Section 2: Gaussian elimination and LU solvers

February 25, 2019

In this section, we will address how to solve the equation:

$$Ax = b$$

where A is a n by n invertible matrix. Given b (or multiple b's), we would like to find x. Many problems of numerical analysis (and math in general) involve some inversion of a linear system. For now, we will look into a direct method of solving a linear system. Later in the course, we will introduce the concept of iterative solvers as well.

1 Solving triangular systems

Before we attempt to solve the general case, it is wise to first see how we can solve $Ax = b$ assuming some special structure to A. The strategy later would be to reduce the general to this special case as a means to solve the problem overall. Let us considering following linear system of equations:

$$\begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \cdots & a_{1,n} \\ 0 & a_{2,2} & a_{2,3} & \cdots & a_{2,n} \\ 0 & 0 & a_{3,3} & \cdots & a_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_{n,n} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{bmatrix}$$

The matrix A is a triangular matrix. All the entries below the main diagonal are equal to zero. Having this particular zero structure certainly simplifies the solution. From the last equation, we can already find an expression for the last variable x_n, i.e.

$$x_n = \frac{b_n}{a_{n,n}}$$

Now that we know what x_n is, looking at second last equation:

$$a_{n-1,n-1}x_{n-1} + a_{n-1,n}x_1 = b_{n-1}$$

we also have an expression for x_{n-1} in terms of known quantities:

$$x_{n-1} = \frac{1}{a_{n-1,n-1}} \left(b_{n-1} - a_{n-1,n}x_1\right)$$

I guess the idea should be clear now. The general process looks as follows:

$$\begin{aligned}
x_n &= \frac{1}{a_{n,n}} b_n \\
x_{n-1} &= \frac{1}{a_{n-1,n-1}} \left(b_{n-1} - a_{n-1,n}x_1\right) \\
&\vdots \\
x_p &= \frac{1}{a_{pp}} \left(b_p - \sum_{k=p+1}^{n} a_{1,k}x_k\right) \\
&\vdots \\
x_1 &= \frac{1}{a_{11}} \left(b_1 - \sum_{k=2}^{n} a_{1,k}x_k\right)
\end{aligned}$$

This whole process is called back-substitution. Notice that if one of the diagonal entries are equal zero, we will run into trouble in the algorithm. This makes sense, because A would be no longer invertible. A similar algorithm can also be conveived for lower triangular matrices:

$$\begin{bmatrix} a_{1,1} & 0 & 0 & \cdots & 0 \\ a_{2,1} & a_{2,2} & 0 & \cdots & 0 \\ a_{3,1} & a_{3,2} & a_{3,3} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & a_{n,3} & \cdots & a_{n,n} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{bmatrix}$$

The corresponding iterations are called forward substitution. We will leave to the reader to work out those details.

2 LU factorization: gaussian elimination in disguise

In a considerable number cases (but certainly not all, as we will see!) it is possible to factor a square invertible matrix into a product of a lower triangular matrix and an upper triangular matrix. The factorization looks like:

$$A = LU = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ l_{2,1} & 1 & 0 & \cdots & 0 \\ l_{3,1} & l_{3,2} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ l_{n,1} & l_{n,2} & l_{n,3} & \cdots & 1 \end{bmatrix} \begin{bmatrix} u_{1,1} & u_{1,2} & u_{1,3} & \cdots & u_{1,n} \\ 0 & u_{2,2} & u_{2,3} & \cdots & u_{2,n} \\ 0 & 0 & u_{3,3} & \cdots & u_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & u_{n,n} \end{bmatrix}.$$

Notice above that the main diagonal entries of L are all equal to one. This factorization of a matrix may appear to you as some new concept, but rest assured, you have been already working with this factorization indirectly in your basic linear algebra classes.

Gaussian elimination. It turns out that the LU factorizatio is basically Gaussian elimination in disguise. To refresh, let us consider the following 3 by 3 system of equations:

$$\begin{array}{rcrcrcr} 2x_1 & + & 3x_2 & + & 5x_3 & = & 4 \\ 4x_1 & + & 2x_2 & + & 2x_3 & = & 1 \\ -2x_1 & + & 3x_2 & + & x_3 & = & -2 \end{array}$$

How do you solve this system? As you may know, one way to go about it is to do a sequence of eliminations to reduce the problem to something simpler. Simultaneously, we can do the following two operations:

- Multiply the first equation by 2 and subtract it from the second equation.

