MA DM

30 January 2024 03:03 PM

BISECTION METHOD

This method is based on Intermediate Value Theorem.

Theorem

If $f \in C[a, b]$ and K is any number between $f(a)$ and $f(b)$, then there exists $c \in (a, b)$ such that $f(c) = K$.

Bisection Method

- Suppose that $f(x)$ is continuous on given interval $[a, b]$.
- The function f satisfies the property $f(a)f(b) < 0$ with $f(a) \neq 0$ and $f(b) \neq 0$.
- By Intermediate Value Theorem, there exists a number c such that $f(c) = 0$.

The Bisection method consists of the following steps:

Step 1: Given an initial interval $[a_0, b_0]$, set $n = 0$.

Step 2: Define $c_n = \dfrac{(a_n + b_n)}{2}$, the mid-point of interval $[a_n, b_n]$.

Step 3:
- If $f(c_n) = 0$, then $x^\star = c_n$ is the root.
- If $f(c_n) \neq 0$, then either

$$f(a_n)f(c_n) < 0 \quad \text{or} \quad f(a_n)f(c_n) > 0.$$

- If $f(a_n)f(c_n) < 0$, then $a_{n+1} = a_n$, $b_{n+1} = c_n$ and the root $x^\star \in [a_{n+1}, b_{n+1}]$.
- If $f(a_n)f(c_n) > 0$, then $f(b_n)f(c_n) < 0$, this implies $a_{n+1} = c_n$, $b_{n+1} = b_n$ and the root $x^\star \in [a_{n+1}, b_{n+1}]$.

Step 4: Repeat

Step 5: If the root is not achieved in **Step 3**, then, find the length of new reduced interval $[a_{n+1}, b_{n+1}]$. If the length of the interval $b_{n+1} - a_{n+1}$ is less than a recommended positive number ε, then take the mid-point of this interval $(x^\star = (b_{n+1} + a_{n+1})/2)$ as the approximate root of the equation $f(x) = 0$, otherwise go to Step 2.

Let $[a_0, b_0] = [a, b]$ be the initial interval with $f(a)f(b) < 0$. Define the approximate root as $c_n = (a_n + b_n)/2$. Then, there exists a root $x^\star \in [a, b]$ such that

$$|c_n - x^\star| \leq (\frac{1}{2})^n (b - a). \tag{2}$$

Moreover, to achieve the accuracy of $|c_n - x^\star| \leq \varepsilon$, it is sufficient to take

$$\frac{|b - a|}{2^n} \leq \varepsilon \quad \text{i.e.} \quad n \geq \frac{log(|b - a|) - log(\varepsilon)}{log 2}. \tag{3}$$

Our goal is to have $|c_n - x^\star| \leq \varepsilon$. This will be satisfied if

$$\begin{aligned} (\frac{1}{2})^n (b - a) \leq \varepsilon &\implies 2^n \geq \frac{b - a}{\varepsilon} \\ &\implies n log_{10} 2 \geq log_{10}(\frac{b - a}{\varepsilon}) \\ &\implies n \geq \frac{log(|b - a|) - log(\varepsilon)}{log 2}. \end{aligned} \tag{7}$$

SECANT METHOD

This method is based on Mean Value Theorem.

- Assume that two initial guesses to α are known. Let these be x_0 and x_1. They may occur on opposite side of α or on the same side of α.
- The two points $(x_0, f(x_0))$ and $(x_1, f(x_1))$ on the graph determine a straight line called a **secant line**.
- Equation of the secant line

$$y - f(x_1) = \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_1) \tag{2}$$

This method is based on Mean Value Theorem.

- The intersection of this line with x-axis is the next approximation to α. Let us denote it by x_2.
- The root x_2 or $(x_2, 0)$ lies on the line (2). So, we have

$$0 - f(x_1) = \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x_2 - x_1)$$

$$\implies \boxed{x_2 = x_1 - \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x_2 - x_1)} \tag{3}$$

- Having found x_2, we can drop x_0 and use x_1, x_2 as a new set of approximate value for α. This leads to an improved value x_3; and this process can be continued indefinitely.

- The general iteration formula for the secant method is

$$x_{n+1} = x_n - f(x_n)\frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})} \quad n \geq 1. \tag{4}$$

- The sequence of iterates does not need to converge to root of the function. In fact it might also diverge.Then, **why does one use secant method instead of bisection method, which gives the security of convergence?**
- It is called a two-point method, since two approximation values are needed to obtain an improved value.
- The Bisection method is also a two-point method, but the Secant method will almost always converge faster than bisection.

