



6.4.8 Gaussian elimination with partial pivoting

Douglas Wilhelm Harder, LEL, M.Math.

dwharder@uwaterloo.ca

dwharder@gmail.com





Introduction

- In this topic, we will
 - Review the steps taken in solving a system of linear equations
 - Determine that it is important to take floating-point numbers into consideration
 - Describe the Gaussian elimination algorithm with partial pivoting
 - Look at a few examples
 - Describe the algorithm in a more human-friendly format
 - Look at many more examples





Solving systems of linear equations

- Up to this point,
 - You have seen backward substitution for finding a solution to a system of linear equations when one exists
 - Either there is one unique solution or infinitely many solutions
 - We have described row operations that will be used to convert an arbitrary augmented matrix into one that is in row-echelon form





Solving systems of linear equations

- In secondary school, when solving a system of linear equations, you did whatever was easiest:

$$\begin{array}{rcl} x_1 + 2x_2 & = & 5 \\ -3x_1 + 4x_2 & = & -6 \end{array} \quad \left(\begin{array}{cc|c} 1 & 2 & 5 \\ -3 & 4 & -6 \end{array} \right)$$

- The obvious step is to add 3 times Eqn 1 to Eqn 2
- In this course, you would perform the row operation $R_{3R1 \rightarrow R2}$

$$\begin{array}{rcl} x_1 + 2x_2 & = & 5 \\ 10x_2 & = & 9 \end{array} \quad \left(\begin{array}{cc|c} 1 & 2 & 5 \\ 0 & 10 & 9 \end{array} \right)$$

- In either case, we get the solution vector $\mathbf{x} = \begin{pmatrix} 3.2 \\ 0.9 \end{pmatrix}$





Numerical instability

- As an engineer, you must be aware of the tools you are using
 - A computer using floating-point numbers
 - These have finite precision,
which means they only store 16 significant digits of precision
 - This will lead to problems
 - As an engineer, you will be responsible for misusing your tools





Numerical instability

- Suppose your computer can only store four digits of precision

- Thus, 59234234.23 would be stored as 5.923×10^7

- 0.000791863153 would be stored as 7.919×10^{-4}

- π would be stored as 3.142×10^0

- Also, each time you perform a calculation, the result can only be stored to four digits:

$$5.923 \times 10^7 + 8.758 \times 10^5 = 6.01058 \times 10^7$$

- but this result would be stored as 6.011×10^7

$$\begin{array}{r} 592.300 \times 10^5 \\ + \quad 8.758 \times 10^5 \\ \hline 601.058 \times 10^5 \end{array}$$





Numerical instability

- If you were in secondary school, and given the system

$$0.00001x_1 + x_2 = 1$$

$$x_1 + x_2 = 2$$

you may just add -100000 times Eqn 1 onto Eqn 2:

$$0.00001x_1 + x_2 = 1$$

$$-99999x_2 = -99998$$

- Solving this, you get that $\mathbf{x} = \begin{pmatrix} \frac{100000}{99999} \\ \frac{99998}{99999} \end{pmatrix} = \begin{pmatrix} \overline{1.00001} \\ \overline{0.99998} \end{pmatrix} \approx \begin{pmatrix} 1 \\ 1 \end{pmatrix}$





Numerical instability

- Suppose, however, you only had 4 digits of precision:

$$0.00001x_1 + x_2 = 1$$

$$x_1 + x_2 = 2$$

you may just add -100000 times Eqn 1 onto Eqn 2:

$$0.00001x_1 + x_2 = 1$$

$$0.00001x_1 + x_2 = 1$$

$$-99999x_2 = -99998$$

$$-100000x_2 = -100000$$

- Solving this, you get that $\mathbf{x} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

– However you dice it, 0 is a horrible approximation of 1





Numerical instability

- Instead, swap the two equations:

$$x_1 + x_2 = 2$$

$$0.00001x_1 + x_2 = 1$$

- Now add -0.00001 times Eqn 1 onto Eqn 2:

$$\begin{aligned} x_1 + x_2 &= 2 \\ 0.99999x_2 &= 0.99998 \end{aligned}$$

$$\begin{aligned} x_1 + x_2 &= 2 \\ x_2 &= 1 \end{aligned}$$

- Solving this, you get that $\mathbf{x} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$
 - To four significant digits,
this is as close as we can get to $\mathbf{x} = \begin{pmatrix} \overline{1.00001} \\ \overline{0.99998} \end{pmatrix}$