- Multiply the first equation by unity and add it to the first equation

This yields the equivalent system:

$$\begin{array}{rcrcrcr} 2x_1 & + & 3x_2 & + & 5x_3 & = & 4 \\ & & -2x_2 & - & 8x_3 & = & -7 \\ & & 6x_2 & + & 6x_3 & = & 2 \end{array}$$

Next we do the following operations:

- Multiply the second equation by three and add it to the second equation.

This yields the equivalent system:

$$\begin{array}{rcrcrcr} 2x_1 & + & 3x_2 & + & 5x_3 & = & 4 \\ & & -2x_2 & - & 8x_3 & = & -7 \\ & & & & 30x_3 & = & 19 \end{array}$$

Now we have an upper triangular system. The back-substitution algorithm can be used to further solve the problem.

The LU factorization. The row reductions steps applied to the system of equations are invertible linear operations and can be expressed in matrix language. Indeed, in matrix notation the example of the previous paragraph is written as:

$$\begin{bmatrix} 2 & 3 & 5 \\ 4 & 2 & 2 \\ -2 & 3 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4 \\ 1 \\ -2 \end{bmatrix}$$

The first elimination round can be expresseds as:

$$\begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 & 3 & 5 \\ 4 & 2 & 2 \\ -2 & 3 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 \\ 1 \\ -2 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 3 & 5 \\ 0 & -2 & -8 \\ 0 & 6 & 6 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4 \\ -7 \\ 2 \end{bmatrix}$$

The second elimination can be expressed as:

$$
\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 3 & 1 \end{bmatrix}
\begin{bmatrix} 2 & 3 & 5 \\ 0 & -2 & -8 \\ 0 & 6 & 6 \end{bmatrix}
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}
=
\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 3 & 1 \end{bmatrix}
\begin{bmatrix} 4 \\ -7 \\ 2 \end{bmatrix}
$$

$$
\begin{bmatrix} 2 & 3 & 5 \\ 0 & -2 & -8 \\ 0 & 0 & 30 \end{bmatrix}
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}
=
\begin{bmatrix} 4 \\ -7 \\ 19 \end{bmatrix}
$$

As we have mentioned before, the LU factorization is basically this process in disguise. It may already be clear what U should be, but where is the L? Well, let us look at the first and second elimination step combined:

$$
\begin{bmatrix} 2 & 3 & 5 \\ 0 & -2 & -8 \\ 0 & 0 & 30 \end{bmatrix}
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}
=
\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 3 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 4 \\ 1 \\ -2 \end{bmatrix}
$$

Let us rewrite this as follows:

$$
\begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}^{-1}
\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 3 & 1 \end{bmatrix}^{-1}
\begin{bmatrix} 2 & 3 & 5 \\ 0 & -2 & -8 \\ 0 & 0 & 30 \end{bmatrix}
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}
=
\begin{bmatrix} 4 \\ 1 \\ -2 \end{bmatrix}
$$

The inverses for the above matrices are easy to find, they are simply:

$$
\begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}^{-1}
=
\begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix},
\qquad
\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 3 & 1 \end{bmatrix}^{-1}
=
\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -3 & 1 \end{bmatrix}
$$

Multiplying these two matrices is also easy:

$$
\begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -3 & 1 \end{bmatrix}
=
\begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & -3 & 1 \end{bmatrix}
$$

We have obtained the LU factorization:

$$
\begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & -3 & 1 \end{bmatrix}
\begin{bmatrix} 2 & 3 & 5 \\ 0 & -2 & -8 \\ 0 & 0 & 30 \end{bmatrix}
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}
=
\begin{bmatrix} 4 \\ 1 \\ -2 \end{bmatrix}
$$

The "general" case. Let us generalize this process. For the matrix:

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & a_{2,3} & \cdots & a_{2,n} \\ a_{3,1} & a_{3,2} & a_{3,3} & \cdots & a_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & a_{n,3} & \cdots & a_{n,n} \end{bmatrix}$$

The first elimination cycle is done by applying the transformation:

$$L_1 = \begin{bmatrix} 1 & 0 & 0 & & 0 \\ \frac{a_{2,1}}{a_{1,1}} & 1 & 0 & \cdots & 0 \\ \frac{a_{3,1}}{a_{1,1}} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{a_{n,1}}{a_{1,1}} & 0 & 0 & \cdots & 1 \end{bmatrix}$$

