ROOT FINDING

Bisection Search

Root

$$X_r = \frac{(X_h + X_l)}{2}$$

Maximum Error

$$E_{max} = \frac{(X_h - X_l)}{2}$$

Approximate Error

$$E_{app} = \frac{(X_r^{new} - X_r^{old})}{X_r^{new}}$$

Relative Error

$$E_{rel} = \frac{(X_r^{original} - X_r)}{X_r^{original}}$$

E_{app} After "n" iterations

$$E_a^n = \frac{({X_u}^0 - {X_l}^0)}{2^n}$$

"N" knowing E_{app}

$$N = \frac{\log\left(\frac{\Delta X^0}{E_{max}}\right)}{\log 2}$$

Example: f(x) = $X^3 - 5$

$\boldsymbol{X_l}$	$\boldsymbol{X_r = \frac{(X_h + X_l)}{2}}$	$\boldsymbol{X_h}$	$\boldsymbol{f(X_l)}$	$\boldsymbol{f(X_r)}$	$\boldsymbol{f(X_h)}$	$\boldsymbol{E_{max} = \frac{(X_h - X_l)}{2}}$
1	1.5	2	-4	-1.625	3	0.5
1.5	1.75	2	-1.625	0.35938	3	0.25
1.5	1.625	1.75	-1.625	-0.70898	0.359375	0.125
1.625	**1.6875**	1.75	-0.708984	-0.19458	0.359375	0.0625

Practice Problem

The function $f\ (x) = x^2 - 12x + 32$ has a root between 1 and 5. Execute the first three steps of Bisection.

F(x) = $x^2 - 12x + 32$

$\boldsymbol{X_l}$	$\boldsymbol{X_r = \frac{(X_h + X_l)}{2}}$	$\boldsymbol{X_h}$	$\boldsymbol{f(X_l)}$	$\boldsymbol{f(X_r)}$	$\boldsymbol{f(X_h)}$	$\boldsymbol{E_{max} = \frac{(X_h - X_l)}{2}}$
1	3	5	21	5	-3	2
3	**4**	5	5	0	-3	1

False Position Search

Root

$$X_r = X_u - \frac{f(X_u)(X_l - X_u)}{f(X_l) - f(X_u)}$$

Maximum Error

$$E_{max} = \frac{(X_h - X_l)}{2}$$

Approximate Error

$$E_{app} = \frac{(X_r^{new} - X_r^{old})}{X_r^{new}}$$

Relative Error

$$E_{rel} = \frac{(X_r^{original} - X_r)}{X_r^{original}}$$

Example: f(x) = $X^3 - 5$

X_l	X_r	X_h	$f(X_l)$	$f(X_r)$	$f(X_h)$	$E_{max} = \frac{(X_h - X_l)}{2}$
1	1.571428	2	-4	-1.119537	3	0.5
1.571428	1.68789	2	-1.119537	-0.191248	3	0.214286
1.68789	**1.70659**	2	-0.1912476	-0.02960	3	0.156055

Practice Problem

The function $f\ (x) = x^{2} - 12x + 32$ has a root between 6 and 9. Execute the first three steps of Regula Falsi.

X_l	X_r	X_h	$f(X_l)$	$f(X_r)$	$f(X_h)$	$E_{max} = \frac{(X_h - X_l)}{2}$
6	10.6667	9	-4	17.778	5	1.5
6	6.85714	10.6667	-4	-3.2653	17.778	2.333
6.85714	7.556848	10.6667	-3.2653	-1.57622	17.778	1.90475
7.55685	7.810118	10.6667	-1.57622	-0.72347	17.778	1.554925
7.810118	7.9216965	10.6667	-0.72347	-0.30708	17.778	1.428291

Newton – Raphson Method

Next Position X_{i+1}

$$X_{i+1} = X_i - \frac{f(X_i)}{f^1(X_i)}$$

Approximate Error

$$E_{app} = |X_{i+1} - X_i|$$

(Or)

$$E_{app} = \frac{(X_i^{new} - X_i^{old})}{X_i^{new}}$$

Example: f(x) $= X^3 - 9$

$f^1(x) = 3X^2$

i	X_{i-1}	X_i	$f(X_{i-1})$	$f^1(X_{i-1})$	E_{max}
1	8	5.3802	503	192	
2	5.3802	3.6904	146.73896	86.8396	1.6898
3	3.6904	**2.680516**	41.26097	40.857156	1.0098

Practice Problem

The function $f(x) = x^2 - 12x + 32$ has a root between 10 and 12. Execute the first three steps of Newton's searches by hand.