Numerical instability

- What happened?
 - In the first technique,
we added a huge multiple of the first equation onto the second
 - This had the effect of “swamping out” any information found in the second equation
- Solution:
 - If we are performing the row operation $R_{\alpha Ri \rightarrow Rj'}$
we want to guarantee $|\alpha| \leq 1$
 - This will ensure that the multiple of Row i does not swamp out the information in Row j

$$\begin{array}{r} 75320.000 \\ + \quad 4.729 \\ \hline 75324.729 \end{array}$$





Important!

- In this course, you do not have to understand the subtleties of working with floating-point numbers
 - This will be covered in a future course on numerical analysis and numerical methods
- In this course, you only need be aware that there are issues with using floating-point numbers, and that the algorithm presented mitigates the weaknesses with floating-point numbers





Gaussian elimination with partial pivoting

- We will introduce an algorithm that is numerically stable
 - It will be different from any implementation taught in first-year mathematics-based courses
 - Do not look this up on-line unless you explicitly include in your search “partial pivoting”





Gaussian elimination with partial pivoting

- Let $(A : \mathbf{b})$ be an $m \times (n + 1)$ augmented matrix
- Set $i \leftarrow 1$ and $j \leftarrow 1$
 1. If Column j has no leading non-zero entries,
so just update $j \leftarrow j + 1$; that is, only go to the next column
 2. Otherwise, Column j has at least one leading non-zero entry
 - a. Find that Row k that has the largest-in-absolute-value leading non-zero entry in Column j
 - b. Swap Row i and Row k
 - c. For each other $\ell \neq i$ such that $a_{\ell,j}$ that is a leading non-zero entry,
add $-\frac{a_{\ell,j}}{a_{i,j}}$ times Row i onto Row ℓ
 - d. Update $i \leftarrow i + 1$ and $j \leftarrow j + 1$
that is, go to the next row and next column
 3. If $i > m$ or $j > n + 1$, we are done and $(A : \mathbf{b})$ is in row-echelon form; otherwise go to Step 1.





Gaussian elimination with partial pivoting

- Note: You can stop as soon as you see that the matrix $(A : \mathbf{b})$ is in row-echelon form ☺
- If you ever find more than one leading non-zero entry that has the maximum absolute value, you can pick either one
- Here is a high-level overview of the algorithm:





Example

- Consider this 2×3 augmented matrix ($m = n = 2$)
 - In Column j , there are two leading non-zero entries, the largest in magnitude is $a_{2,1}$, so swap Rows i and 2
 - Next, add $-\frac{1.2}{4} = -0.3$ times Row 1 onto Row 2 to eliminate the leading non-zero entry at index $(2, 1)$, and increment i and j
 - In Column 2, there is only one leading non-zero entry, and it is already at index (i, j) , so increment i and j
 - $i > m$, so we are done

$$i = 1$$

$$j = 1$$

$$\left(\begin{array}{cc|c} 1.2 & 2.9 & 3.3 \\ 4 & 3 & -9 \end{array} \right) \underset{R_{R1 \leftrightarrow R2}}{\sim} \left(\begin{array}{cc|c} 4 & 3 & -9 \\ 1.2 & 2.9 & 3.3 \end{array} \right) \underset{R_{-0.3 \cdot R1 \rightarrow R2}}{\sim} \left(\begin{array}{cc|c} 4 & 3 & -9 \\ 0 & 2 & 6 \end{array} \right)$$

$$\mathbf{x} = \begin{pmatrix} -4.5 \\ 3 \end{pmatrix}$$





Example

- Consider this 2×3 augmented matrix ($m = n = 2$)
 - In Column j , there are two leading non-zero entries, the largest in magnitude is $a_{2,1}$, so swap Rows i and 2
 - Next, add $-\frac{-2.4}{6} = 0.4$ times Row 1 onto Row 2 to eliminate the leading non-zero entry at index $(2, 1)$, and increment i and j
 - In Column j , there is are no leading non-zero entries, so increment j
 - In Column j , there is are no leading non-zero entries, so increment j
 - Now, $j > n + 1$, so we are done