This is called a *Gauss transform*. The inverse of a Gauss transform is easy to compute:

$$L_1^{-1} = \begin{bmatrix} 1 & 0 & 0 & & 0 \\ -\frac{a_{2,1}}{a_{1,1}} & 1 & 0 & \cdots & 0 \\ -\frac{a_{3,1}}{a_{1,1}} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -\frac{a_{n,1}}{a_{1,1}} & 0 & 0 & \cdots & 1 \end{bmatrix}$$

Now, we apply the oldest trick in the book:

$$A = L_1 L_1^{-1} A = L_1(L_1^{-1} A) = L_1 A^{(1)}.$$

It should be clear that:

$$\begin{aligned} A^{(1)} &= \begin{bmatrix} 1 & 0 & 0 & & 0 \\ \frac{a_{2,1}}{a_{1,1}} & 1 & 0 & \cdots & 0 \\ \frac{a_{3,1}}{a_{1,1}} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{a_{n,1}}{a_{1,1}} & 0 & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & a_{2,3} & \cdots & a_{2,n} \\ a_{3,1} & a_{3,2} & a_{3,3} & \cdots & a_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & a_{n,3} & \cdots & a_{n,n} \end{bmatrix} \\ &= \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \cdots & a_{1,n} \\ 0 & a_{2,2}^{(1)} & a_{2,3}^{(1)} & \cdots & a_{2,n}^{(1)} \\ 0 & a_{3,2}^{(1)} & a_{3,3}^{(1)} & \cdots & a_{3,n}^{(1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n,2}^{(1)} & a_{n,3}^{(1)} & \cdots & a_{n,n}^{(1)} \end{bmatrix} \end{aligned}$$

where:

$$\begin{bmatrix} a_{2,2}^{(1)} & a_{2,3}^{(1)} & \cdots & a_{2,n}^{(1)} \\ a_{3,2}^{(1)} & a_{3,3}^{(1)} & \cdots & a_{3,n}^{(1)} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,2}^{(1)} & a_{n,3}^{(1)} & \cdots & a_{n,n}^{(1)} \end{bmatrix} = \begin{bmatrix} a_{2,2} & a_{2,3} & \cdots & a_{2,n} \\ a_{3,2} & a_{3,3} & \cdots & a_{3,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,2} & a_{n,3} & \cdots & a_{n,n} \end{bmatrix} - \begin{bmatrix} \frac{a_{2,1}}{a_{1,1}} \\ \frac{a_{3,1}}{a_{1,1}} \\ \vdots \\ \frac{a_{n,1}}{a_{1,1}} \end{bmatrix} \begin{bmatrix} a_{1,2} & a_{1,3} & \cdots & a_{1,n} \end{bmatrix}$$

The equation above appears so frequently in linear algebra that is has a name: the *Schur complement*. The first step of Gaussian elimination is complete. The matrix A has been factored as:

$$A = L_1 A^{(1)} = \begin{bmatrix} 1 & 0 & 0 & & 0 \\ \frac{a_{2,1}}{a_{1,1}} & 1 & 0 & \cdots & 0 \\ \frac{a_{3,1}}{a_{1,1}} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{a_{n,1}}{a_{1,1}} & 0 & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \cdots & a_{1,n} \\ 0 & a_{2,2}^{(1)} & a_{2,3}^{(1)} & \cdots & a_{2,n}^{(1)} \\ 0 & a_{3,2}^{(1)} & a_{3,3}^{(1)} & \cdots & a_{3,n}^{(1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n,2}^{(1)} & a_{n,3}^{(1)} & \cdots & a_{n,n}^{(1)} \end{bmatrix}$$

We can now proceed to the second elimination cycle. Our second Gauss transform looks like as follows:

$$L_2 = \begin{bmatrix} 1 & 0 & 0 & & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & \frac{a_{3,2}^{(1)}}{a_{2,2}^{(1)}} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \frac{a_{n,2}^{(1)}}{a_{2,2}^{(1)}} & 0 & \cdots & 1 \end{bmatrix}$$

