Applications

- Robot path planning
- Network flows (transportation, circuits, communication)
- Equilibria (physics, economics)

Topics

- Solutions of systems of linear equations (ALA 1.1)
    - equivalent systems
    - Back substitution & triangular systems

- Matrices & vectors (ALA 1.2 & 1.6)
    - definitions; special matrices, row & col'n vectors
    - Matrix arithmetic & multiplication
    - linear systems in matrix-vector notation

- Gaussian Elimination (Regular Case) (ALA 1.3)
    - Augmented matrices
    - Pivots
    - Upper triangular matrices & triangular systems
    - Regular Gaussian Elimination & A matrices
    - LU-factorization & forward/back substitution

Solution of Linear Systems

Warm up: Three linear equations in three unknowns x, y, z:

$$x + 2y + z = 2 \qquad (1)$$
$$2x + 6y + z = 7 \qquad (2)$$
$$x + y + 4z = 3 \qquad (3)$$

Linear because unknowns only appear to the 1st power, and no product terms like xy or xyz.

How can we solve this linear system of equations for solution (s_x, s_y, s_z) that, when plugged into above, satisfies (1), (2), (3)?

Idea! We know how to solve equations that look like

$$\frac{5}{2}x = 6, \quad 5 - z = 12, \quad 1 + 3y = -4$$

Can we turn (1)–(3) into an equivalent system that has equations we know how to solve?

Tool: Add a multiple of one equation to another (T1)

** You should convince yourself that using (T1) does not change the solution to the linear system (1)–(3). **

We will proceed systematically: first, we try to eliminate x from (2) & (3) by adding multiples of (1) to them:

$$\begin{array}{rl} 2x + 6y + z = 7 & \text{Eqn 2} \\ -2\cdot[\;x + 2y + z = 2\;] & -2\cdot[\text{Eqn 1}] \\ \hline 0x + 2y - z = 3 & \end{array} \qquad \begin{array}{rl} x + y + 4z = 3 & \text{Eqn 3} \\ -1\cdot[\;x + 2y + z = 2\;] & -\text{Eqn 1} \\ \hline 0x - y + 3z = 1 & \end{array}$$

This gives the equivalent system:

$$2x + 6y + z = 2 \qquad (4)$$
$$\boxed{\begin{array}{r} 2y - z = 3 \\ -y + 3z = 1 \end{array}} \qquad \begin{array}{l} (5) \\ (6) \end{array}$$

Progress! The unknown x has been eliminated from the bottom 2 equations

Let's focus on the two equations in the pink square above and try to eliminate y from (6):

$$\begin{array}{rrrl} & -y + 3z & = 1 & \left[\text{Eqn } 6\right. \\ +\frac{1}{2}\cdot & [2y - z & = 3] & \left.+\tfrac{1}{2}[\text{Eqn } 5]\right] \\ \hline & 0y + \frac{5}{2}z & = \frac{5}{2} & \end{array}$$

This gives the following triangular system:

$$\begin{aligned} x + 2y + z &= 2 \\ 2y - z &= 3 \\ \tfrac{5}{2}z &= \tfrac{5}{2} \end{aligned} \qquad \text{(Tri)}$$

The name is self-explanatory, but it has a very important consequence:

Triangular Systems can be easily solved via <u>back substitution</u>!

As the name suggests, back substitution consists of solving the last equation first & then working backwards:

① $\frac{5}{2}z = \frac{5}{2} \Rightarrow z = 1$

② $2y - z = 3,\ z = 1 \Rightarrow 2y - 1 = 3 \Rightarrow 2y = 4 \Rightarrow y = 2$

③ $x + 2y + z = 2,\ y = 2,\ z = 1 \Rightarrow x + 4 + 1 = 2 \Rightarrow x = -3$

$(-3, 2, 1)$ solves the linear System (1)-(3)! You should check!

In this case, $(-3, 2, 1)$ is the unique solution, i.e., one and only, to (1)-(3).

This is, modulo some small details, Gaussian Elimination.

Our goal in this class is to move from small systems you can solve by hand to big ones that you need a computer to solve.

To do this, we need a more convenient way of writing big systems, w/ for example 1000s of variables & equations.

Matrices & Vectors

A matrix is a rectangular array of numbers:

$$\begin{bmatrix} 1 & 0 & 3 \\ -2 & 4 & 1 \end{bmatrix}, \quad \begin{bmatrix} \pi & 0 \\ e & 1/2 \\ -1 & .83 \\ \sqrt{2} & -4/7 \end{bmatrix}, \quad [.2, -1.6, .32], \quad \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

are all examples of matrices.

