10.4.1.2.3

EE24BTECH11007 - Arnav Makarand Yadnopavit

Question: Rohan's mother is 26 years older than him. The product of their ages (in years) 3 years from now will be 360. We would like to find Rohan's present age.
Solution:
Theoretical Solution:
Let Rohan's age, R=x.
Then Mother's age, M=$x+26$ Then after 3 years we get

$$(M+3)(R+3) = 360 \tag{1}$$
$$(x+29)(x+3) = 360 \tag{2}$$
$$x^2 + 32x - 273 = 0 \tag{3}$$
$$\tag{4}$$

Solving the equation we get, $x = 7$ or $x = -39$. Eliminating $x = -39$ (Age considered to be a non-negative value)
Rohan's present age is 7.
Computational Solution:
Two methods for finding the solution of a quadratic equation are:
Matrix-Based Method:
For a polynomial equation of form $x^n + b_{n-1}x^{n-1} + \cdots + b_2x^2 + b_1x + b_0 = 0$ we construct a matrix called companion matrix of form

$$\Lambda = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \vdots & 1 \\ -b_0 & -b_1 & -b_2 & \dots & -b_{n-1} \end{pmatrix} \tag{5}$$

The eigenvalues of this matrix are the roots of the given polynomial equation.
Finding eigenvalues

I. QR Algorithm with Householder technique and Wilkinson Shift

A. QR Decomposition

QR decomposition factors a given matrix A into:

$$A = QR,$$

where Q is an orthogonal matrix ($Q^\top Q = I$), and R is an upper triangular matrix.

B. Householder Transformations

Householder transformations are used to zero out elements below the diagonal of a matrix column. Given a vector v, the Householder matrix is:

$$H = I - 2\frac{vv^\top}{v^\top v} \tag{6}$$

The way it works is:
Initialize Q as Identity matrix. Let $\mathbf{x}$ be the the first column of A, and $\alpha = ||\mathbf{x}||$.

$$\mathbf{u} = \mathbf{x} - \alpha \mathbf{e_1} \tag{7}$$

$$\mathbf{v} = \frac{\mathbf{u}}{||\mathbf{u}||} \tag{8}$$

$$Q = I - 2vv^H \tag{9}$$

By this we obtain Q_1 such that:

$$Q_1 A = \begin{pmatrix} \alpha_1 & * & \dots & * \\ 0 & & & \\ \vdots & & A' & \\ 0 & & & \end{pmatrix}$$

This can be repeated for A' (obtained from $Q_1 A$ by deleting the first row and first column), resulting in a Householder matrix Q'_2. Note that $Q\prime_2$ is smaller than Q_1. Since we want it really to operate on $Q_1 A$ instead of A' we need to expand it to the upper left, filling in a 1, or in general:

$$Q_k = \begin{pmatrix} I_{k-1} & 0 \\ 0 & Q'_k \end{pmatrix}$$

After $n-1$ iterations of this process.

$$R = Q_{n-1} \dots Q_2 Q_1 A \tag{10}$$

$$Q^\top = Q_{n-1} \dots Q_2 Q_1 \tag{11}$$

$$Q = Q_1 Q_2 \dots Q_{n-1} \tag{12}$$

C. QR Algorithm for Eigenvalues

The QR algorithm iteratively applies QR decomposition to a shifted matrix $A - \mu I$ and reconstructs it as:

$$A = RQ + \mu I$$

converging to an upper triangular form with eigenvalues on the diagonal.
where μ can be calculated by:

$$\mu = a_m - \frac{\delta}{|\delta|} \frac{b_{m-1}^2}{|\delta| + \sqrt{\delta^2 + b_{m-1}^2}}$$

where B is the lower rightmost 2×2 matrix of A, B= $\begin{pmatrix} a_{m-1} & b'_{m-1} \\ b'_{m-1} & a_m \end{pmatrix}$
$\delta = \frac{a_{m-1} - a_m}{2}$
If $\delta = 0$, then $\mu = a_m - b_{m-1}$

D. Complex Eigenvalues

In case a matrix has complex eigenvalues a hessenberg matrix (2×2) will be formed along the diagonal of the triangularised matrix A such that:

$$A = \begin{pmatrix} \lambda_1 & \dots & \dots & \dots \\ 0 & a & b & \dots \\ \vdots & c & d & \dots \\ 0 & \dots & \dots & \lambda_n \end{pmatrix}$$

then:

$$\lambda_2 = \frac{a + d + \sqrt{(a+d)^2 - 4(ad - bc)}}{2}$$
$$\lambda_3 = \frac{a + d - \sqrt{(a+d)^2 - 4(ad - bc)}}{2}$$

