Iteration Method. In Jacobi method, we begin with the initial values: $x_1^{(0)}, x_2^{(0)}, x_3^{(0)} \ldots x_n^{(0)}$ and obtain the next approximation$x_1^{(1)}, x_2^{(1)}, x_3^{(1)} \ldots x_n^{(1)}$. Note that, in computing $x_2^{(1)}$, we used $x_1^{(0)}$ and not $x_1^{(1)}$ which has just been computed. Since, at this point, both $x_1^{(0)}$ and $x_1^{(1)}$ are available, we can use $x_1^{(1)}$ which is a better approximation for computing $x_2^{(1)}$. Similarly, for computing $x_3^{(1)}$, we can use $x_1^{(1)}$ and $x_2^{(1)}$ along with $x_4^{(0)}$, … , $x_n^{(0)}$. This idea can be extended to all subsequent computations. This approach is called the Gauss Seidel Method.

Example:

Obtain the solution of the following system using the Jacobi Iteration Method.

$$2x + y + z = 5$$
$$3x + 5y + 2z = 15$$
$$2x + y + 4z = 8$$

Solution:

Rewriting the above equation in the form:

$$x = (5 - y - z)/2$$
$$y = (15 - 3x - 2z)/5$$
$$z = (8 - 2x - y)/4$$

If we assume initial values of x, y and z to be zero, then going by iteration we will get the following results:

n	x	y	Z
0	0	0	0
1	2.5	1.5	0.4
2	1.6	1.9	0.7
3	1.2	2	0.9
4	1.1	2	1
5	1	2	1

Hence, x = 1, y = 2 and z = 1.

Assignment 6

Full Marks: 50
Pass Marks: 25

18. Use Gauss Elimination Method to solve:

$$2w + 2x + y + 2z = 7$$
$$w - 2x - z = 2$$
$$3w - x - 2y - z = 3$$
$$w - 2z = 0$$ [7]

19. A system of simultaneous linear equations are given below:

$$2x + y - z = -1$$
$$x - 2y + 3z = 9$$
$$3x - y + 5z = 14$$

a) Solve by Gauss Elimination with pivoting and [5]
b) Solve by Gauss Jordan Method, Compare two methods. [5]

20. Decompose the matrix $A = \begin{bmatrix} 5 & -2 & 1 \\ 7 & 1 & -5 \\ 3 & 7 & 4 \end{bmatrix}$ into the form DoLittle/Crout LU and hence solve the system AX = B, where B = $[4 \;\; 8 \;\; 10]^T$ [8]

21. Determine the Eigen Values and corresponding Eigen Vectors for the matrix: $A = \begin{bmatrix} 1 & 6 & 1 \\ 1 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix}$. [5]

22. Compute the singular value decomposition of the matrix: $A = \begin{bmatrix} 1 & 2 \\ 2 & 1 \\ 1 & 3 \end{bmatrix}$. [10]

23. Find the solution to three decimals, of the system:

$$83x + 11y - 4z = 95$$
$$7x + 52y + 13z = 104$$
$$3x + 8y + 29z = 71$$

a) Using Jacobi Iteration Method [4]
b) Using Gauss Seidel Iteration Method and [4]
c) Compare the results and execution time. [2]

7
CHAPTER

Numerical Solution of Ordinary Differential Equations

Contents:

- Introduction
- Initial Value Problems
  - Solution by Taylor's Series Method
  - Euler's Method
  - Heun's Method
  - Runge Kutta Method
  - Predictor Corrector Method
  - Solution of Simultaneous & Higher Order Equations
- Boundary Value Problem
  - Finite Difference Method
  - Shooting Method

INTRODUCTION: DIFFERENTIAL EQUATIONS

Most of the real time physical systems are expressed in terms of rate of change. The Mathematical models that describe the state of such systems are often expressed in terms of not only certain system parameters but also their derivatives. Such model which used differential calculus to express relationship between variables is known as differential equation.

Types of Differential Equations:

1.) Ordinary Differential Equations

The equation which involves an independent variable and dependent variable with its ordinary derivatives of any order is called ordinary differential equation. Examples of such equations are:

a) **Law of Cooling:** $\frac{dT(t)}{dx} = K\{T_s - T(t)\}$: The rate of loss of heat from a liquid is proportional to the difference of temperature between the liquid and the surroundings.

b) **Law of Motion**: $m\frac{dv(t)}{dt} = F$: The time rate change of velocity of a moving body is proportion to the force exerted by the body.

c) **Kirchhoff's Law for and Electric Circuit**: $L\frac{di}{dt} = iR = V$: The voltage across an electric circuit containing an inductance L and a resistance R.

The general form of the ordinary differential equation is: $F(x, y, y', y'', \ldots y^n) = 0$. Where, 'x' is independent variable, y is dependent and y', y'' ... are derivatives of 'y' with respect to 'x'.

2.) Partial Differential Equations

The equation which involves two or more independent variables, dependent variables and its partial derivative is called Partial Differential Equation. Examples are:

a) **Heat Flow in a Rectangular Plate:** $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x, y)$ etc.

More about partial differential equations, we will discuss in next chapter.

Order and Degree of Ordinary Differential Equation

The order of a differential equation is the highest derivative that appears in the equation. When the equation contains only a first derivative, it is called a first order differential equation. On the other hand, the degree of a differential equation is the power of the highest order derivative.

For example:

$xy'' + y^2y' = 2y + 3$ is a 2nd order 1st degree equation, while:
$(y''')^2 + 5y' = 0$ is a 3rd order, 2nd degree equation.

Linear and Non Linear Differential Equation

A differential equation is said to be linear when it does not contain terms involving the products of the dependent variable or its derivative.

On the other side, equations avoiding the above rules are known to be non linear equations. For example: $y'' + 3y' = 2y + x^2$ is a 2nd order 1st degree linear equation. But the equation $y'' + (y')^2 = 1$ and $y' = -ay^2$ are non linear equations.

General and Particular Solution

A solution to a differential equation is a relationship between the dependent and independent variables that satisfy the differential equation. The solution that contains arbitrary constants and is not unique is known as the general solution. For example: $y' = 6x + 1$ has number of solutions as, $y = 3x^2 + x + 2$, $y = 3x^2 + x - 10$… Therefore, $y = 3x^2 + x + C$ is considered as general solution.

On the other part, if the values of the constants are known, then on substitution of these values in the general solution, a unique solution is obtained which is known as particular solution.

One Step and Multi Step Method of Solution

In one step methods, we use information from only one preceding point, i.e. to estimate the value of y_i, we need the conditions at the previous point y_{i-1} only. Multistep methods use information at two or more previous steps to estimate a value.

SOLUTION BY TAYLOR'S SERIES METHOD

Let us consider a differential equation in the form: $y' = f(x, y)$ with the initial condition $y(0) = y(x = 0) = y(x_0) = y_0$. If $y(x)$ is the exact solution, then we can expand a function $y(x)$ about a point $x = x_0$ using Taylor's Theorem of Expansion.

$$y(x) = y(x_0) + (x - x_0)y'(x_0) + (x - x_0)^2 \frac{y''(x_0)}{2!} + \cdots + (x - x_0)^n \frac{y^n(x_0)}{n!} \ldots \text{ (i)}$$

Now, we must repeatedly differentiate $f(x, y)$ implicitly with respect to x and evaluate them at x_0. Then equation (i) gives a power series of y.

Example:
From the Taylor's Series for y(x), find y(0.1) correct to four decimal places if y(x) satisfies: $y' = x - y^2$ and $y(0) = 1$.

Solution:
The Taylor's Series for y(x) is given by:

$$y(x) = y(x_0) + (x - x_0)y'(x_0) + (x - x_0)^2 \frac{y''(x_0)}{2!} + \cdots + (x - x_0)^n \frac{y^n(x_0)}{n!}$$

We have:

$y'(x) = x - y^2$	$y'(0) = -1$
$y''(x) = 1 - 2yy'$	$y''(0) = 3$
$y'''(x) = -2yy'' - 2y'^2$	$y'''(0) = -8$
$y^{iv}(x) = -2yy''' - 6y'y''$	$y^{iv}(0) = 34$
$y^{v}(x) = -2yy^{iv} - 8y'y''' - 6y''^2$	$y^{v}(0) = -186$ and so on…

Using these values, equation (i) gives a power series as follows:

$$y(x) = 1 - x + \frac{3}{2!}x^2 - \frac{4}{3}x^3 + \frac{17}{12}x^4 - \frac{31}{20}x^5 + \cdots$$

Now: for the value of y(0.1) correct up to four decimal place, it is found that the terms up to x^4 should be considered, then we will have: $y(0.1) = 0.9138$ with the truncation of $(31/20)x^5 \leq 0.00005$.

/Home Work/
Given the differential equation: $y'' - xy' - y = 0$ with the conditions $y(0) = 1$ and $y'(0)$, use Taylor's Series Method to determine the value of y(0.1).

EULER'S METHOD: (1ST ORDER RUNGE KUTTA METHOD)

The solution of differential equation by Taylor's Series Method gives the equation in the form of power series. We will now discuss the methods which give the solution in the form of a set of tabulated values. Euler's Method is one of the simplest one-step methods and it has limited application because of its low accuracy.

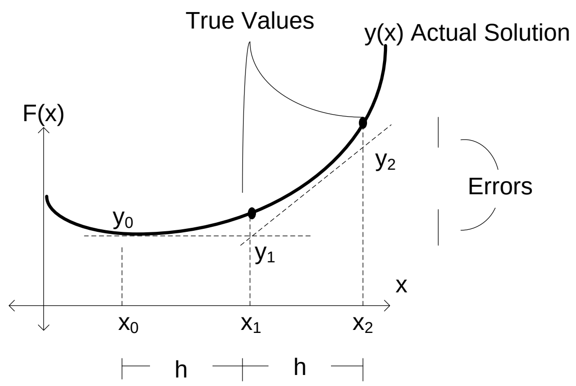


Fig: Illustration of Euler's Method

Let us consider the first two terms of the Taylor's Series Expansion as:

$$y(x) = y(x_0) + y'(x_0)(x - x_0) \ldots (i)$$

If we have given the differential equation $y'(x) = f(x, y)$ with the initial condition by $y(x_0) = y_0$. Then: $y'(x_0) = f(x_0, y_0)$.

Hence, equation (i) becomes: $y(x) = y(x_0) + f(x_0, y_0).(x - x_0)$

And also : $y(x_1) = y(x_0) + f(x_0, y_0).(x_1 - x_0)$

i.e. : $y_1 = y_0 + f(x_0, y_0).h$

Similarly, : $y_2 = y_1 + f(x_1, y_1).h$

In general, : $y_{i+1} = y_i + f(x_i, y_i).h \ldots (ii)$

Which is the main formula given by the Euler's Method.

Example:

Given the equation $y' = 3x^2 + 1$ with $y(1) = 2$, estimate $y(2)$ bye Euler's Method using (i) h = 0.5 and (ii) h = 0.25.