Indeed,

$$\begin{aligned} A &= L_1 A^{(1)} \\ &= L_1 L_2 A^{(2)} \\ &= \begin{bmatrix} 1 & 0 & 0 & & 0 \\ \frac{a_{2,1}}{a_{1,1}} & 1 & 0 & \cdots & 0 \\ \frac{a_{3,1}}{a_{1,1}} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{a_{n,1}}{a_{1,1}} & 0 & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & \frac{a_{3,2}^{(1)}}{a_{2,2}^{(1)}} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \frac{a_{n,2}^{(1)}}{a_{2,2}^{(1)}} & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \cdots & a_{1,n} \\ 0 & a_{2,2}^{(1)} & a_{2,3}^{(1)} & \cdots & a_{2,n}^{(1)} \\ 0 & 0 & a_{3,3}^{(2)} & \cdots & a_{3,n}^{(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & a_{n,3}^{(2)} & \cdots & a_{n,n}^{(2)} \end{bmatrix} \end{aligned}$$

where:

$$\begin{bmatrix} a_{3,3}^{(2)} & \cdots & a_{3,n}^{(2)} \\ \vdots & \ddots & \vdots \\ a_{n,3}^{(2)} & \cdots & a_{n,n}^{(2)} \end{bmatrix} = \begin{bmatrix} a_{3,3}^{(1)} & \cdots & a_{3,n}^{(1)} \\ \vdots & \ddots & \vdots \\ a_{n,3}^{(1)} & \cdots & a_{n,n}^{(1)} \end{bmatrix} - \begin{bmatrix} \frac{a_{3,2}^{(1)}}{a_{2,2}^{(1)}} \\ \vdots \\ \frac{a_{3,2}^{(1)}}{a_{2,2}^{(1)}} \end{bmatrix} \begin{bmatrix} a_{2,3}^{(1)} & \cdots & a_{2,n}^{(1)} \end{bmatrix}$$

Multiplying two Gauss transforms is easy:

$$L_1 L_2 = \begin{bmatrix} 1 & 0 & 0 & & 0 \\ \frac{a_{2,1}}{a_{1,1}} & 1 & 0 & \cdots & 0 \\ \frac{a_{3,1}}{a_{1,1}} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{a_{n,1}}{a_{1,1}} & 0 & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & \frac{a_{3,2}^{(1)}}{a_{2,2}^{(1)}} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \frac{a_{n,2}^{(1)}}{a_{2,2}^{(1)}} & 0 & \cdots & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & & 0 \\ \frac{a_{2,1}}{a_{1,1}} & 1 & 0 & \cdots & 0 \\ \frac{a_{3,1}}{a_{1,1}} & \frac{a_{3,2}^{(1)}}{a_{2,2}^{(1)}} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{a_{n,1}}{a_{1,1}} & \frac{a_{n,2}^{(1)}}{a_{2,2}^{(1)}} & 0 & \cdots & 1 \end{bmatrix}$$

Overview. Let us summarize the first two steps we have done so far. From there it should be clear how the remaining steps are executed. The first two steps were:

$$
\begin{aligned}
A &= \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & a_{2,3} & \cdots & a_{2,n} \\ a_{3,1} & a_{3,2} & a_{3,3} & \cdots & a_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & a_{n,3} & \cdots & a_{n,n} \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ \frac{a_{2,1}}{a_{1,1}} & 1 & 0 & \cdots & 0 \\ \frac{a_{3,1}}{a_{1,1}} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{a_{n,1}}{a_{1,1}} & 0 & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \cdots & a_{1,n} \\ 0 & a_{2,2}^{(1)} & a_{2,3}^{(1)} & \cdots & a_{2,n}^{(1)} \\ 0 & a_{3,2}^{(1)} & a_{3,3}^{(1)} & \cdots & a_{3,n}^{(1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n,2}^{(1)} & a_{n,3}^{(1)} & \cdots & a_{n,n}^{(1)} \end{bmatrix} \\
&= \begin{bmatrix} 1 & 0 & 0 & & 0 \\ \frac{a_{2,1}}{a_{1,1}} & 1 & 0 & \cdots & 0 \\ \frac{a_{3,1}}{a_{1,1}} & \frac{a_{3,2}^{(1)}}{a_{2,2}^{(1)}} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{a_{n,1}}{a_{1,1}} & \frac{a_{n,2}^{(1)}}{a_{2,2}^{(1)}} & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \cdots & a_{1,n} \\ 0 & a_{2,2}^{(1)} & a_{2,3}^{(1)} & \cdots & a_{2,n}^{(1)} \\ 0 & 0 & a_{3,3}^{(2)} & \cdots & a_{3,n}^{(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & a_{n,3}^{(2)} & \cdots & a_{n,n}^{(2)} \end{bmatrix}
\end{aligned}
$$

where:

$$
\begin{bmatrix} a_{2,2}^{(1)} & a_{2,3}^{(1)} & \cdots & a_{2,n}^{(1)} \\ a_{3,2}^{(1)} & a_{3,3}^{(1)} & \cdots & a_{3,n}^{(1)} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,2}^{(1)} & a_{n,3}^{(1)} & \cdots & a_{n,n}^{(1)} \end{bmatrix} = \begin{bmatrix} a_{2,2} & a_{2,3} & \cdots & a_{2,n} \\ a_{3,2} & a_{3,3} & \cdots & a_{3,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,2} & a_{n,3} & \cdots & a_{n,n} \end{bmatrix} - \begin{bmatrix} \frac{a_{2,1}}{a_{1,1}} \\ \frac{a_{3,1}}{a_{1,1}} \\ \vdots \\ \frac{a_{n,1}}{a_{1,1}} \end{bmatrix} \begin{bmatrix} a_{1,2} & a_{1,3} & \cdots & a_{1,n} \end{bmatrix}
$$

$$
\begin{bmatrix} a_{3,3}^{(2)} & \cdots & a_{3,n}^{(2)} \\ \vdots & \ddots & \vdots \\ a_{n,3}^{(2)} & \cdots & a_{n,n}^{(2)} \end{bmatrix} = \begin{bmatrix} a_{3,3}^{(1)} & \cdots & a_{3,n}^{(1)} \\ \vdots & \ddots & \vdots \\ a_{n,3}^{(1)} & \cdots & a_{n,n}^{(1)} \end{bmatrix} - \begin{bmatrix} \frac{a_{3,2}^{(1)}}{a_{2,2}^{(1)}} \\ \vdots \\ \frac{a_{3,2}^{(1)}}{a_{2,2}^{(1)}} \end{bmatrix} \begin{bmatrix} a_{2,3}^{(1)} & \cdots & a_{2,n}^{(1)} \end{bmatrix}
$$

Notice that the identity matrix on the left is slowly shaping into the L matrix, while A is being transformed into U. We will leave the extrapolation of the remaining steps to the reader.

Solving the linear system. So now that we have factored $A = LU$, how to we proceed in solving $Ax = b$? Well, that would be easy. From:

$$LUx = b$$

Introduce $y = Ux$ to obtain:

$$Ly = b$$

First obtain y using forward substitution. Then solve $y = Ux$ using back-substitution.

Disclaimer: LU factorization is not always possible. Look at the following matrix:

$$\begin{bmatrix} 0 & 3 & 5 \\ 4 & 2 & 2 \\ -2 & 3 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

This matrix does not have an LU factorization. Apply the first step of Gaussian elimination, and you will see a division by zero immediately. The matrix is invertible, so what is really going on? Well, nothing really. We just need to re-order the linear equations. Notice that we have the equations:

$$\begin{array}{rcrcrcl} & & 3x_2 & + & 5x_3 & = & 1 \\ 4x_1 & + & 2x_2 & + & 2x_3 & = & 2 \\ -2x_1 & + & 3x_2 & + & x_3 & = & 3 \end{array}$$

But this is the same as:

$$\begin{array}{rcrcrcl} 4x_1 & + & 2x_2 & + & 2x_3 & = & 2 \\ & & 3x_2 & + & 5x_3 & = & 1 \\ -2x_1 & + & 3x_2 & + & x_3 & = & 3 \end{array}$$

This re-ordering trick is the way to go in general (if we cannot fix it with a re-ordering, we actually have non-invertible matrix). The re-ordering of equations can be expressed in matrix language. Let:

$$P = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

What we have done is that we have multiplied our linear system by:

$$P \begin{bmatrix} 0 & 3 & 5 \\ 4 & 2 & 2 \\ -2 & 3 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = P \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}.$$

It turns out that this re-ordering of equations is not only required to make an LU factorization feasible, but also necessary in order to contain numerical issues.

3 Numerical aspects and the need for pivoting

So far we have not discussed the numerical aspect of implementing an LU factorization code in practice. It turns out that there are quite some stability issues to be concerned about. Let us look at the following example:

$$A = \begin{bmatrix} 10^{-20} & 1 \\ 1 & 1 \end{bmatrix}$$

In exact arithmetic, we obtain:

$$L = \begin{bmatrix} 1 & 0 \\ 10^{20} & 1 \end{bmatrix}, \quad U = \begin{bmatrix} 10^{-20} & 1 \\ 0 & 1 - 10^{20} \end{bmatrix}$$

In floating point arithmetic, we obtain

$$\hat{L} = \begin{bmatrix} 1 & 0 \\ 10^{20} & 1 \end{bmatrix}, \quad \hat{U} = \begin{bmatrix} 10^{-20} & 1 \\ 0 & -10^{20} \end{bmatrix}$$

