CPCS212 Applied Math for Computing (I)
Methods for Solving Systems Linear Equations
Gaussian Elimination and Cholesky Decomposition
Group Project - CS3

Group Members:

- **Mahmued Alardawi - 2135209**
- **Afiq Ijlal Ikhtiari Sigarra - 2047931**

- **Submission Date: 11/10/2023**

Table of Content

Introduction

Systems of linear equations are collections of two or more linear equations that use the same set of variables. Each equation in a system provides a constraint on the possible values of the variables. The goal when working with such systems is typically to find the values of the variables that satisfy all the equations in the system simultaneously.

Here's a breakdown of the concepts:

A linear equation is an equation that represents a straight line when graphed on a coordinate plane. It has the general form:

$ax + by = c$

Where x and y variables, and a, b, and c are constants with a and b not both zero.

Heare is an example of a system with three linear equations:

$25x_1 + 15x_2 - 5x_3 = 35$

$15x_1 + 18x_2 + 0 = 35$

$-5x_1 + 0 + 11x_3 = 35$

If you try to solve it, you probably could but imagine solving it when it contains larger numbers or more equations, and that is why we must find a new way to solve them.

Part 01: Gaussian Elimination

Theory

Gaussian Elimination is a systematic method for solving systems of linear equations. It is a type of row reduction that transforms the system into upper triangular form, which can then be solved by backward substitution. The method is named after the German mathematician Carl Friedrich Gauss

Explanation

step-by-step overview of the Gaussian Elimination process:

Example

We will solve the following linear system:

$$
\begin{aligned}
25x_1 + 15x_2 - 5x_3 &= 35 \\
15x_1 + 18x_2 - 0 &= 33 \\
-5x_1 + 0 - 11x_3 &= 6
\end{aligned}
$$

Step 1: Write the Augmented Matrix

The system of equations is written as an augmented matrix, where each row represents an equation and each column represents the coefficients of one of the variables, with the last column representing the constants on the right side of the equations.

For example, the system of equations:

$$
\begin{aligned}
25x_1 + 15x_2 - 5x_3 &= 35 \\
15x_1 + 18x_2 - 0 &= 33 \\
-5x_1 + 0 - 11x_3 &= 6
\end{aligned}
\quad = \quad
\left[\begin{array}{ccc|c}
25 & 15 & -5 & 35 \\
15 & 18 & 0 & 33 \\
-5 & 0 & 11 & 6
\end{array}\right]
$$

Step 2: Row Reduction to Upper Triangular Form

An upper triangular form in the context of a matrix refers to a type of square matrix where all the entries below the main diagonal are zero.

$$\left[\begin{array}{ccc|c} 25 & 15 & -5 & 35 \\ 15 & 18 & 0 & 33 \\ -5 & 0 & 11 & 6 \end{array}\right] \Rightarrow \left[\begin{array}{ccc|c} x_1 & x_2 & x_3 & y_1 \\ 0 & x_2 & x_3 & y_2 \\ 0 & 0 & x_3 & y_3 \end{array}\right]$$

Let's implement it:

iteration 01

Make $a_{21} = 0$

$$\left[\begin{array}{ccc|c} a_{11} & a_{12} & a_{13} & b_1 \\ \boxed{a_{21}} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{22} & a_{33} & b_3 \end{array}\right]$$

$$key = \frac{a_{21}}{a_{11}}$$

$$Formula : r_2 = r_2 - key \times r_1$$

$$a_{21} = a_{21} - key \times a_{11}$$
$$a_{22} = a_{22} - key \times a_{12}$$
$$a_{23} = a_{23} - key \times a_{13}$$
$$b_2 = b_2 - key \times b_1$$

$$start$$

$$\left[\begin{array}{ccc|c} 25 & 15 & -5 & 35 \\ \boxed{15} & 18 & 0 & 33 \\ -5 & 0 & 11 & 6 \end{array}\right]$$

$$end$$

$$\left[\begin{array}{ccc|c} 25 & 15 & -5 & 35 \\ \boxed{0} & 9 & 3 & 12 \\ -5 & 0 & 11 & 6 \end{array}\right]$$