Solution:

Case I: For h = 0.5

$y(1) = 2$

$y(1.5) = 2 + 0.5[3(1.0)^2 + 1] = 4.0$

$y(2.0) = 4.0 + 0.5[3(1.5)^2 + 1] = 7.875$

Case II: For h = 0.25

$y(1) = 2$

$y(1.25) = 2 + 0.25[3(1.0)^2 + 1] = 3.0$

$y(1.50) = 3.0 + 0.25[3(1.25)^2 + 1] = 5.42188$

$y(1.75) = 5.42188 + 0.25[3(1.50)^2 + 1] = 7.35938$

$y(2.0) = 7.35938 + 0.25[3(1.75)^2 + 1] = 9.90626$

/Howe Work/For y' = 2y/x, find y(2) at h = 0.25/

HEUN'S METHOD: (MODIFIED EULER'S METHOD/2ND ORDER RUNGE KUTTA METHOD)

Since, Euler's Method does not require any differentiation and is easy to implement on computers. However, it major weakness is large truncation errors. This is due to its linear characteristic because it uses only the first two terms of the Taylor's Series. So, Heun's Method is considered to be an improvement to Euler's Method.

In Euler's Method, the slope at the beginning of the interval is used to extrapolate y_i to y_{i+1} over the entire interval by the formula:

$$y_{i+1} = y_i + m_1h \qquad \text{... (i)}$$

An alternative is use the line which is parallel to the tangent at the point $(x_{i+1}, y(x_{i+1}))$ to extrapolate from y_i, to y_{i+1} as shown in the figure by the formula:

$$y_{i+1} = y_i + m_2h \qquad \text{... (ii)}$$

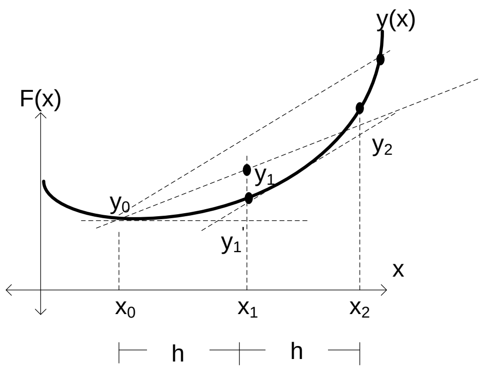


Fig: Illustration of Heun's Method

The estimate given by equation (ii) appears to be overestimated.

Now, a third approach is to use a line whose slope is the average of the slope at the end points of the interval. Then:

$$y_{i+1} = y_i + \frac{m_1+m_2}{2} h \quad \ldots \text{(iii)}$$

As shown in figure, this gives a better approximation to y_{i+1}. This approach is known as Heun's Method. Now from equations (i) and (ii), equation (iii) becomes:

$$y_{i+1} = y_i + \frac{h}{2}[m_1 + m_2] = \frac{h}{2}[f(x_i, y_i) + f(x_{i+1}, y_{i+1})] \ldots \text{(iv)}$$

Here, the term y_{i+1} is appearing on both sides of the equation (iv) and therefore, y_{i+1} cannot be evaluated until the value of y_{i+1} inside the function $f(x_{i+1}, y_{i+1})$ is available. This value can be predicted using the Euler's Formula given in equation (i). So, finally, equation (iv) becomes:

$$y_{i+1} = \frac{h}{2}[f(x_i, y_i) + f(x_{i+2}, y_{i+1}{}^{e})] \ldots \text{(v)}$$

Hence, equation (v) is called improved version of Euler's Method. It is also classified as a one – step Predictor Corrector Method. Equation (i) is known as the predictor equation whereas equation (v) is known as corrector equation.

Example:
Given the equation: y' = 2y/x. Estimate y(2) using Heun's Method at h = 0.25.

Solution:
$y' = f(x, y) = 2y/x$: For: $x_0 = 0, y_0 = 2$: $h = 0.25$

Iteration: 1
$m_1 = (2X2)/1 = 4.0$ $y_e(1.25) = 2 + 0.25(4.0) = 3.0$
$m_2 = (2X3.0)/1.25 = 4.8$ $y(1.25) = 2 + (0.25)/2(4.0 + 4.8) = 3.1$

Iteration: 2
$m_1 = (2X3.1)/1.25 = 4.96$ $y_e(1.5) = 3.1 + 0.25(4.96) = 4.34$
$m_2 = (2X4.34)/1.5 = 5.79$ $y(1.5) = 3.1 + (0.25)/2(4.96 + 5.79) = 4.44$

Iteration: 3
$m_1 = (2X4.4)/1.5 = 5.92$ $y_e(1.75) = 4.44 + 0.25(5.92) = 5.92$
$m_2 = (2X5.92)/1.75 = 6.77$ $y(1.75) = 4.44 + (0.25)/2(5.92 + 6.77) = 6.03$

Iteration: 4
$m_1 = (2X6.03)/1.75 = 6.89$ $y_e(2) = 6.03 + 0.25(6.89) = 7.75$
$m_2 = (2X7.75)/2 = 7.75$ $y(2) = 6.03 + (0.25)/2(6.89 + 7.75) = 7.86$

Therefore: y(2) = 7.86

FOURTH ORDER RUNGE KUTTA METHOD

It is also a family of one step methods used for numerical solution of initial value problems. It is also based on the general form of the extrapolation equation:

$$y_{i+1} = y_i + mh \; \ldots(i)$$

Where, 'm' represents the slope that is weighted averages of the slopes at various points in the interval 'h'. Runge Kutta (RK) methods are known by their order. For example, an RK method is called r^{th} order RK method when slopes at 'r' points are used to construct the weighted average slope 'm'. In Euler's method we used only one slope at (x_i, y_i) and Heun's method employs slopes at two end points of the interval so, they are called first-order and second-order RK method.

Higher the order better would be the accuracy of estimates. Therefore, selection of order for using RK methods depends on the problem under consideration.

So, in 4^{th} order RK method, we use four slopes and they are given by:

$$m_1 = f(x_i, y_i)$$
$$m_2 = f\left(x_i + \frac{h}{2},\ y_i + \frac{m_1 h}{2}\right)$$
$$m_3 = f\left(x_i + \frac{h}{2},\ y_i + \frac{m_2 h}{2}\right)$$
$$m_4 = f(x_i + h,\ y_i + m_3 h)$$

Then, the general form of extrapolation equation (i) will be:

$$y_{i+1} = y_i + \frac{h}{6}(m_1 + 2m_2 + 2m_3 + m_4) \;\ldots\; (ii)$$

Example

Given $y'(x) = x^2 + y^2$, $y(0) = 0$ using 4^{th} order RK method, estimate $y(0.4)$, take $h = 0.2$

Solution:

We have given: $f(x, y) = x^2 + y^2$ and $y_0 = 0$ at $x_0 = 0$

To find: $y_n = ?$ at $x_n = 0.4$ for $h = 0.2$

Total number of iteration (n) = $(x_n - x_0)/h = (0.4 - 0)/0.2 = 2$

For iteration 1, (i =0)

$$m_1 = f(x_i, y_i) = f(x_0, y_0) = f(0,0) = 0^2 + 0^2 = 0$$

$$m_2 = f\left(x_i + \frac{h}{2},\ y_i + \frac{m_1 h}{2}\right) = f\left(x_0 + \frac{h}{2},\ y_0 + \frac{m_1 h}{2}\right) = f\left(0 + \frac{0.2}{2},\ 0 + \frac{0 \times 0.2}{2}\right)$$
$$= f(0.1, 0) = 0.1^2 + 0^2 = 0.01$$

$$m_3 = f\left(x_i + \frac{h}{2},\ y_i + \frac{m_2 h}{2}\right) = f\left(x_0 + \frac{h}{2},\ y_0 + \frac{m_2 h}{2}\right) = f\left(0 + \frac{0.2}{2},\ 0 + \frac{0.01 \times 0.2}{2}\right)$$
$$= f(0.1, 0.001) = 0.1^2 + 0.001^2 = 0.010001$$

$$m_4 = f(x_0 + h,\ y_0 + m_3 h) = f(0 + 0.2,\ 0 + 0.010001 \times 0.2)$$
$$= f(0.2, 0.0020002) = 0.2^2 + 0.002002^2 = 0.04$$

$$y_1 = y(0.2) = y_0 + \frac{h}{6}(m_1 + 2m_2 + 2m_3 + m_4)$$
$$= 0 + \frac{0.2}{6}(0 + 2 \times 0.01 + 2 \times 0.01001 + 0.04) = 0.002667$$

For Iteration 2, (i =1 x_1 = 0.2, y_1 = 0.002667)

$$m_1 = f(x_i, y_i) = f(x_1, y_1) = f(0.2, 0.002667) = 0.2^2 + 0.002667^2 = 0.04$$

$$m_2 = f\left(x_1 + \frac{h}{2},\ y_1 + \frac{m_1 h}{2}\right) = f\left(x_1 + \frac{h}{2},\ y_1 + \frac{m_1 h}{2}\right)$$
$$= f\left(0.2 + \frac{0.2}{2},\ 0.002667 + \frac{0.04 \times 0.2}{2}\right) = f(0.3, 0.006667) = 0.090044$$

$$m_3 = f\left(x_1 + \frac{h}{2},\ y_1 + \frac{m_2 h}{2}\right) = f\left(x_1 + \frac{h}{2},\ y_1 + \frac{m_2 h}{2}\right)$$
$$= f\left(0.2 + \frac{0.2}{2},\ 0.002667 + \frac{0.090044 \times 0.2}{2}\right) = f(0.3,\ 0.0116714) = 0.090136$$

$$m_4 = f(x_1 + h,\ y_1 + m_3 h) = f(0.2 + 0.2,\ 0.002667 + 0.090136 \times 0.2)$$
$$= f(0.4, 0.0206942) = 0.160428$$

$$y_2 = y(0.4) = y_1 + \frac{h}{6}(m_1 + 2m_2 + 2m_3 + m_4)$$
$$= 0.002667 + \frac{0.2}{6}(0.04 + 2 \times 0.090044 + 2 \times 0.090136 + 0.160428)$$
$$= 0.021360$$

SIMULTANEOUS FIRST ORDER DIFFERENTIAL EQUATIONS

Mathematical models of many applications involve a system of several first order differential equations. They may be represented as follows:

$$\frac{dy_1}{dx} = f_1(x, y_1, y_2 \dots, y_n), y_1(x_0) = y_{10}$$
$$\frac{dy_2}{dx} = f_2(x, y_1, y_2 \dots, y_n), y_1(x_0) = y_{20}$$

....

....

$$\frac{dy_n}{dx} = f_n(x, y_1, y_2 \dots, y_n), y_1(x_0) = y_{n0}$$

These equations should be solved for $y_1(x)$, $y_2(x)$ … $y_n(x)$ over interval (a, b).

