Unit 1

Solution of Nonlinear Equations

Errors in Numerical Methods

1. **True Error:** True error is denoted by E_t and is defined as the difference between the true value and approximate value i.e. *True error = True value – Approximate value*

2. **Relative Error:** Relative error is denoted by E_r, and is defined as the ratio between the true error and the true value i.e. *Relative error* $= \frac{True\ Error}{True\ Value}$

3. **Approximate Error:** Approximate error is denoted by E_a and is defined as the difference between the present approximation and previous approximation i.e.
   Approximate error= Present approximation- previous approximation

4. **Relative Approximate Error:** Relative approximate error is denoted by E_{ra}, and is defined as the ratio between the approximate error and the present approximation i.e.
   Relative approximate error $= \frac{Approximate\ Error}{Present\ Approximation}$

Sources of Error

1. **Turncation Errors:** Turncation errors arises from using an approximation in place of exact mathematical procedure. It is the error resulting from the truncation of the numerical process. We often use some finite number of terms to estimate the sum of a finite series. For e.g. $S = \sum_{i=0}^{\infty} a_i x^i$ is replaced by some finite sum which give rise to truncation errors.

2. **Round off Errors:** Round off errors occurs when fixed number of digits are used to represent exact number. Since, the numbers are stored at every stage of computation; round off errors is introduced at the end of every arithmetic operation.

Bracketing & Non-Bracketing Methods

- ***Bracketing Method*** start with two initial guesses that bracket the root and then systematically reduce the width of the bracket until the solution is reached. E.g.
  - Bisection Method.

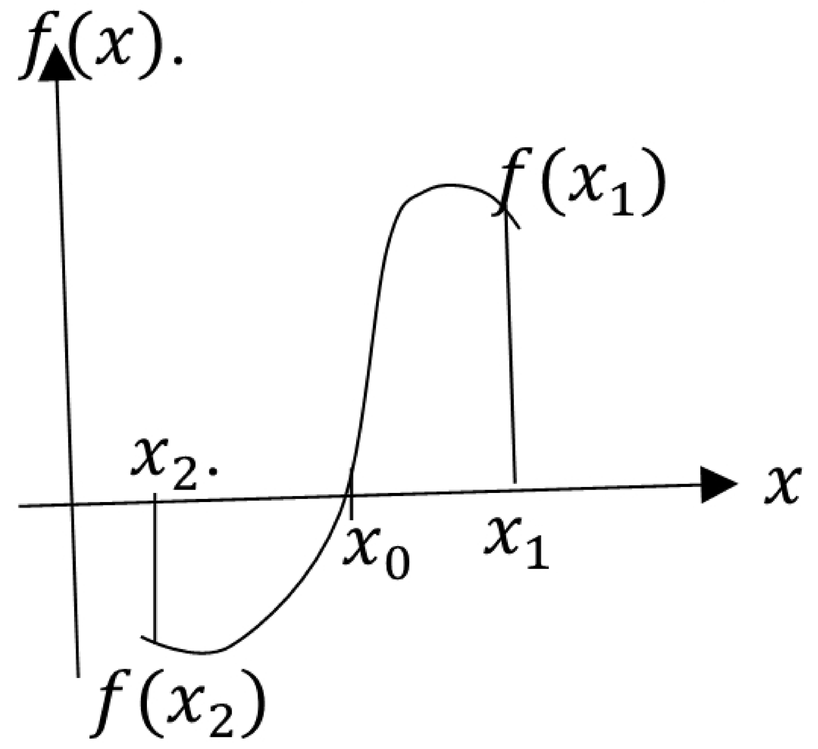
- ***Non-Bracketing Methods*** (Open-end Method) use a single starting value or two values that do not necessarily bracket the root. E.g.
  - Secant method,
  - Newton-Raphson Method.

Different methods for the solution of Non-linear equations are explained below:

Bisection Method

Bisection method is an iterative method used for the solution of non-linear equations, also known as binary chopping or half interval method.

This method is based on the intermediate value theorem which states that if $f(x)$ is continuous in the interval $[x_1, x_2]$ and $f(x_1)$ and $f(x_2)$ has different signs then the equation $f(x) = 0$ has at least one root between x_1 and x_2.