RATE OF CONVERGENCE

Definition 2

Let $\{x_n\}_{n\geq 1}$ be a sequence that converges to α. If positive constants λ and p exist with

$$\lim_{n\to\infty} \frac{|x_{n+1}-\alpha|}{|x_n-\alpha|^p} = \lambda,$$

then $\{x_n\}_{n\geq 1}$ is said to **converge to α of order p, with assymptotic error constant λ**. If $p = 1$, the method is called **linear**. If $p = 2$, the method is called **quadratic**.

- Can we find the exponent p such that

$$|x_{n+1}-\alpha| \approx |x_n-\alpha|^p? \qquad (5)$$

- **Answer:** $p = \frac{1}{2}(1+\sqrt{5}) \approx 1.62$. This is called **super linear convergence** $(1 < p < 2)$.

- The general iteration formula for the secant method is

$$x_{n+1} = x_n - f(x_n)\frac{x_n - x_{n-1}}{f(x_n)-f(x_{n-1})} \qquad n \geq 1. \qquad (6)$$

- Let $\varepsilon_n = x_n - \alpha$ and $\varepsilon_{n+1} = x_{n+1} - \alpha$. So, ε_n and ε_{n+1} denote the errors in the root at the nth and (n+1)th iterations.
- (6) $\Longrightarrow$

$\varepsilon_n - \varepsilon_{n-1}$

$$\varepsilon_{n+1} = \varepsilon_n - \frac{f(\alpha+\varepsilon_n)(\varepsilon_n - \varepsilon_{n-1})}{f(\alpha+\varepsilon_n)-f(\alpha+\varepsilon_{n-1})} \qquad (7)$$

- Let $M = \frac{f''(\alpha)}{2f'(\alpha)}$.
-

$$f(\alpha+\varepsilon_n) \approx \varepsilon_n f'(\alpha)\Big(1+\varepsilon_n M\Big).$$

$$\varepsilon_{n+1} \approx \frac{f''(\alpha)}{2f'(\alpha)}\varepsilon_{n-1}\varepsilon_n \qquad (8)$$

- (8) tells us that, as $n \to \infty$, the error tends to zero faster than a linear function but not quadratically!

error analysis

- Assume that f is twice differentiable and $f'(\alpha), f''(\alpha) \neq 0$. By Taylor's formula (with very small ε)

$$f(\alpha+\varepsilon) = f(\alpha) + \varepsilon f'(\alpha) + \frac{\varepsilon^2}{2}f''(\alpha) + R_2(\varepsilon).$$

Now, $f(\alpha) = 0$, and ε is very small, and $R_2(\varepsilon)$ is the remainder term. $R_2(\varepsilon)$ vanishes as $\varepsilon \to 0$ at a faster rate than ε^2. Therefore,

$$f(\alpha+\varepsilon) \approx \varepsilon f'(\alpha) + \frac{\varepsilon^2}{2}f''(\alpha).$$

By Mean Value Theorem,

$$f(\alpha) - f(x_n) = f'(c_n)(\alpha - x_n) \qquad (9)$$

where c_n lies between x_n and α. So, if $x_n \to \alpha$, then $c_n \approx x_n$ for large n, and we have

error estimate

$$\begin{aligned}\alpha - x_n &\approx -\frac{f(x_n)}{f'(c_n)}\\ &\approx -\frac{f(x_n)}{f'(x_n)}\\ &\approx -f(x_n)\frac{x_n - x_{n-1}}{f(x_n)-f(x_{n-1})}\\ &\approx x_{n+1} - x_n.\end{aligned}$$

Thus, $\boxed{\alpha - x_n \approx x_{n+1} - x_n}$.

NEWTON RAPHSON METHOD

- Consider the sample graph of $y = f(x)$. Let the estimate of the root α be x_0 (initial guess).
- To improve on this estimate, consider the straight line that is tangent to the graph at $(x_0, f(x_0))$.
- If x_0 is near α, the tangent line at x_0 cuts the x-axis at x_1, which is near to α.
- To find a formula for x_1, consider the equation of tangent to the graph of $y = f(x)$ at $(x_0, f(x_0))$. It is simply the graph of

$$y = f(x_0) + f'(x_0)(x - x_0).$$

- $(x_1, 0)$ lies on this line:

$$\Longrightarrow 0 = f(x_0) + f'(x_0)(x_1 - x_0)$$
$$\Longrightarrow x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}.$$

- Since x_1 is expected to be an improvement over x_0 as an estimate of α, this entire procedure can be repeated with x_1 as the initial guess.
- This leads to the new estimate

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

- Repeating this process, we obtain a sequence of numbers $x_1, x_2, \cdots$ that we hope will approach the root α. These numbers are called iterates, and they are defined recursively by the following general iteration formula:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \qquad n = 0, 1, 2, \cdots \qquad (1)$$

This is **Newton's method** for solving $f(x) = 0$.

