Determination of Roots of Algebraic and Transcendental Equations

- Bisection Method
- Fixed Point Iteration Method
- Newton-Raphson Method
- Secant Method

Newton-Raphson Method

In the triangle $x_1 P x_0$:

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

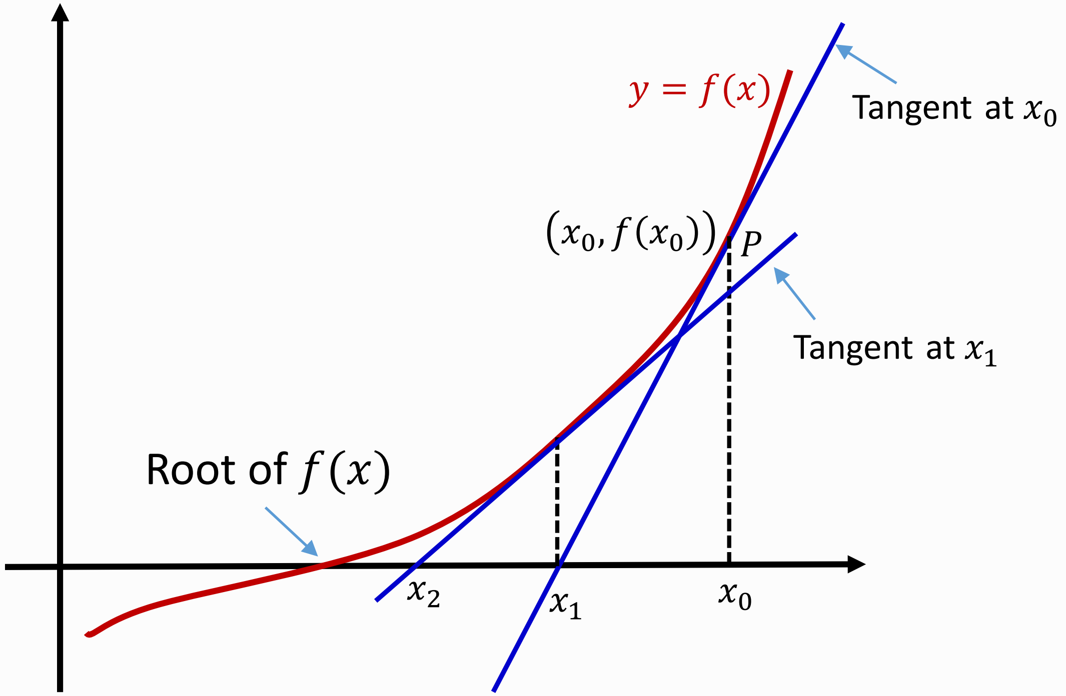
$$\Rightarrow x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

Similarly, the second step :

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

$$\vdots$$

$(k+1)th$ step: $$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}; \quad k = 0, 1, 2, \ldots$$



Alternative Formulation:

Let x_k be an approximation to the solution of $f(x) = 0$.

Let Δx be an increment in x_k such that $x_k + \Delta x$ is an exact root,

i.e.

$$f(x_k + \Delta x) = 0$$

$$\Rightarrow f(x_k) + \Delta x f'(x_k) + \frac{1}{2!}\Delta x^2 f''(x_k) + \cdots = 0$$

Neglecting 2nd and higher order terms of Δx, we get

$$f(x_k) + \Delta x f'(x_k) \approx 0 \quad \Rightarrow \Delta x \approx -\frac{f(x_k)}{f'(x_k)}$$

Hence, the iteration method becomes

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}; \quad k = 0, 1, 2, \ldots$$

Recall Fixed Point Iteration Method:

$$f(x) = 0 \Leftrightarrow x = g(x)$$

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

Convergence is guaranteed if $|g'(x)| \leq \rho < 1$

If we take $g(x) = x - \dfrac{f(x)}{f'(x)}$ assuming $f'(x) \neq 0$

Then $x_{k+1} = x_k - \dfrac{f(x_k)}{f'(x_k)}$ **Newton-Raphson Method**

Convergence of Newton-Raphson Method

Let $x_{k+1} = g(x_k)$ define the newton's method. Let s be a root of $f(x) = 0$, i.e., $s = g(s)$.