$$key = \frac{a_{21}}{a_{11}} = \frac{15}{25} = \frac{3}{5}$$

$$a_{21} = a_{21} - key \times a_{11} = 15 - \frac{3}{5} \times 25 = 0$$

$$a_{22} = a_{22} - key \times a_{12} = 18 - \frac{3}{5} \times 15 = 9$$

$$a_{23} = a_{23} - key \times a_{13} = 0 - \frac{3}{5} \times -(-5) = 3$$

$$b_2 = b_2 - key \times b_1 = 33 - \frac{3}{5} \times 35 = 12$$

Iteration 02

Make $a_{31} = 0$

$$\left[\begin{array}{ccc|c} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ \boxed{a_{31}} & a_{22} & a_{33} & b_3 \end{array}\right]$$

$$key = \frac{a_{31}}{a_{11}}$$

$$Formula : r_3 = r_3 - key \times r_1$$

$$a_{31} = a_{31} - key \times a_{11}$$
$$a_{32} = a_{32} - key \times a_{12}$$
$$a_{33} = a_{33} - key \times a_{13}$$
$$b_3 = b_3 - key \times b_1$$

$$start$$

$$\left[\begin{array}{ccc|c} 25 & 15 & -5 & 35 \\ 0 & 9 & 3 & 12 \\ \boxed{-5} & 0 & 11 & 6 \end{array}\right]$$

$$end$$

$$\left[\begin{array}{ccc|c} 25 & 15 & -5 & 35 \\ 0 & 9 & 3 & 12 \\ \boxed{0} & 3 & 10 & 13 \end{array}\right]$$

$$key = \frac{a_{31}}{a_{11}}$$
$$= \frac{-5}{25} = -\frac{1}{5}$$

$$a_{21} = a_{21} - key \times a_{11}$$
$$= -5 - \left(-\frac{1}{5}\right) \times 25 = 0$$

$$a_{32} = a_{32} - key \times a_{12}$$
$$= 0 - \left(-\frac{1}{5}\right) \times 15 = 3$$

$$a_{33} = a_{33} - key \times a_{13}$$
$$= 11 - \left(-\frac{1}{5}\right) \times -(-5) = 10$$

$$b_3 = b_3 - key \times b_1$$
$$= 6 - \left(-\frac{1}{5}\right) \times 35 = 13$$

Iteration 03

Make $a_{32} = 0$

$$\left[\begin{array}{ccc|c} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & \boxed{a_{22}} & a_{33} & b_3 \end{array}\right]$$

$$Formula : r_3 = r_3 - key \times r_2$$

$$a_{32} = a_{32} - key \times a_{22}$$
$$a_{33} = a_{33} - key \times a_{23}$$
$$b_3 = b_3 - key \times b_2$$

$$key = \frac{a_{32}}{a_{22}}$$

$$start$$

$$\left[\begin{array}{ccc|c} 25 & 15 & -5 & 35 \\ 0 & 9 & 3 & 12 \\ 0 & \boxed{3} & 10 & 13 \end{array}\right]$$

$$end$$

$$\left[\begin{array}{ccc|c} 25 & 15 & -5 & 35 \\ 0 & 9 & 3 & 12 \\ 0 & \boxed{0} & 9 & 9 \end{array}\right]$$

$$key = \frac{a_{32}}{a_{22}}$$
$$key = \frac{3}{9} = \frac{1}{3}$$

$$a_{32} = a_{32} - key \times a_{22}$$
$$= 3 - \frac{1}{3} \times 9 = 0$$

$$a_{33} = a_{33} - key \times a_{23}$$
$$= 10 - \frac{1}{3} \times 3 = 9$$

$$b_3 = b_3 - key \times b_2$$
$$= 13 - \frac{1}{3} \times -12 = 9$$

Step 3: Backward Substitution

Back substitution is a method used to solve a system of linear equations that has been transformed into upper triangular form, it is called "back" substitution because you start with the last equation and work your way backwards to the first equation.

Backward Substitution steps:

1. Start with the Last Equation
2. Substitute and Solve the Second-to-Last Equation
3. Continue Substituting and Solving
4. Finish with the First Equation

Let's implement it:

$$\begin{array}{ccc} x_1 & x_2 & x_3 \end{array}$$

$$\left[\begin{array}{ccc|c} 25 & 15 & -5 & 35 \\ 0 & 9 & 3 & 12 \\ 0 & 0 & 9 & 9 \end{array}\right]$$

$$x_3 : 9x_3 = 0 => x_3 = 1$$

$$x_2 : 9x_2 + 3x_3 = 12 => 9x_2 + 3\,(1) = 12$$
$$=> 9x_2 = 9 => x_2 = 1$$

$$x_1 : 25x_1 + 15x_2 - 5x_3 = 35$$
$$=> 25x_1 + 15\,(1) - 5\,(1) = 35$$
$$=> 25x_1 = 25 => x_1 = 1$$