$$i = 2$$

$$j = 1$$

$$\left(\begin{array}{cc|c} -2.4 & 3.6 & -1.2 \\ 6 & -9 & 3 \end{array} \right) \underset{R_{1 \leftrightarrow R_2}}{\sim} \left(\begin{array}{cc|c} 6 & -9 & 3 \\ -2.4 & 3.6 & -1.2 \end{array} \right) \underset{R_{0.4 \cdot R_1 \rightarrow R_2}}{\sim} \left(\begin{array}{cc|c} 6 & -9 & 3 \\ 0 & 0 & 0 \end{array} \right)$$

$$\mathbf{x} = \begin{pmatrix} 0.5 + 1.5x_2 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0.5 \\ 0 \end{pmatrix} + x_2 \begin{pmatrix} 1.5 \\ 1 \end{pmatrix}$$





Example

- Consider this 2×3 augmented matrix ($m = n = 2$)
 - In Column j , there are two leading non-zero entries, the largest in magnitude is $a_{1,1}$, so there is nothing to do
 - Next, add $-\frac{2.5}{5} = 0.5$ times Row 1 onto Row 2 to eliminate the leading non-zero entry at index $(2, 1)$, and increment i and j
 - In Column j , there are no leading non-zero entries, so increment j
 - In Column j , there is one leading non-zero entry, but it is already in Row i , and there are no other leading non-zero entries, so increment i and j
 - Now $i > m$ and $j > n + 1$, so we are done

$$i = 3$$
$$j = 3$$

$$\left(\begin{array}{cc|c} 5 & 2 & 4 \\ -2.5 & -1 & 3 \end{array} \right) \sim \left(\begin{array}{cc|c} 5 & 2 & 4 \\ 0 & 0 & 5 \end{array} \right)$$

$R_{0.5 \cdot R1 \rightarrow R2}$





Example

- Consider this 3×4 augmented matrix ($m = n = 3$)
 - In Column j , there are three leading non-zero entries, the largest in magnitude is $a_{3,1}$, so swap Rows i and 3
 - Next, add $-\frac{3.6}{4} = -0.9$ times Row 1 onto Row 2 to eliminate the leading non-zero entry at index (2, 1)
 - Also, add $-\frac{-3.2}{4} = 0.8$ times Row 1 onto Row 3 to eliminate the leading non-zero entry at index (3, 1)
 - Increment i and j

$$i = 2$$

$$j = 2$$

$$\left(\begin{array}{ccc|c} -3.2 & 2.6 & -2.4 & -25.4 \\ 3.6 & -0.8 & 11 & 38.8 \\ 4 & 3 & -2 & 3 \end{array} \right) \sim \left(\begin{array}{ccc|c} 4 & 3 & -2 & 3 \\ 3.6 & -0.8 & 11 & 38.8 \\ -3.2 & 2.6 & -2.4 & -25.4 \end{array} \right) \sim \left(\begin{array}{ccc|c} 4 & 3 & -2 & 3 \\ 0 & -3.5 & 12.8 & 36.1 \\ 0 & 5 & -4 & -23 \end{array} \right)$$

$R_{R1 \leftrightarrow R3}$ $R_{-0.9 \cdot R1 \rightarrow R2}$ $R_{0.8 \cdot R1 \rightarrow R3}$





Example

- Consider this 3×4 augmented matrix ($m = n = 3$)
 - In Column j , there are two leading non-zero entries, the largest in magnitude is $a_{3,2}$, so swap Rows i and 3
 - Next, add $-\frac{3.5}{5} = 0.7$ times Row 2 onto Row 3 to eliminate the leading non-zero entry at index $(3, 2)$
 - Increment i and j

 $i = 2$ $j = 2$

$$\left(\begin{array}{ccc|c} 4 & 3 & -2 & 3 \\ 0 & -3.5 & 12.8 & 36.1 \\ 0 & 5 & -4 & -23 \end{array} \right) \underset{R_{R2 \leftrightarrow R3}}{\sim} \left(\begin{array}{ccc|c} 4 & 3 & -2 & 3 \\ 0 & 5 & -4 & -23 \\ 0 & -3.5 & 12.8 & 36.1 \end{array} \right) \underset{R_{0.7 \cdot R2 \rightarrow R3}}{\sim} \left(\begin{array}{ccc|c} 4 & 3 & -2 & 3 \\ 0 & 5 & -4 & -23 \\ 0 & 0 & 10 & 20 \end{array} \right)$$