Here mid-point x_0 is

$$x_0 = \frac{x_1 + x_2}{2}$$

This gives us two new intervals $[x_1, x_0]$ & $[x_0, x_2]$.

If $f(x_0) = 0$ then x_0 is the root of $f(x)$. Otherwise

If $f(x_0) * f(x_1) < 0$ then the root lies between x_0 & x_1 else root lies between x_0 & x_2. Then we bisect the interval as before and continue the process until the desired level of accuracy achieved.

Algorithm

1. *Define function f(x) and error.*
2. *Guess two initial values* x_1 and x_2.
3. *Compute* $f(x_1)$ and $f(x_2)$.
4. *If* $f(x_1)$* $f(x_2)$ *>0*
   goto step 8, otherwise:
5. *Calculate* $x_0 = \frac{x_1+x_2}{2}$
6. *Check if* $|f(x_0)|$>*error*
   if $f(x_0)$* $f(x_1)$ *>0*
   x_1=x_0
   else
   x_2=x_0
   goto step 5
7. *Print root* =x_0
8. *END*

Examples

1. Find the root of equation $\boldsymbol{x^3 - 2x - 5 = 0}$ ***using bisection method.***

Sol^n:

Given that,
$f(x) = x^3 - 2x - 5$
Let the initial guess be 2 and 3.
$f(2) = -1 < 0$ and $f(3) = 16 > 0$. Therefore root lies between 2 and 3.

Now let us calculate root by tabulation method.

n	x_1	x_2	$x_0 = \frac{x_1 + x_2}{2}$	$f(x_0)$
1	2	3	2.5	5.625
2	2	2.5	2.25	1.891
3	2	2.25	2.215	0.346
4	2	2.125	2.0625	-0.35132
5	2.0625	2.125	2.0938	-0.0083
6	2.0938	2.125	2.1094	0.1671
7	2.0938	2.1094	2.1016	0.0789
8	2.0938	2.1016	2.0977	0.0352
9	2.0938	2.0977	2.0958	0.0139
10	2.0938	2.0958	2.0948	0.0027

Since, x_1, x_2 & x_0 are same up to two decimal places, so the root of given equation is 2.0948.

Note: *To find the root of* ***trigonometric equation*** *we have to first make sure that calculator is in* ***radian form****.*

2. ***Estimate a real root of following nonlinear equation using bisection method correct upto two significant figures.***

$$\boldsymbol{x^2 sinx + e^{-x} = 3}$$

Sol^n:

Given that,

$f(x) = x^2 sinx + e^{-x} - 3 = 0$

Let the initial guess be 1 & 2.

$f(1) = -1.7906 < 0$

$f(2) = 0.7725 > 0$. Therefore root lies between 1 & 2.

Now let us calculate root by tabulation method.

n	x_1	x_2	$x_0 = \frac{x_1 + x_2}{2}$	$f(x_0)$
1	1	2	1.5	-0.5325
2	1.5	2	1.75	0.1872
3	1.5	1.75	1.625	-0.1663
4	1.625	1.75	1.6875	0.0133
5	1.625	1.6875	1.6562	-0.0761
6	1.6562	1.6875	1.6718	-0.0314
7	1.6718	1.6875	1.6796	$-9.17x10^{-3}$
8	1.6796	1.6875	1.6835	$1.91x10^{-3}$
9	1.6796	1.6835	1.6815	$-3.77x10^{-3}$
10	1.6815	1.6835	1.6825	$-9.27x10^{-4}$
11	1.6825	1.6835	1.683	$4.93x10^{-4}$

Since, x_1, x_2 & x_0 has same value up to 2 decimal place, so the root of given equation is 1.683.

Convergence of Bisection Method

In bisection method interval is halved every interval. After n^{th} iteration size of interval is reduced to

$\frac{x_2 - x_1}{2^n} = \frac{\Delta x}{2^n}$

Now we can say that maximum error after n^{th} iteration is $E_n = \pm\frac{\Delta x}{2^n}$

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

Similarly, after $(n+1)^{th}$ iteration max^m error is given by

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

This equation shows that error is halved after each iteration of bisection method. Therefore we can say that bisection method converges linearly.

Advantages of Bisection Method

1. The bisection method is always convergent. Since, the method brackets the roots, the method guaranteed to converge.
2. As iteration are conducted, the interval gets halved. So one can guarantee the decrease in the error in the solution of the equation.