F(x) = $x^2 - 12x + 32$

F\`(x) =2x-12

i	X_{i-1}	X_i	$f(X_{i-1})$	$f^1(X_{i-1})$	E_{max}
1	10	8.5	12	8	1.5
2	8.5	8.05	2.25	5	0.45
3	8.05	**8.00006**	0.2025	4.1	0.05

Secant Search

Next Position X_{i+1}

$$X_{i+1} = X_i - \frac{(X_i - X_{i-1})}{f(X_i) - f(X_{i-1})} f(X_i)$$

Approximate Error

$$E_{app} = |X_{i+1} - X_i|$$

(Or)

$$E_{app} = \frac{(X_i^{new} - X_i^{old})}{X_i^{new}}$$

Example: f(x) = $X^3 - 9$

i	X_{i-1}	X_i	X_{i+1}	$f(X_{i-1})$	$f(X_i)$	E_{max}
1	3	5	2.6327	18	116	2.3673
2	5	2.6327	2.42763	116	9.24753	0.20507
3	2.6327	2.42763	2.15836	9.24753	5.306973	0.26927
4	2.42763	2.15836	**2.091564**	5.306973	1.0548066	0.06680

Practice Problem

The function $f(x) = x^2 - 12x + 32$ has a root between 2 and 6. Execute the first three steps of secant.

i	X_{i-1}	X_i	X_{i+1}	$f(X_{i-1})$	$f(X_i)$	E_{max}
1	2	6	5	12	-4	1
2	6	5	2	-4	-3	3
3	5	2	4.4	-3	12	2.24
4	2	4.4	**4.276**	12	-1.44	0.124

Function (Fzero)

Must be given either a range of interest or a ballpark value
fzero (f, [0 4])   % find root between 0 and 4
fzero (f, 4)   % find root in surrounding of 4

OPTIMIZATION

Golden Section Search

$$\varphi = 1.6180339887$$

$$d = (\varphi - 1)(X_u - X_l)$$

$$X_1 = X_l + d$$

$$X_2 = X_u - d$$

Or

$$X_1 = 0.38196601125\, X_l + 0.6180339887\, X_u$$

$$X_2 = 0.38196601125\, X_u + 0.6180339887\, X_l$$

$$\boldsymbol{E_{max}} = \frac{({X_u}^0 - {X_l}^0)(\varphi - 1)^N}{2} \quad \text{Assuming midpoint}$$

***Change the function to find the minimum**

Example: f(x) = $X^2 - 2X - 5$

X_L	X_2	X_1	X_u	$f(X_L)$	$f(X_2)$	$f(X_1)$	$f(X_u)$	E_{max}
0	1.1459	1.8541	3	-5	-5.9787	-5.2705	-2	1.5
0	0.7082	1.1459	1.8541	-5	-5.9149	-5.9787	-5.2705	0.92705
0.7082	1.1459	1.4164	1.8541	-5.9149	-5.9787	-5.8266	-5.2705	0.57295
0.7082	0.9787	1.1458	**1.4164**					**0.354**

$$\boldsymbol{Best\ Guess} = \frac{1.4164 + 0.7082}{2}$$

To get the y max value plug the value back into the original function (i.e. ymax = f (Best Guess))

Practice Problem

The function - sin(x) *exp (-0.5x) has a minimum between 0 and 2. Perform the first three steps of a golden section search by hand.

F(x) = - Sin(x) *exp (-0.5x)

X_L	X_2	X_1	X_u	$f(X_L)$	$f(X_2)$	$f(X_1)$	$f(X_u)$	E_{max}
0	0.763932	1.23607	2	0	-0.4721	-0.50908	-0.3345	1
0.763932	1.236067	1.52786	2	-0.4721	-0.5091	-0.46540	-0.3345	0.6180
0.763932	1.05572	1.23606	**1.52786**	-0.4721	-0.5133	-0.50909	-0.4654	0.3820

Changing Optimization problem into a Root finding Problem

- Take the Derivative of the Original Function (i.e. Fp.)
- Then Using the function fzero to find the maximum (i.e. fzero(Fp,0))

To get the y max value plug the value back into the original function (i.e. ymax = f(X))

Function (fminbnd)

***Change the function to find the minimum**

Must be given a function and their lower limit and Upper limit
Xmax = fminbnd (f, X_L , X_U)   % find max value of x between lower and upper limit

To get the y max value plug the value back into the original function (i.e. ymax = f (Xmax))

SYSTEM OF LINEAR EQUATIONS

Naive Gaussian elimination Method

Pivoting

Step 1: Make necessary row switching to make pivot not equal to zero

If any of the elements below the pivot position have a greater magnitude than the pivot element rows are switched so as to make the magnitude of the pivot element as large as possible.