Step 4: Check answer satisfaction

$$x_1 = 1, x_2 = 1, x_3 = 1$$

$$\left[\begin{array}{ccc|c} 25 & 15 & -5 & 35 \\ 0 & 9 & 3 & 12 \\ 0 & 0 & 9 & 9 \end{array}\right]$$

$$25x_1 + 15x_2 - 5x_3 = 35$$
$$=> 25\,(1) + 15\,(1) - 5x\,(1) = 35$$
$$=> 25 + 15 - 5 = 35$$
$$=> (35 == 35) = True$$

MATLAB code

```
% Gaussian Elimination:

clear;
clc;

syms x1 x2 x3
eq = [
    25*x1 + 15*x2 - 5*x3 == 35,
    15*x1 + 18*x2 + 0*x3  == 33,
    -5*x1 + 0*x2  + 11*x3 == 6,
    ];

[Coefficients, results] = equationsToMatrix(eq)
x = zeros(1, size(Coefficients, 2));
A = [Coefficients results]

% Forward Elimination:

for i=1:size(A, 1)
    for j=i+1:size(A, 1)
        key = A(j,i)./A(i,i);
        A(j, :) = A(j, :) - key.*A(i, :);
    end
end
A
```

```
% Backward Substitution:

for i=size(A, 1):-1:1
    s = sum(A(i, i+1:end-1).*x(i+1:end));
    x(i) = (A(i, end) - s)./A(i, i);
end
x
```

Output

```
>> GaussianElimination

Coefficients =
 [25, 15, -5]
[15, 18,  0]
[-5,  0, 11]

 results =
 35
33
 6

A =
[25, 15, -5, 35]
[15, 18,  0, 33]
[-5,  0, 11,  6]

A =
[25, 15, -5, 35]
[ 0,  9,  3, 12]
[ 0,  0,  9,  9]

x =
     1     1     1
```

Part 02: Cholesky Decomposition

Theory

Cholesky Decomposition is a specialized matrix factorization technique for Hermitian, positive-definite matrices. A matrix is Hermitian if it is equal to its conjugate transpose. In more formal terms $A = A_{conj}^{T}$, and a matrix is positive-definite when all the leading principal minors of the matrix are positive.

The Cholesky Decomposition is used to decompose a matrix into the product of a lower triangular matrix and its conjugate transpose.

Explanation

step-by-step overview of the Cholesky Decomposition process:

Example

$$\underset{A}{\begin{bmatrix} 25 & 15 & -5 \\ 15 & 18 & 0 \\ -5 & 0 & 11 \end{bmatrix}} \quad \underset{B}{\begin{bmatrix} 35 \\ 33 \\ 6 \end{bmatrix}}$$

We will skip the Augmentation part.

Step 1: Applicability

The Cholesky Decomposition only works if the matrix is Hermitian and positive-definite.

Check if Hermitian

All real symmetric matrices are Hermitian matrices

$$\underset{A}{\begin{bmatrix} 25 & 15 & -5 \\ 15 & 18 & 0 \\ -5 & 0 & 11 \end{bmatrix}} = \underset{A^T}{\begin{bmatrix} 25 & 15 & -5 \\ 15 & 18 & 0 \\ -5 & 0 & 11 \end{bmatrix}}$$

Since A equals its transpose that means the matrix is symmerical.

Check if positive-definite

Method 1

Check if all minor determinants and main determinants are positive:

$$A_1 = [25], det\,(A_1) = 25 > 0$$

$$A_2 = \begin{bmatrix} 25 & 15 \\ 15 & 18 \end{bmatrix}, det\,(A_2) = 25 \times 18 - 15 \times 15$$
$$= 450 - 225 = 225 > 0$$

$$A_3 = \begin{bmatrix} 25 & 15 & -5 \\ 15 & 18 & 0 \\ -5 & 0 & 11 \end{bmatrix}, 25 \times det \begin{bmatrix} 18 & 0 \\ 0 & 11 \end{bmatrix} - 15 \times det \begin{bmatrix} 15 & 0 \\ -5 & 11 \end{bmatrix} - (-5) \times det \begin{bmatrix} 15 & 18 \\ -5 & 0 \end{bmatrix}$$

$$= 25\,(18 \times 11 - 0) - 15\,(15 \times 11 - (-5) \times 0) - (-5)\,(15 \times 0 - (-5) \times 18)$$
$$= 25\,(189) - 15\,(165) + 5\,(90) = 4950 - 2475 + 450 = 2925 > 0$$

Since all determinants are greater than zero then the matrix is positive-definite.