Let us consider two such simultaneous differential equations, given by:

$$\frac{dy}{dx} = f(x, y, z) \qquad \& \qquad \frac{dz}{dx} = g(x, y, z)$$

Given that: $y(x_0) = y_0$ and $z(x_0) = z_0$

Then we follow the solution approach given by the following algorithm.

$$m_1 = f(x_i, y_i, z_i)$$
$$m_2 = f\left(x_i + \frac{h}{2},\ y_i + \frac{m_1 h}{2}, z_i + \frac{l_1 h}{2}\right)$$
$$m_3 = f\left(x_i + \frac{h}{2},\ y_i + \frac{m_2 h}{2}, z_i + \frac{l_2 h}{2}\right)$$
$$m_4 = f(x_i + h,\ y_i + m_3 h + z_i + l_3 h)$$

Then:

$$y_{i+1} = y_i + \frac{h}{6}(m_1 + 2m_2 + 2m_3 + m_4) \dots \text{(i)}$$

$$l_1 = f(x_i, y_i, z_i)$$
$$l_2 = f\left(x_i + \frac{h}{2},\ y_i + \frac{m_1 h}{2}, z_i + \frac{l_1 h}{2}\right)$$
$$l_3 = f\left(x_i + \frac{h}{2},\ y_i + \frac{m_2 h}{2}, z_i + \frac{l_2 h}{2}\right)$$
$$l_4 = f(x_i + h,\ y_i + m_3 h + z_i + l_3 h)$$

Then:

$$z_{i+1} = z_i + \frac{h}{6}(l_1 + 2l_2 + 2l_3 + l_4) \dots \text{(ii)}$$

Example:

Find y(0.1) and z(0.1) from the system of equations: $y' = x + z$ and $z' = x - y^2$ with $y(0) = 2$, $z(0) = 1$ and $h = 0.1$

Solution:

We have, $x_0 = 0$, $y_0 = 2$, $z_0 = 1$ and $h = 0.05$

$m_1 = f(x_0, y_0, z_0) = f(0, 2, 1) = (x_0 + z_0) = 0 + 1 = 1$

$l_1 = g(x_0, y_0, z_0) = g(0, 2, 1) = (x_0 - y_0^2) = 0 - 2^2 = -4$

$m_2 = f(0.05, 2.05, 0.8) = 0.85$

$l_2 = g(0.05, 2.05, 0.8) = -4.1525$

Similarly,

$m_3 = 0.84 \quad l_3 = -4.22$

$m_4 = 0.687 \quad l_4 = -4.244$

Finally,

$$y_1 = y(0.1) = y_0 + \frac{h}{6}(m_1 + 2m_2 + 2m_3 + m_4)$$

$$= 2 + \frac{0.1}{6}(1 + 2 \times 0.85 + 2 \times 0.84 + 0.687)$$

$$= 2.08445$$

And

$$z_1 = z(0.1) = z_0 + \frac{h}{6}(l_1 + 2l_2 + 2l_3 + l_4)$$

$$= 1 + \frac{0.1}{6}(-4 - 2 \times 4.1525 - 2 \times 4.22 - 4.244)$$

$$= -1.416483$$

SOLUTION OF ORDINARY DIFFERENTIAL EQUATION AS A BOUNDARY VALUE PROBLEM:

We have seen that we require 'm' conditions to be specified in order to solve 'm' order differential equation. And all the 'm' conditions were specified at one point, $x = x_0$ and, therefore we call this problem as an initial value problem. It is not always necessary to specify the condition at one point of the independent variable. They can be specified at different points in the interval (a, b) and, therefore, such problems are called the boundary value problems. A large number of problems fall into this category.

In this topic, we will only discuss two special methods to solve such ordinary differential equations they are:

a) Finite Difference Method
b) Shooting Method

FINITE DIFFERENCE METHOD

In finite difference method, the derivatives are replaced by their finite difference equivalents, thus converting the differential equation into a system of algebraic equations. Here, we will use the following "Central Difference Approximations".

$$y_i' = \frac{y_{i+1} - y_{i-1}}{2h} \quad \text{... (i)}$$

$$y_i'' = \frac{y_{i+1} - 2y_i - y_{i-1}}{h^2} \quad \text{... (ii)}$$

These are the second order equations and the accuracy of estimates can be improved by using higher order equations. The interval (a, b) is divided into 'n' sub-intervals, each of width 'h'. Then:

$$x_i = x_0 + ih = a + ih$$

$$y_i = y(x_i) = y(a + ih)$$

$$y_0 = y(a)$$

$$y_n = y(a + nh) = y(b)$$

Now, the difference equation is written for each of the interval points i = 1 to (n-1). If the differential equation is linear, it has (n-1) unknowns $y_1, y_2 \dots y_{n-1}$, which can be solved by any of the elimination methods. Smaller the size of 'h', more the sub-intervals and, therefore, more are the equations to be solved. However, a smaller 'h' yields better estimates.

Example:

Given the equation: y'' = e^{x^2} with y(0) = 0 and y(1) = 0. Estimate the values of y(x) at x = 0.25, 0.5 and 0.75

Solution:

We have: h = 0.25 and

y_0 = y(0) = 0
y_1 = y(0.25)
y_2 = y(0.5)
y_3 = y(0.75)
y_4 = y(1) = 0

We also have: $$y_i'' = \frac{y_{i+1} - 2y_i - y_{i-1}}{h^2} = e^{x^2}$$

For, i =1, x = 0.25: $$y_i'' = \frac{y_2 - 2y_1 - y_0}{0.25^2} = e^{0.25^2}$$

Then: $y_2 - 2y_1 = 0.0665$ ... (i)

For, i =2, x = 0.5: $$y_i'' = \frac{y_3 - 2y_2 - y_1}{0.25^2} = e^{0.5^2}$$

Then: $y_3 - 2y_2 + y_1 = 0.0803$ ... (ii)

For, i =3, x = 0.75: $$y_i'' = \frac{y_4 - 2y_3 - y_2}{0.25^2} = e^{0.75^2}$$

Then: $-2y_3 + y_2 = 0.0665$ ... (iii)

Solving, these three simultaneous algebraic equations, we will get:

y_1 = y(0.25) = -0.1175
y_2 = y(0.5) = -0.1684
y_3 = y(0.75) = -0.1391

SHOOTING METHOD:

In shooting method, the given boundary value problem is first converted into an equivalent initial value problem and then solved using any of the methods discussed in the previous chapter.

Let us consider the equation:

$y'' = f(x, y, y')$ with $y(x_0) = y(a) = A$ and $y(x_n) = y(b) = B$

By letting $y' = z$, we obtain the following set of two equations:

$$y' = z = f_1(x, y, z) \text{ and } z' = y'' = f_2(x, y, z)$$

In order to solve this set as an initial value problem, we need two conditions at $x = a$. We have one condition $y(a) = A$ and, therefore, require another condition for z at $x = a$.

Let us assume: $z(x_0) = z(a) = M_1$(Just a guess but represent the slope $y'(x)$ at $x =a$) Thus, the problem is reduced to a system of first order equation with the initial conditions as:

$$\begin{aligned} y' = z = f_1(x, y, z) &\qquad \text{with } y(a) = A \\ z' = y'' = f(x, y, z) &\qquad \text{with } z(a) = M_1 \ (= y'(a)) \end{aligned}$$

Now, using $z(a) = z(x_0) = M_1$, estimate $y(b) = y(x_n) = B_1$.

If $B_1 = B$ (Exact Solution), else
Suppose $z(a) = z(x_0) = M_2$(Another Guess) and estimate $y(b) = y(x_n) = B_2$

If $B_2 = B$ (Exact Solution), else
Estimate 'M' using the following relation:

$$\frac{M - M_1}{B - B_1} = \frac{M_2 - M_1}{B_2 - B_1}$$

Now, with $z(x_0) = z(a) = M_3$, we can again obtain the solution of $y(x)$.

Example:

Using shooting method, solve the equation: $y'' = 6x^2$, with $y(0) = 1$ and $y(1) = 2$ in the interval (0, 1) for $y(0.5)$ taking $h = 0.5$

Solution:

We have given:

For $x_0(=x(a)) = 0$, $y_0(=y(0)) = 1$,

For $x_n(=x(b)) = 1$, $y_0(=y(1)) = 2 = B$ and $h = 0.5$

Assume: $y' = z = f_1(x, y, z)$

Then: $y'' = z' = 6x^2 = f_2(x, y, z)$

Let: $z(0) = 1.5$ $(=M_1)$

So, using Euler's Method:

$y_1 = y(0.5) = y_0 + hf_1(x_0, y_0, z_0) = 1 + 0.5X1.5 = 1.75$ and

$z_1 = z(0.5) = z_0 + hf_2(x_0, y_0, z_0) = 1.5 + 0.5X6X0^2 = 1.5$

Therefore:

$y_2 = y(1) = y_1 + hf_1(x_1, y_1, z_1) = 1.75 + 0.5X1.5 = 2.50 = B_1$

Since, $B_1 > B$, assume another guess: $z(0) = 0.5$ $(=M_2)$

So, again using Euler's Method:

$y_1 = y(0.5) = y_0 + hf_1(x_0, y_0, z_0) = 1 + 0.5X0.5 = 1.25$ and

$z_1 = z(0.5) = z_0 + hf_2(x_0, y_0, z_0) = 0.5 + 0.5X6X0^2 = 0.5$

Therefore:

$y_2 = y(1) = y_1 + hf_1(x_1, y_1, z_1) = 1.25 + 0.5X0.5 = 1.50 = B_2$

Again, $B_2 < B$, so finding M using the following relation:

$$\frac{M - M_1}{B - B_1} = \frac{M_2 - M_1}{B_2 - B_1}$$

$$\frac{M - 1.5}{2 - 2.5} = \frac{0.5 - 1.5}{1.5 - 2.5}$$

$M = 1 = z(0)$

Finally, $y_1 = y(0.5) = y_0 + hf_1(x_0, y_0, z_0) = 1 + 0.5X1 = 1.5$

Assignment 7

Full Marks: 40
Pass Marks: 20

24. Given the differential equation $y'' - xy' - y = 0$ with initial conditions $y(0) = 1$ and $y'(0) = 0$, use Taylor's series method to determine the value of $y(0.1)$. [5]

25. Use Euler's method to solve the differential equation $y' = - y'$ with the initial condition $y(0) = 1$ with step size 0.01 and find the value of $y(0.04)$. [5]

26. Solve the differential equation $10y'' + (y')^2 + 6x = 0$ with $y(0) = 1$ and $y'(0) = 0$ by Heun's method to estimate $y(0.2)$ using $h = 0.1$ [7]

27. Use 4th order Runge Kutta method to estimate $y(0.5)$ of the differential equation $y' = y + \_/y$ with $y(0) = 1$. Take $h = 0.25$. [5]

28. Solve the following simultaneous first order differential equations to estimate $y(0.2)$ and $z(0.2)$ using any method of your choice.

    $y' = z$ with $y(0) = 0$ and $z' = yz + x^2 + 1$ with $z(0) = 0$ [8]

29. A body of mass 2 Kg is attached to a spring with a spring constant of 10. The differential equation governing the displacement of the body 'y' and time 't' is given by: $y'' + 2y' + 5y = 0$. Find the displacement 'y' at time t = 0.5, 1, 1.5 using finite difference method. Given that $y(0) = 2$, $y'(0) = -4$ and $y(2) = 4$

    (Note: Make Correction if needed) [10]

8

CHAPTER

Numerical Solution of Partial Differential Equations

Contents:

- Introduction
- Finite Difference Approximation
- Solution of Elliptic Equations
  - Laplace's Equation
  - Poisson's Equation
- Solution Parabolic Equation
- Solution of Hyperbolic Equation
  - Finite Difference Method
  - Shooting Method

INTRODUCTION: PARTIAL DIFFERENTIAL EQUATION

Physical phenomena in applied science and engineering when formulated into mathematical models fall into a category of systems known as partial differential equations that involves more than one independent variable which determine the behaviour of the dependent variable as described by their partial derivative contained in the equation.