Naive Gaussian elimination

Step 2: Combine the matrix A and B to get the augmented matrix C = [(A| B)]

Step 3: Row = Row – $\left(\frac{\text{Eliment Below Pivot}}{\text{Pivot Element}}\right)$ Pivot Row

Step 4: After several steps of using the above formula solve the equation from bottom to top

Gauss – Jordan elimination Method

Step 5: After all the steps use back the formula from bottom to top such that the matrix is diagonal

Practice Problem

Use Gaussian eliminating (with pivoting) to solve the following series of equations by hand.

After the coefficient matrix has been reduced to upper triangular form, complete the job in two ways.
i) By back substitution
ii) By back elimination (Gauss Jordan)

Naive Gaussian elimination

$$\begin{bmatrix} 2 & 3 & -1 \\ 5 & -2 & 2 \\ 3 & 1 & -3 \end{bmatrix} \begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \begin{Bmatrix} 16 \\ 6 \\ 2 \end{Bmatrix}$$

$$C = \begin{bmatrix} 2 & 3 & -1 & 16 \\ 5 & -2 & 2 & 6 \\ 3 & 1 & -3 & 2 \end{bmatrix}$$

$$\text{R2} = \text{R2} - \left(\frac{5}{2}\right) \text{R1}$$

$$C = \begin{bmatrix} 2 & 3 & -1 & 16 \\ 0 & -9.5 & 4.5 & -34 \\ 3 & 1 & -3 & 2 \end{bmatrix}$$

$$R3 = R3 - \left(\frac{3}{2}\right) R1$$

$$C = \begin{bmatrix} 2 & 3 & -1 & 16 \\ 0 & -9.5 & 4.5 & -34 \\ 0 & -3.5 & -1.5 & -22 \end{bmatrix}$$

$$R3 = R3 - \left(\frac{3.5}{9.5}\right) R2$$

$$C = \begin{bmatrix} 2 & 3 & -1 & 16 \\ 0 & -9.5 & 4.5 & -34 \\ 0 & 0 & -3.16 & -9.47 \end{bmatrix}$$

Gauss – Jordan elimination Method

$$R2 = R2 - \left(\frac{4.5}{3.16}\right) R3$$

$$C = \begin{bmatrix} 2 & 3 & -1 & 16 \\ 0 & -9.5 & 4.5 & -34 \\ 0 & 0 & -3.16 & -9.47 \end{bmatrix}$$

$$R1 = R1 + \left(\frac{1}{3.16}\right) R3$$

$$C = \begin{bmatrix} 2 & 3 & -1 & 16 \\ 0 & -9.5 & 0 & -47.49 \\ 0 & 0 & -3.16 & -9.47 \end{bmatrix}$$

$$R1 = R1 + \left(\frac{3}{9.5}\right) R2$$

$$C = \begin{bmatrix} 2 & 0 & 0 & 4 \\ 0 & -9.5 & 0 & -47.49 \\ 0 & 0 & 3.16 & -9.47 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 2 \\ 0 & 1 & 0 & 5 \\ 0 & 0 & 1 & 3 \end{bmatrix}$$

Cost of Gaussian elimination

Computational time for elimination process is O (n3) ("order n cubed").Follows from the fact that the process involves three nested loops and that in each case the number of iterations depends upon n.

Computational time for back substitution is O (n2) ("order n squared").Follows from the fact that the process involves two nested loops and that in each case the number of iterations depends upon n. Again the last loop is implicit. Computational time for the complete process is O(n3) (as n increases n3soon swamps n2).In crude terms every doubling of n increased the time required by a factor of 23= 8.If solving a system of 100 equations takes 1 second solving a system of 800 equations (three doublings) will take approximately 512 seconds = 8.533 minutes.