A general matrix of size $\underset{\#\text{rows}}{m} \times \underset{\#\text{col'ns}}{n}$ is written as

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}, \qquad a_{ij}: \text{ entry in row } i \text{ \& col'n } j \text{ of matrix } A$$

A $m \times 1$ matrix is called a column vector:

2×1 column vectors: $\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} -1 \\ 12 \end{bmatrix}, \begin{bmatrix} \pi \\ e \end{bmatrix}$ 2-vectors

3×1 column vectors: $\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \begin{bmatrix} \sqrt{3} \\ -12 \\ \sqrt{\pi} \end{bmatrix}$ 3-vectors.

Since column vectors are going to be very important to us, we will often call a $m \times 1$ column vector a m-vector.

Less important to us, but still needed will be row vectors. These are $1 \times n$ matrices:

1×2 row vectors: $[0 \;\; 0]$, $[-1 \;\; 12]$ $[\pi \;\; e]$

1×3 row vectors: $[0 \;\; 0 \;\; 0]$, $[1 \;\; 2 \;\; 3]$, $[\sqrt{3} \;\; -12 \;\; \sqrt{\pi}]$

Although similar, row and column vectors <u>are not the same</u>! We'll see why this distinction matters later in the course.

Matrix Arithmetic

Three basic operations: matrix addition, scalar multiplication, matrix multiplication.

Matrix addition:

To add two matrices A and B, they must be the same size. For A, B the same size, $C = A+B$ is computed entry by entry, that is

$$C = A+B \iff c_{ij} = a_{ij} + b_{ij}$$

Example: $\begin{bmatrix} 1 & 2 \\ -1 & 0 \end{bmatrix} + \begin{bmatrix} 3 & -5 \\ 2 & 1 \end{bmatrix} = \begin{bmatrix} 1+3 & 2-5 \\ -1+2 & 0+1 \end{bmatrix} = \begin{bmatrix} 4 & -3 \\ 1 & 1 \end{bmatrix}$

Matrix addition otherwise behaves just like scalar addition:

- Commutative: $A+B = B+A$
- Associative: $A+(B+C) = (A+B)+C$

Scalar multiplication:

A scalar is a fancy name for an ordinary number. 95% of this class will consider real valued scalars. To tell you that a number c is a real scalar, we will write $c \in \mathbb{R}$ ← symbol for real line

means "lives in"

Scalar multiplication takes a scalar $c \in \mathbb{R}$ and an $m \times n$ matrix A and computes the $m \times n$ matrix $B = cA$ by multiplying (or "scaling") each entry of A by c.

Example: $c = 3$, $A = \begin{bmatrix} 1 & 2 \\ -1 & 0 \end{bmatrix}$, $cA = 3\begin{bmatrix} 1 & 2 \\ -1 & 0 \end{bmatrix} = \begin{bmatrix} 3 & 6 \\ -3 & 0 \end{bmatrix}$.

In general, if $B = cA$, then $b_{ij} = ca_{ij}$.

Matrix multiplication

Warm up: a row vector with a column vector

Let $\underline{a}$ be a $1 \times n$ row vector & $\underline{x}$ a $n \times 1$ col'n vector

** We will underline symbols for vectors to help distinguish them from scalars. Capital letters (A, B, C) will be reserved for matrices. **

The product $\underline{a}\,\underline{x}$ is a scalar defined as follows:

$$\underline{a}\,\underline{x} = [a_1 \ a_2 \ \cdots \ a_n] \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = a_1x_1 + a_2x_2 + \cdots + a_nx_n = \sum_{k=1}^{n} a_k x_k$$

Matrix - Matrix multiplication:

To multiply two matrices A and B, they must be of compatible size: A must have the same number of columns as B has rows, so if A is a $m \times n$ matrix, B must be a $n \times p$ matrix. The resulting product $C = AB$ is then a $m \times p$ matrix, defined by

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$$

The (i,j) entry of $C = AB$ is the product of the i^{th} row of A and the j^{th} column of B.

Another convenient way of computing $C = AB$ is as follows. We denote the columns of B by $\underline{b_1}, \underline{b_2}, \dots, \underline{b_p}$ so that $B = [\underline{b_1} \ \underline{b_2} \ \cdots \ \underline{b_p}]$. Then

$$C = AB = A[\underline{b_1} \ \underline{b_2} \ \cdots \ \underline{b_p}] = [A\underline{b_1} \ A\underline{b_2} \ \cdots \ A\underline{b_p}],$$

i.e., the k^{th} column of $C = AB$ is computed by the matrix-vector product of A and the k^{th} column $\underline{b_k}$ of B.

