Least Square regression:

$$Q = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n}[y_i - f(x_i)]^2$$

$$= \sum_{i=1}^{n}(y_i - a - bx_i)^2 \quad \text{— (v)}$$

In the 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

$$\frac{\partial Q}{\partial a} = 0 \quad \text{and} \quad \frac{\partial Q}{\partial b} = 0$$

Then;

$$\frac{\partial Q}{\partial a} = -2\sum_{i=1}^{n}(y_i - a - bx_i) = 0 \quad \text{— (vi)}$$

$$\frac{\partial Q}{\partial b} = -2\sum_{i=1}^{n}x_i(y_i - a - bx_i) = 0 \quad \text{— (vii)}$$

Thus;

$$\sum y_i = na + b\sum x_i \quad \text{— from (vi)}$$

$$\sum x_i y_i = a\sum x_i + b\sum x_i^2 \quad \text{— from (vii)}$$

Solving for a & b.

$$a = \frac{\sum y_i}{n} - b\frac{\sum x_i}{n} = \bar{y} - b\bar{x} \quad \text{— (viii)}$$

$$b = \frac{n\sum x_i y_i - \sum x_i \cdot \sum y_i}{n\sum x_i^2 - (\sum x_i)^2} \quad \text{— (ix)}$$

Thus, eqn (viii) & (ix) are normal equation where $\bar{x}$ & $\bar{y}$ are the average of x and y value respectively.

Algorithm :-

Linear regression

1. Read data values.
2. Compute sum of powers & product
   $\Sigma x_i, \Sigma y_i, \Sigma x_i^2, \Sigma x_i y_i$
3. Check whether the denominator of the equation for b is zero.
4. Compute b and a
5. print out the equation
6. Interpolate data if required

Exponential Regression :-

As, we know that exponential equation be;

$$y = ae^{bx} \quad \text{—} \quad (1)$$

Taking log on both side;

$$\ln y = \ln ae^{bx}$$

$$\log y = \log a + \log e^{bx}$$

$$\log y = \log a + bx \log e$$

$$\log y = \log a + bx \quad \text{—} \quad (11)$$

Eq^n (11) is similar in form of linear equation and therefore solving eq^n (11) we get,

$$b = \frac{n \sum x \log y - \sum x \cdot \sum \log y}{n \sum x^2 - (\sum x)^2}$$

&

$$\log a = \frac{\sum \log y}{n} - b \frac{\sum x}{n}$$

For Quardratic Equation :-

$$y = a_1 + a_2 x + a_3 x^2$$

$$= f(x) \quad \text{—} \quad (1)$$

Sum of square of error;

$$Q = \sum [y - f(x)]^2 \quad \text{—} \quad (2)$$

Since $f(x)$ is a quardratic and contain coefficient a_1, a_2, a_3 we have to estimate all coefficient.

$$\frac{\partial Q}{\partial a_1} = 0$$

$$\frac{\partial Q}{\partial a_2} = 0$$

$$\frac{\partial Q}{\partial a_3} = 0$$

Consider;

$$\frac{\partial Q}{\partial a_j} = -2 \sum_{i=1}^{3} [y_i - f(x_i)] \cdot \frac{\partial f(x_i)}{\partial a_j} = 0$$

$$\frac{\partial f(x_i)}{\partial a_j} = x_i^{j-1}$$

Thus, we have;

$$\sum_{i=1}^{3} [y_i - f(x_i)] \, x_i^{j-1} = 0 \quad \text{for } j = 1 \text{ to } 3$$

$$\sum [y_i x_i^{j-1} - x_j^{j-1} f(x_i)] = 0 \quad \text{---} \text{ (iii)}$$

Now;

substituting value of $f(x_i)$ in (iii)

$$\sum_{i=1}^{3} x_i^{j-1} (a_1 + a_2 x + a_3 x^2) = \sum_{i=1}^{3} y_i x_i^{j-1}$$

After simplification we get;

$$\left.\begin{aligned} a_1 n + a_2 \sum x_i + a_3 \sum x_i^2 &= \sum y_i \\ a_1 \sum x_i + a_2 \sum x_i^2 + a_3 \sum x_i^3 &= \sum x_i y_i \\ a_1 \sum x_i^2 + a_2 \sum x_i^3 + a_3 \sum x_i^4 &= \sum x_i^2 y_i \end{aligned}\right\} \text{---} \text{ (iv)}$$

#

Matrix representation of eq^n (iv) as.