Iterative Methods

$$C_{ii} = 0$$

$$C_{ij} = -\frac{a_{ij}}{a_{ii}}$$

$$d_i = \frac{b_i}{a_{ii}}$$

$$X_{k+1} = CX_k + d$$

Example

$$\begin{array}{llll} -2x & +5y & +z & = 6 \\ 3x & +y & -z & =16 \\ x & +y & -3z & =2 \end{array}$$

$$\begin{bmatrix} -2 & 5 & 1 \\ 3 & 1 & -1 \\ 1 & 1 & -3 \end{bmatrix} \begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \begin{Bmatrix} 6 \\ 16 \\ 2 \end{Bmatrix}$$

$$\begin{bmatrix} 3 & 1 & -1 \\ -2 & 5 & 1 \\ 1 & 1 & -3 \end{bmatrix} \begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \begin{Bmatrix} 16 \\ 6 \\ 2 \end{Bmatrix}$$

$$\begin{bmatrix} 1 & 1/3 & -1/3 \\ -2/5 & 1 & 1/5 \\ -1/3 & -1/3 & 1 \end{bmatrix} \begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \begin{Bmatrix} 16/3 \\ 6/5 \\ 2/3 \end{Bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x \\ y \\ z \end{Bmatrix} + \begin{bmatrix} 0 & 1/3 & -1/3 \\ -2/5 & 0 & 1/5 \\ -1/3 & -1/3 & 0 \end{bmatrix} \begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \begin{Bmatrix} 16/3 \\ 6/5 \\ 2/3 \end{Bmatrix}$$

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \begin{bmatrix} 0 & -1/3 & 1/3 \\ 2/5 & 0 & -1/5 \\ 1/3 & 1/3 & 0 \end{bmatrix} \begin{Bmatrix} x \\ y \\ z \end{Bmatrix} + \begin{Bmatrix} 16/3 \\ 6/5 \\ 2/3 \end{Bmatrix}$$

Let $\begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix}$

Jacobi Method

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \begin{bmatrix} 0 & -1/3 & 1/3 \\ 2/5 & 0 & -1/5 \\ 1/3 & 1/3 & 0 \end{bmatrix} \begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix} + \begin{Bmatrix} 16/3 \\ 6/5 \\ 2/3 \end{Bmatrix}$$

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \begin{Bmatrix} 16/3 \\ 7/5 \\ 0 \end{Bmatrix}$$

Use this as your new guess and solve it again

Gauss – Seidel Method

$$x_1 = \begin{bmatrix} 0 & -1/3 & 1/3 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} + \frac{16}{3} = \frac{16}{3}$$

$$y_1 = \begin{bmatrix} 2/5 & 0 & -1/5 \end{bmatrix} \begin{bmatrix} 16/3 \\ 1 \\ 1 \end{bmatrix} + \frac{6}{5} = \frac{47}{15}$$

$$z_1 = \begin{bmatrix} 1/3 & 1/3 & 0 \end{bmatrix} \begin{bmatrix} 16/3 \\ 47/5 \\ 1 \end{bmatrix} - \frac{2}{3} = \frac{191}{45}$$

$$\begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \begin{Bmatrix} 16/3 \\ 47/15 \\ 191/45 \end{Bmatrix}$$

Difference between Jacobi and Gauss-Seidel method is Gauss- Seidel Method use the value of x_1 to find y_1 andx_1, $y_1$to find z_1

Function (lu)

$$\mathrm{Ax = b} \ \rightarrow \mathrm{LUx = b} \rightarrow \mathrm{Ld = b} \rightarrow \mathrm{Ux = d}$$

Ax = b

[L, U] = lu (A)

d = L\b

x = U\d

$$\mathrm{Ax = b} \ \rightarrow \mathrm{P^{-1}LUx = b} \rightarrow \mathrm{LUx = Pb} \rightarrow \mathrm{Ld = Pb} \rightarrow \text{and } \mathrm{Ux = d}$$