Notice that the (2,2)-entry of U has been modified to 10^{20}. The effect of the small number 1 has been absorbed by the larger number 10^{20}. If the machine precision is 10^{-16}, the relative error between 10^{20} and $1 - 10^{20}$ is after all too small to be captured with floating point numbers. On first hand, this little perturbation does not seem like an issue. But if we multiply $\hat{L}$ and $\hat{U}$ with each other retrieve our original matrix, we see that (under exact arithmetic again):

$$\hat{A} = \hat{L}\hat{U} = \begin{bmatrix} 10^{-20} & 1 \\ 1 & 0 \end{bmatrix}$$

This is a drastic change! Catostrophic cancellation has occured. The culprit causing these huge discrepencies is division by small numbers. These divisions generate large entries in L and U, causing all kinds of headaches.

The numbers which we divide by are called *pivots*. The problem can be resolved by selecting another pivot through exchanging rows or columns, in fact). This is called *pivoting*. Indeed, by choosing:

$$P = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$PA = \begin{bmatrix} 1 & 1 \\ 10^{-20} & 1 \end{bmatrix} = LU = \begin{bmatrix} 1 & 0 \\ 10^{-20} & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 1 - 10^{-20} \end{bmatrix}$$

Under floating point arithmetic, we would have:

$$\hat{L} = \begin{bmatrix} 1 & 0 \\ 10^{-20} & 1 \end{bmatrix}, \quad \hat{U} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

Again in the (2,2)-entry of U, the smaller number 10^{-20} is absorbed by the much larger number 1. But this time around, things are not as bad (under exact arithmetic again):

$$P\hat{A} = \hat{L}\hat{U} = \begin{bmatrix} 1 & 0 \\ 10^{-20} & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 10^{-20} & 10^{-20} + 1 \end{bmatrix}$$

It is important to note that this phenomen we see here is not a consequence of ill conditioning. You can check for yourself that the matrix in our example is well conditioned. The problem is really in the algorithm that we have proposed. In fact, the standard LU factorization algorithm is numerically unstable. Pivoting is really needed to stabilize the algorithm. This is what we will discuss next.

4 LU factorization with partial pivoting

The minimize the effects of floating point arithmetic, we must choose the largest pivot possible through an appropriate row interchange. This should be done at every step of Gaussian elimination. When this is done, it turns out that we can find a permutation matrix P such that:

$$PA = LU \tag{1}$$

Let us talk about permutation matrices first.

Permutation matrices. When we would like to describe row interchanges or column interchanges of matrices, permutation matrices offer a convenient notation to describe them. A permutation matrix is basically some re-ordering (i.e. a permutation) of the columns of the idenity identity matrix. Here is an example:

$$I = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad P = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

The fourth column of the identity matrix is placed first, the third column is placed second, the first column is placed third, and the second column is placed fourth. Multiplying a matrix A by P from the right will exchange the columns of A in that second specific order. Multiplying a matrix from the left would exchange the rows. Please try it yourself to be convinced.

Permutation matrices are nice. The inverse of a permutation matrix is its transpose, i.e. $P^T P = I$. Also, multiplying two permutation matrices yields another permutation matrix. The P matrix in (1) is, in fact, obtained from multiplications of simple permutation matrices: so-called exchange permutations. These permutations just exchange or "swap" to columns/rows of matrix. Here is an example of an exchange permutation:

$$P = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

It swaps the first two columns/rows of a matrix. Our previous example was not an exchange permutation. Exchange permutations are used to describe the pivoting.

Partial pivoting through an example. Let us look at the following matrix:

$$A = \begin{bmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ 6 & 7 & 9 & 8 \end{bmatrix}$$

as an example to illustrate how partial pivoting works. The general algorithm should be clear from the steps we apply to this specifc example.

$$\begin{aligned}
A &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ 6 & 7 & 9 & 8 \end{bmatrix} \\
&= P_1^T P_1 \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} P_1^T P_1 \begin{bmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ 6 & 7 & 9 & 8 \end{bmatrix}, \qquad P_1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
&= P_1^T \left(P_1 \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} P_1^T \right) \left(P_1 \begin{bmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ 6 & 7 & 9 & 8 \end{bmatrix} \right) \\
&= P_1^T \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 8 & 7 & 9 & 5 \\ 4 & 3 & 3 & 1 \\ 2 & 1 & 1 & 0 \\ 6 & 7 & 9 & 8 \end{bmatrix}
\end{aligned}$$