The solution given by the code is

$$x_1 = 7.000000 + 0.000000i \tag{13}$$
$$x_2 = -39.000000 + 0.000000i \tag{14}$$

Newton-Raphson Method:
Start with an initial guess x_0, and then run the following logical loop,

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{15}$$

where,

$$f(x) = x^2 + 32x - 273 \tag{16}$$
$$f'(x) = 2x - 32 \tag{17}$$

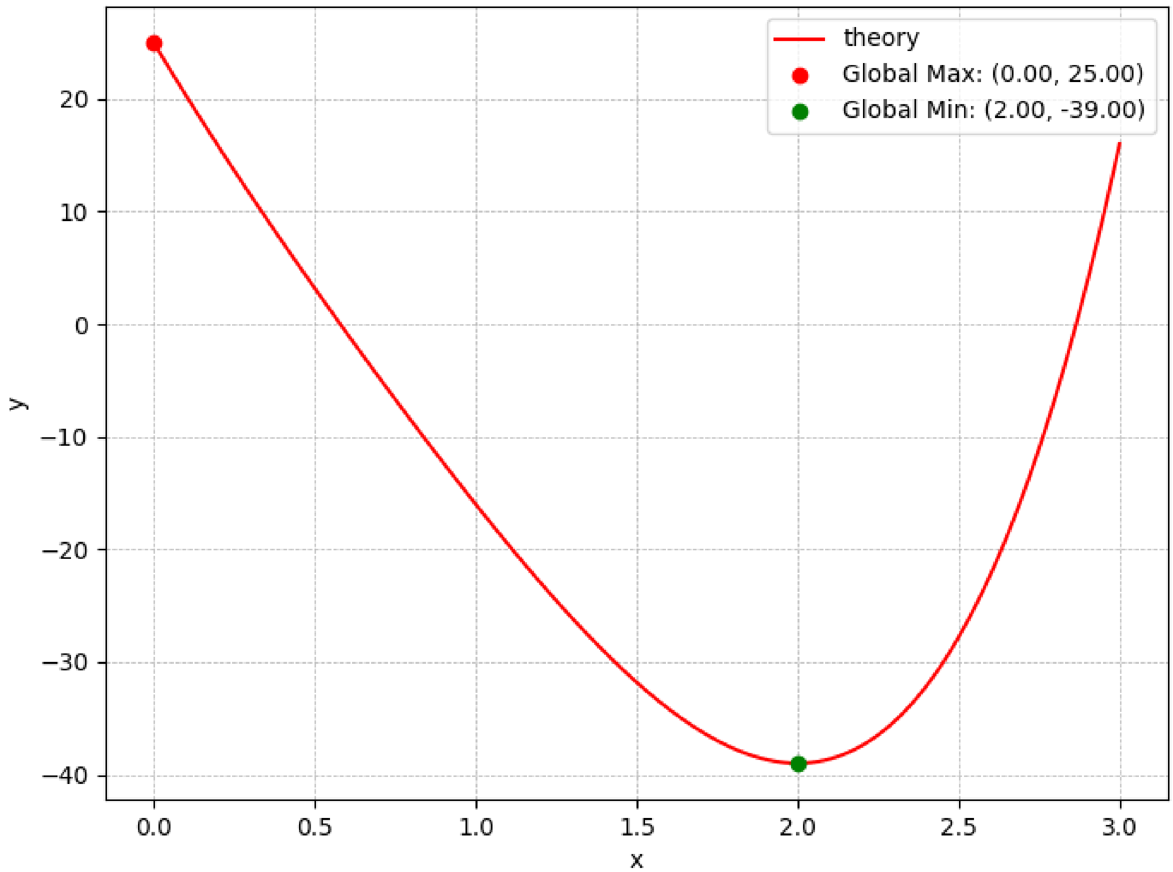
The update equation will be

$$x_{n+1} = x_n - \frac{2{x_n}^2 - 13x_n + 9}{4x_n - 13} \tag{18}$$

$$\tag{19}$$

The problem with this method is if the roots are complex but the coefficients are real, x_n either converges to an extrema or grows continuously without any bound. However, to obtain complex solutions, we can just take the initial guess point to be a random complex number.
The output of a program written to find roots is shown below:

$$r_1 = 0.7878 \tag{20}$$
$$r_2 = 5.7122 \tag{21}$$


theory
Global Max: (0.00, 25.00)
Global Min: (2.00, -39.00)
20
10
0
−10
−20
−30
−40
y
0.0
0.5
1.0
1.5
2.0
2.5
3.0
x