Ax = b

[L, U, P] = lu (A)

d = L\ (P*b)

x = U\d

Left Division

Ans = A\B

REGRESSION

Transformation of Equations

For $y = \alpha e^{\beta x}$

$$y` = ax` + b$$

$$x` = x$$

$$y` = ln(y)$$

$$\alpha = e^{b}$$

$$\beta = a$$

For $y = \alpha x^{\beta}$

$$x` = \log(x)$$

$$y` = log(y)$$

$$\alpha = 10^{b}$$

$$\beta = a$$

For $y = \alpha \frac{x}{\beta + x}$

$$x` = \frac{1}{\mathrm{x}}$$

$$y` = \frac{1}{y}$$

$$\alpha = \frac{1}{b}$$

$$\beta = \frac{a}{b}$$

Straight line fit

$$y = mx + b$$

$$m = \frac{n\sum X_i Y_i - \sum X_i \sum Y_i}{n\sum X_i^2 - (\sum X_i)^2}$$

$$b = \frac{\sum {X_i}^2 \sum Y_i - \sum X_i \sum Y_i}{n\sum X_i^2 - (\sum X_i)^2} = \bar{y} - m\bar{x}$$

Correlation of Coefficient

$$r^2 = \frac{S_t - S_r}{S_t}$$

$$S_t = \sum (Y_i - \bar{Y})^2$$

$$S_r = \sum (Y_i - f(X_i))^2$$

For a straight line fit only

$$r = \frac{n\sum X_i Y_i - \sum X_i \sum Y_i}{\sqrt{n\sum X_i^2 - (\sum X_i)^2}\sqrt{n\sum Y_i^2 - (\sum Y_i)^2}}$$

General least squares

$$y = a_0 g_0(x) + a_1 g_1(x) + a_2 g_2(x) + \cdots + a_n g_n(x)$$

Basic Solution

$$Z_{ij} = g_j(x_i)$$

$$Z^T Z a = Z^T y$$

QR decomposition

$$Z = QR$$

$$Ra = Q^T y$$

Practice Problem

Manually find the "best fit" straight line for the four points. Confirm your answer using "polyfit". What is the correlation coefficient?

$$y = mx + b$$

$$m = \frac{n\sum X_iY_i - \sum X_i \sum Y_i}{n\sum X_i^2 - (\sum X_i)^2}$$

$$b = \frac{\sum {X_i}^2 \sum Y_i - \sum X_i \sum X_iY_i}{n\sum X_i^2 - (\sum X_i)^2} = \bar{y} - m\bar{x}$$

$$m = \frac{4(22122) - 48360}{4(146) - 400} = 218.087$$

$$b = \bar{y} - m\bar{x} = -485.93$$

$$y = 218.087x - 485.93$$

$$r = \frac{n\sum X_iY_i - \sum X_i \sum Y_i}{\sqrt{n\sum X_i^2 - (\sum X_i)^2}\sqrt{n\sum Y_i^2 - (\sum Y_i)^2}}$$

$$r = \frac{4(22122) - 48360}{\sqrt{4(146) - 400}\sqrt{4(3958452) - 5846724}} = \frac{40128}{42868.59} = 0.936070083$$

INTERPOLATION

Lagrange Polynomial

$$p(x) = L_1(x)y_1 + L_2(x)y_2 + L_3(x)y_3 + \cdots + L_N(x)y_N$$

$$L_k(x) = \frac{(x - x_1)(x - x_2)\ldots(x - x_{k-1})(x - x_{k+1})\ldots(x - x_N)}{(x_k - x_1)(x_k - x_2)\ldots(x_k - x_{k-1})(x_k - x_{k+1})\ldots(x_k - x_N)}$$

Numerator of $L_k(x)$ is the product of all $(x - x_i)$ except for $(x - x_k)$

Denominator of $L_k(x)$ is the product of all $(x_k - x_i)$ except for $(x_k - x_k)$

Newton's Polynomial

$$p(x) = a_1 + a_2(x - x_1) + a_3(x - x_1)(x - x_2) + \cdots + a_N(x - x_1)\ldots(x - x_{N-1})$$

$$a_1 = y_1 \qquad a_2 = Dy_1 \qquad a_3 = D^2y_1 \qquad a_N = D^{N-1}y_1$$

$$Dy_1 = \frac{y_{i+1} - y_i}{x_{i+1} - x_i} \qquad D^2y_1 = \frac{Dy_{i+1} - Dy_i}{x_{i+2} - x_i} \qquad D^Ky_1 = \frac{D^{K-1}y_{i+1} - D^{K-1}y_i}{x_{i+K} - x_i}$$

