Section 3.2 Pivoting Strategies

Key terms

- Common pivoting strategies
  - Natural order of pivots
  - Partial Pivoting
  - Scaled partial pivoting
  - Full or complete pivoting
- Partial pivoting without physically interchanging rows
- MATLAB’s linear system solver
Section 3.2 Pivoting Strategies

It is sometimes necessary during Gaussian elimination to interchange rows while solving a system of linear equations so as to avoid a zero pivot element. When performing calculations in finite precision arithmetic, it may also be necessary to interchange rows to reduce the effect of roundoff error on the computed solution. In this section, we will first illustrate the problems which can arise when solving linear systems using finite precision arithmetic.

**We use row interchanges only to avoid zero pivots or small pivots.** The reason for this is that pivots become denominators of the scalar multipliers used in row operations as we move toward upper triangular form. Division by small values is a floating point arithmetic pitfall.

The selective use of row interchanges is referred to as a **PIVOTING STRATEGY**.
There are several **common pivoting strategies**:

- **Natural order of pivots**: no row interchanges permitted. With this strategy not every nonsingular linear system can be solved.

- **Partial pivoting** (also called Maximal column pivots)

- **Scaled partial pivoting**

- **Full (complete) pivoting**

It is considered a strategic blunder **NOT** to use a partial or full pivoting strategy.

**The choice of pivots is a computational stability issue!**
Motivational Example

Consider the linear system

First we work with exact arithmetic using row operations;

\[
\begin{align*}
\frac{-1}{2} R_1 + R_2 & \rightarrow R_2 \\
-3 \quad R_1 + R_3 & \rightarrow R_3
\end{align*}
\]

We get equivalent system

Interchanging rows 2 and 3 we get an upper triangular system whose solution by back substitution is \( x_1 = 1, x_2 = 7, x_3 = 1 \).

Note that the (2,2) entry was zero using exact arithmetic.
Now consider using **4 decimal digit rounding arithmetic** to solve this system. The augmented matrix is now

\[
\begin{pmatrix}
6.667e-001 & 2.857e-001 & 2.000e-001 & 2.867e+000 \\
3.333e-001 & 1.429e-001 & -5.000e-001 & 8.333e-001 \\
2.000e-001 & -4.286e-001 & 4.000e-001 & -2.400e+000
\end{pmatrix}
\]

Sweeping out below the (1,1) entry we get the next matrix

\[
\begin{pmatrix}
6.667e-001 & 2.857e-001 & 2.000e-001 & 2.867e+000 \\
0.000e+000 & 1.000e-004 & -6.000e-001 & -5.997e-001 \\
0.000e+000 & -5.143e-001 & 3.400e-001 & -3.260e+000
\end{pmatrix}
\]

Sweeping out below the (2,2) entry we get

\[
\begin{pmatrix}
6.667e-001 & 2.857e-001 & 2.000e-001 & 2.867e+000 \\
0.000e+000 & 1.000e-004 & -6.000e-001 & -5.997e-001 \\
0.000e+000 & 0.000e+000 & -3.086e+003 & -3.087e+003
\end{pmatrix}
\]

\(\leftarrow\) This is in upper triangular form (for the coefficient matrix) so we use back substitution.

We get

\[
\begin{align*}
x_1 &= 2.715e+000 \\
x_2 &= 3.000e+000 \\
x_3 &= 1.000e+000
\end{align*}
\]
The true solution is \( x_1 = 1, x_2 = 7, x_3 = 1 \).

Using 4-digit arithmetic we get:
- \( x_1 = 2.715e+000 \)
- \( x_2 = 3.000e+000 \)
- \( x_3 = 1.000e+000 \)

The percent of error when using 4 digit rounding arithmetic is approximately 171% in \( x_1 \), 57% in \( x_2 \), and 0% in \( x_3 \).

**What caused such error?**

Look carefully at the first row operation:

\[
\begin{bmatrix}
-3.333e-001 \\
6.667e-001
\end{bmatrix} \times \text{Row}_1 + \text{Row}_2
\]

we get \(-4.999e-001 \begin{bmatrix} 2.857e-001 & 2.000e-001 & 2.867e+000 \end{bmatrix}\)

\(+ \begin{bmatrix} 1.429e-001 & -5.000e-001 & 8.333e-001 \end{bmatrix}\)

Look at behavior of the sum of first entries:

\[-4.999e-001 \times 2.857e-001 + 1.429e-001 \Rightarrow -1.428e-001 + 1.429e-001 = 1.000e-004\]

which is cancellation error.