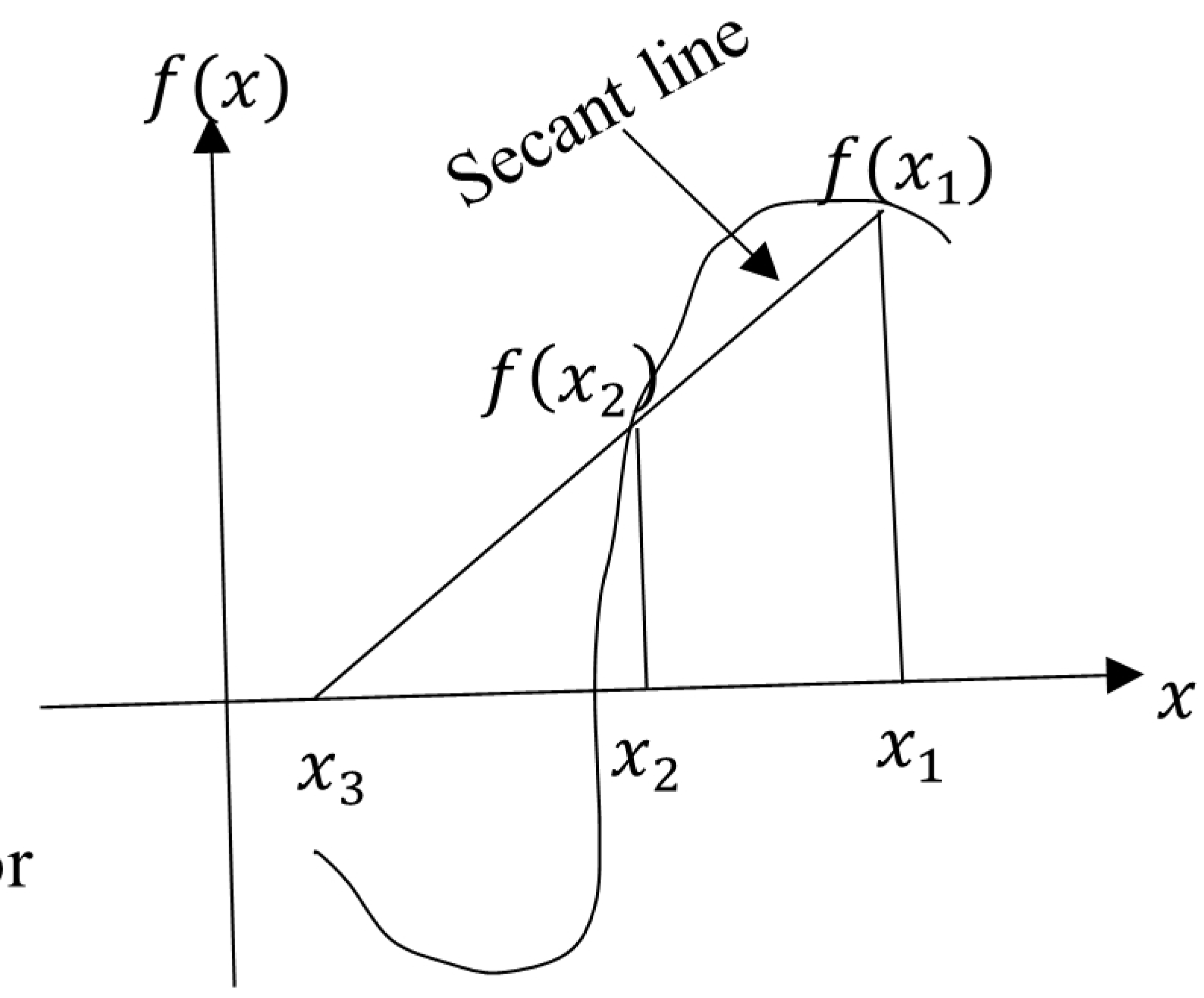
Disadvantages of Bisection Method

1. The convergence of bisection method is slow as it is simply based on halving the interval.
2. If one of the initial guesses is closer to the root, it will take larger number of iterations to reach the root.