X_1	y_1	$Dy_1 = \frac{y_2 - y_1}{x_2 - x_1}$	$D^2y_1 = \frac{Dy_2 - Dy_1}{x_3 - x_1}$	$D^3y_1 = \frac{D^2y_2 - D^2y_1}{x_4 - x_1}$
X_2	y_2	$Dy_2 = \frac{y_3 - y_2}{x_3 - x_2}$	$D^2y_2 = \frac{Dy_3 - Dy_2}{x_4 - x_2}$	
X_3	y_3	$Dy_3 = \frac{y_4 - y_3}{x_4 - x_3}$		
X_4	y_4			

Practice Problems

Find the polynomial that passes through the following four points:

Obtain an answer using

i) Lagrange's form and
ii) Newton's form.

X	Y
1	0
3	54
6	420
10	1944

Lagrange's form and

$$\text{L1(x)} = \frac{(x-3)(x-6)(x-10)}{(1-3)(1-6)(1-10)}$$

$$\text{L2(x)} = \frac{(x-1)(x-6)(x-10)}{(3-1)(3-6)(3-10)}$$

$$\text{L3(x)} = \frac{(x-1)(x-3)(x-10)}{(6-1)(6-3)(6-10)}$$

$$\text{L4(x)} = \frac{(x-1)(x-3)(x-6)}{(10-1)(10-3)(10-6)}$$

$$p(x) = L_1(x)y_1 + L_2(x)y_2 + L_3(x)y_3 + \cdots + L_N(x)y_N$$

$$p(x) = 2x^3 - x^2 + 5x - 6$$

Newton's form.

1	0	27	19	2
3	54	122	37	
6	420	381		
10	1944			

$$p(x) = 0 + 27(x-1) + 19(x-1)(x-3) + 2(x-1)(x-3)(x-6)$$

$$p(x) = 2x^3 - x^2 + 5x - 6$$

Function (polyfit, polyval)

Must be given set of X and Y values of same order and the order of the polynomial

Maximum number of n that can be used is ((order of X and Y)-1)

P= polyfit (X, Y, N)

Must be given a polynomial function or the coefficient matrix (i.e. P) and the number of interest

A =polyval (p, n)

Function (spline, interp1)

Spline: Spline is a form of interpolation. Instead of using a single function (as in polynomial interpolation) separate functions are used between each pair of data points. Name comes from flexible *splines* used in drafting to draw a smooth curve point through a series of points (though linear splines won't look much like this, cubic splines will).

Spline(x, y, xin)
Interp1(x, y, xin, 'spline')

NUMERICAL INTEGRATION

Note: When using trapezoidal, or Simpson's method make sure the units of the points are in the form that you can apply the formula directly

Trapezoidal Integration

'n' intervals
'n+1' data points
h = (b-a)/n

$$I = \frac{\mathrm{h}}{2}(f(x_0) + f(x_1))$$

$$I = \frac{\mathrm{h}}{2}\left(f(x_0) + 2f(x_1) + 2f(x_2) + 2f(x_3) + \cdots + 2f(x_{N-1}) + f(x_N)\right)$$

Simpson's 1/3 Rule

'n' intervals
'n+1' data points
h = (b-a)/n

$$I = \frac{\mathrm{h}}{3}(f(x_0) + 4f(x_1) + f(x_2))$$

$$I = \frac{\mathrm{h}}{3}\left(f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + 2f(x_4) + \cdots + 4f(x_{N-1}) + f(x_N)\right)$$

Simpson's 3/8 Rule

'n' intervals
'n+1' data points
h = (b-a)/n

$$I = \frac{3\mathrm{h}}{8}(f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3))$$

Practice Problem

The table below gives velocities of an object from t = 0 to t = 10 seconds.

Estimate the distance travelled by the object between t = 0 and t = 10. Use trapezoidal integration and Simpson's 1/3 rule. Estimate the acceleration of the object at t = 4 using forward, central, and backward differences.