Convergence Analysis

assumptions: i) f has continuous derivatives of order 2 $\forall x$ around α

ii) $f'(\alpha) \neq 0$

$\therefore$ $f'(\alpha) \neq 0$ $\forall x$ near α

- By Taylor's theorem,

$$f(\alpha) = f(x_n + \alpha - x_n)$$
$$= f(x_n) + (\alpha - x_n)f'(x_n) + \frac{(\alpha - x_n)^2}{2} f''(c_n) \quad (3)$$

where c_n is an unkown point between α and x_n.

- Note that $f(\alpha) = 0$. Using $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$ This implies

$$0 = f(x_n) + (\alpha - x_n)f'(x_n) + \frac{1}{2}(\alpha - x_n)^2 f''(c_n)$$
$$\implies 0 = \frac{f(x_n)}{f'(x_n)} + (\alpha - x_n) + \frac{1}{2}(\alpha - x_n)^2 \frac{f''(c_n)}{f'(x_n)}$$
$$\implies 0 = x_n - x_{n+1} + \alpha - x_n + (\alpha - x_n)^2 \frac{f''(c_n)}{f'(x_n)} \quad (4)$$

- $\boxed{\alpha - x_{n+1} = (\alpha - x_n)^2 \left[-\frac{f''(c_n)}{2f'(x_n)} \right]}$ Convergence analysis

$x_{n+1} \propto x_n^2$

error analysis

- Noting that $f(\alpha) = 0$, by Mean Value Theorem

$$f(x_n) = f(x_n) - f(\alpha) = f'(\xi_n)(x_n - \alpha).$$

Thus, error $\varepsilon_n = \alpha - x_n = -\frac{f(x_n)}{f'(x_n)}$ provided that x_n is close to α that $f'(x_n) \approx f'(\xi_n)$. This implies $\boxed{\alpha - x_n \approx x_{n+1} - x_n}$.

formula to estimate error
Not valid when $f'(\alpha) = 0$

Case when $f'(\alpha) = 0$

- The zero of the function f is said to be of multiplicity m if

$$f(x) = (x - \alpha)^m g(x)$$

for some continuous function g with $g(\alpha) \neq 0$, m is a positive integer.
- If we assume that f is sufficiently differentiable, an equivalent definition is that

$$f(\alpha) = f'(\alpha) = \cdots = f^{(m-1)}(\alpha) = 0, f^{(m)}(\alpha) \neq 0.$$

- A zero of multiplicity 1 is called a simple root or a simple zero.

- One method of handling the problem of multiple roots of a function f is to define

$$\mu(x) = \frac{f(x)}{f'(x)}.$$

- If α is a zero of f of multiplicity m. Then, one can write

$$f(x) = (x - \alpha)^m g(x), \quad g(\alpha) \neq 0.$$

- This implies

$$\mu(x) = \frac{(x - \alpha)^m g(x)}{m(x - \alpha)^{m-1} g(x) + (x - \alpha)^m g'(x)}$$
$$= (x - \alpha) \frac{g(x)}{mg(x) + (x - \alpha)g'(x)}$$

also has a zero at α. However, $g(\alpha) \neq 0$.

- Thus, $\mu'(\alpha) = \frac{1}{m} \neq 0$, hence α is called a simple zero of μ.
- Newton's method can be applied to $\mu(x)$ to give

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

- This is called Newton's modified method. This has quadratic convergence regardless of multiplicity of the zeros of f.
- For a simple zero, the original Newton's method requires significantly low computations.

POLYNOMIAL INTERPOLATION

Problem Description

Given, $(n+1)$ points, say (x_i, y_i) where $i = 0, 1, 2, \cdots, n$ with distinct x_i, not necessarily sorted, we want to find a polynomial of degree n,

$$P_n(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$$

such that it interpolates these points, i.e.,

$$P_n(x_i) = y_i, \quad i = 0, 1, 2, \cdots, n.$$

Our goal: is to determine the coefficients $a_n, a_{n-1}, \cdots, a_1, a_0$.
Note: The total number of data points is 1 larger than the degree of the polynomial.