$CA = B$

$$C = \begin{bmatrix} n & \sum x_i & \sum x_i^2 \\ \sum x_i & \sum x_i^2 & \sum x_i^3 \\ \sum x_i^2 & \sum x_i^3 & \sum x_i^4 \end{bmatrix} \quad A = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \quad B = \begin{bmatrix} \sum y_i \\ \sum x_i y_i \\ \sum x_i^2 y_i \end{bmatrix}$$

Chapter:-4 Numerical Differentiation and Integration

* What do you mean by numerical differentiation? Write an algorithm for central difference quotient formula with example.

Soln:-
The method of obtaining the derivative of a function using a numeric ~~integration~~ technique is known as numerical differentiation. There are essentially two situations where numerical differentiation is required. They are:

(i) The functional value are known but the function is unknown. Such function are called tabulated function.

(ii) The function to be differentiated is complicated and therefore it is difficult to differentiate.

For central difference quotient formula:-

Consider small increment $\Delta x = h$ in x. According to Taylor's theorem;

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2!}f''(x) + \frac{h^3}{3!}f'''(\theta_1) \quad \text{--- (1)}$$

Similarly;

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

Subtracting eqn (2) from eqn (1)

$$f(x+h) - f(x-h) = 2hf'(x) + \frac{h^3}{3}[f'''(\theta_1) + f'''(\theta_2)] \quad \text{--- (3)}$$

Thus, we have;

$$\boxed{f'(x) = \frac{f(x+h) - f(x-h)}{2h}} \quad \text{--- (4)}$$

Eqn (4) is second order central difference quotient.

Example:-

$f(x) = x^2$ at $x = 1$ for $h = 0.2, 0.1, 0.05, 0.01$

using central Difference formula

h	$f'(1)$	Error
0.2	2.0	0
0.1	2.0	0
0.05	2.0	0
0.01	2.0	0

<u>For forward difference Quotient & Backword</u>

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

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2} f''(\theta) \quad - ①$$

for $x \le \theta \le x+h$. By rearranging the term, we get;

$$f'(x) = \frac{f(x+h) - f(x)}{h} - \frac{h^2}{2} f''(\theta) \quad - ②$$

Thus, if h is choose to be sufficiently small, $f'(x)$ can be approximated by;

$$f'(x) = \frac{f(x+h) - f(x)}{h} \quad - ③$$

with a truncation error of;

$$E_t(h) = -\frac{h}{2} f''(\theta) \quad - ④$$

Eqn ③ is called first order forward difference constant. This is also known as two point formula.

The truncation error is in the order of h and can be decreased by decreasing h.

For Backward Difference Quotient:

By using Taylor's expansion;

$$f(x-h) = f(x) - hf'(x) + \frac{h^2}{2} f''(\theta) \quad \text{--- } (1)$$

for; $x-h \leq \theta \leq x$

$$f'(x) = \frac{f(x) - f(x-h)}{h} + \frac{h^2}{2} f''(\theta) \quad \text{--- } (2)$$

If h is choosen sufficiently small;

$$f'(x) = \frac{f(x) - f(x-h)}{h} \quad \text{--- } (3)$$

Eq^n (3) is the Backward Difference Quotient.

Newton cotes Methods:-

Newton cotes formula is the most popular and widely used numerical Integration formula.