Trapezoidal Method

$$I = \frac{1}{2}(452 + 2(410) + 2(377) + 2(350) + 2(322) + 2(301) + 2(283) + 2(269) + 2(257) + 2(251) + 250) = 3171$$

Simpson's 1/3 Rule

$$I = \frac{1}{3}(452 + 4(410) + 2(377) + 4(350) + 2(322) + 4(301) + 2(283) + 4(269) + 2(257) + 4(251) + 250) = 3168$$

Central

$$\frac{301 - 350}{1} = -49$$

Forward

$$\frac{301 - 322}{1} = -21$$

Backward

$$\frac{322 - 350}{1} = -28$$

Gaussian Quadrature - Direct Integration (Only for -1 to 1)

To change the limits to -1 to 1

$$x = \frac{b+a}{2} + \frac{b-a}{2} X_d$$

$$dx = \frac{b-a}{2} dX_d$$

Then, Use the table to integrate the function

N	C_0	X_0	C_1	X_1	C_2	X_2	C_3	X_3
2	1	-0.57735	1	0.57735				
3	0.55555	-0.77459	0.88888	0	0.55555	0.77459		
4	0.34785	-0.86117	0.652145	-0.339981	0.652145	0.339981	0.34785	-0.86117

$$I = C_0 f(X_0) + C_1 f(X_1) + \cdots$$

Romberg Integration

h_1	I_{11}	I_{12}= (4(I_{21}) - I_{11})/3	I_{13}= (16(I_{22}) - I_{12})/15	**I_{14}= (64(I_{23}) - I_{13})/63**
h_2= h_1/2	I_{21}	I_{22}= (4(I_{31}) - I_{21})/3	I_{23}= (16(I_{32}) – I_{22})/15	
h_3= h_2/2	I_{31}	I_{32}= (4(I_{41}) - I_{31})/3		
h_4= h_3/2	I_{41}			

% Error = $\left|\frac{I_{21}-I_{12}}{I_{12}}\right| \times 100$

(i.e. (Best estimation at the previous column – New estimation) / (New estimation))

To calculate I_{11}, Use Trapezoidal Integration

$$I = \frac{h}{2}\left(f(x_0) + 2f(x_1) + 2f(x_2) + 2f(x_3) + \cdots + 2f(x_{N-1}) + f(x_N)\right)$$

NUMERICAL DIFFERENTIATION

Forward, Central, and Backward differentiation

First Derivative	First Order	Second Order (better)
Forward	$\frac{f(X_{i+1}) - f(X_i)}{h}$	$\frac{-f(X_{i+2}) + 4f(X_{i+1}) - 3f(X_i)}{2h}$
Backward	$\frac{f(X_i) - f(X_{i-1})}{h}$	$\frac{3f(X_i) - 4f(X_{i-1}) - 3f(X_{i-2})}{2h}$
Central	$\frac{f(X_{i+1}) - f(X_{i-1})}{2h}$	$\frac{-f(X_{i+2}) + 8f(X_{i+1}) - 8f(X_{i-1}) - f(X_{i-2})}{12h}$

Practice Problem

Manually determine dp/dt for each data point. Use first order central differences where this is possible and a first order forward or backward difference otherwise.

Year	Population
1950	2555
1955	2780
1960	3040
1965	3346
1970	3708
1975	4087
1980	4454
1985	4850
1990	5276
1995	5686
2000	6079

Forward for the first two points

$$\frac{2780 - 2555}{5} = 45$$

Central for the (n-1) points

$$\frac{3040-2555}{5}+\frac{3346-2780}{5}+\frac{3708-3040}{5}+\frac{4087-3346}{5}+\frac{4454-3708}{5}+\frac{4850-4087}{5}+\frac{5276-4454}{5}+\frac{5686-4850}{5}+\frac{6079-5276}{5}=1286$$

Backward for last point

$$\frac{6079-5686}{5}=78.6$$

Adding all

$$45+1286+78.6=1409.6$$

Richardson extrapolation and differentiation

h_1	D_{11}	$D_{12}= (4(D_{21}) - D_{11})/3$	$D_{13}= (16(D_{22}) - D_{12})/15$	**$D_{14}= (64(D_{23}) - D_{13})/63$**
$h_2= h_1/2$	D_{21}	$D_{22}= (4(D_{31}) - D_{21})/3$	$D_{23}= (16(D_{32}) – D_{22})/15$	
$h_3= h_2/2$	D_{31}	$D_{32}= (4(D_{41}) - D_{31})/3$		
$h_4= h_3/2$	D_{41}			

% Error = $\left|\frac{D_{21}-D_{12}}{D_{12}}\right| \times 100$

(i.e. (Best estimation at the previous column – New estimation) / (New estimation))