Secant Method

This method uses two initial estimates but doesn't require that they must bracket the root. Let us consider the following figure:

The point x_1 and x_2 are starting point, they do not bracket the root. Slope of line (secant line) passing through x_1 and x_2 is given by



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

$$f(x_1)(x_2 - x_3) = f(x_2)(x_1 - x_3)$$

$$x_3[f(x_2) - f(x_1)] = f(x_2).x_1 - f(x_1).x_2$$

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

Now, by adding and subtracting $f(x_2)x_2$ to the numerator and rearranging the terms; we get

$$x_3 = x_2 - \frac{f(x_2)(x_2 - x_1)}{f(x_2) - f(x_1)} \ldots\ldots\ldots\ldots\ldots. (1)$$

This equation is called secant formula. Here, x_3 represents the approximate root of f(x). The approximate value of root can be refined by repeating this procedure by replacing x_1 by x_2 and x_2 by x_3 in equation (1) i.e. next approximate value is given by

$$x_4 = x_3 - \frac{f(x_3)(x_3 - x_2)}{f(x_3) - f(x_2)}$$

This procedure is continued till the desired level of accuracy is obtained.

The secant formula in general form is

$$x_{i+1} = x_i - \frac{f(x_i)(x_i - x_{i-1})}{f(x_i) - f(x_{i-1})}$$

Algorithm

1. *Define function f(x) and error*
2. *Guess two initial values x_1 and x_2*
3. *Calculate* $f(x_1)$ and $f(x_2)$.
4. *Compute* $x_3 = x_2 - \frac{f(x_2)(x_2-x_1)}{f(x_2)-f(x_1)}$
5. *Set* x_1=x_2 *&* x_2=x_3
6. *Check if* $|f(x_3)| >$ *error*
   goto step 4, otherwise,
7. *Print root =* x_3
8. *END*

Examples

1. Use the secant method to estimate the root of the equation sinx-2x+1.

Sol^n:

Given that,
f(x)= sinx-2x+1
Let the initial guess be 0 and 1.

Now, let us calculate the root using tabular form.

n	x_1	x_2	$f(x_1)$	$f(x_2)$	$x_3 = x_2 - \frac{f(x_2)(x_2 - x_1)}{f(x_2) - f(x_1)}$
1	1	0	-0.1585	1	0.863185
2	0	0.863185	1	0.03355	0.89315
3	0.863185	0.89315	0.03355	-0.00725	0.88782
4	0.89315	0.88782	-0.00725	0.000577	0.88786

Hence, the value of x_3 in 3rd and 4th iteration is similar up to 3 decimal places so the root of given equation is 0.088786.

2. *Estimate a real root of following nonlinear equation using secant method correct up to three decimal places*.

$$x^2 + lnx = 3$$

Sol**[n]**:

Given that,

$f(x) = x^2 + lnx - 3$

Let the initial guess be 1 and 2.

Now, let us calculate the root using tabular form.

n	x_1	x_2	$f(x_1)$	$f(x_2)$	$x_3 = x_2 - \frac{f(x_2)(x_2 - x_1)}{f(x_2) - f(x_1)}$
1	1	2	-2	1.6932	1.54154
2	2	1.54154	1.6932	-0.19087	1.58799
3	1.54154	1.58799	-0.19087	-0.01582	1.59219
4	1.58799	1.59219	-0.01582	0.00018	1.59214

Hence, the value of x_3 in 3rd & 4th iteration is similar up to 3 decimal places so the root of given equation is 1.59214.

3. *Using the secant method, estimate the root of the equation $x^2 - 4x - 10 = 0$ with the initial estimates of $x_1 = 4$ & $x_2 = 2$. Do these points bracket a root?*

Sol**[n]**:

Given that,

$f(x) = x^2 - 4x - 10$

Initial estimates be $x_1 = 4$ & $x_2 = 2$.

Now, let us calculate the root using tabular form.

n	x_1	x_2	$f(x_1)$	$f(x_2)$	$x_3 = x_2 - \frac{f(x_2)(x_2 - x_1)}{f(x_2) - f(x_1)}$
1	4	2	-10	-14	9
2	2	9	-14	35	4
3	9	4	35	-10	5.11
4	4	5.11	-10	-4.36	5.97
5	5.11	5.97	-4.36	1.761	5.72
6	5.97	5.72	1.761	-0.1616	5.74
7	5.72	5.74	-0.1616	-0.0124	5.74

Hence, the value of x_3 in 6th & 7th iteration is similar up to 2 decimal places so the root of given equation is 5.74.