An important special case is matrix-vector products. Let A be $m \times n$ matrix and $\underline{x}$ a $n \times 1$ column vector. Then, the matrix-vector product $\underline{b} = A\underline{x}$ is an $m \times 1$ column vector, with entries

(Since $\underline{b}$ is a column vector, we do not write the column index since it is always $j=1$)

$$b_i = \sum_{k=1}^{n} a_{ik} b_i$$

** If we let $\underline{a}_1, \underline{a}_2, \ldots, \underline{a}_n$ be the columns of A so that $A = [\underline{a}_1\ \underline{a}_2\ \cdots\ \underline{a}_n]$
Then another formula for $\underline{b} = A\underline{x}$ is

$$\underline{b} = x_1 \underline{a}_1 + x_2 \underline{a}_2 + \cdots + x_n \underline{a}_n,$$

i.e., $\underline{b}$ is computed by adding the columns of A together, weighted by the entries of $\underline{x}$.

This will be useful to us later when we think about the column span of a matrix **

Two special matrices:

Identity matrix: $I = I_n = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 \end{bmatrix}$ $n \times n$ matrix w/ $I_{ii} = 1$ & $I_{ij} = 0$ for $i \neq j$

Key property: $IA = AI = A$ (note I & I may be different sizes if A is not square)

Zero matrix: $O = O_{m \times n}$ is the all zeros matrix

** You will often need to infer dimensions from context. It may take some practice, but you'll get used to it in no time. **

Basic Matrix Arithmetic

Matrix Addition:	Commutativity	$A + B = B + A$
	Associativity	$(A + B) + C = A + (B + C)$
	Zero Matrix	$A + \mathrm{O} = A = \mathrm{O} + A$
	Additive Inverse	$A + (-A) = \mathrm{O}, \quad -A = (-1)A$
Scalar Multiplication:	Associativity	$c(dA) = (cd)A$
	Distributivity	$c(A + B) = (cA) + (cB)$ $(c + d)A = (cA) + (dA)$
	Unit Scalar	$1A = A$
	Zero Scalar	$0A = \mathrm{O}$
Matrix Multiplication:	Associativity	$(AB)C = A(BC)$
	Distributivity	$A(B + C) = AB + AC,$ $(A + B)C = AC + BC,$
	Compatibility	$c(AB) = (cA)B = A(cB)$
	Identity Matrix	$A\,\mathrm{I} = A = \mathrm{I}\,A$
	Zero Matrix	$A\,\mathrm{O} = \mathrm{O}, \quad \mathrm{O}\,A = \mathrm{O}$

WARNING: $AB \neq BA$ in general. If $AB = BA$, we say A and B commute

WARNING: $AB = AC \not\Rightarrow B = C$

WARNING: $AB = 0 \not\Rightarrow A = 0$ or $B = 0$

For now, these are just quirks of how we defined matrix multiplication. By the end of this semester, these will make perfect sense to you, as will the (seemingly arbitrary) definition of matrix multiplication.

Example: $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$, $B = \begin{bmatrix} 1 & 2 \\ 1 & 2 \\ 1 & 2 \end{bmatrix}$, $\underline{x} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$, $\underline{y} = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}$

$$BA = \begin{bmatrix} 1 & 2 \\ 1 & 2 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 7 & 10 \\ 7 & 10 \\ 7 & 10 \end{bmatrix},$$ AB does not exist

$$A\underline{x} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \end{bmatrix} = -1 \begin{bmatrix} 1 \\ 3 \end{bmatrix} + 1 \cdot \begin{bmatrix} 2 \\ 4 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$BA\underline{x} = \underbrace{\begin{bmatrix} 7 & 10 \\ 7 & 10 \\ 7 & 10 \end{bmatrix}}_{BA} \underbrace{\begin{bmatrix} -1 \\ 1 \end{bmatrix}}_{\underline{x}} = \underbrace{\begin{bmatrix} 1 & 2 \\ 1 & 2 \\ 1 & 2 \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} 1 \\ 1 \end{bmatrix}}_{A\underline{x}} = \begin{bmatrix} 3 \\ 3 \\ 3 \end{bmatrix}$$

$$\underline{y}B = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 1 & 2 \\ 1 & 2 \end{bmatrix} = \begin{bmatrix} 6 & 12 \end{bmatrix}$$

$$\underline{y}B\underline{x} = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 1 & 2 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 6 & 12 \end{bmatrix}}_{\underline{y}B} \underbrace{\begin{bmatrix} -1 \\ 1 \end{bmatrix}}_{\underline{x}} = 6$$