Method 2

Check if all eigenvalues λ are positive:

$$\underbrace{\begin{bmatrix} 25 & 15 & -5 \\ 15 & 18 & 0 \\ -5 & 0 & 11 \end{bmatrix}}_{A}, det\,[A - \lambda I] = 0 => \begin{bmatrix} 25 & 15 & -5 \\ 15 & 18 & 0 \\ -5 & 0 & 11 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = 0$$

$$=> det \begin{bmatrix} 25-\lambda & 15 & -5 \\ 15 & 18-\lambda & 0 \\ -5 & 0 & 11-\lambda \end{bmatrix} = 0$$

$$= 25 - \lambda \times det \begin{bmatrix} 18-\lambda & 0 \\ 0 & 11-\lambda \end{bmatrix} - 15 \times det \begin{bmatrix} 15 & 0 \\ -5 & 11-\lambda \end{bmatrix} - (-5) \times det \begin{bmatrix} 15 & 18-\lambda \\ -5 & 0 \end{bmatrix} = 0$$

$$\lambda = 37.4889, 4.4955, 12.0157$$

Since all eigenvalues are greater than zero then the matrix is positive-definite.

The Cholesky Decomposition is applicable for the matrix.

Step 2: Compute the Cholesky factor *L*

The Cholesky factor L is obtained from the following formula:

$$LL^T = \underbrace{\begin{bmatrix} a & 0 & 0 \\ b & c & 0 \\ d & e & f \end{bmatrix}}_{L} \underbrace{\begin{bmatrix} a & b & d \\ 0 & c & e \\ 0 & 0 & f \end{bmatrix}}_{L^T} = \begin{bmatrix} a^2 & ab & ad \\ ab & b^2+c^2 & bd+ce \\ ad & bd+ce & d^2+e^2+f^2 \end{bmatrix}$$

Find the Cholesky factor L:

$$\underbrace{\begin{bmatrix} 25 & 15 & -5 \\ 15 & 18 & 0 \\ -5 & 0 & 11 \end{bmatrix}}_{A} = \underbrace{\begin{bmatrix} a^2 & ab & ad \\ ab & b^2+c^2 & bd+ce \\ ad & bd+ce & d^2+e^2+f^2 \end{bmatrix}}_{LL^T}$$

$$a^2 = 25 => \boxed{a = 5}$$

$$ab = 15 => (5)b = 15 => \boxed{b = 3}$$

$$ad = -5 => (5)d = -5 => \boxed{d = -1}$$

$$b^2 + c^2 = 18 => (3)^2 + c^2 = 18 => c^2 = 9 => \boxed{c = 3}$$

$$bd = ce = 0 => (3)(-1) + (3)e = 0 => 3e = 3 => \boxed{e = 1}$$

$$d^2 + e^2 + f^2 = 11 => (-1)^2 + (1)^2 + f^2 = 11 => f^2 = 9 => \boxed{f = 3}$$

We get the following formula:

$$LL^T = \underbrace{\begin{bmatrix} a & 0 & 0 \\ b & c & 0 \\ d & e & f \end{bmatrix}}_{L} \underbrace{\begin{bmatrix} a & b & d \\ 0 & c & e \\ 0 & 0 & f \end{bmatrix}}_{L^T} = \underbrace{\begin{bmatrix} 5 & 0 & 0 \\ 3 & 3 & 0 \\ -1 & 1 & 3 \end{bmatrix}}_{L} \underbrace{\begin{bmatrix} 5 & 3 & -1 \\ 0 & 3 & 1 \\ 0 & 0 & 3 \end{bmatrix}}_{L^T}$$

Step 3: $\boldsymbol{Solve\ L_y = b\ for\ y\ using\ forward\ substitution}$

$$\underbrace{\begin{bmatrix} 5 & 0 & 0 \\ 3 & 3 & 0 \\ -1 & 1 & 3 \end{bmatrix}\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}}_{Ly} = \underbrace{\begin{bmatrix} 35 \\ 33 \\ 6 \end{bmatrix}}_{b}$$

$$y_1 : 5y_1 = 35 => \boxed{y_1 = 7}$$

$$y_2 : 3y_1 + 3y_2 = 33 => 3\,(7) + 3y_2 = 33 => 3y_2 = 12 => \boxed{y_2 = 4}$$

$$y_3 : -y_1 + y_2 + 3y_3 = 6 => -\,(7) + (4) + 3y_3 = 6$$
$$=> 3y_3 = 9 => \boxed{y_3 = 3}$$

