FCM Project 2

Chase Wiederstein

October, 25th 2023

1 Executive Summary

In this report, the focus is on solving the system $Ax = b$ using LU decomposition. I will analyze solving this system with different types of matrices using the no pivoting and partial pivoting methods. The types of matrices included are symmetric positive definite and randomly generated. To do this, I will employ LU decomposition algorithms that separates the given matrix A into a lower triangular matrix L and an upper triangular matrix U. Note that the LU decomposition algorithms will not save the 0's and 1's to save computational space. I will also implement forward and backward substitution to solve for our computed solution $\tilde{x}$. Next, I will show relative errors of these methods using the one, two, and Frobenius norms to analyze the accuracy of each method and the type of matrix used. Finally, I implement a correctness test for a simple 3×3 matrix with a given b vector to solve for the solution x.

2 Description of the Algorithms and Implementation

I was first tasked with randomly generating a non-singular matrix $A \in \mathbb{R}^{n \times n}$ of size $n = 20$ and $n = 200$. I then created a different matrix A such that A is symmetric positive definite (SPD) and generated from a nonsingular lower triangular matrix L_1 where $A = L_1 L_1^T$. The reason for creating two different matrices is to compare the results by various tests which will be shown below. I then randomly generated a vector x of size n to solve for b in the equation $Ax = b$. In order to perform LU decomposition on either matrix, I first needed to determine the correct method of LU decomposition to use. If the matrix is diagonally dominant by rows, I perform the no-pivoting method. If the matrix is not diagonally dominant, I perform partial pivoting.

2.1 No Pivoting Method

For the no-pivoting method, I translate my SPD and randomly generated matrix into a lower triangular matrix L and an upper triangular matrix U. Next, with the vector b I solved for above, I perform forward substitution ($Ly = b$) so solve for y and then solve for $\tilde{x}$ using backward substitution ($U\tilde{x} = y$). To check the factorization accuracy, I computed $\frac{||A-LU||}{||A||}$ for both the one norm and the Frobenius norm. I also computed the relative error $\frac{||x-\tilde{x}||}{||x||}$ and checked the accuracy via the residual $b - A\tilde{x}$ and $\frac{||b-A\tilde{x}||}{||b||}$ for the vector one and two norm. The results for each matrix are found in tables 1-3 in section 5.

2.2 Partial Pivoting

The partial pivoting method will be employed for non-diagonally dominant and well-condtioned matrices. This algorithm works by finding the maximum absolute value of each column and moving each of the largest values to the diagonal. While performing these row swaps, I also update my permutation matrix P. During this process, we also update our vector b. Once we have our permutation matrix filled out, we then perform LU decomposition such that $PA = LU$. However, to avoid saving unnecessary values for P, we simply create a permutation vector p that saves each of the row swaps. Just as I did for the no pivoting method, I checked the factorization accuracy with $\frac{||PA-LU||}{||A||}$ for both the one norm and the Frobenius norm. I also computed the relative error $\frac{||x-\tilde{x}||}{||x||}$ and checked the accuracy via the residual $b - A\tilde{x}$ and $\frac{||b-A\tilde{x}||}{||b||}$ for the vector one and two norm. The results for the partial pivoting matrices are found in tables 4-6 in section 6.

2.3 Correctness Test Task

In order to check our code, we were tasked with solving the system below.

$$A_{\text{test}} = \begin{bmatrix} 2 & 1 & 0 \\ -4 & 0 & 4 \\ 2 & 5 & -10 \end{bmatrix} b_{\text{test}} = \begin{bmatrix} 3 \\ 0 \\ 17 \end{bmatrix}$$

For this system, we will implement the partial pivoting method. For our output, we get:

$$Matrix_{\text{p}} = \begin{bmatrix} -4 & 0 & 4 \\ -.5 & 5 & -8 \\ -.5 & .2 & 3.6 \end{bmatrix} p = [2, 3, 1]$$

Where $Matrix_p$ is the combination of the L and U from LU decomposition and p is the pivoting vector created from the permutation matrix P. The computed solution yields $\tilde{x} = \begin{bmatrix} -1/9 \\ 29/9 \\ -1/9 \end{bmatrix}$.

