10.4.2.3

EE24BTECH11021 - Eshan Ray

January 23, 2025

Problem Statement

Question:
Find two numbers whose sum is 27 and product is 182.

Solution 1: Using Algebra

Let one of the numbers be x
So, the other number is $27 - x$
Given,

$$x\,(27 - x) = 182 \tag{1}$$
$$27x - x^2 = 182 \tag{2}$$
$$x^2 - 27x + 182 = 0 \tag{3}$$
$$(x - 13)\,(x - 14) = 0 \tag{4}$$
$$\implies x = 13, 14 \tag{5}$$

So, the solutions are $x = 13$ and $x = 14$.

Computational Solution: Newton-Raphson Method

To find the roots of the quadratic equation $x^2 - 27x + 182 = 0$, we use the Newton-Raphson method.

$$x_{n+1} = x_n - \frac{f(x_n)}{f\prime(x_n)} \tag{6}$$

$$f(x) = x^2 - 27x + 182 \tag{7}$$

$$f\prime(x) = 2x - 27 \tag{8}$$

$$x_{n+1} = x_n - \frac{x_n^2 - 27x_n + 182}{2x_n - 27} \tag{9}$$

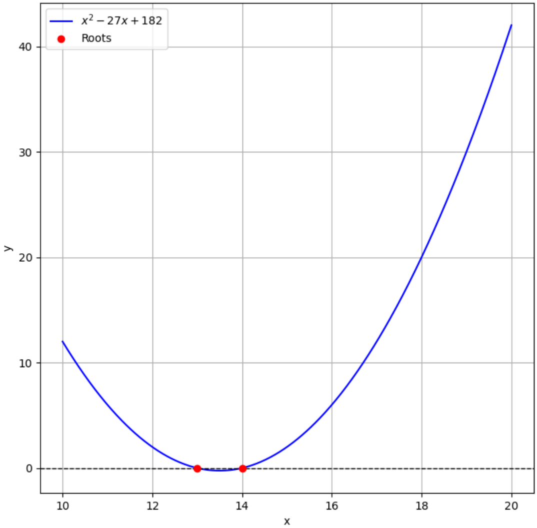
Computational Solution: Results

After running the Newton-Raphson method, we get the following roots:

$$\text{Root 1: } 14.00000000 \tag{10}$$

$$\text{Root 2: } 13.00000000 \tag{11}$$

Newton Raphson Plot



Alternate Method: Eigenvalues of Companion Matrix

In this method, we find the roots of any polynomial of the form $x^n + a_{n-1}x^{n-1} \dots ax + a_0 = 0$ by finding the eigenvalues of the Companion Matrix (C) given below:-

$$C = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \vdots & 1 \\ -a_0 & -a_1 & -a_2 & \dots & -a_{n-1} \end{pmatrix} \tag{12}$$

For the Quadratic Equation $x^2 - 27x + 182 - 0$, we get the following companion Matrix

$$C = \begin{pmatrix} 0 & 1 \\ -182 & 27 \end{pmatrix} \tag{13}$$

The roots of the equation is the eigenvalues of the matrix C which has been calculated using the QR Decomposition process.

QR Decomposition : Gram-schmidt Process

In the QR Decomposition, the matrix A is decomposed into matrices Q and R as:

$$A = QR \tag{14}$$

where ,Q is an orthogonal matrix and R is an upper triangular matrix. We start by producing an orthogonal set of column vectors of Q $\{q_1, q_2, \ldots, q_n\}$ from a set of column vectors of A $\{a_1, a_2, \ldots, a_n\}$. For orthogonalization we subtract each vector a_i with the projections of all previously obtained orthogonal vectors $q_1, q_2, \ldots, q_{i-1}$ to make q_i orthogonal to them.

The projection of a_i onto a vector q_j is calculated as:

$$proj_{q_j}(a_i) = \frac{\langle a_i, q_j \rangle}{\langle q_j, q_j \rangle} q_j \tag{15}$$

Then q_i is computed as:

$$q_i = a_i - \sum_{j=1}^{i-1} proj_{q_j}(a_i) \tag{16}$$

Then all the q_i's are normalized by :

$$q_i = \frac{q_i}{||q_i||} \tag{17}$$

The process is repeated for all the colums of A

3) As Q is an orthonormal matrix

$$Q^\top Q = I \tag{18}$$

So, R can be represented as follows

$$R = Q^\top A \tag{19}$$

$$r_{ij} = \langle a_j, q_i \rangle \text{ , for } i \leq j \tag{20}$$

QR Algorithm

In the QR algorithm, the matrix A_n is decomposed into matrices Q_n and R_n as:

$$A_n = Q_n R_n \tag{21}$$

Then, the new matrix A_{n+1} is computed as:

$$A_{n+1} = R_n Q_n \tag{22}$$

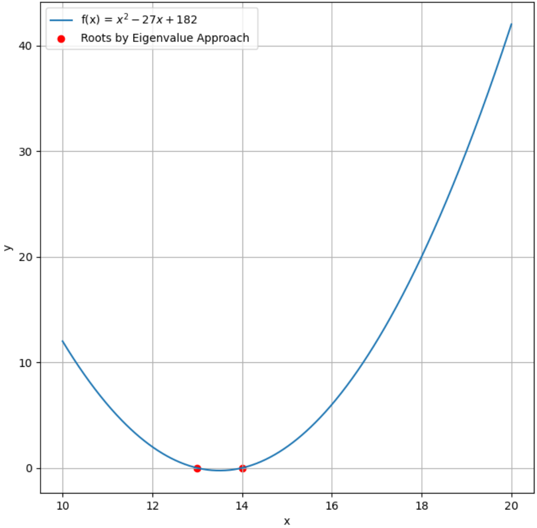
This process is repeated until the off-diagonal elements of the matrix become negligibly small, at which point the diagonal elements approximate the eigenvalues of the original matrix.

Eigenvalue Approach: Results

After applying the QR algorithm to the companion matrix, the eigenvalues are computed to be:

$$\text{Eigenvalues: } 14.0, 13.0$$

Eigenvalue Approach Plot



Conclusion

The problem was solved using two methods: algebraic factorization and computational methods (Newton-Raphson and Eigenvalue approach).

Both methods resulted in the same roots: 13 and 14.

The eigenvalue method uses matrix operations to find roots, while Newton-Raphson provides a more direct approach.

GitHub Repository

C code for Newton-Raphson:
https://github.com/eshan810/ee1003/blob/main/Assignments/5/codes/code.c
Python code for Newton-Raphson:
https://github.com/eshan810/ee1003/blob/main/Assignments/5/codes/root.py
C code for Eigenvalue Approach:
https://github.com/eshan810/ee1003/blob/main/Assignments/5/codes/eigen.c
Python code for Eigenvalue Approach:
https://github.com/eshan810/ee1003/blob/main/Assignments/5/codes/eigen.py