Note that $g(x) = x - \dfrac{f(x)}{f'(x)} \implies g'(x) = 1 - \dfrac{\left(f'(x)\right)^2 - f(x)f''(x)}{\left(f'(x)\right)^2}$

$$g'(s) = 1 - \left(\frac{\left(f'(s)\right)^2 - f(s)f''(s)}{\left(f'(s)\right)^2}\right) = \frac{f(s)f''(s)}{(f'(s))^2} = 0 \text{ as } f(s) = 0.$$

Using Taylor's formula for expanding $g(x_k)$ around s in the scheme $x_{k+1} = g(x_k)$:

$$x_{k+1} = g(x_k) = g(s) + g'(s)(x_k - s) + \frac{1}{2}g''(\xi)(x_k - s)^2 \quad \text{where } \xi \in (x_k,\ s)$$

Convergence of Newton-Raphson Method

$$x_{k+1} = g(x_k) = g(s) + g'(s)(x_k - s) + \frac{1}{2}g''(\xi)(x_k - s)^2 \quad \text{where } \xi \in (x_k,\ s)$$

$$\Rightarrow x_{k+1} - s = \frac{1}{2}g''(\xi)(x_k - s)^2$$

$$\Rightarrow e_{k+1} = \frac{1}{2}g''(\xi)\, {e_k}^2$$

Each successive error term is proportional to the square of the previous error.

Hence, Newton-Raphson method converges quadratically.

Note: In the case of fixed point iteration method $g'(x) \neq 0$ (in general), and hence the method converges linearly.

Moreover the size of $|g'(x)|$ matters and it has to be less than 1 for convergence. Note that $g'(s) = 0$ in the case of Newton's method and therefore convergence is guaranteed for x_0 sufficiently close to s.

As discussed, in the case of Newton's, the method converges quadratically for x_0 sufficiently close to s.

Example : Perform four iterations of the Newton-Raphson method to find the smallest positive root of the equation $f(x) = x^3 - 5x + 1 = 0$.

Solution: Take $x_0 = 0.5$

$$f'(x) = 3x^2 - 5$$

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

$$= x_k - \frac{(x_k^3 - 5x_k + 1)}{3x_k^2 - 5}$$

$$= \frac{2x_k^3 - 1}{3x_k^2 - 5}; \quad k = 0, 1, 2, \ldots$$

$x_0 = 0.5$

$x_1 = 0.176470588$

$x_2 = 0.2015680743$

$x_3 = 0.2016396750$

$x_4 = 0.2016396757$

Example : Apply Newton-Raphson method to determine a root of the equation $f(x) = \cos x - xe^x = 0$ such that $|f(x^*)| < 10^{-8}$ where x^* is the approximation to the root. Take the initial approximation as $x_0 = 1$.

Iteration Scheme : $$x_{k+1} = x_k - \frac{(\cos x_k - x_k e^{x_k})}{(-\sin x_k - e^{x_k} - x_k e^{x_k})}$$

k	0	1	2	3	4	5
x_k	1	0.6531	0.5313	0.5179	0.5178	0.5178
$f(x_k)$	-2.1780	-0.4606	-0.0418	-4.6×10^{-4}	-5.9×10^{-8}	-8.8×10^{-16}

Secant Method :

Note that the newton's method is very powerful but it has the disadvantage of evaluating f' which may be computationally very expensive.

The Secant method is a variant of Newton's method where $f'(x_k)$ is replaced by the following differences:

$$f'(x_k) \approx \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}$$

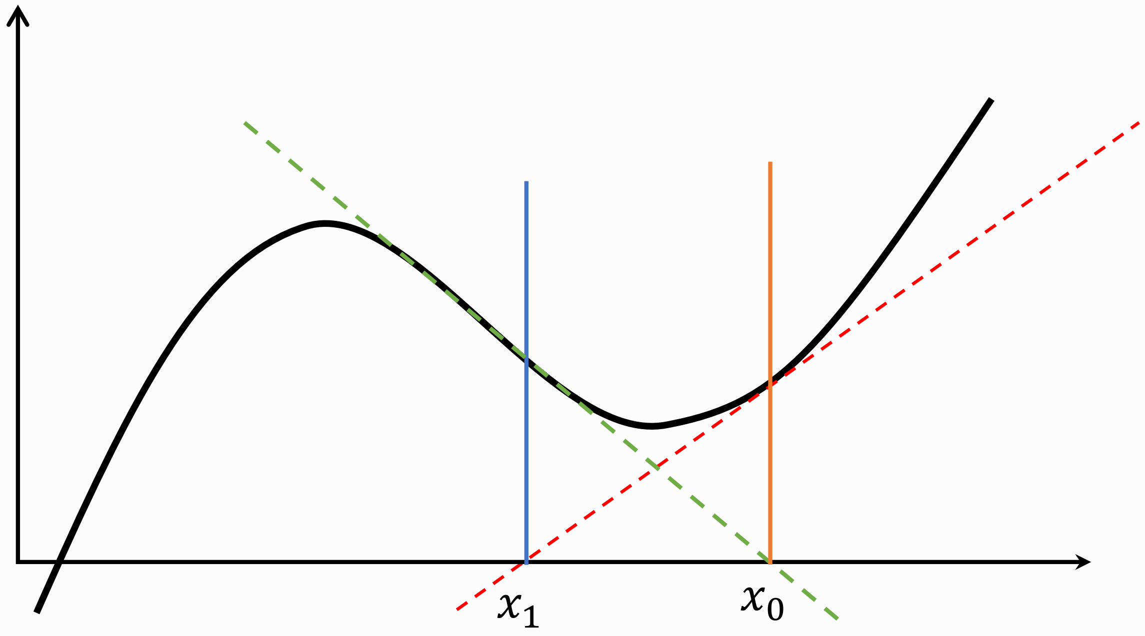
$$x_{k+1} = x_k - \frac{f(x_k)(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})}; \quad k = 0, 1, 2, \ldots$$

Pitfalls: Newton-Raphson Method

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}; \quad k = 0,1,2, \dots.$$

1. The method fails if f' becomes 0 at any approximation $x_k, k = 0,1,2, \dots$

2. Cycling behavior leads to complete failure of the method



Pitfalls: Newton-Raphson Method

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}; \quad k = 0,1,2,\ldots.$$

3. Consider $f(x) = x^3 - 23x^2 + 135x - 225$

Actual Roots are $3, 5, 15,$

Initial Guess	$\mathbf{4}$	$\mathbf{4.2}$	$\mathbf{3.9}$
Iteration 1	15	6.1636	-5.0341
Iteration 2	15	5.2223	-1.3851
Iteration 3		5.0159	0.8586
Iteration 4		5.0001	2.1420
Iteration 5		5.0000	2.7697
Iteration 6		5.0000	2.9749
Iteration 7			2.9996
Iteration 8			3.0000
Iteration 9			3.0000

Pitfalls: Newton-Raphson Method

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}; \quad k = 0,1,2,\dots.$$

4. Consider $f(x) = x^3 + 94x^2 - 389x + 294$

Actual Roots are: $1, 3, -98$

Initial Guess	$\mathbf{2}$	$\mathbf{2.4}$	$\mathbf{3.9}$
Iteration 1	-98	3.4611	0.2061
Iteration 2	-98	3.0742	0.8282
Iteration 3		3.0026	0.9877
Iteration 4		3.0000	0.9999
Iteration 5		3.0000	1.0000
Iteration 6			1.0000

CONCLUSIONS

- **Newton-Raphson Method**

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}; \quad k = 0, 1, 2, \ldots$$

- **Secant Method**

$$x_{k+1} = x_k - \frac{f(x_k)(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})}; \quad k = 0, 1, 2, \ldots$$

Polynomial Interpolation

- Polynomial Interpolation
- Existence and Uniqueness
- Error in Interpolating Polynomials

Interpolation

Interpolation is a process of estimating values between known data points or approximating complicated functions by simple polynomials or determining a polynomial that fits a set of given points.

Applications :

1. Constructing the function when it is not given explicitly and only the values of function are given at some points.

2. Replacing complicated function by an interpolating simpler function (usually polynomials) so that many operations such as determination of roots, differentiation, integrations or other such operations may be performed.