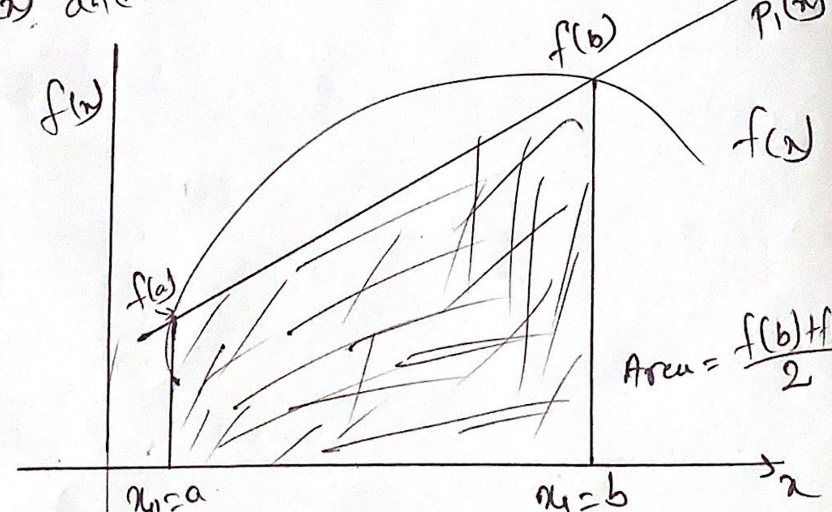
The derivation of Newton-cotes formula is based on polynomial ~~interpolated~~ interpolation. i.e;

$$I = \int_a^b f(x)\,dx$$

We only determined for closed form method:-

1. Trapazoidal rule
2. Simpson 1/3 rule
3. Simpson 3/8 rule
4. Bode's rule

(1) Trapazoidal rule:- It is first and simplest form of newton cotes formulae. Since it is two point formula, it uses the first order interpolation polynomial $P_1(x)$ for approximating function $f(x)$ and assume $x_0 = a$ and $x_1 = b$



Representation of ~~trapazoidal~~ trapazoidal rule

$$I_t = \int_a^b (T_0 + T_1)\,dx$$

$$= \int_a^b T_0\, dx + \int_a^b T_1\, dx = I_{t1} + I_{t2}$$

Since T_i are expressed in term of S, we need to use the following transformation.

$$dx = h \times ds$$

$x_0 = a$, $x_1 = b$ and $h = b - a$

At $x = a$ $\quad S = (a - x_0)/h = 0$

At $x = b$ $\quad S = (b - x_0)/h = 1$

$$\therefore \quad I_{t1} = \int_a^b T_0\, dx = \int_0^1 h f_0\, ds = h f_0.$$

$$I_{t2} = \int_a^b T_1\, dx = \int_0^1 \Delta f_0\, S h\, ds = h \frac{\Delta f_0}{2}$$

Therefore,

$$I_t = h\left[f_0 + \frac{\Delta f_0}{2}\right] = h\left[\frac{f_0 + f_1}{2}\right]$$

Since; $f_0 = f(a)$ and $f_1 = f(b)$ we have

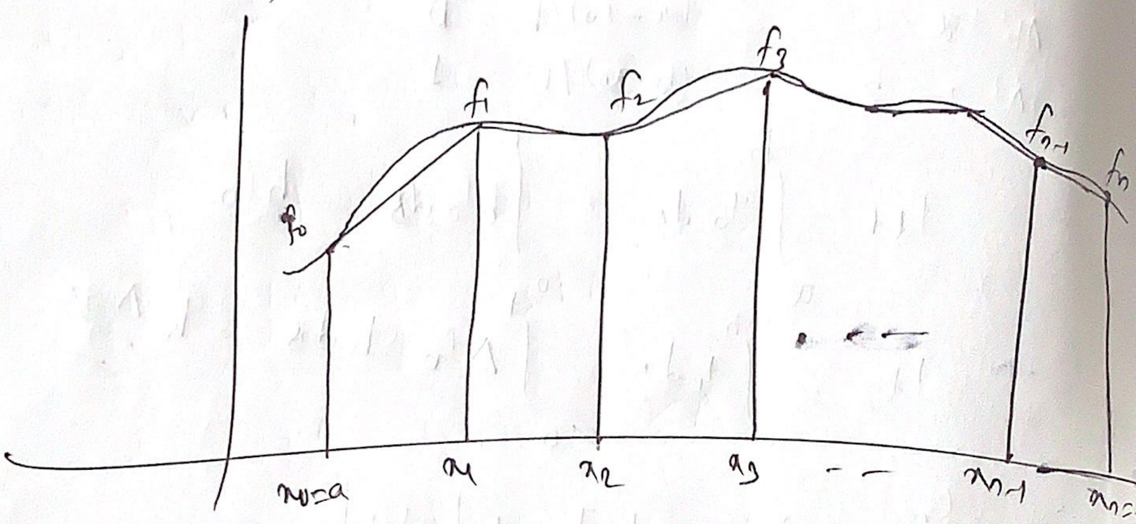
$$I_t = h\left[\frac{f(a) + f(b)}{2}\right] = (b-a)\left[\frac{f(a)+f(b)}{2}\right]$$