To find D_{11} to D_{41} use

$$f'(x) \sim \frac{f(x+h)-f(x)}{h}$$

ORDINARY DIFFERENTIAL EQUATIONS (ODE)

Euler's Method

$$V_{i+1} = V_i + \varphi(t_i, v_i)\Delta t$$

Where,

$$V_i \rightarrow Intial\ Velocity$$

$$\varphi(t_i, v_i) \ \rightarrow Original\ Function$$

$$\Delta t \ \rightarrow Change\ in\ time\ period$$

Example

$$V_0 \rightarrow 10\ m/s$$

$$\varphi(t_i, v_i) \ \rightarrow -\frac{1}{200}(0.5v + 500\ (1 - \exp(-0.5t))$$

$$\Delta t \ \rightarrow 0.5$$

$$V_{i+1} = V_i + \varphi(t_i, v_i)\Delta t$$

$\boldsymbol{i}$	$\boldsymbol{t_i}$	$\boldsymbol{V_i}$	$\boldsymbol{\varphi(t_i, v_i)}$	$\boldsymbol{V_{i+1}}$	$\boldsymbol{Exact}$
0	0	10	-0.025	9.9875	9.8436
1	0.5	9.9875	-0.5780	9.6985	9.4428
2	1	9.6985	-1.00792	9.19454	8.8522

Heun's Method

Heun's Method uses the same formula as Euler's Method; only difference is Heun's Method has a correction step.

$$V_{i+1} = V_i + \varphi(t_i, v_i)\Delta t$$

$\boldsymbol{i}$	$\boldsymbol{t_i}$	$\boldsymbol{V_i}$	$\boldsymbol{\varphi(t_i, v_i)}$	$\boldsymbol{V_{i+1}}$	$\boldsymbol{t_i{}'}$	$\boldsymbol{\varphi'(t_i, v_i)}$	$\boldsymbol{\frac{\varphi + \varphi'}{2}}$	$\boldsymbol{V_{i+1}{}'}$	$\boldsymbol{Exact}$
0	0	10	-0.025	9.9875	0.5	-0.5780	-0.3015	9.84925	
1	0.5	9.84925	-0.5776	9.56045	1	-1.0076	-0.7926	9.45295	
2	1	9.45295	-1.0073	8.9493	1.5	-1.34146	-1.1744	8.86575	

Practice Problem

Assuming that y (0) = 1, manually find y (t) for t = 0 to t = 2 seconds
a) Using Euler's method with h = 0.5 and h = 0.25
b) Using Heun's method (no iteration) with h = 0.5

Euler's method

i	t_i	Y_i	$\varphi(t_i, y_i)$	Y_{i+1}	*Exact*
0	0	1	-1.1	0.45	
1	0.5	0.45	-0.3825	0.25875	
2	1	0.25875	-0.025875	0.2458125	
3	1.5	0.2458125	0.282684375	0.3871547	
4	2	0.3871547	1.12274863	1.0008979	

Heun's method

i	t_i	Y_i	$\varphi(t_i, y_i)$	Y_{i+1}	$t_i{}'$	$\varphi'(t_i, y_i)$	$\frac{\varphi + \varphi'}{2}$	$Y_{i+1}{}'$	*Exact*
0	0	1	-1.1	0.45	0.5	-0.3825	-0.7413	0.62935	
1	0.5	0.62935	-0.5349	0.3619	1	-0.03619	-0.2855	0.4866	
2	1	0.4866	-0.0487	0.46225	1.5	0.531588	0.24144	0.60732	
3	1.5	0.60732	0.69842	0.95653	2	2.77394	1.73618	1.47541	
4	2	1.47541	4.27869	3.614755	2.5	18.6160	11.447	7.1991	

Function (ode45)

[t, y] = ode45 (@slope, [0 20], [X_0, V_0]

Step 1: Define slope function

>> Slope =@ (t, v) –(c*v + b*t)/m;

Step 2: Involve ode45

>> [t, y] = ode45 (slope, [0 30], v_0] ;

GENERAL FUNCTIONS

Linspace:

linspace (first Number, Last Number, Number of points)

Colon:

First Number: Difference between the Numbers: Last Number

Plot:

plot (x, y, 'options');

fplot (function, from, to);

Vector zings an equation:

= a.*b; a. ^b; a. /b;

Function (Roots)

Roots(p) % p is coefficient matrix