Now we apply the Gauss transform:

$$L_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{4}{8} & 1 & 0 & 0 \\ \frac{2}{8} & 0 & 1 & 0 \\ \frac{6}{8} & 0 & 0 & 1 \end{bmatrix}$$

Notice that all factors in the Gauss transform are of magnitude less than 1. This is a consequence of partial pivoting. Anyway,

$$
\begin{aligned}
A &= P_1^T \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 8 & 7 & 9 & 5 \\ 4 & 3 & 3 & 1 \\ 2 & 1 & 1 & 0 \\ 6 & 7 & 9 & 8 \end{bmatrix} \\
&= P_1^T \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} L_1 L_1^{-1} \begin{bmatrix} 8 & 7 & 9 & 5 \\ 4 & 3 & 3 & 1 \\ 2 & 1 & 1 & 0 \\ 6 & 7 & 9 & 8 \end{bmatrix} \\
&= P_1^T \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{4}{8} & 1 & 0 & 0 \\ \frac{2}{8} & 0 & 1 & 0 \\ \frac{6}{8} & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 8 & 7 & 9 & 5 \\ 0 & -\frac{1}{2} & -\frac{3}{2} & -\frac{3}{2} \\ 0 & -\frac{3}{4} & -\frac{5}{4} & -\frac{5}{4} \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \end{bmatrix}
\end{aligned}
$$

Now we do another exchange permutation.

$$
\begin{aligned}
A &= P_1^T \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{4}{8} & 1 & 0 & 0 \\ \frac{2}{8} & 0 & 1 & 0 \\ \frac{6}{8} & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 8 & 7 & 9 & 5 \\ 0 & -\frac{1}{2} & -\frac{3}{2} & -\frac{3}{2} \\ 0 & -\frac{3}{4} & -\frac{5}{4} & -\frac{5}{4} \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \end{bmatrix} \\
&= P_1^T P_2^T P_2 \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{4}{8} & 1 & 0 & 0 \\ \frac{2}{8} & 0 & 1 & 0 \\ \frac{6}{8} & 0 & 0 & 1 \end{bmatrix} P_2^T P_2 \begin{bmatrix} 8 & 7 & 9 & 5 \\ 0 & -\frac{1}{2} & -\frac{3}{2} & -\frac{3}{2} \\ 0 & -\frac{3}{4} & -\frac{5}{4} & -\frac{5}{4} \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \end{bmatrix}, \quad P_2 = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \\
&= (P_2 P_1)^T \left(P_2 \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{4}{8} & 1 & 0 & 0 \\ \frac{2}{8} & 0 & 1 & 0 \\ \frac{6}{8} & 0 & 0 & 1 \end{bmatrix} P_2^T \right) \left(P_2 \begin{bmatrix} 8 & 7 & 9 & 5 \\ 0 & -\frac{1}{2} & -\frac{3}{2} & -\frac{3}{2} \\ 0 & -\frac{3}{4} & -\frac{5}{4} & -\frac{5}{4} \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \end{bmatrix} \right) \\
&= (P_2 P_1)^T \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{6}{8} & 1 & 0 & 0 \\ \frac{2}{8} & 0 & 1 & 0 \\ \frac{4}{8} & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & -\frac{3}{4} & -\frac{5}{4} & -\frac{5}{4} \\ 0 & -\frac{1}{2} & -\frac{3}{2} & -\frac{3}{2} \end{bmatrix}
\end{aligned}
$$

We are ready to apply another Gauss transform:

$$L_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -\frac{3}{7} & 1 & 0 \\ 0 & -\frac{2}{7} & 0 & 1 \end{bmatrix}.$$

This yields:

$$\begin{aligned}
A &= (P_2P_1)^T \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{6}{8} & 1 & 0 & 0 \\ \frac{2}{8} & 0 & 1 & 0 \\ \frac{4}{8} & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & -\frac{3}{4} & -\frac{5}{4} & -\frac{5}{4} \\ 0 & -\frac{1}{2} & -\frac{3}{2} & -\frac{3}{2} \end{bmatrix} \\
&= (P_2P_1)^T \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{6}{8} & 1 & 0 & 0 \\ \frac{2}{8} & 0 & 1 & 0 \\ \frac{4}{8} & 0 & 0 & 1 \end{bmatrix} L_2 L_2^{-1} \begin{bmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & -\frac{3}{4} & -\frac{5}{4} & -\frac{5}{4} \\ 0 & -\frac{1}{2} & -\frac{3}{2} & -\frac{3}{2} \end{bmatrix} \\
&= (P_2P_1)^T \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{6}{8} & 1 & 0 & 0 \\ \frac{2}{8} & -\frac{3}{7} & 1 & 0 \\ \frac{4}{8} & -\frac{2}{7} & 0 & 1 \end{bmatrix} \begin{bmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & 0 & -\frac{2}{7} & \frac{4}{7} \\ 0 & 0 & -\frac{6}{7} & -\frac{2}{7} \end{bmatrix}
\end{aligned}$$