** Make sure you understand how we computed all of these products. **

** A trick for sizes: $A_{m \times n} \, B_{n \times p} = C_{m \times p}$ **

inner must match and eat each other

outer dimensions survive

Systems of Linear Equations: $A\underline{x} = \underline{b}$

Let us rewrite our example linear system

$$\begin{aligned} x + 2y + z &= 2 \\ 2x + 6y + z &= 7 \\ x + y + 4z &= 3 \end{aligned}$$

as a vector equation $A\underline{x} = \underline{b}$. We first notice we can enforce this by stacking the left & right hand sides into vectors

$$\underbrace{\begin{bmatrix} x + 2y + z \\ 2x + 6y + z \\ x + y + 4z \end{bmatrix}}_{A\underline{x}} = \underbrace{\begin{bmatrix} 2 \\ 7 \\ 3 \end{bmatrix}}_{\underline{b}}.$$

we call $\underline{b}$ the right hand side or RHS

It is easy to read off $\underline{b}$ & $A\underline{x}$. Next we need to split $A\underline{x}$ into a coefficient matrix A and unknowns vector $\underline{x}$. It is also easy to read these off as

$$A = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 6 & 1 \\ 1 & 1 & 4 \end{bmatrix} \quad \text{and} \quad \underline{x} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}.$$

You can check that $A\underline{x}$ indeed gives the left hand side of our vector equation.

Gaussian Elimination: Regular Case

For a vector equation $A\underline{x} = \underline{b}$, we define the augmented matrix

$$M = [A \mid b] = \left[\begin{array}{cccc|c} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} & b_m \end{array}\right]$$

just to remind us that last col'n is special.

If A is an $m \times n$ matrix, M is the $m \times (n+1)$ matrix obtained by concatenating $\underline{b}$ on the end.

Example:

$$\begin{aligned} x + 2y + z &= 2 \\ 2x + 6y + z &= 7 \\ x + y + 4z &= 3 \end{aligned} \Rightarrow \underbrace{\begin{bmatrix} 1 & 2 & 1 \\ 2 & 6 & 1 \\ 1 & 1 & 4 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x \\ y \\ z \end{bmatrix}}_{\underline{x}} = \underbrace{\begin{bmatrix} 2 \\ 7 \\ 3 \end{bmatrix}}_{\underline{b}} \Rightarrow M = \left[\begin{array}{ccc|c} 1 & 2 & 1 & 2 \\ 2 & 6 & 1 & 7 \\ 1 & 1 & 4 & 3 \end{array}\right]$$

Tool: adding a scalar multiple of one row of M to another.

** You should convince yourself this is doing the same thing as (T1) **

Example: add -2 times the first row to the second row gives:

$$-2[1 \quad 2 \quad 1 \quad 2] + [2 \quad 6 \quad 1 \quad 7] = [0 \quad 2 \quad -1 \quad 3]$$

giving the new augmented matrix

$$\left[\begin{array}{ccc|c} 1 & 2 & 1 & 2 \\ 0 & 2 & -1 & 3 \\ 1 & 1 & 4 & 3 \end{array}\right]$$

We call the $(1,1)$ entry of A the first pivot.

** Key requirement: pivots must be nonzero **

Eliminating x from 2^{nd} & 3^{rd} equation means making all entries below the first pivot zero. $(2,1)$ entry is already zero, & we can eliminate $(3,1)$ entry by adding -1 times the 1^{st} equation to the last, giving us:

$$\left[\begin{array}{ccc|c} 1 & 2 & 1 & 2 \\ 0 & 2 & -1 & 3 \\ 0 & -1 & 3 & 1 \end{array}\right]$$

The second pivot is the $(2,2)$ entry, which is 2 (important: it is nonzero)

We add $\frac{1}{2}$ the 2^{nd} row to the 3^{rd} row to zero out everything below it.

$$\left[\begin{array}{ccc|c} 1 & 2 & 1 & 2 \\ 0 & 2 & -1 & 3 \\ 0 & 0 & \frac{5}{2} & \frac{5}{2} \end{array}\right]$$

which maps exactly onto the triangular system (Tri) we saw earlier!

We'll write this as $N = \left[\begin{array}{c|c} \underbrace{\begin{matrix} 1 & 2 & 1 \\ 0 & 2 & -1 \\ 0 & 0 & 5/2 \end{matrix}}_{U} & \underbrace{\begin{matrix} 2 \\ 3 \\ 5/2 \end{matrix}}_{\underline{c}} \end{array}\right]$

which defines the corresponding linear system $U\underline{x} = \underline{c}$.