For the second sweep this becomes a “SMALL” pivot and then dividing by this small value propagates the error.

\[
\begin{bmatrix}
6.667e-001 & 2.857e-001 & 2.000e-001 & 2.867e+000 \\
0.000e+000 & 1.000e-004 & -6.000e-001 & -5.997e-001 \\
0.000e+000 & -5.143e-001 & 3.400e-001 & -3.260e+000
\end{bmatrix}
\]
If we start the same way, and now note that the (2,2) entry is the result of cancellation error

\[
\begin{align*}
6.667e-001 & \quad 2.857e-001 & \quad 2.000e-001 & \quad 2.867e+000 \\
0.000e+000 & \quad 1.000e-004 & \quad -6.000e-001 & \quad -5.997e-001 \\
0.000e+000 & \quad -5.143e-001 & \quad 3.400e-001 & \quad -3.260e+000 \\
0.000e+000 & \quad 0.000e+000 & \quad -5.997e-001 & \quad -6.003e-001
\end{align*}
\]

Now we interchange rows 2 and 3 before the second sweep to get the following system.

\[
\begin{align*}
6.667e-001 & \quad 2.857e-001 & \quad 2.000e-001 & \quad 2.867e+000 \\
0.000e+000 & \quad -5.143e-001 & \quad 3.400e-001 & \quad -3.260e+000 \\
0.000e+000 & \quad 1.000e-004 & \quad -6.000e-001 & \quad -5.997e-001 \\
0.000e+000 & \quad 0.000e+000 & \quad -5.999e-001 & \quad -6.003e-001
\end{align*}
\]

Eliminating the (3,2)-entry we get

\[
\begin{align*}
6.667e-001 & \quad 2.857e-001 & \quad 2.000e-001 & \quad 2.867e+000 \\
0.000e+000 & \quad -5.143e-001 & \quad 3.400e-001 & \quad -3.260e+000 \\
0.000e+000 & \quad 0.000e+000 & \quad -5.999e-001 & \quad -6.003e-001 \\
0.000e+000 & \quad 0.000e+000 & \quad 0.000e+000 & \quad -5.999e-001
\end{align*}
\]

Applying back substitution using 4 digit rounding arithmetic we get

\[
\begin{align*}
\mathbf{x}_1 &= 1.000e+000 \\
\mathbf{x}_2 &= 7.000e+000 \\
\mathbf{x}_3 &= 1.001e+000
\end{align*}
\]

This result is virtually identical to the true solution.
The example clearly shows that we should avoid using small pivots.

**PARTIAL PIVOTING STRATEGY**
Starting with the current diagonal entry search the "remainder" of the column for the entry of **largest magnitude**. Perform appropriate row interchanges to move that entry to the current pivot position. Then use row operations to "zero out" below the pivot.

**Example.** Solve the linear system whose augmented matrix is

\[
\begin{bmatrix}
1 & 2 & 3 & -2 & 5 \\
2 & 4 & 1 & 0 & 0 \\
3 & 3 & 2 & 5 & 4 \\
1 & 0 & 2 & 1 & 3 \\
\end{bmatrix}
\]

using **reduce** and partial pivoting. Make a list of the pivots.

What is the relationship between the absolute value of the product of the pivots and the coefficient matrix?

Solution is \[
\begin{bmatrix}
1 \\
-1 \\
2 \\
0 \\
\end{bmatrix}
\]

Do this real time in MATLAB.
Partial pivoting without row interchanges.

When partial pivoting is applied we often need to interchange rows to maintain the upper triangular form of the augmented matrix. These row interchanges are a convenience for our hand calculations so we can better organize our reduction process. They are by no means required and in fact when using a computer algorithm they waste time since a physical interchange of storage location must be performed.