Examples: Heat flow in a rectangular plate

The model for heat flow in a rectangular plate that is heated is given by:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x, y)$$

Where u(x, y) denotes the temperature at point (x, y) and f(x, y) is the heat source. Here, the rate of change of a variable is expressed as a function of variables and parameters. Although most of the differential equations may be solved analytically in their simplest form, analytical techniques fail when the models are modified to take into account the effect of other conditions of real life situations. In all such cases, numerical approximation of the solution may be considered as a possible approach.

If we represent the dependent variable as 'f' and the two independent variables as 'x' and 'y', then we will have three possible second order partial derivatives:

$$\frac{\partial^2 f}{\partial x^2}, \frac{\partial^2 f}{\partial x\, \partial y} \text{ and } \frac{\partial^2 f}{\partial y^2}$$

Then we can write a second order equation involving two independent variables in general form as:

$$a\frac{\partial^2 f}{\partial x^2} + b\,\frac{\partial^2 f}{\partial x\,\partial y} + c\frac{\partial^2 f}{\partial y^2} = F(x, y, f, \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}) \quad \ldots \text{(i)}$$

Where the coefficients a, b and c may be constants or function of 'x' and 'y'. Depending on the values of these coefficients equation (i) can be classified into one of the three types of equations, namely:

a) Elliptic Equation: If $b^2 - 4ac < 0$
b) Parabolic Equation: If $b^2 - 4ac = 0$
c) Hyperbolic Equation: If $b^2 - 4ac > 0$

FINITE DIFFERENCE APPROXIMATION

In finite difference method, we replace derivatives that occur in the partial differential equation by their finite difference equivalents. We then write the difference equation corresponding to each 'grid point' (where derivative is required) using functional values at the surrounding grid points. Solving these equations simultaneously gives the values for the function at grid point.

Consider a two-dimensional solution domain as shown in the figure below. The domain is split into regular rectangular grids of width 'h' and height 'k'. The pivotal values at the points of intersection (grid points or node) are denoted by f_{ij} which is a function of the two-space variable 'x' and 'y'.

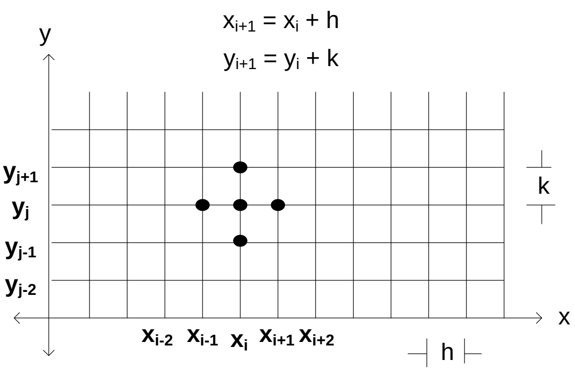


Fig: Two Dimensional Finite Difference Grid

If the function f(x) has continuous fourth derivative, then its first and second derivatives are given by the following central difference approximations.

$$f'(x_i) = \frac{f(x_i+h)-f(x_i-h)}{2h} = \frac{f_{i+1}-f_{i-1}}{2h}$$

$$f''(x_i) = \frac{f(x_i+h)-2f(x_i)+f(x_i-h)}{h^2} = \frac{f_{i+1}-2f_i+f_{i-1}}{h^2}$$

When 'f' is a function of two variables 'x' and 'y', the partial derivatives of 'f' with respect to 'x' (or 'y') are the ordinary derivatives of 'f' with respect to 'x' (or 'y') when 'y' (or 'x') does not change. So, we can use above equation to determine derivatives with respect to 'x' and in the 'y' direction. Thus, we have:

$$\frac{\partial f(x_i,y_j)}{\partial x} = f_x(x_i,y_j) = \frac{f(x_{i+1},y_j)-f(x_{i-1},y_j)}{2h}$$

$$\frac{\partial f(x_i,y_j)}{\partial y} = f_y(x_i, y_j) = \frac{f(x_i,y_{j+1}) - f(x_i,y_{j-1})}{2k}$$

$$\frac{\partial^2 f(x_i,y_j)}{\partial x^2} = f_{xx}(x_i, y_j) = \frac{f(x_{i+1},y_j) - 2f(x_i,y_j) + f(x_{i-1},y_j)}{h^2}$$

$$\frac{\partial^2 f(x_i,y_j)}{\partial y^2} = f_{yy}(x_i, y_j) = \frac{f(x_i,y_{j+1}) - 2f(x_i,y_j) + f(x_i,y_{j-1})}{k^2}$$

$$\frac{\partial^2 f(x_i,y_j)}{\partial x\, \partial y} = f_{xy}(x_i, y_j) = \frac{f(x_{i+1},y_{j+1}) - f(x_{i+1},y_{j-1}) - f(x_{i-1},y_{j+1}) - f(x_{i-1},y_{j-1})}{4hk}$$

SOLUTION OF ELLIPTIC EQUATIONS

Elliptic equations are governed by conditions on the boundary of closed domain. We consider here the two most commonly encountered elliptic equations, namely: Laplace's Equation and Poisson's Equation.

Laplace Equation:

The general second order partial differential equation (i), when a = 1, b = 0, c = 1 and $F(x, y, f_x, f_y) = 0$ becomes:

$$\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = \nabla^2 f = 0 \ldots \text{(ii)} \quad \left\{\therefore\ \nabla^2 = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)\right\}$$

The operator ∇^2 is called the Laplacian Operator and equation (ii) is called Laplace's Equation. To solve the Laplace's Equation on a region in the xy-plane, we subdivide the region as shown in the figure below. Consider the portion of the region near (x_i, y_i). We have to approximate: $\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = \nabla^2 f = 0$.

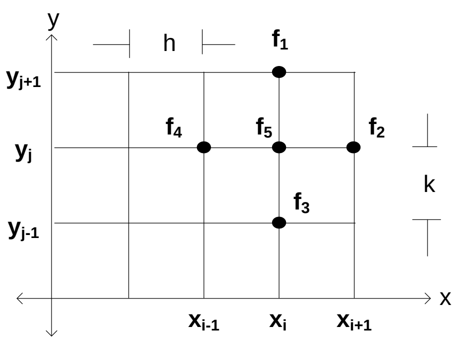


Fig: Solution of Laplace's Equation

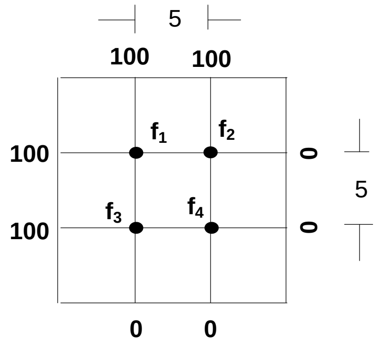
Replacing the second order derivatives by their finite difference equivalents at point (x_i, y_i), we obtain.

$$\frac{f_2-2f_5+f_4}{h^2}+\frac{f_1-2f_5+f_3}{k^2}=0$$

i.e. $f_2 + f_4 + f_1 + f_3 - 4f_5 = 0$ (for h = k)

Example:

Consider a steel plate of size 15X15 cm. If two of the sides are held at 100°C and the other two sides are held at 0°C. What is the steady state value of temperature at interior knots, assuming a grid size of 5X5 cm.



Solution:

At point 1:

$f_2 + f_3 + 100 + 100 - 4f_1 = 0 \gg f_2 + f_3 - 4f_1 + 200 = 0$ ... (i)

At point 2:

$f_1 + f_4 + 100 + 0 - 4f_2 = 0 \gg f_1 - 4f_2 + f_4 + 100 = 0$ ... (ii)

At point 3:

$f_1 + f_4 + 100 + 0 - 4f_3 = 0 \gg f_1 - 4f_3 + f_4 + 100 = 0$ ... (iii)

At point 4:

$f_3 + 0 + 0 + f_2 - 4f_4 = 0 \gg f_2 + f_3 - 4f_4 = 0 \gg f_2 = 4f_4 - f_3$ ... (iv)

Putting the values of of f_2 from equation (iv) into equations (i) and (ii), we get:

$f_2 + f_3 - 4f_1 + 200 = 0 \gg 4f_4 - f_3 + f_3 - 4f_1 + 200 = 0 \gg f_1 - f_4 = 50$ ... (vi)
$f_1 - 4f_2 + f_4 + 100 = 0 \gg f_1 - 4(4f_4 - f_3) + f_4 + 100 = 0 \gg f_1 - 4f_3 - 15f_4 = -100$ ... (vii)
$f_1 - 4f_3 + f_4 + 100 = 0$ ... (viii)

Now, solving equations (vi), (vii) and (viii) we will get: $f_1 = 75$, $f_3 = 50$ and $f_4 = 25$
Putting f_3 and f_4 in equation (iv), we get: $f_2 = 50$.

Poisson's Equation

The general second order partial differential equation, when a = 1, b = 0, c = 1 and F(x, y, f_x, f_y) = g(x, y) becomes:

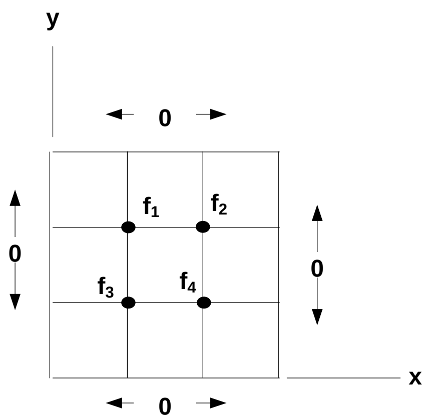
$$\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = \nabla^2 f = g(x, y) \ \ldots \text{(iii)} \quad \left\{\therefore\ \nabla^2 = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)\right\}$$

Equation (ii) is called Poisson's Equation. Now, Laplace's equation may be modified to solve Poisson's Equation. Then the finite difference formula for soving Poisson's Equation takes the form:

i.e. $f_2 + f_4 + f_1 + f_3 - 4f_5 = h^2 g_{ij}$ (for h = k)

Example:

Solve the Poisson's Equation $\nabla^2 f = 2x^2y^2$ over the square domain $0 \le x \le 3$ and $0 \le y \le 3$ with f = 0 on the boundary and h = 1.



Solution:

At point 1: $0 + 0 + f_2 + f_3 - 4f_1 = 2(1)^2(2)^2 \gg f_2 + f_3 - 4f_1 = 8$ ... (i)

At point 2: $0 + 0 + f_1 + f_4 - 4f_2 = 2(2)^2(2)^2 \gg f_1 - 4f_2 + f_4 = 32$ ... (ii)

At point 3: $0 + 0 + f_1 + f_4 - 4f_3 = 2(1)^2(1)^2 \gg f_1 - 4f_3 + f_4 = 2$ ... (iii)

At point 4: $0 + 0 + f_2 + f_3 - 4f_4 = 2(2)^2(1)^2 \gg f_2 + f_3 - 4f_4 = 8$ ... (iv)

On solving these simultaneous equations by elimination method, we will get the answers.