$$\therefore \quad \boxed{I_t = (b-a)\left[\frac{f(a)+f(b)}{2}\right]} \quad \text{—— (1)}$$

Eqn (1) is trapezoidal rule to compute functional value between two points.

Composite Trapezoidal rule

If the range to be integrated is large, the trapezoidal rule can be improved by dividing the interval (a,b) into a number of small intervals & applying the rule as simple trapezoidal rule.



$$h = \frac{b-a}{n}$$

$$x_i = a + ih \qquad i = 0, 1, \ldots n$$

As we known that

$$I_t = h\,\frac{f(a)+f(b)}{2}$$

Here,

$$I_i = \int_{x_{i-1}}^{x_i} P_1(x)\,dx = \frac{h}{2}\left[f(x_{i-1}) + f(x_i)\right] \quad \text{—} \; (2)$$

The total area of n segment as

$$I_{ct} = \sum_{i=1}^{n} \frac{h}{2}\left[f(x_{i-1}) + f[x_i]\right]$$

$$= \frac{h}{2}\left[f(x_0) + f(x_1)\right] + \frac{h}{2}\left[f(x_1) + f(x_2)\right] + \ldots$$

$$\ldots + \frac{h}{2}\left[f(x_{n-1}) + f(x_n)\right]$$

$$A_{ct} = \frac{h}{2}\left[f_0 + 2\sum_{i=1}^{n-1} f(x_i) + f_n\right] \quad \text{—— (3)}$$

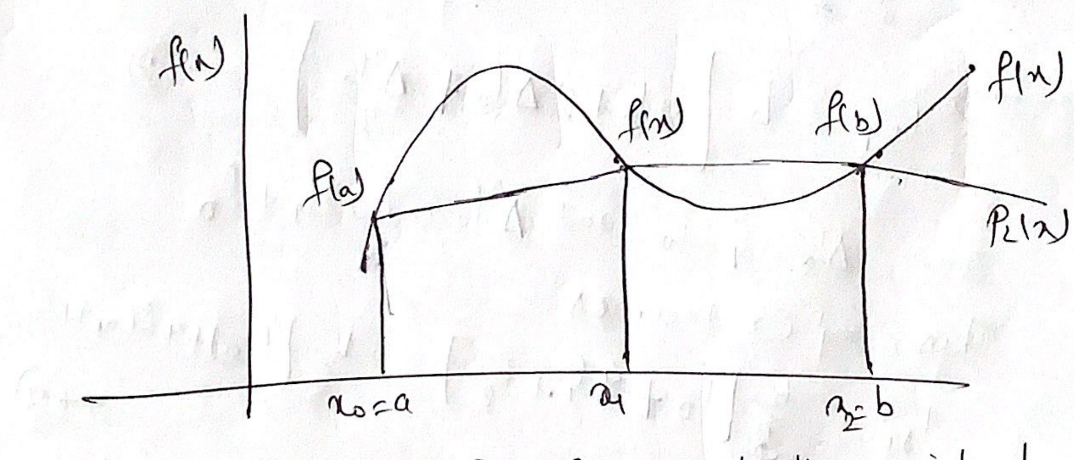
Eqⁿ (3) is composite trapezoidal formula.