It is possible to use partial pivoting on an augmented matrix and not physically make row interchanges. In order to accomplish this we need a device for keeping track of the order in which rows are used as pivot rows. A simple device called a pivot vector will accomplish this book keeping process. At the start of the reduction process for an $n \times n$ linear system we initialize the pivot vector $\mathbf{p}$ as

$$
\mathbf{p} = \begin{bmatrix}
1 \\
2 \\
\vdots \\
n
\end{bmatrix}
$$

We denote the entries of $\mathbf{p}$ by $p_i$. If at some point we interchange row 2 and row 5 we interchange the contents of $p_2$ and $p_5$. We use the contents of the pivot vector as an indirect addressing scheme for row numbers.
Example  Solve $Ax = b$ using partial pivoting without row interchanges where

$$A = \begin{bmatrix} 2 & -1 & 2 \\ -3 & 4 & 1 \\ 6 & 0 & 3 \end{bmatrix}, \quad b = \begin{bmatrix} 4 \\ 7 \\ 12 \end{bmatrix}$$

We form the augmented matrix $[A \mid b]$ and initialize the pivot vector $p = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$. The first pivot is $a_{31} = 6$.

Rather than physically interchange rows 1 and 3 we interchange the contents of $p_1$ and $p_3$ giving the updated pivot vector $p = \begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix}$. Now location $p_1$ contains the address of the first pivot row and the other row numbers are in entries $p_2$ and $p_3$. Using these addresses we do row operations

$$\frac{-a_{p(2)}}{a_{p(1)}} \text{Row}(p(1)) + \text{Row}(p(2)) \rightarrow \frac{-(-3)}{6} \text{Row}(3) + \text{Row}(2)$$

$$\frac{-a_{p(3)}}{a_{p(1)}} \text{Row}(p(1)) + \text{Row}(p(3)) \rightarrow \frac{-2}{6} \text{Row}(3) + \text{Row}(1)$$

The resulting matrix is $\begin{bmatrix} 0 & -1 & 1 & | & 0 \\ 0 & 4 & 5/2 & | & 13 \\ 6 & 0 & 3 & | & 12 \end{bmatrix}$ and the current pivot vector is $p = \begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix}$. Applying partial pivoting to select the second pivot we look for the max of $|a_{p(2)}|$ and $|a_{p(3)}|$; that is here max $\{ |-1|, |4| \}$. Obviously the max has value 4 so $a_{p(2)}$ is the second pivot. Technically we interchange the contents of $p(2)$ with $p(2)$. (There is no change in the pivot vector this time.)
Scaled Partial Pivoting

Scaled partial pivoting not only seeks to avoid "small" pivot values but also takes into account the size of coefficients in a row. The process **scaled partial pivoting** is described as follows.

**SCALED PARTIAL PIVOTING STRATEGY**

Define $s_i$ to be the **absolute value** of the coefficient in the $i^{th}$ equation that is largest in absolute value; that is,

$$s_i = \max_{j=1,2,\ldots,n} |a_{ij}|, \text{ for } i = 1,2, \ldots, n.$$  

(Note: we do not include the augmented column in this procedure.)

$s_i$ is called the **size** or **scale factor** of the $i^{th}$ equation. These are computed at the start of the algorithm and are **not** recomputed. If row interchanges are made we must keep the scale factor associated with the equation.

Starting with the current diagonal entry search the "remainder" of the column for the ratio of an entry divided by the equation's scale factor which is largest in magnitude. Perform appropriate row interchanges to move that entry to the current pivot position. Then use row operations to "zero out" below the pivot.  
(Note that the equation scale factor is not part of the pivot value used, but only plays a role in selecting the pivot.)
Example Consider the linear system which has coefficient matrix \( A = \begin{bmatrix} 3 & 5 & -6 \\ 2 & 0 & 3 \\ 4 & 9 & 0 \end{bmatrix} \) and right side \( b = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \). Solve the system \( Ax = b \) using scaled partial pivoting.