$f_1 = -22/4$ $f_2 = -43/4$

$f_3 = -13/4$ $f_4 = -22/4$

SOLUTION OF PARABOLIC EQUATIONS

Elliptic equations describe the problems that are time independent. Such problems are known as steady state problems. But we come across problems that are not steady state. This means that the function is dependent on both space and time. Parabolic equations, for which $b^2 - 4ac = 0$ describe the problems that depend on space and time variables. The popular case for parabolic type equation is the study of heat flow in one dimensional direction in an insulated rod. Such problems are governed by both boundary and initial conditions.

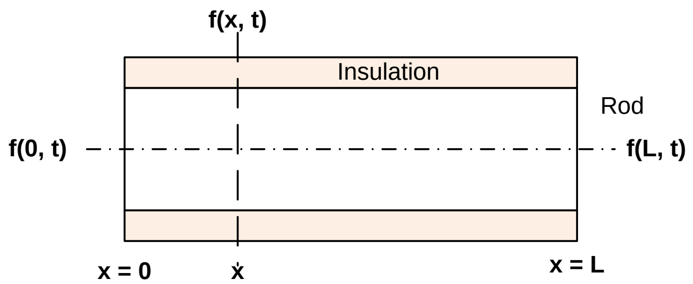


Fig: Heat flow in a rod

If 'f' represent the temperature at any point in rod whose distance from the left end is 'x = 0' to 'x = L'. Heat is flowing from left to right under the influence of temperature gradient. The temperature f(x, t) in the rod at the position 'x' and time 't', is governed by the heat equation:

$$K_1 \frac{\partial^2 y}{\partial x^2} = K_2 K_3 \frac{\partial f}{\partial x} \quad \dots \text{(i)}$$

Where, K_1 = Coefficient of thermal conductivity
K_2 = Specific Heat
K_3 = Density of the material

Equation (i) can be written as: $K.f_{xx}(x, t) = f_t(x, t)$ … (ii), where $K = K_1/(K_2K_3)$

The initial condition will be the initial temperatures at all points along the rod i.e. $f(x, 0) = f(x)$ for $0 \leq x \leq L$.

The boundary conditions f(0, t) and f(L, t) describe the temperature at each end of the rod as functions of time. If they are held constant, then:

$f(0, t) = C_1$, for $0 \leq t \leq \infty$ and $f(L, t) = C_2$, for $0 \leq t \leq \infty$

We can solve the heat equation given by equation (ii) using finite difference formula as below:

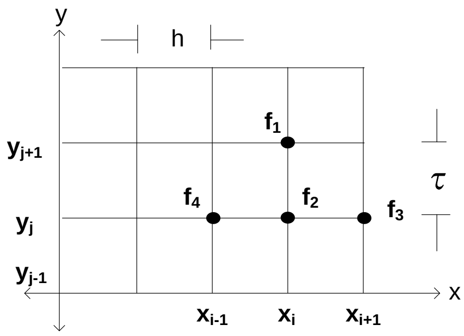


Fig: Solution of Parabolic Equation

As: $K.f_{xx}(x, t) = f_t(x, t)$

i.e. $K\frac{\partial^2 f}{\partial x^2} = \frac{\partial f}{\partial t}$

So, at point f_2:

i.e. $K\left(\frac{f_3 - 2f_2 + f_4}{h^2}\right) = \frac{f_1 - f_2}{\tau}$

i.e. $\frac{\tau K}{h^2}(f_3 + f_4) - \frac{2\tau K}{h^2} f_2 = f_1 - f_2$

i.e. $\left[1 - \frac{2\tau K}{h^2}\right] f_2 = f_1 - \frac{\tau K}{h^2}(f_3 + f_4)$

For, $\frac{2\tau K}{h^2} = 1$, $0 = f_1 - \frac{1}{2}(f_3 + f_4)$

i.e. $f_1 = \frac{1}{2}(f_3 + f_4)$

Example:

Solve the equation: $2f_{xx}(x, t) = f_t(x, t)$ for $0 < t < 1.5$ and $0 < x < 4$.

Given, | |
---|---|---
Given, | **$f(x, 0) = 50(4 - x)$** | **for $0 < x < 4$;**
 | **$f(0, t) = 0$** | **for $0 < t < 1.5$;**
 | **$f(4, t) = 0$** | **for $0 < t < 1.5$.**

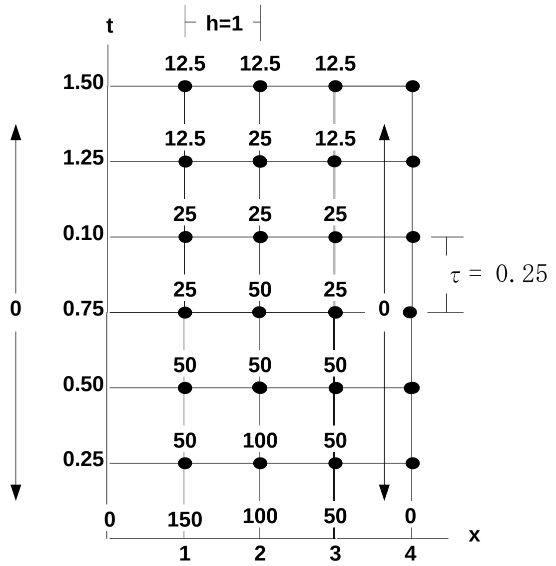
Take h = 1.

Solution:

Here, we have: K = 2 and h = 1

So, for: $\frac{2\tau K}{h^2} = 1$, $\tau = 1/2K = 1/(2X2) = 0.25$

Now constructing the grid with h = 1 and τ = 0.25



t\x	0.0	1.0	2.0	3.0	4.0
0.00	0	150	100	50	0
0.25	0	50	100	50	0
0.50	0	50	50	50	0
0.75	0	25	50	25	0
1.00	0	25	25	25	0
1.25	0	12.5	25	12.5	0
1.50	0	12.5	12.5	12.5	0

SOLUTION OF HYPERBOLIC EQUATIONS

Hyperbolic equations model the vibration of structures such as buildings, beams and machines. We consider here the case of a vibrating string that is fixed at both the ends as shown in figure below:

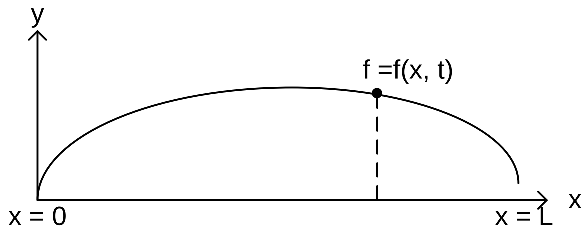


Fig: Displacement of vibrating spring

The lateral displacement of string 'f' varies with time 't' and distance 'x' along the string. The displacement f(x, t) is governed by the wave equation:

$$T\frac{\partial^2 f}{\partial x^2} = \rho\frac{\partial^2 f}{\partial t^2} \ldots \text{(i)}$$

Where, T is the tension in the string and ρ is the mass per unit length. Hyperbolic problems are also governed by both boundary and initial conditions, if time is one of the independent variables. Two boundary conditions for the vibrating string problem under consideration are:

$f(0, t) = 0$ for $0 \leq t \leq b$ and $f(L, t) = 0$ for $0 \leq t \leq b$

Two initial conditions are:

$f(x, 0) = f(x)$ for $0 \leq x \leq a$ and $f_t(x, 0) = g(x)$ for $0 \leq x \leq a$

Now, we can solve the heat equation given by equation (ii) using finite difference formula as below:

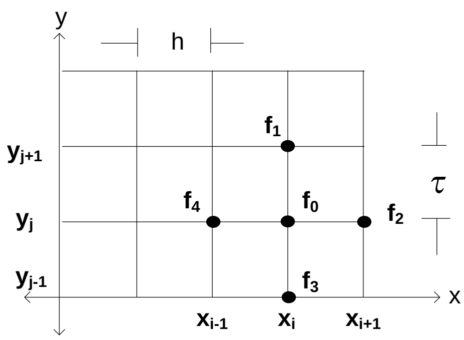


Fig: Solution of Hyperbolic Equation

As the wave equation is given by: $T\frac{\partial^2 f}{\partial x^2} = \rho\frac{\partial^2 f}{\partial t^2}$

So, at point f_0

i.e. $T\left(\frac{f_2-2f_0+f_4}{h^2}\right) = \rho\left(\frac{f_1-2f_0+f_3}{\tau^2}\right)$

i.e. $\frac{T\tau^2 K}{\rho h^2}(f_2 + f_4) - 2\frac{T\tau^2}{\rho h^2}f_0 = f_1 + f_3 - 2f_0$

i.e. $\left[1 - \frac{T\tau^2}{\rho h^2}\right]f_0 = f_1 + f_3 - \frac{T\tau^2}{\rho h^2}(f_2 + f_4)$

For, $\frac{T\tau^2}{\rho h^2} = 1$, $0 = f_1 + f_3 - (f_2 + f_4)$

i.e. $f_1 = (f_2 + f_4 - f_3)$

This can be also written in coordinate form as:

$f_{i,j+1} = \left(f_{i+1,j} + f_{i-1,j} - f_{i,j-1}\right)$ ...(ii)

Starting Values

Here, we need two row of starting values, corresponding to j = 1 and j = 2 in order to compute the values at the third row. First row is obtained using the condition: f(x, 0) = f(x). The second row can be obtained using the second initial condition as given by: $f_t(x, 0) = g(x)$

As we know: $f_t(x,0) = \frac{f_{i,0+1} - f_{i,0-1}}{2\tau} = g_i$ (for t = 0 only). By this, substituting the value of $f_{i,-1}$ in equation (ii) for j = 0, we will get at t = t_1:

$f_t(x,1) = \frac{1}{2}\left(f_{i+1,0} - f_{i-1,0}\right) + \tau g_i$

As for many cases: $g_i = 0$, therefore: $f_{i,1} = \frac{1}{2}\left(f_{i+1,0} - f_{i-1,0}\right)$

Example:

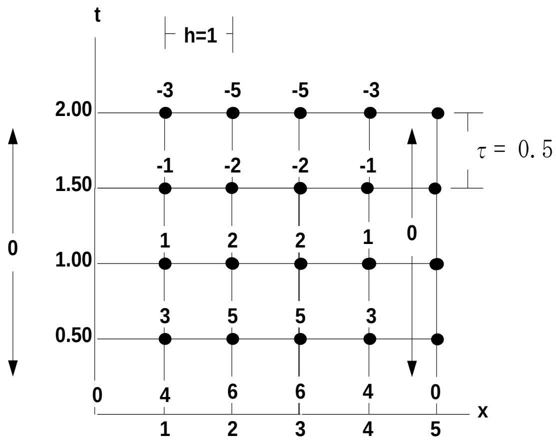
Solve the wave equation: $4f_{xx} = f_{tt}$ for $0 \le x \le 5$, take h = 1.
Given: f(0, t) = 0 and f(5, t) = 0
f(x, 0) = x(5 – x)
$f_t(x, 0) = 0 = g_i$

Solution:

Here we have, h = 1, T = 4 and ρ = 1

So, for: $\frac{T\tau^2}{\rho h^2} = 1, \tau^2 = \frac{1}{4} = 0.5$

Hence, constructing the table, we will have:



t\x	0.0	1.0	2.0	3.0	4.0	5.0
0.00	0	4	6	6	4	0
0.50	0	3	5	5	3	0
1.00	0	1	2	2	1	0
1.50	0	-1	-2	-2	-1	0
2.00	0	-3	-5	-5	-3	0

Assignment 8

Full Marks: 30
Pass Marks: 15
Grace Mark: 4

30. Solve the steady-state temperature in a rectangular plate of 8cm x 10 cm, if one 10 cm side is held at 50°C, and the other 10 cm side is held at 30°C and the other two sides are held at 10°C. Assume square grids of size 2cm x 2 cm. [5]

    (Ref: Elliptic Equation – Solution of Laplace's Equation)

31. Solve the equation: $\nabla^2 f = F(x, y)$ with F(x, y) = xy and f = 0 on boundary. The domain is a square with corners at (0, 0) and (4, 4). Use h = 1. [5]

    (Ref: Elliptic Equation – Solution of Poisson's Equation)

32. Estimate the values at grid points of the following equation using Bender Schmidt Recurrence Equation. Assume h = 1.

    a) $0.5f_{xx} - f_t = 0$ [8]
       Given: f(0, t) = -5, f(5, t) = 5
       f(x, 0) = -5, for x: [0, 2.5]
       = 5, for x: [2.5, 5]

    (Ref: Parabolic Equation)

33. Solve the following wave equation using finite difference method.

    $4f_{xx} = f_{tt}$
    Given:
    f(0, t) = 0 and f(1, t) = 0
    f(x, 0) = $\sin(\pi x) + \sin(2\pi x)$ [8]
    $f_t(x, 0) = 0$

    (Ref: Hyperbolic Equation)

9

CHAPTER

Numerical Solution of Integral Equations

Contents:

INTRODUCTION:

Any equation in which the unknown function appears under the integral sign is known as an integral equation. General integral equation of first and second kind can be represented by:

$$\int_a^b K(x,t)f(t)dt = \emptyset(x) \ldots \text{(i)}$$

$$\gamma \int_a^b K(x,t)f(t)dt = f(x) + \emptyset(x) \;\; \ldots. \text{(ii)}$$

Here, the limits of integrals are constants; therefore it is called **Fredholm Integral Equation** of first and second kinds respectively. In each case, f(x) is unknown and occurs to the first degree but $\emptyset(x)$ and the Kernel K(x, t) are known functions.

If the constant 'b' in above equations is replaced by 'x', the variable of integration, the equations are called **Volterra Integral Equation** given by:

$$\gamma \int_a^x K(x,t)f(t)dt = f(x) + \emptyset(x) \;\; \ldots. \text{(iii)}$$

If $\emptyset(x) = 0$ in equation (ii), then the equation is called homogeneous, otherwise non-homogeneous. For non-homogeneous equations 'γ' is a numerical parameter whereas for homogeneous equations, it is an eigen-value parameter.

If the kernel: K(x, t) is bounded and continuous, then the integral equation is said to be non-singular. If the range of integration is infinite, or if the kernel violates the above conditions, then the equation is said to be singular.

To solve an integral equation of any type is to find the unknown function satisfying that equation. In this chapter we deal with the Fredholm Integral Equation, particularly those of the second kind since, it occurs quite frequently in practice.

METHOD OF DEGENERATE KERNELS

This method is important in the theory of integral equations but does not seem to be much useful in the numerical work, since the kernel is unlikely to have the simple form in practical problems. In general, however, it is possible to take a partial sum of Taylor's series for the kernel.

Let us consider the integral equation of type:

$$f(x) - \int_a^b K(x,t)f(t)dt = \emptyset(x) \ ...(i)$$

A kernel K(x, t) is said to be degenerate if it can be expressed in the form:

$$K(x,t) = \sum_{i=1}^{n} u_i(x)v_i(t) \ ...(ii)$$

Substituting equation (ii) in equation (i), we obtain:

$$f(x) - \int_a^b \{\sum_{i=1}^{n} u_i(x)v_i(t)\, f(t)dt\} = \emptyset(x)$$

$$f(x) - \sum_{i=1}^{n} \int_a^b u_i(x)v_i(t)f(t)dt = \emptyset(x) \ \ ... (iii)$$

Let us suppose: $\int_a^b v_i(t)f(t)dt = A_i$

Then equation (iii) becomes:

$$f(x) = \sum_{i=1}^{n} \int_a^b A_i u_i(x) + \emptyset(x) \ ... (iv)$$

Hence, the coefficients A_i are determined from equations (iv) and (iii), which will give a system of 'n' equations with 'n' unknowns A_1, A_2 … A_n. When A_i are determined, equation (iv) gives f(x).

Example:

Solve the integral equation: $\mathbf{f(x) = \frac{1}{2}(e^{-x} + 3x - 1) + \int_0^1 \left(e^{-xt^2} - 1\right) x. f(t)dt.}$

Solution:

Comparing the given integral equation with the original equation, we get:

K(x, t) = $\left(e^{-xt^2} - 1\right)x = \left(1 - xt^2 + \frac{x^2t^4}{2} + \cdots - 1\right)x = -x^2t^2 + \frac{1}{2}x^3t^4$+(neglecting the other terms of the Taylor's Series)

i.e. $K(x, t) = -x^2t^2 + \frac{1}{2}x^3t^4$ ... (i)

Hence, the given integral equation becomes:

$$f(x) = \frac{1}{2}(e^{-x} + 3x - 1) + \int_0^1 \left(-x^2t^2 + \frac{1}{2}x^3t^4\right). f(t)dt$$

$$= \frac{1}{2}(e^{-x} + 3x - 1) - x^2 \int_0^1 t^2. f(t)dt + \frac{1}{2}x^3 \int_0^1 t^4. f(t)dt$$

$$= \frac{1}{2}(e^{-x} + 3x - 1) - K_1x^2 + \frac{1}{2}K_2x^3 \text{ ... (ii)}$$

Now, generating simultaneous equations:

$$K_1 = \int_0^1 t^2. f(t)dt = \int_0^1 t^2. \left\{\frac{1}{2}(e^{-t} + 3t - 1) - K_1t^2 + \frac{1}{2}K_2t^3\right\} dt$$

$$\frac{6K_1}{5} - \frac{K_2}{12} = -\frac{5}{2e} + \frac{29}{24} \text{ ... (iii)} \qquad \therefore \int_0^1 t^2e^{-t}dt = 2 - \frac{5}{e}$$

Again:

$$K_2 = \int_0^1 t^4. f(t)dt = \int_0^1 t^4. \left\{\frac{1}{2}(e^{-t} + 3t - 1) - K_1t^2 + \frac{1}{2}K_2t^3\right\} dt$$

$$\frac{K_1}{7} - \frac{15}{16}K_2 = -\frac{65}{2e} + \frac{243}{20} \text{ ... (iv)}$$

Solving equations (iii) and (iv) we get: $K_1 = 0.2522$ and $K_2 = 0.1685$

Hence, the solution of the given integral equation is:

$$f(x) = \frac{1}{2}(e^{-x} + 3x - 1) - 0.2522x^2 + \frac{1}{2}0.1685x^3$$

METHOD OF USING GENERALIZED QUADRATURE

Let us consider the integral equation of type:

$$f(x) - \int_a^b K(x,t)f(t)dt = \emptyset(x) \ldots \text{(i)}$$

Since, a definite integral can be closely approximated by a Quadrature Formula and obviously, different types of Quadrature Formulas can be employed (e.g. Trapezoidal, Simpson etc). We approximate the integral term of equation (i) by a formula of the form:

$$\int_a^b F(x)dx = \sum_{m=1}^{n} A_m F(x_m) \ldots \text{(ii)}$$

Where A_m and x_m are the weights and abscissa, respectively. Consequently eq^n (ii) can be written as:

$$f(x) - \sum_{m=1}^{n} A_m K(x,t_m)F(t_m) = \emptyset(x) \ldots \text{(iii)}$$

Where t_1, t_2 … , t_n are the points in which the interval (a, b) is subdivided. Further equation (iii) must hold for all values of 'x' in the interval (a, b); in particular, it must hold for $x = t_1$, $x = t_2$, … , $x = t_n$. Hence we obtain:

$$f(t_i) - \sum_{m=1}^{n} A_m K(t_i,t_m)F(t_m) = \emptyset(t_i) \text{ for } i = 1, 2, \ldots, n \quad \ldots \text{(iv)}$$

Which is a system of 'n' linear equations in the 'n' unknowns $f(t_1)$, $f(t_2)$ … $f(t_n)$. When the $f(t_i)$ are determined, equation (iii) gives an approximation for f(x).

Example:

Solve: $f(x) - \int_0^1 (x+t)f(t)dt = \frac{3}{2}x - \frac{5}{6}$

Solution:

Here, K(x, t) = (x + t)

The integral term can be directly approximated by the Quadrature Formula. For the numerical solution, we divide the range [0, 1] into two equal subintervals so that h = 0.5. Let us apply composite trapezoidal rule to approximate the integral term, we obtain:

$$f(x) - \frac{h}{2}\{f_0 + 2f_1 + f_3\} = \frac{3}{2}x - \frac{5}{6}$$

$$f(x) - \frac{1}{4}\left\{xf_0 + 2\left(x + \frac{1}{2}\right)f_1 + (x+1)f_3\right\} = \frac{3}{2}x - \frac{5}{6} \text{ ... (i)}$$

Setting $x = t_i$ ($t_0 = 0$, $t_1 = 0.5$, $t_1 = 1$) and $f(t_i) = f_{i,}$ we will have:

a) For $x = t_0 = 0$

$$f(t_i) - \frac{1}{4}\left\{t_i f_0 + 2\left(t_i + \frac{1}{2}\right)f_1 + (t_i+1)f_3\right\} = \frac{3}{2}t_i - \frac{5}{6}$$

$$f(t_0) - \frac{1}{4}\left\{t_0 f_0 + 2\left(t_0 + \frac{1}{2}\right)f_1 + (t_0+1)f_3\right\} = \frac{3}{2}t_0 - \frac{5}{6}$$

$$f_0 - \frac{1}{4}\left\{0 \times f_0 + 2\left(0 + \frac{1}{2}\right)f_1 + (0+1)f_3\right\} = \frac{3}{2} \times 0 - \frac{5}{6}$$

$$12f_0 - 3f_1 - 3f_2 = -10 \text{ ... (a)}$$

b) Similarly, for $x = t_1 = 0.5$

$$-3f_0 + 12f_1 - 9f_2 = 2 \text{ ... (b)}$$

c) And for $x = t_2 = 1$

$$-3f_0 - 9f_1 + 6f_2 = 8 \text{ ... (c)}$$

On solving, we get: $f_0 = -1/2$, $f_1 = -5/6$ and $f_2 = ½$

Now, using these values in equation (i), we get the solution as:

$$f(x) - \frac{1}{4}\left\{x\left(-\frac{1}{2}\right) + 2\left(x + \frac{1}{2}\right)\left(-\frac{5}{6}\right) + (x+1)\left(\frac{1}{2}\right)\right\} = \frac{3}{2}x - \frac{5}{6}$$

i.e. $f(x) = \frac{1}{4}\left\{x\left(-\frac{1}{2}\right) - 2\left(x + \frac{1}{2}\right)\left(-\frac{5}{6}\right) - (x+1)\left(\frac{1}{2}\right)\right\} + \frac{3}{2}x - \frac{5}{6} = 1.6x - 0.75$

Note: Since, the integrand is a second degree polynomial in‘t’ so, if we implement the composite Simpson’s 1/3 rule, it will give the exact solution i.e. f(x) = x - 1.