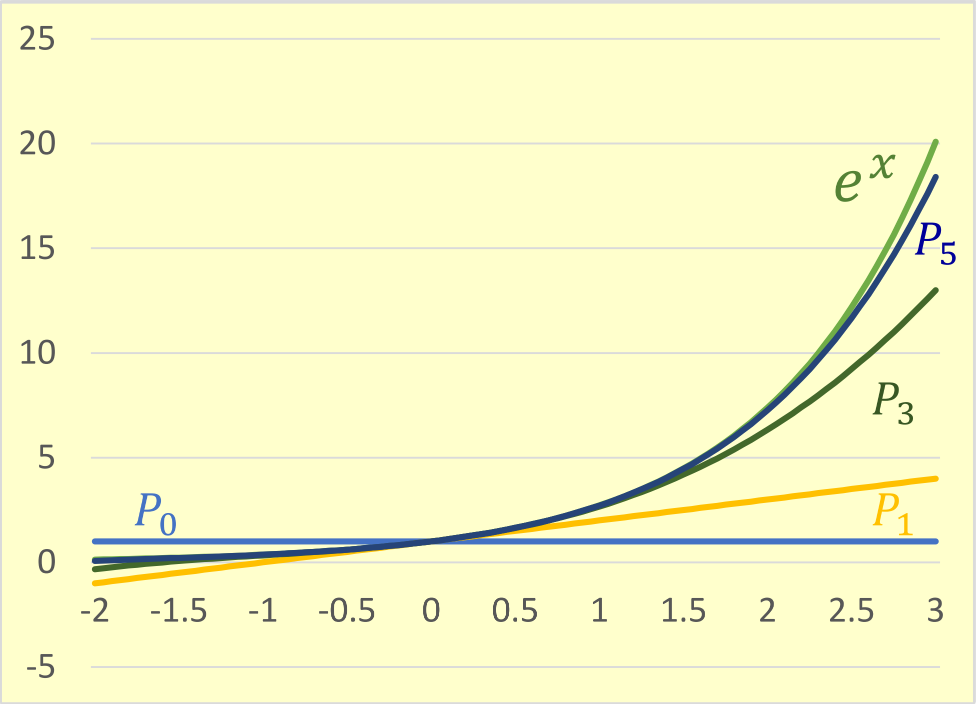
Fundamental principle behind polynomial interpolation

Weirstrass Approximation Theorem

Suppose that f is defined and continuous on $[a, b]$. For each $\epsilon > 0$, there exists a polynomial $P(x)$ with the property that

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

Why not Taylor's Polynomial ? Consider Taylor's Polynomial of e^x around $x = 0$.



$$P_5(x) = 1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24} + \frac{x^5}{120}$$

$$P_4(x) = 1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24}$$

$$P_2(x) = 1 + x + \frac{x^2}{2} \qquad P_3(x) = 1 + x + \frac{x^2}{2} + \frac{x^3}{6}$$

$$P_0(x) = 1; \qquad P_1(x) = 1 + x$$

Taylor's polynomials agree as closely as possible with a given function at a specific point, so they concentrate their accuracy near that point.

For ordinary computation purposes it is more efficient to use methods that include information at various points.

Existence and uniqueness for polynomial interpolation

For $(n+1)$ data points there is one and only one polynomial of order $\leq n$ that passes through all the points.

For example, there is only one straight line (a first order polynomial) that passes through two points.

Consider for simplicity, a second order polynomial

$$f(x) = a_0 + a_1 x + a_2 x^2$$

A straight forward method for computing the coefficients of a polynomial of degree n is based on the fact that $(n+1)$ data points are required (3 data points in this example) to determine $(n+1)$ unknowns (3 unknowns $a_0,\ a_1,\ a_2$ in this example)

Polynomial to be fitted with the given data $f(x) = a_0 + a_1 x + a_2 x^2$

Suppose that there are 3 given data points $(x_0,\ f(x_0))$, $(x_1,\ f(x_1))$, $(x_2,\ f(x_2))$.

If the given polynomial passes through the given data points then it must satisfy them, i.e.,

$$\left.\begin{aligned} f(x_0) &= a_0 + a_1 x_0 + a_2 x_0^2 \\ f(x_1) &= a_0 + a_1 x_1 + a_2 x_1^2 \\ f(x_2) &= a_0 + a_1 x_2 + a_2 x_2^2 \end{aligned}\right\} \Leftrightarrow \begin{pmatrix} 1 & x_0 & x_0^2 \\ 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f(x_1) \\ f(x_2) \end{pmatrix}$$

$$\begin{pmatrix} 1 & x_0 & x_0^2 \\ 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f(x_1) \\ f(x_2) \end{pmatrix}$$

This system of equations has a unique solution as

$$\text{Det}\begin{pmatrix} 1 & x_0 & x_0^2 \\ 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \end{pmatrix} = (x_0 - x_1)(x_1 - x_2)(x_2 - x_0) \neq 0 \text{ if } x_0,\ x_1,\ x_2 \text{ are distinct.}$$

In practice, it is observed that the above system of equations is *ill-conditioned*.