Solution:

\[
\begin{align*}
s_1 &= \max \{|3|, |5|, |-6|\} = 6 \\
s_2 &= \max \{|2|, |0|, |3|\} = 3 \\
s_3 &= \max \{|4|, |9|, |0|\} = 9
\end{align*}
\]

For the first pivot selection we form the ratios of the magnitudes of the entries from the first column and the equation scale factors then determine the max.

\[
\max \left\{ \frac{3}{s_1}, \frac{2}{s_2}, \frac{4}{s_3} \right\} = \max \left\{ \frac{3}{6}, \frac{2}{3}, \frac{4}{9} \right\} = \max \left\{ \frac{3}{6}, \frac{2}{3}, \frac{4}{9} \right\} = \frac{2}{3}
\]

Thus the first pivot is the \((2,1)\)-entry which has value 2.
Since the first pivot is the (2,1)-entry which has value 2. We need to interchange rows 1 and 2 to get this choice to the first pivot position. In order to keep the scale factors associated with their original equations we will append a column of scale factors to the augmented matrix as shown next. Then any row interchanges will automatically keep the scale factors associated with the correct equations.

$$\begin{bmatrix} 3 & 5 & -6 & 1 & s_1 \\ 2 & 0 & 3 & 2 & s_2 \\ 4 & 9 & 0 & 1 & s_3 \end{bmatrix} \xrightarrow{\text{row interchange}} \begin{bmatrix} 2 & 0 & 3 & 2 & 3 \\ 3 & 5 & -6 & 1 & 6 \\ 4 & 9 & 0 & 1 & 9 \end{bmatrix}$$

Original.  

After row interchange.

Now perform row operations but \textbf{not} on the column of scale factors. We get

$$\begin{bmatrix} 2 & 0 & 3 & 2 & 3 \\ 0 & 5 & -21/2 & -2 & 6 \\ 0 & 9 & -6 & -3 & 9 \end{bmatrix}$$

To select the second pivot we determine $\max \left\{ \frac{5}{6} , \frac{9}{9} \right\} = \max \left\{ \frac{5}{6} , 1 \right\} = 1$. Thus the second pivot is the (3,2)-entry which has value 9.
So we interchange rows 2 and 3 to get

\[
\begin{bmatrix}
2 & 0 & 3 & 2 & 3 \\
0 & 9 & -6 & -3 & 9 \\
0 & 5 & -21/2 & -2 & 6 \\
\end{bmatrix}
\]

Zero out below the second pivot to get

\[
\begin{bmatrix}
2 & 0 & 3 & 2 & 3 \\
0 & 9 & -6 & -3 & 9 \\
0 & 0 & -129/18 & -1/3 & 6 \\
\end{bmatrix}
\]

Performing back substitution we get \( \mathbf{x} = \begin{bmatrix} 40/43 \\ -13/43 \\ 2/43 \end{bmatrix} \).

\[
\begin{array}{l}
0.930232558139535 \\
-0.302325581395349 \\
0.046511627906977 \\
\end{array}
\]

It is exact to the working precision of MATLAB.
**FULL (COMPLETE) PIVOTING STRATEGY**

Starting with the current pivot **search the entire submatrix of the coefficient matrix downward and to the right for the entry of largest magnitude.** Perform appropriate row and column interchanges to move that entry to the current pivot position. Then use row operations to "zero out" below the pivot.

Note that column interchanges "reorder" the unknowns.

**How do you keep track of this?**

**Example.** Consider a linear system whose augmented matrix is

\[
\begin{bmatrix}
6 & -2 & -4 & 4 & 10 \\
3 & -3 & -6 & 1 & 0 \\
-12 & 8 & 21 & -8 & -16 \\
-6 & 0 & -10 & 7 & 21
\end{bmatrix}
\]

a) Determine the first pivot using the complete pivoting strategy.
b) Perform the appropriate interchanges to get the selected entry to the pivot position.
c) Do the "zeroing out" below the first pivot.
d) Determine the second pivot.
Later we discuss MATLAB’s linear system solver.

Answers: first pivot is 21; the "rearranged" matrix is
\[
\begin{bmatrix}
21 & 8 & -12 & -8 & -16 \\
-6 & -3 & 3 & 1 & 0 \\
-4 & -2 & 6 & 4 & 10 \\
-10 & 0 & -6 & 7 & 21
\end{bmatrix}
\]

"zeroing out" in column 1 the system has the form
\[
\begin{bmatrix}
21 & 8 & -12 & -8 & -16 \\
0 & -0.7143 & -0.4286 & -1.2857 & -4.5714 \\
0 & -0.4762 & 3.7143 & 2.4762 & 6.9524 \\
0 & 3.8095 & -11.7143 & 3.1905 & 13.3810
\end{bmatrix}
\]
the second pivot is \(-11.7143\).