Solving Systems of Equations, Errors and Explorations

David Tran and Spencer Kelly

March 9, 2024

Abstract

1 Introduction

2 The PA = LU factorization method for linear systems

2.1 Why is PA = LU needed for solving linear systems approximately?

When solving linear systems of the form $Ax = b$, we begin by gaussian elimination of the matrix A, followed by back substitution, and ultimately arrive at our solution. However, when a particular matrix A is being used for multiple iterations, the overhead involved can become quite an obstacle. This is because the process of Gaussian elimination is a computationally expensive process, with complexity on the order $O(n^3)$. But with $PA = LU$ factorization, we essentially remove the overhead involved with Gaussian elimination, for all but the first iteration, by rewriting the matrix A in terms of the upper and lower matrices L, and U, respectively. Thus, for every subsequent iteration involving the same matrix, we need not perform gaussian elimination, since L and U allow us to immediately begin performing the second step of solving; back-substitution, which only has complexity $O(n^2)$.

However, when performing naive Gaussian elimination to form the matrices L and U, we are at risk of swamping, or the existence of a zero-pivot. With the help of a permutation matrix P, we can now swap rows and columns, to mitigate the propogation of errors due to multiplying rows by large values, and avoid zero-pivots. The permutation matrix P keeps track of the swapping of rows and columns, so that the linear system itself remains unperturbed (however it is now written $PAx = Pb$).

A great example would be the system $Ax = b$ with $A = \begin{pmatrix} 1 & 2 & 4 \\ 3 & 8 & 14 \\ 2 & 6 & 13 \end{pmatrix}$ and $b = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$ To begin, we swap rows 1 and 2, since we want our multiplication to be by the smallest values possible during Gaussian elimination. When doing this, we update our permutation matrix: $P = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$, and we can subsequently perform gaussian elimination to yield the matrix $\begin{pmatrix} 3 & 8 & 14 \\ (\frac{1}{3}) & \frac{-2}{3} & \frac{-2}{3} \\ (\frac{2}{3}) & \frac{10}{3} & \frac{25}{3} \end{pmatrix}$, where the brackets around the values in the place of what should be 0 represents the multiplier for the elimination of the row, (important for bookkeeping when doing permutations).

Next we permute the rows 2 and 3, so that $P = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$, and once again performing Gaussian elimination we obtain the matrix $\begin{pmatrix} 3 & 8 & 14 \\ (\frac{2}{3}) & \frac{10}{3}\frac{25}{3} & \\ (\frac{1}{3})(-\frac{1}{5}) & & \end{pmatrix}$

2.2 How to identify systems Ax = b for which PA = LU is not suited

2.3 Larger applications of PA = LU factorization

3 Iterative solution of systems of linear equations

3.1 Solving an equation for n = 100,000

3.2 Comparison of PA = LU and Jacobi Iteration

3.3 Why is solving such large systems important in applications?

4 Implement Newton's method for multiple variables

4.1 Implement Newton's method for systems using vectorization

4.2 Testing

4.3 Challenging Example

5 Summary

6 Appendices

6.1 Code

6.2 Plots

7 Code

8 Summary

8.1 Results

8.2 Team Description

8.3 Future Explorations

8.4 References

Appendix