Step 4: $\boldsymbol{Solve\ L^T x = y\ for\ x\ using\ backward\ substitution}$

$$\underbrace{\begin{bmatrix} 5 & 3 & -1 \\ 0 & 3 & 1 \\ 0 & 0 & 3 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{L^T x} = \underbrace{\begin{bmatrix} 7 \\ 4 \\ 3 \end{bmatrix}}_{y}$$

$$x_3 : 3x_3 = 3 => \boxed{x_3 = 1}$$

$$x_2 : 3x_2 + x_1 = 4 => 3x_2 + (1) = 4 => 3x_2 = 3 => \boxed{x_2 = 1}$$

$$x_1 : 5x_1 + 3x_2 - x_3 = 7 => 5x_1 + 3\,(1) - (-1) = 7$$
$$=> 5x_1 = 5 => \boxed{x_1 = 1}$$

Step 5: Check answer satisfaction

$$x_1 = 1, x_2 = 1, x_3 = 1$$

$$\underbrace{\begin{bmatrix} 25 & 15 & -5 \\ 15 & 18 & 0 \\ -5 & 0 & 11 \end{bmatrix}}_{A} \quad \underbrace{\begin{bmatrix} 35 \\ 33 \\ 6 \end{bmatrix}}_{B}$$

$$25x_1 + 15x_2 - 5x_3 = 35$$
$$=> 25\,(1) + 15\,(1) - 5x\,(1) = 35$$
$$=> 25 + 15 - 5 = 35$$
$$=> (35 == 35) = True$$

MATLAB code

```
% Cholesky Decomposition:
clear;
clc;

syms x1 x2 x3
eq = [
    25*x1 + 15*x2 - 5*x3 == 35,
    15*x1 + 18*x2 + 0*x3  == 33,
    -5*x1 + 0*x2  + 11*x3 == 6,
    ];

[A, b] = equationsToMatrix(eq)
y = zeros(1, size(A, 2));
x = zeros(1, size(A, 2));

if ishermitian(logical(A))
    if all(real(eval(eig(A))) > 0)

        Ly_b = [chol(A, "lower") b]

        for i=1:size(Ly_b, 1)
            s = sum(Ly_b(i, 1:i-1).*y(1:i-1));
            y(i) = (Ly_b(i, end) - s)./Ly_b(i,i);
        end
```

```
    LTx_y = [chol(A, "upper") transpose(y)]

    for i=size(LTx_y, 1):-1:1
      s = sum(LTx_y(i, i+1:end-1).*x(i+1:end));
      x(i) = (LTx_y(i, end) - s)./LTx_y(i, i);
    end
  end
end
x
```

Output

```
>> CholeskyDecomposition

A =
[25, 15, -5]
[15, 18,  0]
[-5,  0, 11]

b =
35
33
 6

Ly_b =
[ 5, 0, 0, 35]
[ 3, 3, 0, 33]
[-1, 1, 3,  6]

LTx_y =
[5, 3, -1, 7]
[0, 3,  1, 4]
[0, 0,  3, 3]

x =
     1     1     1
```

Part 03: Comparing

We will compare the two algorithms based on time complexity and space complexity.

Gaussian Elimination

Time Complexity

To analyze its time complexity, we'll consider the algorithm in its basic form, which involves two main steps: Forward Elimination, Backward Substitution.

Forward Elimination

In this phase, the algorithm transforms the matrix into an upper triangular form. This is achieved by performing elementary row operations. For a matrix of size $n \times n$, the process involves the following steps for each row:

- Selecting a pivot element (typically the diagonal element in the current row).
- Making all elements below the pivot element zero, which involves subtracting a suitable multiple of the pivot row from each of the rows below it.

For each row i (from 1 to $n-1$), you need to perform operations on the subsequent $n-i$ rows. Each operation involves $n-i$ multiplications and subtractions. Therefore, the total number of operations for forward elimination is approximately:

$\sum_{i=1}^{n-1}(n-i)^2$

This summation is a series that can be simplified to $\frac{n^3}{3}$.