The coefficient matrix U is upper triangular, named for obvious reasons.
The three nonzero entries along its diagonal, $1, 2, 5/2$, are the three pivots.

We can solve $U\underline{x} = \underline{c}$ easily via Back Substitution.

This procedure for solving a linear system of n equations in n unknowns is called regular Gaussian Elimination.

A square matrix A will be called regular if the above algorithm succeeds and produces a U with all nonzeros along the diagonal

The LU-Factorization

Idea! Can we encode the row operations above as a matrix product, i.e., can I find a matrix E so that

$$EM = EN, \text{ i.e., } EA = U \ \& \ E\underline{b} = \underline{c}$$

We will show how to build E from elementary matrices.

The elementary matrix for a row operation on a matrix with m rows is the $m \times m$ matrix obtained by applying the row operation to I_m.

For example, adding -2 times row 1 to row 2 of I_3 gives

$$E_1 = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Let's check this does what we want it to:

$$E_1 A = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 6 & 1 \\ 1 & 1 & 4 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 2 & -1 \\ 1 & 1 & 4 \end{bmatrix}$$

It does! So let's compactly encode step 2 (adding $-1 \cdot$ (row 1) to row 3) & step 3 (adding $\frac{1}{2}$(row 2) to row 3) via

$$E_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}, \qquad E_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1/2 & 1 \end{bmatrix}$$

and then importantly, apply them in the right order, we get

$$\underbrace{E_3 E_2 E_1}_{E} A = U = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 2 & -1 \\ 0 & 0 & 5/2 \end{bmatrix}$$

- The way to read $E_3 E_2 E_1 A$ is from right to left. We start w/ A, then apply E_1, then E_2, then E_3.
- Because of associativity of matrix multiplication, we don't need to compute things in that order. For example, we could just compute

$$E = E_3 E_2 E_1$$

and then $U = EA$.

The LU-Factorization

We can also "undo" the action of an elementary matrix using the corresponding inverse elementary matrix. For example to undo the action of E_1, which adds $-2 \cdot (\text{row } 1)$ to row 2, we simply add $2 \cdot (\text{row } 1)$ back to row 2. The corresponding matrix is

$$L_1 = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Notice $L_1 E_1 = I_3$. This makes sense as we are "undoing" the action of E_1 w/ L_1, so the end result should be to do nothing.

We can define appropriate inverses for E_2 & E_3:

$$L_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}, \quad L_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1/2 & 1 \end{bmatrix}$$

Now we start with $U = E_3 E_2 E_1 A$ & "undo" our actions by left-multiplying both sides by $L_1 L_2 L_3$ (notice the order!)

$$\begin{aligned} (L_1 L_2 L_3) U &= L_1 L_2 \underbrace{\cancel{L_3 E_3}}_{I} E_2 E_1 A \\ &= L_1 \underbrace{\cancel{L_2 E_2}}_{I} E_1 A \\ &= \underbrace{\cancel{L_1 E_1}}_{I} A \end{aligned}$$

- Calling $L = L_1 L_2 L_3$, we just showed that $A = LU$. We end by noting that because L_1, L_2, L_3 are all lower triangular, so is their product.

- This is the LU-factorization of A, which decomposes A as a product of a lower and upper-triangular matrix.

- This is important because implementing LU-factorizations of A using computer code is easy to do, letting us use these ideas on very big linear systems.

- Next we show how LU-factorization leads to an easy solving $A\underline{x} = \underline{b}$.

Forward and Back Substitution

Given LU-factorization $A = LU$, we solve $\underline{A\underline{x}} = \underline{b}$ in two steps. The idea is to rewrite $A\underline{x} = \underline{b}$ as two linear systems:

$$A\underline{x} = L\underbrace{U\underline{x}}_{\underline{z}} = \underline{b} \iff \begin{array}{l} L\underline{z} = \underline{b} \\ U\underline{x} = \underline{z} \end{array}$$

① Solve $L\underline{z} = \underline{b}$ via forward substitution. L is lower-triangular so this is just like Back substitution, but starting from the top instead of bottom.

② Using the $\underline{z}$ we just found, solve $U\underline{x} = \underline{z}$ via Back substitution

This works because if $U\underline{x} = \underline{z}$ and $L\underline{z} = \underline{b}$ then $A\underline{x} = LU\underline{x} = L\underline{z} = \underline{b}$.

See online notes for worked examples.