relative error $= \frac{f(x) - P_2(x)}{f(x)}$
$E_r(P_2(x))$

For the general case with $(n+1)$ points, we have

$$P_n(x_i) = y_i, \quad i = 0, 1, 2, \cdots, n.$$

We will have $(n+1)$ equations and $(n+1)$ unknowns:

$$\begin{aligned}
P_n(x_0) = y_0 \quad &: \quad x_0^n a_n + x_0^{n-1} a_{n-1} + \cdots + x_0 a_1 + a_0 = y_0 \\
P_n(x_1) = y_1 \quad &: \quad x_1^n a_n + x_1^{n-1} a_{n-1} + \cdots + x_1 a_1 + a_0 = y_1 \\
&\vdots \\
P_n(x_n) = y_n \quad &: \quad x_n^n a_n + x_n^{n-1} a_n + \cdots + x_n a_1 + a_0 = y_n.
\end{aligned}$$

Putting this in matrix-vector form

$$\begin{pmatrix} x_0^n & x_0^{n-1} & \cdots x_0 & 1 \\ x_1^n & x_1^{n-1} & \cdots x_1 & 1 \\ \vdots & \vdots & \vdots & \\ x_n^n & x_n^{n-1} & \cdots x_n & 1 \end{pmatrix} \begin{pmatrix} a_n \\ a_{n-1} \\ \vdots \\ a_0 \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}$$

$$\mathbf{X}\vec{a} = \vec{y}$$

- X : $(n+1) \times (n+1)$ matrix, given (Van der Monde matrix)
- $\vec{a}$: unknown vector, with length $(n+1)$
- $\vec{y}$: given vector, with length $(n+1)$

Theorem 1

If x_i's are distinct, then $\mathbf{X}$ is invertible, therefore $\vec{a}$ has a unique solution.

In other words,

Given $n+1$ distinct points $x_0, x_1, \cdots, x_n$ and $n+1$ ordinates $y_0, \cdots, y_n$, there is a polynomial $p(x)$ of degree $\leq n$ that interpolates y_i at x_i, $i = 0, 1, \cdots, n$. This polynomial $p(x)$ is unique among the set of all polynomials of degree at most n.

Recall the Vandermonde matrix $\mathbf{X}$ is given by

$$V_n(x) = \det \begin{pmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ \vdots & & & & \\ 1 & x_{n-1} & x_{n-1}^2 & \cdots & x_{n-1}^n \\ 1 & x & x^2 & \cdots & x^n \end{pmatrix} \tag{1}$$

turns out to be too complicated for larger n.

- One can show that $V_n(x)$ is a polynomial of degree n, and that its roots are $x_0, \cdots, x_{n-1}$. We can obtain the formula
$$V_n(x) = (x - x_0) \cdots (x - x_{n-1}) V_{n-1}(x_{n-1}).$$
- Expand the last row of $V_n(x)$ by minors to show that $V_n(x)$ is a polynomial of degree n and to find the coefficient of the term x^n.
- One can show that
$$\det(X) = V_n(x_n) = \prod_{0 \leq i \leq j \leq n} (x_i - x_j)$$

Lagrange Interpolation Method.

Given points: $x_0, x_1, \cdots, x_n$
Define the cardinal functions $l_0, l_1, \cdots, l_n :\in \mathcal{P}^n$, satisfying the properties

$$l_i(x_j) = \delta_{ij} = \begin{cases} 1, i = j \\ 0, i \neq j, \end{cases} \quad i = 0, 1, \cdots, n.$$

Here, δ_{ij} is called the Kronecker's delta.
Locally supported in discrete sense. The cardinal functions $l_i(x)$ can be written as

$$\begin{aligned}
l_i(x) &= \prod_{j=0, j\neq i}^{n} \left(\frac{x - x_j}{x_i - x_j} \right) \\
&= \frac{x - x_0}{x_i - x_0} \frac{x - x_1}{x_i - x_1} \cdots \frac{x - x_{i+1}}{x_i - x_{i+1}} \cdots \frac{x - x_n}{x_i - x_n}.
\end{aligned}$$

Lagrange form of the interpolation polynomial

Lagrange form of the interpolation polynomial can be simply expressed as

$$P_n(x) = \sum_{i=0}^{n} l_i(x) y_i.$$

It is easy to check the interpolating property:

$$P_n(x_j) = \sum_{i=0}^{n} l_i(x) y_i = y_j, \quad \text{for every } j.$$

Newton's Dividend Differences.

Given $(n+1)$ data set, we will describe an algorithm in a recursive form.