Simpson 1/3 rule

Here, the function is approximated by a second order polynomial $P_2(x)$ which passes through sampling point as shown in figure below. The three points include the end points a and b and a mid-point between them i.e; $x_0 = a$, $x_2 = b$ and $x_1 = (a+b)/2$.

The width of the segments h is given by

$$h = \frac{b-a}{2}$$



Representing of simpson's three point rule

The integral of Simpson's $\frac{1}{3}$ rule is obtained by integrating first three terms of the eqⁿ:-

$$I_{S_1} = \int_a^b P_2(x)\,dx = \int_a^b (T_0 + T_1 + T_2)\,dx$$

$$= \int_a^b T_0\,dx + \int_a^b T_1\,dx + \int_a^b T_2\,dx$$

$$= I_{S11} + I_{S12} + I_{S13} \quad \text{— (1)}$$

where; $I_{S11} = \int_a^b f_0 \, dx$

$I_{S12} = \int_a^b \Delta f_0 S \, dx$

$I_{S13} = \int_a^b \frac{\Delta^2 f_0}{2} s(s-1) \, dx$

$dx = h \times ds$

and s varies from 0 to 2. Thus;

$I_{S11} = \int_0^2 f_0 h \, ds = 2hf_0$

$I_{S12} = \int_0^2 \Delta f_0 \, h \, S \, ds = 2h \Delta f_0$

$I_{S13} = \int_0^2 \frac{\Delta^2 f_0}{2} s(s-1) h \, ds = \frac{h}{3} \Delta^2 f_0$

Therefore, $I_{S1} = h \left[2f_0 + 2\Delta f_0 + \frac{\Delta^2 f_0}{3} \right]$ — ②

Since; $\Delta f_0 = f_1 - f_0$ and $\Delta^2 f_0 = f_2 - 2f_1 + f_0$

Thus, eq^n ② becomes;

$I_{S1} = \frac{h}{3}\left[f_0 + 4f_1 + f_2\right] = \frac{h}{3}\left[f(a) + 4f(x_1) + f(b)\right]$ — ③

Eq^n ③ is Simpson's $\frac{1}{3}$ rule. Eq^n ③ can be expressed as

$I_{S1} = \frac{b-a}{3 \times 2}\left[f(a) + 4f(x_1) + f(b)\right]$

$= \frac{(b-a)\left(f(a) + 4f(x_1) + f(b)\right)}{6}$ #

Composite Simpson's 1/3 rule:-

Here, the integration interval is divided into n number of segments of equal width, where n is even number

$$h = \frac{b-a}{n}$$

As usual; $x_i = a + ih$, $i = 0, 1, \ldots n$

According to Simpson 1/3 formula

$$I_{S1} = \frac{h}{3}\left[f(a) + 4f(x_1) + f(b)\right] \quad - (1)$$

Applying the eqn (1) to each of $n/2$ pairs of segments or subinterval;

$(x_{2i-2}, x_{2i-1}), (x_{2i-1}, x_{2i})$. Thus gives

$$I_{cs1} = \frac{h}{3}\sum_{i=1}^{n/2}\left[f(x_{2i-2}) + 4f(x_{2i-1}) + f(x_{2i})\right]$$

$$= \frac{h}{3}\left[f(a) + 4f_1 + 2f_2 + 4f_3 + \cdots 2f_{n-2} + 4f_{n-1} + f(b)\right]$$

on regrouping.

$$\boxed{I_{cs1} = \frac{h}{3}\left[f(a) + 4\sum_{i=1}^{n/2} f(x_{2i-1}) + 2\sum_{i=1}^{n/2-1} f(x_{2i}) + f(b)\right]} \quad (2)$$

Simpson 3/8 Rule

$$I_{S2} = \frac{3h}{8}\left[f(a) + 3f(x_1) + 3f(x_2) + f(b)\right] \quad -(1)$$

here; $h = \frac{b-a}{3}$. The eqn (1) is known as Simpson's 3/8 rule.