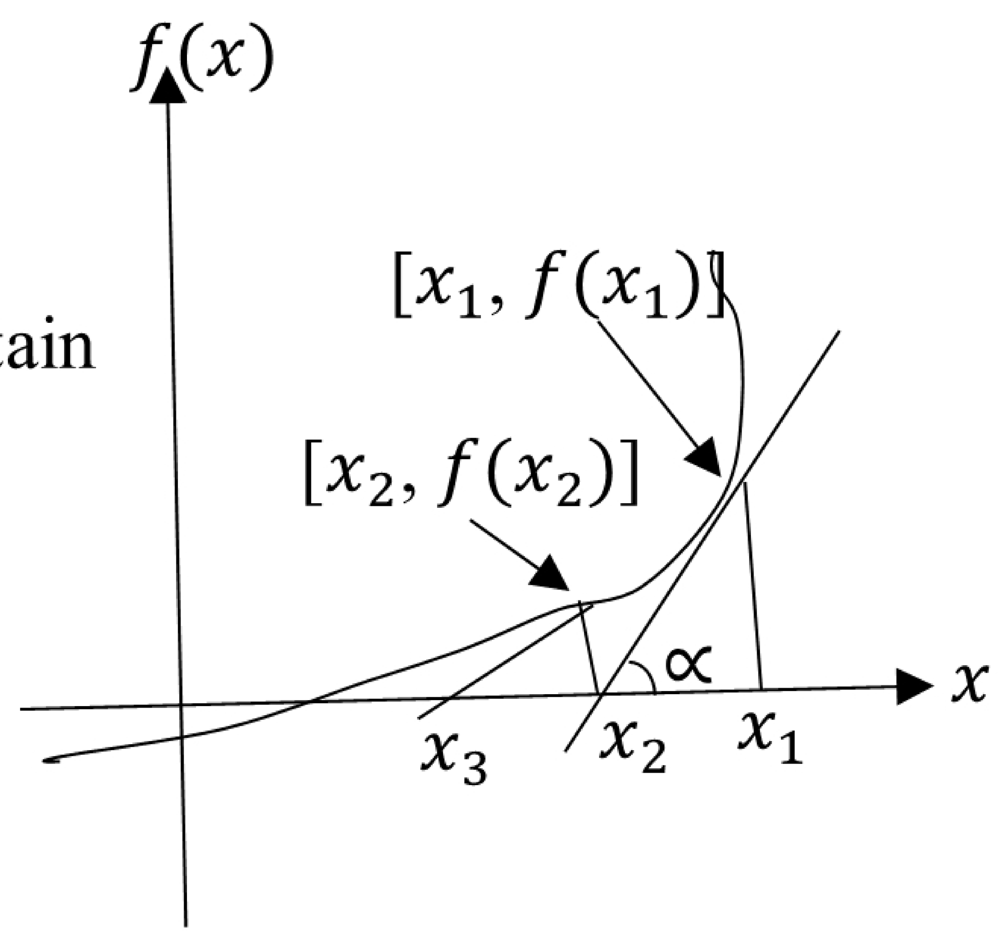
Here, the root of given equation doesn't lie between given initial estimates $x_1 = 4$ & $x_2 = 2$. So, given points doesn't bracket a root.

Convergence of Secant Method

Read yourself (page no. 155, E Balagurusamy's Numerical Methods)

Newton-Raphson Method

Consider a graph below of $f(x)$. Let us consider 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 figure. 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;



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

Where $f(x_1)$ is the slope of $f(x)$ at $x = x_1$. Solving for x_2 we obtain

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

This is called the Newton-Raphson formula.

The next approximation would be

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

In general,

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

Repeat this process until the difference between two successive values is within a prescribed limit.

Algorithm

1. *Guess initial root=x_1 and define stopping criteria error.*
2. *Evaluate $f(x_1)$ & $f'(x_1)$*
3. *Compute new root*

   $$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}$$
4. *Set x_1=x_2*
5. *Check if $|f(x_2)|$ > error*

   goto step 3,otherwise;
6. *Print root = x_2*
7. *END*

Examples

1. Find the root of the equation $e^x - 3x = 0$ using Newton Raphson method correct up to 3 decimal places.

Soln:

Given that,
$f(x) = e^x - 3x$
$f'(x) = e^x - 3$
Let the initial guess be 0.5.