Main idea: Given $P_k(x)$ that interpolates $k+1$ data points $\{x_i, y_i\}$, $i = 0, 1, 2, \cdots, k$, compute $P_{k+1}(x)$ that interpolates one extra point, $\{x_{k+1}, y_{k+1}\}$, by using P_k and adding an extra term.

$$P_0(x) = y_0$$
$$P_1(x) = P_0(x) + a_1(x - x_0)$$
$$P_2(x) = P_1(x) + a_2(x - x_0)(x - x_1)$$
$$P_3(x) = P_2(x) + a_3(x - x_0)(x - x_1)(x - x_2)$$
$$\vdots$$

$$\begin{aligned}
P_n(x) = a_0 &+ a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \cdots \\
&+ a_n(x - x_0)(x - x_1) \cdots (x - x_{n-1}).
\end{aligned}$$

Substitute (x_i, y_i) in $P_i(x)$ to find a_i

ERRORS IN INTERPOLATION

Given a function $f(x)$ on $x \in [a,b]$, and a set of distinct points $x_i \in [a,b]$, $i = 0, 1, \cdots, n$. Let $P_n(x) \in \mathcal{P}_n$ s.t.

$$P_n(x_i) = f(x_i), \quad i = 0, 1, 2, \cdots, n.$$

Error function: $e(x) = f(x) - P_n(x)$, $x \in [a,b]$.

Theorem 1

There exists some value $\xi \in (a,b)$, such that

$$e(x) = \frac{1}{(n+1)!} f^{(n+1)}(\xi) \prod_{i=0}^{n} (x - x_i), \quad \text{for all} \quad x \in [a,b]. \quad (1)$$

$$W(x) = \prod_{i=0}^{n} (x - x_i) \in \mathcal{P}_{n+1},$$

Fix an x such that $a \le x \le b$ and $x \ne x_i$ for any i. We define a constant

$$c = \frac{f(x) - P_n(x)}{W(x)},$$

and another function

$$\varphi(y) = f(y) - P_n(y) - cW(y).$$

We find all the zeros for $\varphi(y)$. We see that x_i's are zeros since

$$\varphi(x_i) = f(x_i) - P_n(x_i) - cW(x_i) = 0.$$

Also, x is a zero because

$$\varphi(x) = f(x) - P_n(x) - cW(x) = 0.$$

Here goes our deduction:

$\varphi(x)$ has atleast $(n+2)$ zeros on $[a,b]$.
$\varphi'(x)$ has atleast $(n+1)$ zeros on $[a,b]$.
$\varphi''(x)$ has atleast n zeros on $[a,b]$.
$\cdots$
$\varphi^{(n+1)}(x)$ has atleast 1 zero on $[a,b]$.
$\boxed{\text{Call it } \xi \ \ s.t. \ \ \varphi^{(n+1)}(\xi) = 0.}$ So, we have

$$\varphi^{(n+1)}(\xi) = f^{(n+1)}(\xi) - 0 - cW^{(n+1)}(\xi) = 0.$$

Recall $W^{n+1} = (n+1)!$, we have, for every y,

$$f^{(n+1)}(\xi) = cW^{(n+1)}(\xi) = \frac{f(y) - P_n(y)}{W(y)}(n+1)!$$

SPLINES

x	t_0	t_1	$\cdots t_n$
y	y_0	y_1	$\cdots y_n$

Find a function $\mathcal{S}(x)$ which interpolates the point $(t_i, y_i)_{i=0}^n$. The set $t_0 < t_1 < \cdots < t_n$ are called knots. Note that they need to be ordered. $\mathcal{S}(x)$ consists of piecewise polynomial

$$\mathcal{S}(x) = \begin{cases} \mathcal{S}_0(x), & t_0 \le x \le t_1 \\ \mathcal{S}_1(x), & t_1 \le x \le t_2 \\ \vdots \\ \mathcal{S}_{n-1}(x), & t_{n-1} \le x \le t_n. \end{cases} \quad (1)$$

$\mathcal{S}(x)$ is called a **spline of degree k**, if

- $\mathcal{S}_i(x)$ is a polynomial of degree k;
- $\mathcal{S}(x)$ is $(k-1)$ times continuously differentiable, i.e., for $i = 1, 2, \cdots, k-1$ we have

$$\mathcal{S}_{i-1}(t_i) = \mathcal{S}_i(t_i),$$
$$\mathcal{S}'_{i-1}(t_i) = \mathcal{S}'_i(t_i),$$
$$\vdots$$
$$\mathcal{S}^{(k-1)}_{i-1}(t_i) = \mathcal{S}^{(k-1)}_i(t_i).$$

Linear Spline.