It is also known as newton's three eighths rule.

chapter-5

Gauss Jordan Method

Gauss-Jordan method is another popular method used for solving a system of linear equation. Like gauss elimination Gauss Jordan method also uses the process of elimination of variables, but there is major difference between them

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22} & a_{23} \\ 0 & 0 & a_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \qquad \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1' \\ b_2' \\ b_3' \end{bmatrix}$$

Result of Gauss Elimination | Gauss Jordan elimination

Algorithm for Gauss Jordan Method

1. Normalise the first equation by dividing it by its pivot element.
2. Eliminate x_1 term from all the other equation.
3. Now, normalize the second equation by dividing its pivot element
4. Eliminate x_2 from all the equation, above and below the normalised pivotal equation.
5. Repeat this process until x_n is eliminated from all the last equation.
6. The resultant b vector is the solution vector.

JACOBI ITERATION METHOD

Jacobi method is one of the simple iterative method.
Recall the equation of 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 follows:

$$x_{i+1} = g(x_i) \quad \text{for } i = 0, 1, 2 \ldots$$

Let us consider a system of n equation in n unknown

$$\left.\begin{aligned}
a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1 \\
a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= b_2 \\
\vdots \qquad \vdots \qquad & \quad \vdots \\
a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n &= b_n
\end{aligned}\right\} - \text{eq } (1)$$

we rewrite the original system as;

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

An iteration for x_i can be obtained for the ith equation as follows:-

$$x_i^{(k+1)} = \frac{b_i - (a_{i1}x_1^{(k)} + a_{ii-1}x_{i-1}^{(k)} + a_{ii+1}x_{i+1}^{(k)} + \cdots a_{in}x_n^{(k)})}{a_{ii}}$$

chapter

Jacobi iteration method :-

(i) obtain n, a_{ij} and b_i value

(ii) Set $x_{0i} = b_i / a_{ij}$ for $i = 1, \dots n$

(iii) Set key = 0

(iv) for $i = 1, 2 \dots n$

  (i) set sum = b_i

  (ii) for $j = 1, 2 \dots n$ $(j \neq i)$

    Set sum = sum $- a_{ij} x_{oj}$

  Repeat j

  (iii) Set x_i = sum / a_{ij}

  (iv) If key = 0 then

    if $\left| \frac{x_i - x_{oi}}{x_i} \right|$ > error then.

    Set key = 1

  Repeat i

(v) If Key = 1 then

  Set $x_{oi} = x_i$

  go to step (iii)

(vi) Write result.

Gauss - Seidel Method :-

Gauss Seidal method is an improved version of Jacobi iteration method. In Jacobi method, we begin with the initial values;

$$x_1^{(0)}, x_2^{(0)}, \ldots x_n^{(0)}$$

and next approximation;

$$x_1^{(1)}, x_2^{(1)}, \ldots x_n^{(1)}.$$

$$x_i^{(k+1)} = \frac{b_i - (a_{i1} x_1^{(k+1)} + \cdots + a_{ii-1} x_{i-1}^{(k+1)} + a_{ii-1} x_{i+1}^{(k)} + \cdots + a_{in} x_n^{(k)})}{b_{ii}}$$

Gauss ~~Jordan~~ Seidal Method :-

① obtain n, a_{ij} and b_i value

② set $x_i = b_i / a_{ii}$ for $i = 1$ to n

③ set key $= 0$

④ For $i = 1$ to n

(i) set sum $= b_i$

(ii) For $j = 1$ to n $(j \neq i)$

set sum = sum $- a_{ij} x_j$

Repeat j

(iii) set dummy = sum / a_{ii}

(iv) If key $= 0$ then

if $\left| \frac{\text{dumy} - x_i}{\text{dummy}} \right| >$ error then

set key $= 1$

(v) set x_i = dummy

Repeat i

⑤ If key $== 1$ then go to step 3

⑥ Write results.