Example

- Consider this 3×4 augmented matrix ($m = n = 3$)
 - In Column j , there is one leading non-zero entry, and it is already at index $(3, 3)$, there is nothing to do
 - There are no other leading non-zero entries in Column j
 - Increment i and j
 - $i > m$, so we're done

$$i = 4$$

$$j = 4$$

$$\left(\begin{array}{ccc|c} -3.2 & 2.6 & -2.4 & -25.4 \\ 3.6 & -0.8 & 11 & 38.8 \\ 4 & 3 & -2 & 3 \end{array} \right) \sim \left(\begin{array}{ccc|c} 4 & 3 & -2 & 3 \\ 0 & 5 & -4 & -23 \\ 0 & 0 & 10 & 20 \end{array} \right)$$

$$\mathbf{x} = \begin{pmatrix} 4 \\ -3 \\ 2 \end{pmatrix}$$





Example

$$i = 1$$

$$j = 1$$

- Consider this 3×4 augmented matrix ($m = n = 3$)
 - In Column j , there are three leading non-zero entries, the largest in magnitude is $a_{2,1}$, so swap Rows i and 2
 - Next, add $-\frac{1}{5} = -0.2$ times Row 1 onto Row 2 to eliminate the leading non-zero entry at index (2, 1)
 - Next, add $-\frac{-0.5}{5} = 0.1$ times Row 1 onto Row 3 to eliminate the leading non-zero entry at index (3, 1)
 - Increment i and j

$$\left(\begin{array}{ccc|c} 1 & 1.4 & -9.6 & 76.2 \\ 5 & 7 & -3 & 66 \\ -0.5 & -0.7 & 3 & -25.5 \end{array} \right) \underset{R_{1 \leftrightarrow 2}}{\sim} \left(\begin{array}{ccc|c} 5 & 7 & -3 & 66 \\ 1 & 1.4 & -9.6 & 76.2 \\ -0.5 & -0.7 & 3 & -25.5 \end{array} \right) \underset{R_{-0.2 \cdot R1 \rightarrow R2}}{\sim} \left(\begin{array}{ccc|c} 5 & 7 & -3 & 66 \\ 0 & 0 & -9 & 63 \\ 0 & 0 & 2.7 & -18.9 \end{array} \right)$$
$$\underset{R_{0.1 \cdot R1 \rightarrow R3}}$$





Example

$$i = 2$$

$$j = 2$$

- Consider this 3×4 augmented matrix ($m = n = 3$)
 - In Column j , there are no leading non-zero entries, so only increment j (go to the next column only)

$$\left(\begin{array}{ccc|c} 5 & 7 & -3 & 66 \\ 0 & 0 & -9 & 63 \\ 0 & 0 & 2.7 & -18.9 \end{array} \right)$$





Example

$$i = 2$$

$$j = 3$$

- Consider this 3×4 augmented matrix ($m = n = 3$)
 - In Column j , there are two leading non-zero entries, the largest in magnitude is $a_{2,3}$, thus, no swap required
 - Next, add $-\frac{2.7}{-9} = 0.3$ times Row 2 onto Row 3 to eliminate the leading non-zero entry at index $(3, 3)$
 - Increment i and j
 - There are no leading non-zero entries in Column j , so increment j
 - As $j > n + 1$, we are done

$$\begin{pmatrix} 5 & 7 & -3 & \vdots & 66 \\ 0 & 0 & -9 & \vdots & 63 \\ 0 & 0 & 2.7 & \vdots & -18.9 \end{pmatrix} \xrightarrow{R_{0.3 \cdot R2 \rightarrow R3}} \begin{pmatrix} 5 & 7 & -3 & \vdots & 66 \\ 0 & 0 & -9 & \vdots & 63 \\ 0 & 0 & 0 & \vdots & 0 \end{pmatrix} \quad \mathbf{x} = \begin{pmatrix} 9 - 1.4x_2 \\ x_2 \\ -7 \end{pmatrix}$$

$$= \begin{pmatrix} 9 \\ 0 \\ -7 \end{pmatrix} + x_2 \begin{pmatrix} -1.4 \\ 1 \\ 0 \end{pmatrix}$$