CHEBYSHEV SERIES METHOD & CUBIC SPLINE METHOD

Chebyshev Series Method: The Fredholm Integral Equation can also be manipulated by Chebyshev Series Method, which is somewhat laborious. For example the integral equation of type:

$$y(x) + \int_{-1}^{1} K(x,s)y(s)ds = 1; \qquad \text{where } K(x,s) = \frac{1}{\pi}\frac{d}{d^2(x-s)^2}$$

- can be solved. However, this method can give better accuracy than the trapezoidal method but for the smaller values of d, this method is unsuitable. Because, for example with 32 subdivisions and d = 0.0001, the value obtained for x = 0 is 0. 04782 compared to the true value of 0.50015 that means: somewhat large fluctuation in the results.

Cubic Spline Method: In contrast with the previous methods, cubic spline method can be applied when the values of d are small. However, for large value of d, the convergence would rather slowly to give the exact solution.

The spline method for the numerical solution of Fredholm Integral Equations is potentially useful. Its application to more complicated problems will have to be examined together with estimation to error in the method. It seems probable that the condition of continuity of the kernel may be relaxed, and the advantage to be achieved by using unequal intervals may also be explored. Finally the solution obtained by the spline method can be improved upon by regarding it as the initial iterate in an iterative method of higher order convergence.

Assignment 9

Full Marks: 30
Pass Marks: 15
Grace Marks: 4

1. Solve the following integral equations with degenerate kernels:
   a) $f(x) - \gamma \int_0^{\frac{\pi}{2}} \sin(x) \cos(t) f(t)dt = \sin(x)$ [10]

2. Solve the integral equation given in problem 1 by using general Quadrature method for using:

   a) Composite Trapezoidal Rule [8]
   b) Composite Simpson's Rule [8]

   (Note: Make necessary assumptions)

1
CHAPTER

Programming Exercises

Contents:

- Horner's Method
- Bisection Method
- Regular Falsi Method
- Newton Raphson Method
- Secant Method
- Fixed Point Iteration
- Linear Interpolation
- Lagrange Interpolation

LAB 1

HORNER'S METHOD OF FINDING POLYNOMIAL AT GIVE VALUE

```
#include<stdio.h>
#include<conio.h>
#include<math.h>

#define G(x) (x)*(x)*(x)-4*(x)*(x)+(x)+6

void main()
{
	int n,i;
	float x, a[10],p;
	printf("------------------------------------------------------------\n");
	printf("HORNER'S METHOD OF FINDING THE POLYNOMIAL VALUE AT GIVEN POINT\n");
	printf("------------------------------------------------------------\n\n");
	printf("\nINPUT DEGREE OF POLYNOMIAL:\t");
	scanf("%d", &n);
	printf("\nINPUT POLYNOMIAL COEFFICENTS WITH ORDER  a[%d] to a[0]:\t", n);
	for(i=n; i>=0; i--)
	scanf("%f", &a[i]);
	printf("\nTHE POLYNOMIAL IS:\t");
	for(i=n; i>=0; i--)
	{
		printf("%f(x^%d) ",a[i],i);
	}
	printf("\n\nINPUT VALUE OF 'X' (EVALUATION POINT):\t");
	scanf("%f", &x);
	p=a[n];
	for(i=n-1; i>=0; i--)
	{
		p=p*x+a[i];
	}
	printf("\nTHE POLYNOMIAL VALUE: f(x) = %f at x = %f\t", p,x);
}
```

OUTPUT

```
--------------------------------------------------------------
HORNER'S METHOD OF FINDING THE POLYNOMIAL VALUE AT GIVEN POINT
--------------------------------------------------------------

INPUT DEGREE OF POLYNOMIAL:     3

INPUT POLYNOMIAL COEFFICENTS WITH ORDER  a[3] to a[0]:  1 -2 3 4

THE POLYNOMIAL IS:      1.000000(x^3) -2.000000(x^2) 3.000000(x^1) 4.000000(x^0)

INPUT VALUE OF 'X' (EVALUATION POINT):  1.54

THE POLYNOMIAL VALUE: f(x) = 7.529064 at x = 1.540000
```

LAB 2

SOLUTION OF A GIVEN EQUATION BY BISECTION METHOD

```
#include<stdio.h>
#include<conio.h>
#include<math.h>

#define EPS 0.000001
#define F(x) log(x)-1

void main()
{
	float xn,xp,xm,a,b,c,d;
	int count;

  begin:
	printf("Enter Bracketing Values:\t");
	scanf("%f%f",&xn,&xp);

	a=F(xn);
	b=F(xp);

	if((a*b)>0)
	{
		printf("Entered Values Didnot Bracket The ROOT\n");
		goto begin;
	}
	else
	{
		count=1;
		 printf("\ncount\t xn\t\t xp\t\t xm\t\t F(xm)\n");

		iteration:

			xm=((xn+xp)/2);

			c=F(xm);

			printf("\n%d\t %f\t %f\t %f\t %f\t", count, xn, xp, xm, F(xm));

			if(c==0)
			{
				printf("\n\nROOT: %f\t", xm);
				printf("\nITERATION: %d\t", count);
			}
			else
			{
				if(c<0)
					xn=xm;
				else
					xp=xm;
```

```
                d=fabs((xn-xp)/xn);
                if(d<EPS)
                {
                        xm=((xn+xp)/2);
                        printf("\n\nROOT: %f\t", xm);
                        printf("\nITERATION: %d\t", count);
                }
                else
                {
                        count=count++;
                        goto iteration;
                }
            }
    }
}
```

OUTPUT

```
----------------------
TITLE: BISECTION METHOD
----------------------

--------------------------------------------------------------------
OBJECTIVE: TO FIND THE SOLUTION OF GIVEN EQUATION BY BISECTION METHOD
--------------------------------------------------------------------

Enter Bracketing Values:        1 2

Entered Values Didnot Bracket The ROOT
Enter Bracketing Values:        1 3

count    xn              xp              xm              F(xm)

1        1.000000        3.000000        2.000000        -0.306853
2        2.000000        3.000000        2.500000        -0.083709
3        2.500000        3.000000        2.750000        0.011601
4        2.500000        2.750000        2.625000        -0.034919
5        2.625000        2.750000        2.687500        -0.011389
6        2.687500        2.750000        2.718750        0.000172
7        2.687500        2.718750        2.703125        -0.005591
8        2.703125        2.718750        2.710938        -0.002705
9        2.710938        2.718750        2.714844        -0.001266
10       2.714844        2.718750        2.716797        -0.000546
11       2.716797        2.718750        2.717773        -0.000187
12       2.717773        2.718750        2.718262        -0.000007
13       2.718262        2.718750        2.718506        0.000082
14       2.718262        2.718506        2.718384        0.000038
15       2.718262        2.718384        2.718323        0.000015
16       2.718262        2.718323        2.718292        0.000004
17       2.718262        2.718292        2.718277        -0.000002
18       2.718277        2.718292        2.718285        0.000001
19       2.718277        2.718285        2.718281        0.000000
20       2.718281        2.718285        2.718283        0.000000

ROOT: 2.718282
ITERATION: 20
Functional Value: 0.000000
```

LAB 3

--
SOLUTION OF GIVEN EQUATION BY REGULAR FALSI METHOD
--

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
#define EPS 0.000001
#define F(x) x*log(x)-1

void main()
{
	float a,b,c,d,e,fa,fb;
	int count;

	begin:
		printf("Enter Bracketing Values:\t");
		scanf("%f%f",&a,&b);

		fa=F(a);
		fb=F(b);

		if((fa*fb)>0)
		{
			printf("Entered Values Didnot Bracket The ROOT\n");
			goto begin;
		}
		else
		{
			count=1;
			 printf("\ncount\t a\t\t b\t\t c\t\t F(c)\n");

			iteration:

				c=((a*fb)-(b*fa))/(fb-fa);
				e=F(c);
				 printf("\n%d\t %f\t %f\t %f\t %f\t", count, a, b, c, e);

				if(e==0)
				{
					printf("\n\nROOT: %f\t",c);
					printf("\nITERATION: %d\t", count);
					printf("\nFunction Value: %f\t", F(c));
					}
				else
				{
					if(e<0)
						a=c;
					else
						b=c;
```

```
                                        d=fabs((b-a)/b);

                                        if(d<EPS)
                                        {
                                                c=((a*fb)-(b*fa))/(fb-fa);

                                                printf("\n\nROOT: %f\t", c);
                                                printf("\nITERATION: %d\t", count);
                                                printf("\nFunction Value: %f\t", F(c));
                                        }
                                        else
                                        {
                                                count=count++;
                                                goto iteration;
                                        }
                                }
                }
}
```

OUTPUT

```
--------------------------
TITLE: REGULAR FALSI METHOD
--------------------------

------------------------------------------------------------------------
OBJECTIVE: TO FIND THE SOLUTION OF GIVEN EQUATION BY REGULAR FALSI METHOD
------------------------------------------------------------------------

Enter Bracketing Values:        4 5
Entered Values Didnot Bracket The ROOT
Enter Bracketing Values:        1 3

count     a               b               c               F(c)

1         1.000000        3.000000        1.606826        -0.237945
2         1.606826        3.000000        2.029533        0.436516
3         1.606826        2.029533        1.735081        -0.043876
4         1.735081        2.029533        1.824422        0.096957
5         1.735081        1.824422        1.762188        -0.001621
6         1.762188        1.824422        1.781071        0.028060
7         1.762188        1.781071        1.767917        0.007363
8         1.762188        1.767917        1.763927        0.001103
9         1.762188        1.763927        1.762716        -0.000795
10        1.762716        1.763927        1.763083        -0.000219
11        1.763083        1.763927        1.763339        0.000182
12        1.763083        1.763339        1.763161        -0.000098
13        1.763161        1.763339        1.763215        -0.000013
14        1.763215        1.763339        1.763252        0.000046
15        1.763215        1.763252        1.763226        0.000005
16        1.763215        1.763226        1.763218        -0.000007
17        1.763218        1.763226        1.763221        -0.000004
18        1.763221        1.763226        1.763222        -0.000001
19        1.763222        1.763226        1.763223        0.000001