Whether they are solved with an elimination method or with a more efficient algorithm, the resulting coefficient can be highly inaccurate, in particular for, large n.

Therefore, we have some mathematical formats (interpolating formats) in which such calculation can be avoided.

Error in Interpolating Polynomials

Let $x_0,\ x_1,\ \dots,\ x_n$ be $(n+1)$ points and let x be a point belonging to the domain of a given function f.

Assume that $f \in C^{(n+1)}(I_x)$, where I_x is the smallest interval containing the nodes $x_0,\ x_1, \dots, x_n$ and x.

Then the interpolation error at the point x is given by

$$E_n(x) = f(x) - P_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x - x_0)(x - x_1)\cdots(x - x_n) \quad \text{where } \xi \in I_x.$$

Error in Interpolating Polynomials

$$E_n(x) = f(x) - P_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x-x_0)(x-x_1)\cdots(x-x_n) \quad \text{where } \xi \in I_x.$$

Proof : Note that the result is obviously true if x coincides with any of the interpolating nodes.

For simplicity, let us assume $w_{n+1}(x) = (x-x_0)(x-x_1)\cdots(x-x_n)$

Now, define for any $x \in I_x$, the function

$$G(t) = E_n(t) - \frac{w_{n+1}(t)E_n(x)}{w_{n+1}(x)}; \quad t \in I_x$$

Since $f \in C^{(n+1)}(I_x)$ and w_{n+1} is a polynomial, then $G \in C^{(n+1)}(I_x)$.

$$G(t) = E_n(t) - \frac{w_{n+1}(t)E_n(x)}{w_{n+1}(x)}; \;\; t \in I_x$$

$$E_n(x) = f(x) - P_n(x)$$

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

Note that $G(t)$ has $(n + 2)$ distinct zeros in I_x since

$$G(x_i) = E_n(x_i) - \frac{w_{n+1}(x_i)E_n(x)}{w_{n+1}(x)} = 0; \;\; i = 0, 1, 2, \dots, n.$$

$$G(x) = E_n(x) - \frac{w_{n+1}(x)E_n(x)}{w_{n+1}(x)} = 0$$

Then using **Rolle's theorem**, G' has atleast $(n + 1)$ distinct zeros.

By recursion it follows that $G^{(j)}$ admits at least $(n + 2) - j$ distinct zeros.

$\Rightarrow$ $G^{(n+1)}$ has atleast one zero, which we denote by ξ, i.e., $G^{(n+1)}(\xi) = 0$

$$G(t) = E_n(t) - \frac{w_{n+1}(t)E_n(x)}{w_{n+1}(x)}; \quad t \in I_x$$

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

$$G^{(n+1)}(\xi) = 0$$

$$\Longrightarrow G^{(n+1)}(t) = E_n^{(n+1)}(t) - \frac{w_{n+1}^{(n+1)}(t)E_n(x)}{w_{n+1}(x)}$$

Note that $E_n(t) = f(t) - P_n(t) \Longrightarrow E_n^{(n+1)}(t) = f^{(n+1)}(t)$ as $P_n^{(n+1)}(t) = 0$

$$w_{n+1}^{(n+1)}(t) = (n+1)! \quad \& \quad G^{(n+1)}(\xi) = 0$$

$$\Longrightarrow 0 = f^{(n+1)}(\xi) - \frac{(n+1)!\,E_n(x)}{w_{n+1}(x)} \Longrightarrow E_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}w_{n+1}(x)$$

Polynomial Interpolation

Polynomial interpolation is the method of determining a polynomial that fits a set of given points

Existence and Uniqueness

For $(n+1)$ data points there is one and only one polynomial of order $\leq n$ that passes through all the points.

Error in Interpolating Polynomials

$$E_n(x) = f(x) - P_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x - x_0)(x - x_1) \dots (x - x_n)$$