Simplified algorithm

- The described algorithm can be programmed into a computer
- Here is a simpler human-implementable version:
 - Stop as soon as the matrix is in row-echelon form
 - Visit each column, from left to right
 - The first time you find a column with at least one leading non-zero entry
 - Find the Row k that contains the maximum-in-magnitude leading non-zero entry in that column
 - Swap Row k and Row 1
 - Proceed to eliminate all other leading non-zero entries in that column by adding appropriate multiples of Row 1 onto subsequent rows





Simplified algorithm

- The next time you find a column with at least one leading non-zero entry
 - Find the Row k that contains the maximum-in-magnitude leading non-zero entry in that column
 - Swap Row k and Row 2
 - Proceed to eliminate all other leading non-zero entries in that column by adding appropriate multiples of Row 2 onto subsequent rows
- Continue this general procedure





Example

- Consider this example:

$$\left(\begin{array}{ccc|c} -3 & 2 & 4 & 3 \\ 1.8 & -2.7 & -2.1 & -2 \\ -0.6 & -4.6 & 1.8 & -3.4 \end{array} \right) \sim \left(\begin{array}{ccc|c} -3 & 2 & 4 & 3 \\ 0 & -1.5 & 0.3 & -0.2 \\ 0 & -5 & 1 & -4 \end{array} \right)$$

$$\sim \left(\begin{array}{ccc|c} -3 & 2 & 4 & 3 \\ 0 & -5 & 1 & -4 \\ 0 & -1.5 & 0.3 & -0.2 \end{array} \right) \sim \left(\begin{array}{ccc|c} -3 & 2 & 4 & 3 \\ 0 & -5 & 1 & -4 \\ 0 & 0 & 0 & 1 \end{array} \right)$$





Example

- Consider this example:

$$\left(\begin{array}{ccc|c} 0 & 2 & -5 & 9 \\ 0 & 3 & 4.8 & -1.26 \\ 0 & -5 & 2 & -9.9 \end{array} \right) \sim \left(\begin{array}{ccc|c} 0 & -5 & 2 & -9.9 \\ 0 & 3 & 4.8 & -1.26 \\ 0 & 2 & -5 & 9 \end{array} \right)$$

$$\sim \left(\begin{array}{ccc|c} 0 & -5 & 2 & -9 \\ 0 & 0 & 6 & -7.2 \\ 0 & 0 & -4.2 & 5.04 \end{array} \right) \sim \left(\begin{array}{ccc|c} 0 & -5 & 2 & -9 \\ 0 & 0 & 6 & -7.2 \\ 0 & 0 & 0 & 0 \end{array} \right)$$

$$\mathbf{x} = \begin{pmatrix} x_1 \\ 1.5 \\ -1.2 \end{pmatrix} = \begin{pmatrix} 0 \\ 1.5 \\ -1.2 \end{pmatrix} + x_1 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$





Example

- Consider this 4×5 augmented matrix ($m = n = 4$)

$$\left(\begin{array}{cccc|c} 3.6 & 3.9 & -5.1 & 4.2 & 13.84 \\ -2.4 & -0.9 & 3.7 & 6.4 & -4.34 \\ 4.6 & 3.9 & -5.1 & 4.2 & 13.84 \\ -2.8 & 0.9 & 2.7 & 3 & -3.06 \end{array} \right) \sim \left(\begin{array}{cccc|c} 4 & 3 & -1 & 0 & 4.8 \\ 0 & 0.9 & 2.1 & 6.4 & -0.46 \\ 0 & 1.2 & -4.2 & 4.2 & 8.82 \\ 0 & 0.9 & 2.1 & 6.4 & -0.46 \end{array} \right)$$

$$\sim \left(\begin{array}{cccc|c} 4 & 3 & -1 & 0 & 4.8 \\ 0 & 3 & 2 & 3 & 0.3 \\ 0 & 0 & -5 & 3 & 8.7 \\ 0 & 0 & 2.5 & 5.5 & -1.55 \end{array} \right) \sim \left(\begin{array}{cccc|c} 4 & 3 & -1 & 0 & 4.8 \\ 0 & 3 & 2 & 3 & 0.3 \\ 0 & 0 & -5 & 3 & 8.7 \\ 0 & 0 & 0 & 7 & 2.8 \end{array} \right)$$