$n = 1$: piecewise linear interpolation, i.e., straight line between 2 neighboring points.
Requirements:

$$\mathcal{S}_0(t_0) = y_0 \quad (4)$$
$$\mathcal{S}_{i-1}(t_i) = \mathcal{S}_i(t_i) = y_i, \quad i = 1, 2, \cdots, n-1 \quad (5)$$
$$\mathcal{S}_{n-1}(t_n) = y_n. \quad (6)$$

Easy to find: write the equation for a line through two points: (t_i, y_i) and (t_{i+1}, y_{i+1})

$$\mathcal{S}_i(x) = y_i + \frac{y_{i+1} - y_i}{t_{i+1} - t_i}(x - t_i), \quad i = 0, 1, \cdots, n-1. \quad (7)$$

Accuracy Theorem for linear spline

- Assume $t_0 < t_1 < \cdots < t_n$, and let $h_i = t_{i+1} - t_i$, $h = \max_i h_i$.
- $f(x)$: given function, $\mathcal{S}(x)$: a linear spline
- $\mathcal{S}(t_i) = f(t_i)$, $i = 0, 1, \cdots, n$.

We have the following, for $x \in [t_0, t_n]$.

(a) If f'' exists and is continuous, then,

$$|f(x) - \mathcal{S}(x)| \le \max \left\{ \frac{1}{8} h_i^2, \max_{t_i \le x \le t_{i+1}} |f''(x)| \right\} \le \frac{1}{8} h^2 \max_x |f''(x)|.$$

(b) If f' exists and is continuous, then

$$|f(x) - \mathcal{S}(x)| \le \max_i \left\{ \frac{1}{2} h, \max_{t_i \le x \le t_{i+1}} |f'(x)| \right\} \le \frac{1}{2} h \max_x |f'(x)|.$$

To minimize the error, it is obvious that one should add more knots where the function has large first or second derivative.

Natural Cubic Splines

Equations we have:

equation		number
$\mathcal{S}_i(t_i) = y_i,$	$i = 0, 1, \cdots, n-1$	n
$\mathcal{S}_{i+1}(t_{i+1}) = y_{i+1},$	$i = 0, 1, \cdots, n-1$	n
$\mathcal{S}'_i(t_{i+1}) = \mathcal{S}'_{i+1}(t_{i+1}),$	$i = 0, 1, \cdots, n-2$	$n-1$
$\mathcal{S}''_i(t_{i+1}) = \mathcal{S}''_{i+1}(t_{i+1}),$	$i = 0, 1, \cdots, n-2$	$n-1$
$\underbrace{\mathcal{S}''_0(t_0) = 0, \ \mathcal{S}''_{n-1}(t_n) = 0,}_{\text{Natural}}$		2

Natural Cubic Splines Algo.

$z_i = S''(t_i) \quad i = 1, 2, \cdots n \qquad z_0 = 0 \qquad z_n = 0$

tridiagonal
symmetrical
diagonal dominent

$h_i = t_{i+1} - t_i$

$$S''_i(x) = \frac{z_{i+1}}{h_i}(x - t_i) - \frac{z_i}{h_i}(x - t_{i+1})$$

$$S_i'(x) = \frac{z_{i+1}}{2h_i}(x-t_i)^2 - \frac{z_i}{2h_i}(x-t_{i+1})^2 + C_i - D_i$$

$$S_i(x) = \frac{z_{i+1}}{6h_i}(x-t_i)^3 - \frac{z_i}{6h_i}(x-t_{i+1})^3 + c_i(x-t_i) - D_i(x-t_{i+1})$$

Substitute $S_i(t_i) = y_i$

$$D_i = \frac{y_i}{h_i} - \frac{h_i}{6} z_i$$

Substitute $S_i(t_{i+1}) = y_{i+1}$

$$c_i = \frac{y_{i+1}}{h_i} - \frac{h_i}{6} z_{i+1}$$

Continuity of S_i' $\quad S_{i-1}'(t_i) = S_i'(t_i) \quad i = 1, 2 \ldots n-1$

$$\therefore \; h_{i-1} z_{i-1} + 2(h_{i-1} + h_i) z_i + h_i z_{i+1} = 6(b_i - b_{i-1})$$

$$z_0 = z_n = 0$$

where $b_i = \dfrac{y_{i+1} - y_i}{h_i}$

$$H\vec{z} = \vec{b}$$

$$\vec{z} = \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_{n-1} \end{pmatrix} \quad \vec{b} = \begin{pmatrix} 6(b_1 - b_0) \\ 6(b_2 - b_1) \\ \vdots \\ 6(b_{n-1} - b_n) \end{pmatrix} \quad H = \begin{pmatrix} 2(h_0+h_1) & h_1 & 0 & 0 \\ h_1 & 2(h_1+h_2) & h_2 & \\ 0 \quad 0 & h_2 & \ddots & \ddots \\ & & \ddots & h_{n-2} \\ & & h_{n-2} & 2(h_{n-1}+h_n) \end{pmatrix}$$

$$2|h_{i-1} + h_i| > |h_i| + |h_{i-1}|$$

Newton Forward Diff Formulas

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

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

$$f[x_i, x_{i+1}, x_{i+2}] = f[x_{i+1}, x_{i+2}] - f[x_i, x_{i+1}]$$

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

$$f[x_0, \dots x_K] = \frac{f[x_1 \dots x_K] - f[x_0 \cdots x_{K-1}]}{x_K - x_0} = a_K$$

3

Let α be a simple root of $f(x)=0$.