3 Conclusion

In summary, we can assume my LU decomposition algorithms are accurate and are able to effectively solve the system $Ax = b$. All of the relative errors remained small and consistent for each of the various methods, types of matrices, sizes and norms. The LU decomposition methods of no pivoting and partial pivoting will produce accurate results $\tilde{x}$ that are good representations of the true solution x.

4 Tables and Figures

Table 1: Results for $\frac{||A-LU||}{||A||}$ for No Pivoting

Matrix:	Size:	Norm:	Average Error:
SPD	20×20	Frobenius Norm	7.43546e-16
SPD	20×20	One Norm	2.1083e-16
SPD	200×200	Frobenius Norm	6.4783e-16
SPD	200×200	One Norm	8.75015e-16
Random	20×20	Frobenius Norm	6.52208e-16
Random	20×20	One Norm	1.44066e-16
Random	200×200	Frobenius Norm	7.98105e-16
Random	200×200	One Norm	9.8488e-16

Table 2: Results for $\frac{||x-\tilde{x}||}{||x||}$ for No Pivoting

Matrix:	Size:	Norm:	Average Error:
SPD	20×20	Two Norm	3.3716e-16
SPD	20×20	One Norm	2.89004e-16
SPD	200×200	Two Norm	8.0095e-16
SPD	200×200	One Norm	6.44461e-16
Random	20×20	Two Norm	2.35737e-16
Random	20×20	One Norm	1.96457e-16
Random	200×200	Two Norm	8.60258e-16
Random	200×200	One Norm	6.68582e-16

Table 3: Results for $\frac{||b-A\tilde{x}||}{||b||}$ for No Pivoting

Matrix:	Size:	Norm:	Average Error:
SPD	20×20	Two Norm	3.30888e-16
SPD	20×20	One Norm	2.39233e-16
SPD	200×200	Two Norm	8.02795e-16
SPD	200×200	One Norm	6.53053e-16
Random	20×20	Two Norm	3.00639e-16
Random	20×20	One Norm	2.15093e-16
Random	200×200	Two Norm	8.51683e-16
Random	200×200	One Norm	6.68946e-16

Table 4: Results for $\frac{||PA-LU||}{||A||}$ for Partial Pivoting

Matrix:	Size:	Norm:	Average Error:
SPD	20×20	Frobenius Norm	6.66581e-16
SPD	20×20	One Norm	2.58143e-16
SPD	200×200	Frobenius Norm	7.30837e-16
SPD	200×200	One Norm	6.01651e-16
Random	20×20	Frobenius Norm	9.24872e-16
Random	20×20	One Norm	1.96926e-16
Random	200×200	Frobenius Norm	7.32694e-16
Random	200×200	One Norm	5.6667e-16

Table 5: Results for $\frac{||x-\tilde{x}||}{||x||}$ for Partial Pivoting

Matrix:	Size:	Norm:	Average Error:
SPD	20×20	Two Norm	5.1126e-14
SPD	20×20	One Norm	4.31394e-14
SPD	200×200	Two Norm	1.142e-11
SPD	200×200	One Norm	9.51299e-12
Random	20×20	Two Norm	2.29419e-16
Random	20×20	One Norm	1.98135e-16
Random	200×200	Two Norm	1.54518e-15
Random	200×200	One Norm	1.365e-15

Table 6: Results for $\frac{||b-A\tilde{x}||}{||b||}$ for Partial Pivoting

Matrix:	Size:	Norm:	Average Error:
SPD	20×20	Two Norm	1.52665e-16
SPD	20×20	One Norm	1.2521e-16
SPD	200×200	Two Norm	1.54056e-15
SPD	200×200	One Norm	1.30724e-15
Random	20×20	Two Norm	1.86458e-16
Random	20×20	One Norm	1.39832e-16
Random	200×200	Two Norm	7.76046e-16
Random	200×200	One Norm	8.71519e-16