Now, let us calculate the root using tabular form.

n	x_1	$f(x_1)$	$f'(x_1)$	$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}$
1	0.5	0.14872	-1.351278	0.61005
2	0.61005	0.010373	-1.15947	0.61899
3	0.61899	0.0081472	-1.14294	0.61273
4	0.61273	0.0035755	-1.14287	0.61903
5	0.61903	0.00003575	-1.14287	0.61906

Here, the 4th and 5th iteration has same value of x_2 up to 3 decimal place, so that root of given equation is 0.61906.

2. Use the Newton method to estimate the root of the equation $x^2 + 2x - 2 = 0$.

Soln:

Given that,
$f(x) = x^2 + 2x - 2$

$f'(x) = 2x + 2$

Let the initial guess be 0.

Now, let us calculate the root using tabular form.

n	x_1	$f(x_1)$	$f'(x_1)$	$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}$
1	0	-2	2	1
2	1	1	4	0.75
3	0.75	0.0625	3.5	0.73214
4	0.73214	0.000309	3.464	0.73205
5	0.73205	$-2.7975x10^{-6}$	3.464	0.73205

Here, the 4th and 5th iteration has same value of x_2, so that root of given equation is 0.73205.

3. *Find the root of equation $xcosx - x^2 = 0$ using newton's method up to 5 decimal places.*

Sol^n:

Given that,
$f(x) = xcosx - x^2$

$f'(x) = cosx - xsinx - 2x$

Let the initial guess be 1.

Now, let us calculate the root using tabular form.

n	x_1	$f(x_1)$	$f'(x_1)$	$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}$
1	1	-0.4597	-2.3012	0.80023
2	0.80023	-0.0829	-1.4781	0.74414
3	0.74414	$-6.302x10^{-3}$	-1.2566	0.73912
4	0.73912	$-4.313x10^{-5}$	-1.2371	0.73908
5	0.73908	$6.349x10^{-6}$	-1.2369	0.73908

Here, the 4th and 5th iteration has same value of x_2 up to 5 decimal places, so that root of given equation is 0.73908.

4. *Find the roots of the following equations using Newton's method.*

log x – cos x = 0

Solution:
Given,
$f(x) = logx - cosx$
$f'(x) = \frac{1}{x} + sinx$
Let the initial guess be 0.5.

n	x_1	$f(x_1)$	$f'(x_1)$	$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}$
1	0.5	-1.17	2.47	1.62
2	1.62	0.26	1.62	1.45
3	1.45	0.04	1.68	1.43
4	1.43	0.01	1.68	1.42
5	1.42	0.002	1.69	1.42

Here, the 4th and 5th iteration has same value of x_2 up to 2 decimal place, so that root of given equation is 1.42.

Q. Derive the Newton-Raphson formula using the Taylor series expansion.

Sol^n:

Assume that x_0 be the approximate root of $f(x) = 0$ and x_1=$x_0 + h$ be correct root so that $f(x_1) = f(x_0 + h) = 0$

Where, h is a small interval i.e. $h = x_1 - x_0$

We can express $f(x_1)$ using Taylor series expansion as follows:

$$f(x_1) = f(x_0) + f'(x_0)h + f''(x_0)\frac{h^2}{2!} + \cdots \ldots \ldots \ldots.$$

If we neglect the terms containing the second order and higher derivatives, we get

$$f(x_1) = f(x_0) + f'(x_0)h = 0$$

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

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

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

Similarly,

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

Thus,

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

Which is the required formula for Newton Raphson method.

Convergence of Newton-Raphson Method

Read yourself (page no. 147, E Balagurusamy's Numerical Methods)

Limitations of Newton-Raphson Method

1. Division by zero may occur if $f'(x_n)$ is zero or very close to zero.
2. If the initial guess is too far away from the required root, the process may converge to some other root.
3. Calculating the required derivative for every iteration may be costly tasks for some functions.

Fixed Point Method

Any function in the form

$f(x) = 0$……….. (1)

can be manipulated such that x is on the left hand side of the equation as shown below

$x = g(x)$……….. (2)

Equations (1) & (2) are equivalent and therefore a root of equation (2) is a root of equation (1).