3(a) By Taylor's theorem, $(\epsilon_n = \alpha - x_n)$

$$f(\alpha) = f(x_n + \alpha - x_n) = f(x_n + \epsilon_n)$$
$$= f(x_n) + \epsilon_n f'(x_n) + \frac{\epsilon_n^2}{2!} f''(c_n),$$

where c_n is an unknown point between α and x_n.

$\because$ α is the actual root of $f(x)=0$, $f(\alpha)=0$.

Therefore
$$0 = f(x_n) + \epsilon_n f'(x_n) + \frac{\epsilon_n^2}{2} f''(c_n)$$

Since $f'(\alpha) \neq 0$, $f'(x) \neq 0$ in sufficiently small neighbourhood of α.

Therefore, ~~Assuming that~~ $f'(x_n) \neq 0$ (at for x_n's close to the root) in Newton's method, we have

$$0 = \frac{f(x_n)}{f'(x_n)} + \epsilon_n + \epsilon_n^2 \frac{f''(c_n)}{2f'(x_n)} \quad \text{--- (1)}$$

By Newton's method:
$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \Rightarrow \frac{f(x_n)}{f'(x_n)} = x_n - x_{n+1}$$
$$= (\alpha - x_{n+1}) - (\alpha - x_n)$$
$$= \epsilon_{n+1} - \epsilon_n.$$

Using this (1) becomes
$$0 = \epsilon_{n+1} + \epsilon_n^2 \frac{f''(c_n)}{2f'(x_n)}$$

or $\epsilon_{n+1} = \epsilon_n^2 \left[-\frac{f''(c_n)}{2f'(x_n)}\right]$ or $\frac{|\epsilon_{n+1}|}{|\epsilon_n|^2} = \left|\frac{f''(c_n)}{2f'(x_n)}\right|$

as $n \to \infty$, $x_n \to \alpha$ and hence $c_n \to \alpha$. Therefore,

$$\lim_{n\to\infty} \frac{|\epsilon_{n+1}|}{|\epsilon_n|^2} = \left|\frac{f''(\alpha)}{2f'(\alpha)}\right|.$$

Hence, for Newton's method, the order of convergence $p=2$ and the asymptotic error constant $C = \left|\frac{f''(\alpha)}{2f'(\alpha)}\right|$.

4

3(b) The secant method gives
$$x_{n+1} = x_n - f(x_n)\frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}, \quad n = 1, 2, 3, \dots$$

Let $\epsilon_n = \alpha - x_n$. Hence the secant method gives
$$\alpha - x_{n+1} = \alpha - \epsilon_n - f(\alpha - \epsilon_n) \frac{\alpha - \epsilon_n - (\alpha - \epsilon_{n-1})}{f(\alpha - \epsilon_n) - f(\alpha - \epsilon_{n-1})}$$

$$\Rightarrow \epsilon_{n+1} = \epsilon_n - f(\alpha - \epsilon_n)\frac{\epsilon_n - \epsilon_{n-1}}{f(\alpha - \epsilon_n) - f(\alpha - \epsilon_{n-1})} \quad \text{--- (1)}$$

For any small number ϵ, Taylor's formula gives
$$f(\alpha + \epsilon) = f(\alpha) + \epsilon f'(\alpha) + \frac{\epsilon^2}{2} f''(\alpha) + R_2(\epsilon)$$

where $R_2(\epsilon)$ is the remainder term that vanishes at a faster rate than ϵ^2 as $\epsilon \to 0$.