$$\left(\begin{array}{cccc|c} 3.6 & 3.9 & -5.1 & 4.2 & 13.14 \\ -2.4 & -0.9 & 3.7 & 6.4 & -4.34 \\ 4 & 3 & -1 & 0 & 4.8 \\ -2.8 & 0.9 & 2.7 & 3 & -3.06 \end{array} \right) \sim \left(\begin{array}{cccc|c} 4 & 3 & -1 & 0 & 4.8 \\ 0 & 3 & 2 & 3 & 0.3 \\ 0 & 0 & -5 & 3 & 8.7 \\ 0 & 0 & 0 & 7 & 2.8 \end{array} \right) \quad \mathbf{x} = \begin{pmatrix} 0.3 \\ 0.7 \\ -1.5 \\ 0.4 \end{pmatrix}$$





Example

- Consider this 3×6 augmented matrix ($m = 3$ and $n = 5$)

$$\left(\begin{array}{ccccc|c} 2 & 4 & -1 & 3 & 5 & 11 \\ 1.6 & 3.2 & 2.2 & 1.2 & 5.8 & 16.6 \\ -0.6 & -1.2 & 5.3 & -2.9 & 1.5 & 9.7 \end{array} \right) \sim \left(\begin{array}{ccccc|c} 2 & 4 & -1 & 3 & 5 & 11 \\ 0 & 0 & 3 & -2.2 & 3.8 & 13.8 \\ 0 & 0 & 3 & -2.2 & 3.8 & 13.8 \end{array} \right)$$

$$\sim \left(\begin{array}{ccccc|c} 2 & 4 & -1 & 3 & 5 & 11 \\ 0 & 0 & 5 & -2 & 3 & 13 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right)$$

$$\mathbf{x} = \begin{pmatrix} 6.8 - 2x_2 - 1.3x_4 - 2.8x_5 \\ x_2 \\ 2.6 + 0.4x_4 - 0.6x_5 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} 6.8 \\ 0 \\ 2.6 \\ 0 \\ 0 \end{pmatrix} + x_2 \begin{pmatrix} -2 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + x_4 \begin{pmatrix} -1.3 \\ 0 \\ 0.4 \\ 1 \\ 0 \end{pmatrix} + x_5 \begin{pmatrix} -2.8 \\ 0 \\ -0.6 \\ 0 \\ 1 \end{pmatrix}$$





Example

- Consider this 5×4 augmented matrix ($m = 5$ and $n = 3$)

$$\left(\begin{array}{ccc|c} 6 & 3.2 & -2.4 & -18.84 \\ 1.5 & 0.9 & 10.3 & 17.67 \\ 3.5 & 2.9 & 0.1 & 3.97 \\ 6 & 3.2 & -2.4 & -18.84 \\ -2 & -9.2 & 5.6 & 19.04 \end{array} \right) \sim \left(\begin{array}{ccc|c} 5 & 3 & 1 & 8.9 \\ 0 & -8 & 16 & 22.6 \\ 0 & 0.8 & -0.6 & -2.26 \\ 0 & 7.2 & -2.4 & -15.84 \\ 0 & -8 & 16 & 22.6 \end{array} \right)$$

$$\sim \left(\begin{array}{ccc|c} 5 & 3 & 1 & 8.9 \\ 0 & -8 & 6 & 22.6 \\ 0 & 0 & 10 & 16 \\ 0 & 0 & 3 & 4.5 \\ 0 & 0 & 10 & 16 \end{array} \right) \sim \left(\begin{array}{ccc|c} 5 & 3 & 1 & 8.9 \\ 0 & -8 & 6 & 22.6 \\ 0 & 0 & 10 & 15 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array} \right) \quad \mathbf{x} = \begin{pmatrix} 2.5 \\ -1.7 \\ 1.5 \end{pmatrix}$$