Our final exchange permutation goes as follows.

$$\begin{aligned}
A &= (P_2P_1)^T \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{6}{8} & 1 & 0 & 0 \\ \frac{2}{8} & -\frac{3}{7} & 1 & 0 \\ \frac{4}{8} & -\frac{2}{7} & 0 & 1 \end{bmatrix} \begin{bmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & 0 & -\frac{2}{7} & \frac{4}{7} \\ 0 & 0 & -\frac{6}{7} & -\frac{2}{7} \end{bmatrix} \\
&= (P_2P_1)^T\, P_3^T P_3 \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{6}{8} & 1 & 0 & 0 \\ \frac{2}{8} & -\frac{3}{7} & 1 & 0 \\ \frac{4}{8} & -\frac{2}{7} & 0 & 1 \end{bmatrix} P_3^T P_3 \begin{bmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & 0 & -\frac{2}{7} & \frac{4}{7} \\ 0 & 0 & -\frac{6}{7} & -\frac{2}{7} \end{bmatrix}, \quad P_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \\
&= (P_3P_2P_1)^T \left(P_3 \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{6}{8} & 1 & 0 & 0 \\ \frac{2}{8} & -\frac{3}{7} & 1 & 0 \\ \frac{4}{8} & -\frac{2}{7} & 0 & 1 \end{bmatrix} P_3^T \right) \left(P_3 \begin{bmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & 0 & -\frac{2}{7} & \frac{4}{7} \\ 0 & 0 & -\frac{6}{7} & -\frac{2}{7} \end{bmatrix} \right) \\
&= (P_3P_2P_1)^T \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{6}{8} & 1 & 0 & 0 \\ \frac{4}{8} & -\frac{2}{7} & 1 & 0 \\ \frac{2}{8} & -\frac{3}{7} & 0 & 1 \end{bmatrix} \begin{bmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & 0 & -\frac{6}{7} & -\frac{2}{7} \\ 0 & 0 & -\frac{2}{7} & \frac{4}{7} \end{bmatrix}
\end{aligned}$$

The last Gauss transform is:

$$L_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{3} & 1 \end{bmatrix}$$

This yields:

$$\begin{aligned} A &= (P_3P_2P_1)^T \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{6}{8} & 1 & 0 & 0 \\ \frac{4}{8} & -\frac{2}{7} & 1 & 0 \\ \frac{2}{8} & -\frac{3}{7} & 0 & 1 \end{bmatrix} \begin{bmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & 0 & -\frac{6}{7} & -\frac{2}{7} \\ 0 & 0 & -\frac{2}{7} & \frac{4}{7} \end{bmatrix} \\ &= (P_3P_2P_1)^T \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{6}{8} & 1 & 0 & 0 \\ \frac{4}{8} & -\frac{2}{7} & 1 & 0 \\ \frac{2}{8} & -\frac{3}{7} & 0 & 1 \end{bmatrix} L_3L_3^{-1} \begin{bmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & 0 & -\frac{6}{7} & -\frac{2}{7} \\ 0 & 0 & -\frac{2}{7} & \frac{4}{7} \end{bmatrix} \\ &= (P_3P_2P_1)^T \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{6}{8} & 1 & 0 & 0 \\ \frac{4}{8} & -\frac{2}{7} & 1 & 0 \\ \frac{2}{8} & -\frac{3}{7} & \frac{1}{3} & 1 \end{bmatrix} \begin{bmatrix} 8 & 7 & 9 & 5 \\ 0 & \frac{7}{4} & \frac{9}{4} & \frac{17}{4} \\ 0 & 0 & -\frac{6}{7} & -\frac{2}{7} \\ 0 & 0 & 0 & \frac{2}{3} \end{bmatrix} \end{aligned}$$

$$P = P_3P_2P_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Solving the linear system. Once we have obtained P, L and U. Solving the linear system is not much different. Simply multiply $Ax = b$ by P from the left on both sides to yield:

$$PAx = Pb$$

. But since $PA = LU$, we have $LUx = Pb$. Now continue as we did before!