If α is the root of $f(x)=0$, then $f(\alpha)=0$, therefore
$$f(\alpha + \epsilon) \approx \epsilon f'(\alpha) + \frac{\epsilon^2}{2} f''(\alpha) = \epsilon f'(\alpha)(1 + \epsilon M)$$

where $M = \frac{f''(\alpha)}{2f'(\alpha)}$.

$$\therefore \; f(\alpha - \epsilon_n) \approx -\epsilon_n f'(\alpha)(1 - \epsilon_n M) \quad \text{--- (2)}$$

and $f(\alpha - \epsilon_n) - f(\alpha - \epsilon_{n-1}) \approx -\epsilon_n f'(\alpha)(1 - \epsilon_n M) + \epsilon_{n-1} f'(\alpha)(1 - \epsilon_{n-1} M)$

$$= f'(\alpha)\left[-\epsilon_n + \epsilon_{n-1} + (\epsilon_n^2 - \epsilon_{n-1}^2)M\right]$$

$$= -(\epsilon_n - \epsilon_{n-1})\, f'(\alpha)\left[1 - (\epsilon_n + \epsilon_{n-1})M\right] \quad \text{— (3)}$$

From (1), (2) & (3),

$$\epsilon_{n+1} = \epsilon_n - \frac{\left[\epsilon_n \cancel{f'(\alpha)}(1-\epsilon_n M)\right]\cancel{(\epsilon_n - \epsilon_{n-1})}}{\left[-\cancel{(\epsilon_n - \epsilon_{n-1})}\cancel{f'(\alpha)}\{1 - (\epsilon_n + \epsilon_{n-1})M\}\right]} = \epsilon_n - \frac{\epsilon_n(1-\epsilon_n M)}{1-(\epsilon_n+\epsilon_{n-1})M}$$

$$= \frac{\cancel{\epsilon_n} - (\cancel{\epsilon_n^2} + \epsilon_n\epsilon_{n-1})M - \cancel{\epsilon_n} + \cancel{\epsilon_n^2 M}}{1-(\epsilon_n+\epsilon_{n-1})M} = -\frac{\epsilon_{n-1}\epsilon_n M}{1-(\epsilon_n+\epsilon_{n-1})M}$$

$$\approx -\epsilon_{n-1}\epsilon_n M \quad \text{— (4)}$$

For finding the order of convergence,

let $|\epsilon_{n+1}| \approx C|\epsilon_n|^P$. — (5) as $n \to \infty$. $\left[\text{equivalent to } \lim_{n\to\infty} \frac{|\epsilon_{n+1}|}{|\epsilon_n|^P} = C\right]$

Hence (4) implies that

$$C|\epsilon_n|^P \approx |\epsilon_{n-1}||\epsilon_n||M|$$

$$\Rightarrow |\epsilon_n|^{P-1} \approx \frac{|M|}{C}|\epsilon_{n-1}|$$

$$\Rightarrow |\epsilon_n| \approx \left(\frac{|M|}{C}\right)^{\frac{1}{P-1}} |\epsilon_{n-1}|^{\frac{1}{P-1}} \quad \text{— (6)}$$

Comparing (5) & (6),

$$P = \frac{1}{P-1} \Rightarrow P^2 - P - 1 = 0 \Rightarrow P = \frac{1 \pm \sqrt{5}}{2}$$

but since $P > 0$, $\boxed{P = \frac{1+\sqrt{5}}{2} \approx 1.618}$

and $C = \left(\frac{|M|}{C}\right)^{\frac{1}{P-1}} \Rightarrow C^{P-1} = \frac{|M|}{C} \Rightarrow C^P = |M|$

$$\Rightarrow C = |M|^{1/P} \Rightarrow C = \left|\frac{f''(\alpha)}{2f'(\alpha)}\right|^{\frac{2}{1+\sqrt{5}}} \Rightarrow \boxed{C = \left|\frac{f''(\alpha)}{2f'(\alpha)}\right|^{\frac{\sqrt{5}-1}{2}}}$$

0	$\boxed{1}$			
1	0	$\boxed{-1}$		
2/3	0.5	-1.5	$\boxed{-0.75}$	
1/3	0.8660	-1.0981	-0.6029	$\boxed{0.4413}$

these are the coeffs of Newtons method.

So we have

$a_0 = 1$, $a_1 = -1$, $a_2 = -0.75$, $a_3 = 0.4413$.

Then,

$$P_3(x) = \boxed{1} + \boxed{-1}x + \boxed{-0.75}\,x(x-1) + 0.4413\,x(x-1)(x-\tfrac{2}{3})$$

Statement of the Weierstrass Approximation Theorem

Let f: [a,b] → **R** be a real valued continuous function. Then we can find polynomials $p_n(x)$ such that every p_n converges uniformly to x on [a,b].

In other words, if f is a continuous real-valued function on [a, b] and if any ε > 0 is given, then there exist a polynomial P on [a, b] such that |f(x) – P(x)| < ε, for every x in [a, b].

The essence of this theorem is that no matter how much complicated the function f is given, we can always find a polynomial that is as close to f as we desire.