Example

- Consider this 5×6 augmented matrix ($m = n = 5$)

$$\left(\begin{array}{ccccc|c} 3.1 & 1.5 & -0.9 & 0.5 & 0.4 & -0.72 \\ 3.1 & 1.5 & -0.9 & 0.5 & 0.4 & -0.72 \\ -2.4 & -2 & 5.6 & 1 & 0 & 11.66 \\ 0.9 & -3.7 & 2 & 0 & 0 & 11.49 \\ -1.5 & 2.7 & 2.9 & -9.1 & -2 & -2.32 \end{array} \right) \sim \left(\begin{array}{ccccc|c} 3 & 1 & 0 & 0 & 0 & -1.7 \\ 0 & -0.8 & -2.9 & 0.5 & 0.4 & 10.47 \\ 0 & -1.2 & 5.6 & 1 & 0 & 10.3 \\ 0 & -0.8 & -2.9 & 0.5 & 0.4 & 10.47 \\ 0 & 3.2 & 2.9 & -9.1 & -2 & -3.17 \end{array} \right)$$

$$\sim \left(\begin{array}{ccccc|c} 3 & 1 & 0 & 0 & 0 & -1.7 \\ 0 & -4 & -2 & 0 & 0 & 12 \\ 0 & 0 & 5 & 1 & 0 & 6.7 \\ 0 & 0 & -2.5 & 6.5 & 6.4 & 2.87 \\ 0 & 0 & 4.5 & -9.1 & -2 & 6.43 \end{array} \right) \sim \left(\begin{array}{ccccc|c} 3 & 1 & 0 & 0 & 0 & -1.7 \\ 0 & -4 & -2 & 0 & 0 & 12 \\ 0 & 0 & 5 & 1 & 0 & 6.7 \\ 0 & 0 & 0 & -10 & -0.4 & 0.42 \\ 0 & 0 & 0 & -10 & -0.4 & 0.42 \end{array} \right)$$

$$\sim \left(\begin{array}{ccccc|c} 3 & 1 & 0 & 0 & 0 & -1.7 \\ 0 & -4 & -2 & 0 & 0 & 12 \\ 0 & 0 & 5 & 1 & 0 & 6.7 \\ 0 & 0 & 0 & -10 & -2 & 0.4 \\ 0 & 0 & 0 & 0 & 5 & 6.5 \end{array} \right) \quad \mathbf{x} = \begin{pmatrix} 0.2 \\ -2.3 \\ 1.4 \\ -0.3 \\ 1.3 \end{pmatrix}$$





Important point!

- Partial pivoting, as we have defined it, is absolutely necessary when programming the Gaussian elimination algorithm
- If you are solving a system of linear equations by hand, you are 100% welcome to pick the easiest pivot

$$\left(\begin{array}{cc|c} 3 & 5 & -0.9 \\ -1 & 2 & -1.9 \end{array} \right) \sim \left(\begin{array}{cc|c} -1 & 2 & -1.9 \\ 3 & 5 & -0.9 \end{array} \right) \sim \left(\begin{array}{cc|c} -1 & 2 & -1.9 \\ 0 & 11 & -6.6 \end{array} \right)$$

$$\mathbf{x} = \begin{pmatrix} 0.7 \\ -0.6 \end{pmatrix}$$

- There will be times in this course where we will not apply partial pivoting
 - I will explicitly state that partial pivoting is not being used





Summary

- Following this topic, you now
 - Understand that we must take floating-point numbers into consideration
 - Know the Gaussian elimination algorithm with partial pivoting
 - Can apply the steps to convert a matrix into row-echelon form using this algorithm
 - Indicating that the partial pivoting aspect is unnecessary for smaller back-of-the-envelope calculations





References

- [1] https://en.wikipedia.org/wiki/Gaussian_elimination
- [2] https://en.wikipedia.org/wiki/Pivot_element





Acknowledgments

Tina Hanna for noting a mistake in Slide 27.





Colophon

These slides were prepared using the Cambria typeface. Mathematical equations use Times New Roman, and source code is presented using Consolas. Mathematical equations are prepared in MathType by Design Science, Inc. Examples may be formulated and checked using Maple by Maplesoft, Inc.

The photographs of flowers and a monarch butter appearing on the title slide and accenting the top of each other slide were taken at the Royal Botanical Gardens in October of 2017 by Douglas Wilhelm Harder. Please see

<https://www.rbg.ca/>

for more information.





Disclaimer

These slides are provided for the NE 112 *Linear algebra for nanotechnology engineering* course taught at the University of Waterloo. The material in it reflects the authors' best judgment in light of the information available to them at the time of preparation. Any reliance on these course slides by any party for any other purpose are the responsibility of such parties. The authors accept no responsibility for damages, if any, suffered by any party as a result of decisions made or actions based on these course slides for any other purpose than that for which it was intended.