Backward Substitution

Once the matrix is in upper triangular form, the algorithm solves for the unknowns in reverse order. For each row i (from n down to 1), it involves a few operations per element to the left of the diagonal, leading to a total of about $\frac{n^2}{3}$.

Overall, the dominant factor in Gaussian elimination is the forward elimination phase, with a cubic time complexity of $O(n^3)$.

Space Complexity
The space complexity of Gaussian elimination is $O(n^2)$. since it requires storing an $n \times n$ matrix.

Cholesky Decomposition

Time Complexity
To analyze its time complexity, we'll consider the algorithm in its basic form, which involves three main steps: Decomposition, Forward Substitution, Backward Substitution.

Decomposition
The key operation in the decomposition is the computation of each element of L. For each row i (from 1 to n), the algorithm computes i elements (since L is lower triangular). Each element computation involves approximately i multiplications and i subtractions. Therefore, the total number of operations is approximately $\sum_{i=1}^{n} i^2$ which simplifies to $\frac{n(n+1)(2n+1)}{6}$. For large n, this summation indicates $O(n^3)$.

Forward/Backward Substitution
Both forward and backward substitutions have a time complexity of $O(n^2)$.

Space Complexity
The space complexity of Cholesky Decomposition is $O(n^2)$. since it requires storing an $n \times n$ matrix.

Gaussian Elimination vs Cholesky Decomposition

Since the time complexity and space complexity failed to determine which algorithm is better, we will compare them based on practical use.

Efficiency Comparison

- Both methods have cubic time complexity, but Cholesky Decomposition is generally faster in practice for matrices that meet its requirements (Hermitian, positive-definite).
- Cholesky is approximately twice as efficient as Gaussian elimination in terms of the number of operations because it only works with half of the matrix (due to its symmetry).

Use Comparison

- **Gaussian Elimination** is a more general method and can be applied to any square matrix. It's widely used in various applications but can be less stable numerically, especially for ill-conditioned matrices.
- **Cholesky Decomposition** is limited to Hermitian, positive-definite matrices. It's more efficient and numerically stable within its applicable domain, making it preferable for problems like optimization, simulations, and statistical computations where such matrices are common.

While both Gaussian Elimination and Cholesky Decomposition have similar time and space complexities, Cholesky Decomposition is generally more efficient for its specific use case. The choice between the two methods should be based on the nature of the matrix involved and the specific requirements of the problem at hand.

Conclusion

In this report, we have explored two algorithms for solving systems of linear equations, Gaussian Elimination and Cholesky Decomposition. These methods, each with their unique characteristics and computational complexities, play a crucial role in the field of numerical linear algebra and have wide-ranging applications in engineering, physics, and computer science.

Both algorithms share a cubic time complexity $O(n^3)$ and a quadratic space complexity $O(n^2)$, but Cholesky Decomposition often requires fewer operations, making it faster in practice for its applicable cases. The choice between these methods should be guided by the nature of the matrix involved and the specific requirements of the problem.

References

- Alexey Frolov. (n.d.) ‘Cholesky decomposition’, AlgoWiki, Available at: https://algowiki-project.org/en/Cholesky_decomposition (Accessed: 6 November 2023).

- Dr. Harish Garg. (n.d.), ‘MATLAB Code of Gauss Elimination Method’, YouTube, Available at: https://youtu.be/U_n9-YCURI0?si=kgqVN4PtAkav25DP (Accessed: 6 November 2023).

- The Organic Chemistry Tutor. (n.d.), ‘Gaussian Elimination & Row Echelon Form’, YouTube, Available at: https://youtu.be/eDb6iugi6Uk?si=KX2q93HhN9BE_rxp (Accessed: 7 November 2023).

- SkanCity Academy. (n.d.), ‘🟢04 - Cholesky Decomposition Method (Algorithm)’, YouTube, Available at: https://youtu.be/CBkdQSu7NnY?si=3aLWtgr3ti3hDpcW (Accessed: 10 November 2023).

- Counting on Justice. (n.d.), ‘Gaussian Elimination and Complexity’, YouTube, Available at: https://youtu.be/OfWjaa6aACM?si=sBOFLcoD-CXktqR3 (Accessed: 10 November 2023).

- The MathWorks, Inc. (1994-2023), Available at: https://www.mathworks.com/?s_tid=gn_logo (Accessed: 10 November 2023).