ROOT: 1.763223
ITERATION: 19
Function Value: 0.000000
```

LAB 4

SOLUTION OF GIVEN EQUATION BY NEWTON RAPHSON METHOD

```
#include<stdio.h>
#include<conio.h>
#include<math.h>

#define EPS 0.000001
#define F(x) (x)*(x)-3*(x)+2
#define FD(x) 2*(x)-3


void main()
{
	int count;
	float x0, xn, fx,fn,fdx,e;

	printf("\n");
	 printf("----------------------------\n");
	printf("TITLE: NEWTON RAPHSON METHOD\n");
	printf("----------------------------\n\n");

	 printf("------------------------------------------------------------------------\n");
	printf("OBJECTIVE: TO FIND THE SOLUTION OF GIVEN EQUATION BY NEWTON RAPHSON METHOD\n");
	printf("------------------------------------------------------------------------\n\n");

	printf("\nINPUT INITIAL GUESS (X0):\t");
	scanf("%f", &x0);

	printf("\n\n");

	count=1;
	printf("Count\tx0\t\tx1\t\tError\t\tFunction Value\N");

	begin:
		fx=F(x0);
		fdx=FD(x0);

		xn=x0-(fx/fdx);
		fn=F(xn);

		e=fabs((xn-x0)/xn);

		printf("\n%d\t%f\t%f\t%f\t%f",count,x0,xn,e,fn);
```

```
        if(e<EPS)
         {
                  printf("\n\n");
                  printf("\nRoot = %f\t", xn);
                  printf("\nFunction Value = %f\t", F(xn));
                  printf("\nIteration = %d\t", count);
         }
         else
         {
                  x0=xn;
                  count++;
                  goto begin;
         }
}
```

OUTPUT

```
----------------------------
TITLE: NEWTON RAPHSON METHOD
----------------------------

---------------------------------------------------------------------------
OBJECTIVE: TO FIND THE SOLUTION OF GIVEN EQUATION BY NEWTON RAPHSON METHOD
---------------------------------------------------------------------------


INPUT INITIAL GUESS (X0):       3


Count   x0              x1              Error           Function Value

1       3.000000        2.333333        0.285714        0.444444
2       2.333333        2.066667        0.129032        0.071111
3       2.066667        2.003922        0.031311        0.003937
4       2.003922        2.000015        0.001953        0.000015
5       2.000015        2.000000        0.000008        0.000000
6       2.000000        2.000000        0.000000        0.000000


Root = 2.000000
Function Value = 0.000000
Iteration = 6
```

LAB 5

SOLUTION OF GIVEN EQUATION BY SECANT METHOD

```
#include<stdio.h>
#include<math.h>
#include<conio.h>

#define EPS 0.000001
#define F(x) (x)*(x)+4*(x)-10

void main()
{
        int  count;
        float x1,x2,x3,f1,f2,f3;

        printf("\n");
        printf("-----------------------\n");
        printf("TITLE: SECANT METHOD\n");
        printf("-----------------------\n\n");

        printf("--------------------------------------------------------------------\n");
        printf("OBJECTIVE: TO FIND THE SOLUTION OF GIVEN EQUATION BY SECANT METHOD\n");
        printf("--------------------------------------------------------------------\n\n");

        printf("\nInput two starting values (x1 & x2):\t");
        scanf("%f%f", &x1, &x2);

        f1=F(x1);
        f2=F(x2);

        count=1;
        printf("\nCount\t\tx1\t\tx2\t\tx3\t\tf(x3)\n");

         begin:
                x3=(f2*x1-f1*x2)/(f2-f1);
                f3=F(x3);

                printf("\n%d\t\t%f\t%f\t%f\t%f",count,x1,x2,x3,f3);

                if(fabs((x3-x2)/x3)<EPS)
                {
                        printf("\n\nRoot = %f\t", x3);
                        printf("\nFunction Value at %f\t = %f\t", x3, F(x3));
                        printf("\nNo of Iteration = %d\t", count);
                 }
                 else
                 {
                        x1=x2;
                        f1=f2;
                        x2=x3;
                        f2=F(x3);
```

```
                count++;
                goto begin;
        }
}
```

OUTPUT

```
-----------------------
TITLE: SECANT METHOD
-----------------------

---------------------------------------------------------------------
OBJECTIVE: TO FIND THE SOLUTION OF GIVEN EQUATION BY SECANT METHOD
---------------------------------------------------------------------


Input two starting values (x1 & x2):    1 3

Count           x1              x2              x3              f(x3)

1               1.000000        3.000000        1.625000        -0.859375
2               3.000000        1.625000        1.724638        -0.127075
3               1.625000        1.724638        1.741928        0.002021
4               1.724638        1.741928        1.741657        -0.000005
5               1.741928        1.741657        1.741657        0.000000

Root = 1.741657
Function Value at 1.741657       = 0.000000
No of Iteration = 5
```

LAB 6

SOLUTION OF GIVEN EQUATION BY FIXED POINT ITERATION METHOD

```
#include<stdio.h>
#include<math.h>
#include<conio.h>

#define EPS 0.000001
//#define G(x) pow(x,2)-5       //#define G(x) 5/x       //#define G(x) pow(x,2)+x-5
#define G(x) (pow(x,2)+5)/(2*(x))

void main()
{
        int count;
        float x0, x, e, f;

        printf("\n");
        printf("-----------------------------------\n");
        printf("TITLE: FIXED POINT ITERATION METHOD\n");
        printf("-----------------------------------\n\n");

        printf("------------------------------------------------------------------------------\n");
        printf("OBJECTIVE: TO FIND THE SOLUTION OF GIVEN EQUATION BY FIXED PNT ITERATION METHOD\n");
        printf("------------------------------------------------------------------------------\n\n");

        printf("\nInput Initial Estimate(x0):\t");
        scanf("%f", &x0);

        count=1;

        printf("\nitn\tx0\t\tx1\t\tError\t\tF(x)");
        printf("\n");

        iteration:

        x=G(x0);
        e=fabs((x-x0)/x);

         f=G(x);

        printf("\n%d\t%f\t%f\t%f\t%f",count,x0,x,e,f);

        if(e>EPS)
        {
                x0=x;
                count = count++;
                goto iteration;
        }
        else
        {
                printf("\n\nRoot is: %f\t", x);
```

```
        printf("\nFunction value: %f\t", G(x));
        printf("\nIteration: %d\t", count);
    }
}
```

OUTPUT

```
-----------------------------------
TITLE: FIXED POINT ITERATION METHOD
-----------------------------------

-------------------------------------------------------------------------------
OBJECTIVE: TO FIND THE SOLUTION OF GIVEN EQUATION BY FIXED PNT ITERATION METHOD
-------------------------------------------------------------------------------


Input Initial Estimate(x0):     9

Count   x0              x1              Error           F(x)

1       9.000000        4.777778        0.883721        2.912145
2       4.777778        2.912145        0.640639        2.314546
3       2.912145        2.314546        0.258193        2.237398
4       2.314546        2.237398        0.034481        2.236068
5       2.237398        2.236068        0.000595        2.236068
6       2.236068        2.236068        0.000000        2.236068

Root is: 2.236068
Function value: 2.236068
Iteration: 6
```

LAB 7

LINEAR INTERPOLATION TECHNIQUES

```
#include<stdio.h>
#include<math.h>
#include<conio.h>

#define F(x) sqrt(x)

void main()
{
	float x1, x2, fx1, fx2, fx, f, x, e;

	printf("\n------------------------------------\n");
	printf("TITLE: LINEAR INTERPOLATION TECHNIQUE\n");
	printf("------------------------------------\n\n");

	printf("-----------------------------------------------------------------------------\n");
	printf("OBJECTIVE: TO FIND THE FUNCTIONAL VALUE AT GIVEN PNT USING LINEAR INTERPOLATION\n");
	printf("-----------------------------------------------------------------------------\n\n");

	printf("\nFunction F(x): sqrt(x)\n");
	printf("\nAt Which Point You Want To Evaluate Function (x):\t");
	scanf("%f", &x);

	x1 = x-1;		x2 = x+1;
	fx1 = F(x1);	fx2 = F(x2);

	f = fx1 + (x-x1)*(fx2-fx1)/(x2-x1);

	fx = F(x);		e = fabs(f-fx);

	printf("\nInterpolating Points (x1, x2):(%f, %f)", x1, x2);
	printf("\nEvaluation Point (x):%f", x);
	printf("\nFunctional Value F(x):%f", f);
	printf("\nError From Exact value (Error):%f", e);
}
```

OUTPUT

```
-------------------------------------
TITLE: LINEAR INTERPOLATION TECHNIQUE
-------------------------------------

-------------------------------------------------------------------------------
OBJECTIVE: TO FIND THE FUNCTIONAL VALUE AT GIVEN PNT USING LINEAR INTERPOLATION
-------------------------------------------------------------------------------


Function F(x): sqrt(x)

At Which Point You Want To Evaluate Function (x):       9

Interpolating Points (x1, x2):(8.000000, 10.000000)
Evaluation Point (x):9.000000
Functional Value F(x):2.995352
Error From Exact value (Error):0.004648
```

LAB 8

LAGRANGE INTERPOLATION TECHNIQUE

```
#include<stdio.h>
#include<math.h>
#include<conio.h>

#define F(x) sqrt(x)

void main()
{
	int n, i, j;
	float x[10], y[10], lbp, lp, xp, sum;

	printf("\n----------------------------------------\n");
	printf("TITLE: LAGRANGE'S INTERPOLATION TECHNIQUE\n");
	printf("----------------------------------------\n\n");

	printf("---------------------------------------------------------------------------\n");
	printf("OBJECTIVE: TO FIND THE FUNCNL VALUE AT GIVEN PNT USING LAGRANGE INTERPOLATION\n");
	printf("---------------------------------------------------------------------------\n\n");

	printf("\nEnter The Number Of Data Points (n):\t");
	scanf("%d", &n);
	printf("\nEnter Data Points & Values (x[i], y[i]):\t");
	for(i=1; i<=n; i++)
	scanf("%f%f", &x[i], &y[i]);

	printf("\nAt Which Point You Want To Evaluate Function (xp):\t");
	scanf("%f", &xp);

	sum = 0;
	for(i=1; i<=n; i++)
	{
		lbp = 1;
		for(j=1; j<=n; j++)
		{
			if(i!=j)
			lbp = lbp*(xp-x[j])/(x[i]-x[j]);
		}
		sum = sum + lbp*y[i];
	}
	lp = sum;
	printf("\n--------------------");
	printf("\nInterpolating Points\n");
	printf("--------------------\n");
	printf("\nx[i]\t\ty[i]\n");
	for(i=1; i<=n; i++)
	printf("\n%f\t%f", x[i], y[i]);
	printf("\n\nEvaluation Point (xp):%f", xp);
	printf("\nFunctional Value F(x):%f", lp);
}
```


Prepared By
Er. Shree Krishna Khadka

OUTPUT

```
-----------------------------------------
TITLE: LAGRANGE'S INTERPOLATION TECHNIQUE
-----------------------------------------

--------------------------------------------------------------------------------
OBJECTIVE: TO FIND THE FUNCNL VALUE AT GIVEN PNT USING LAGRANGE'S INTERPOLATION
--------------------------------------------------------------------------------


Enter The Number Of Data Points (n):    4

Enter Data Points & Values (x[i], y[i]):        0 0 1 1.7183 2 6.3891 3 19.0855

At Which Point You Want To Evaluate Function (xp):      1.5

--------------------
Interpolating Points
--------------------

x[i]            y[i]

0.000000        0.000000
1.000000        1.718300
2.000000        6.389100
3.000000        19.085501

Evaluation Point (xp):1.500000
Functional Value F(x):3.367568
```