If x_0 is the initial guess to a root, the next approximation is given by;

$$x_1 = g(x_0)$$

Further approximation is given by;

$$x_2 = g(x_1)$$

Thus iteration process can be expressed in general form as;

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

Which is fixed point iteration formula.

The iteration process would be terminated when two successive approximation agree within some specified error.

Algorithm

1. *Define function f(x) and error*
2. *Convert the function f(x) =0 in the form x =g(x).*
3. *Guess initial values x_0.*
4. *Calculate $x_{i+1} = g(x_i)$*
5. *If $\left|\frac{x_{i+1}-x_i}{x_{i+1}}\right|$ <= error*
   goto step 7, otherwise;
6. *Assign $x_i = x_{i+1}$*
   goto step 4
7. *Display x_i as the root*
8. *END*

Examples

1. Find the one root of the equation $x^2 + x - 2 = 0$ using the fixed point method.

Sol^n^:

Given that,

$f(x) = x^2 + x - 2 = 0$…………. (1)

For fixed point iteration method, arranging equation (1) in terms of *g(x)*.

$x^2 + x - 2 = 0$
or, $x(x+1) = 2$
or, $x = \frac{2}{x+1}$
$\therefore g(x) = \dfrac{2}{x+1}$

Let the initial guess be 0.

Now calculating the root using tabular form

n	x_i	$x_{i+1} = g(x_i)$
1	0	2
2	2	0.667
3	0.667	1.199
4	1.199	0.91
5	0.91	1.04
6	1.04	0.98
7	0.98	1.01
8	1.01	0.99
9	0.99	1.00
10	1.00	1.00

Since, the value of x_{i+1} in 9th and 10th iteration has similar value. So the root of the given equation is 1.00.

2. Find the root of equation $sinx = 5x - 2$ up to 4 decimal places with initial guess 0.5 using fixed point iteration method.

Sol^n:

Given that,

$f(x) = sinx - 5x + 2 = 0$ ……………*(1)*

For fixed point iteration method, arranging equation (1) in terms of *g(x)*.

$sinx - 5x + 2 = 0$
or, $sinx = 5x - 2$
or, $x = \frac{1}{5}(sinx + 2)$
$\therefore g(x) = \dfrac{1}{5}(sinx + 2)$

Now calculating the root using tabular form

n	x_i	$x_{i+1} = g(x_i)$
1	0.5	0.49589
2	0.49589	0.49516
3	0.49516	0.49503
4	0.49503	0.49501

Since, the value of x_{i+1} in 3rd and 4th iteration has similar value up to 4 decimal place. So the root of the given equation is 0.49503.

Horner's Method

Let $f(x) = a_n x^n + a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \cdots \dots\dots\dots\dots\dots. + a_1 x + a_0$

$$= ((\dots((a_n x + a_{n-1})x + a_{n-2})x + \cdots \dots\dots\dots. + a_1)x + a_0)$$

Let $p_n = a_n$

$p_{n-1} = a_n x + a_{n-1}$

$p_{n-2} = p_{n-1} x + a_{n-2}$

$\dots$

$\dots$

$p_i = p_{i+1} x + a_i$

$\dots$

$p_1 = p_2 x + a_1$

$f(x) = p_0 = p_1 x + a_0$

Algorithm

1. *Read the order of function 'n' and coefficient a_i.*
2. *Read the value for x where value should be determined.*
3. *Set $P_n = a_n$*
4. *For i=n-1 to 0*
   $p_i = p_{i+1} * x + a_i$
5. *Print functional value at x=p_0*
6. *END*

Example

***Q.* Evaluate the polynomial $f(x) = 5x^3+4x^2+3x+9$ using Horner's rule at x=2.**

Sol^n: *$n=3$, $a_3=5$, $a_2=4$, $a_1=3$, $a_0=9$*

$P_3 = a_3 = 5$

$P_2 = 5*2+4 = 14$

$P_1 = 14*2+3 = 31$

$P_0 = 31*2+9 = 71$

$\therefore f(2) = 71$

References:

- *E. Balagurusamy, Numerical Methods, Tata McGraw-Hill*

Please let me know if I missed anything or anything is incorrect.

Poudeljayanta99@gmail.com