MCEN 3030

13 Feb 2024

First midterm
Tues Feb 27?

HW#3 due this week

Last time: Linear Systems $\rightarrow$ $Ax = b$

Today: Gaussian Elimination

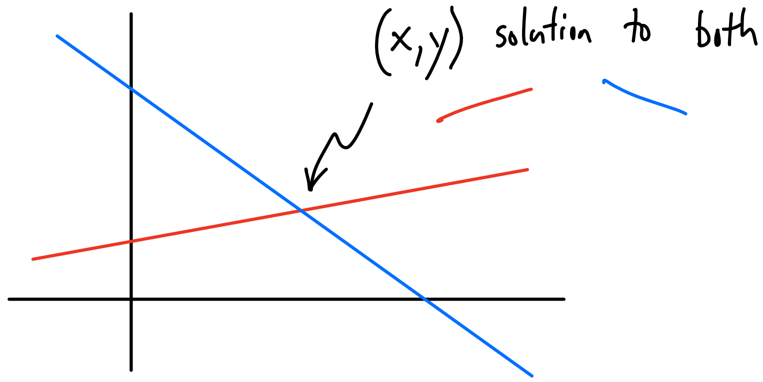
$\rightarrow$ LU Decomposition

Last time: Linear Systems

A 2D system like

$$\begin{aligned} a_{11}x + a_{12}y &= b_1 \\ a_{21}x + a_{22}y &= b_2 \end{aligned}$$

can be represented as two lines in the (x,y)-plane. The solution to this system is the intersection point.



it is entirely analogous in higher-dimensional systems:

$$(x, y) \longrightarrow (x_1, x_2, x_3, \dots, x_n)$$

to determine a unique solution we need n linearly independent equations

Gaussian Elimination

Explain with an example

$$\begin{aligned} 3x_1 + 4x_2 + 5x_3 &= 1 \\ x_1 + x_2 + x_3 &= 2 \\ 3x_1 - 4x_2 + x_3 &= 2 \end{aligned} \longrightarrow \underset{A}{\begin{bmatrix} 3 & 4 & 5 \\ 1 & 1 & 1 \\ 3 & -4 & 1 \end{bmatrix}} \underset{x}{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}} = \underset{b}{\begin{bmatrix} 1 \\ 2 \\ 2 \end{bmatrix}}$$

Things that are totally legal:

- Multiply any equation by a constant
- Add or subtract lines from each other, e.g. replace (row 2) with (row 2) − (row 1)
- Combine these operations: (row 2) − 3(row 1)

So, we might solve for $\underline{x}$ via the following steps:

- Subtract $\frac{1}{3}$(first row) from (second row)

$$\begin{array}{rcl} x_1 + x_2 + x_3 & = & 2 \\ -\frac{1}{3}(3x_1 + 4x_2 + 5x_3) & = & -\frac{1}{3}(1) \\ \hline 0x_1 - \frac{1}{3}x_2 - \frac{2}{3}x_3 & = & \frac{5}{3} \end{array}$$

- And subtract $\frac{3}{3}$(first row) from (third row)

$$
\begin{array}{rl}
3x_1 - 4x_2 + x_3 & = 2 \\
-\frac{3}{3}(3x_1 + 4x_2 + 5x_3) & = -\frac{3}{3}(1) \\
\hline
0x_1 - 8x_2 - 4x_3 & = 1
\end{array}
$$

Intermediate result: the system has become

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

- Subtract 24(new second row) from (new third row)

$$
\begin{array}{rl}
-8x_2 - 4x_3 & = 1 \\
-24\left(\frac{-1}{3}x_2 - \frac{2}{3}x_3\right) & = -24 \cdot \frac{5}{3} \\
\hline
0x_2 + 12x_3 & = -39
\end{array}
$$

So we now have

$$
\begin{bmatrix} 3 & 4 & 5 \\ 0 & -\frac{1}{3} & -\frac{2}{3} \\ 0 & 0 & 12 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ \frac{5}{3} \\ -39 \end{bmatrix}
$$

We now can solve using the "back-substitution algorithm":

$$12x_3 = -39 \implies x_3 = \frac{-39}{12}$$

$$-\frac{1}{3}x_2 - \frac{2}{3}\underset{\substack{\uparrow \\ \text{known}}}{x_3} = \frac{5}{3} \implies x_2 = \frac{3}{2}$$

$$3x_1 + 4x_2 + 5x_3 = 1 \implies x_1 = \frac{15}{4}$$

More generally:

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

$$A \quad x \quad = \quad b$$

a_{24}

row column

Idea: convert this system into one represented by an "upper-triangular matrix"

$\underline{d} \neq \underline{b}$

$$\begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{1n} \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & u_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix}$$

$$U \quad x \quad = \quad d$$

must figure this out

same x

must figure this out

Once we are here, we can use the "back-substitution algorithm" to determine $\underline{x}$

known, or at least generated from a known

Last row: $$u_{nn} x_n = d_n$$

$$\Rightarrow \quad x_n = \frac{d_n}{u_{nn}}$$

Second-to-last row: $$u_{n(n-1)} x_{n-1} + u_{nn} x_n = d_{n-1}$$

$$\Rightarrow \quad x_{n-1} = \frac{d_{n-1} - u_{nn} x_n}{u_{n(n-1)}}$$

size of matrix $n \times n$

General: $$\boxed{x_i = \frac{1}{u_{ii}} \left(d_i - \sum_{j=i+1}^{n} u_{ij} x_j \right)}$$

So, what is the algorithm to generate U & d?

From "augmented matrix"

$$\left[\begin{array}{cccc:c} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{1n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & b_n \end{array}\right]$$

$$(\text{second row}) \text{ becomes } (\text{second row}) - \frac{a_{21}}{a_{11}}(\text{first row})$$

$$(\text{third row}) \text{ becomes } (\text{third row}) - \frac{a_{31}}{a_{11}}(\text{first row})$$

$$\vdots \text{ to } n^{\text{th}} \text{ row}$$

Intermediate result:

first row stays the same

first column zeros except first row

$$\left[\begin{array}{cccc:c} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ 0 & a'_{22} & \cdots & a'_{1n} & b'_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & a'_{n2} & \cdots & a'_{nn} & b'_n \end{array}\right]$$

these have changed

(note: we have done $n-1$ row replacements)

Then, from this intermediate matrix...

$$(\text{new third row}) \text{ becomes } (\text{new third row}) - \frac{a'_{32}}{a'_{22}}(\text{new second row})$$

$$(\text{new fourth row}) \text{ becomes } (\text{new fourth row}) - \frac{a'_{42}}{a'_{22}}(\text{new second row})$$

$$\vdots \quad \text{to } n^{\text{th}} \text{ row}$$

Another intermediate matrix:

first two rows stay the same

first column $n-1$ zeros

$$\left[\begin{array}{cccccc|c}
a_{11} & a_{12} & a_{13} & a_{14} & \cdots & & b_1 \\
0 & a'_{22} & a'_{23} & a'_{24} & \cdots & & b'_2 \\
0 & \boxed{0} & a''_{33} & a''_{34} & \cdots & & b''_3 \\
0 & 0 & a''_{43} & a''_{44} & \cdots & & b''_4 \\
\vdots & \vdots & \vdots & \vdots & \ddots & & \vdots
\end{array}\right]$$

these have changed

second column $n-2$ zeros

So this step took $n-2$ row operations.

We continue on until we have an upper-diagonal matrix, U.

Slow process! Total number of row operations:

$$(n-1) + (n-2) + \dots + \underbrace{(n-(n-1))}_{1}$$

$$= \frac{n(n-1)}{2}$$

Is there a way to speed up the creation of U?

Not really... can't know, say, $a_{(17)(23)}^{'''''}$ without having calculated. But! There are many applications in which we are interested in seeing the effect of different forcing functions, b, and the matrix A is the same in each case.

→ Can determine U <u>once</u>, and apply it to all the forcing functions.

(But, $Ux = d$, so need to figure out if it is easy to get d from b)

LU Decomposition

$$\underbrace{A}_{LU} x = b$$

row operations encapsulated in a matrix

You can generate the matrix L as a by-product of the creation of U

Then, if $Ux = d$

$$\Rightarrow \underbrace{LU}_{A}x = \underbrace{Ld}_{b} \quad \Rightarrow \quad Ld = b$$

... and L is a lower-triangular matrix

$\Rightarrow$ it is cheap/fast to create L

With L & b known we can determine d via a "forward-substitution algorithm"

$$\begin{bmatrix} l_{11} & 0 & 0 & \cdots \\ l_{21} & l_{22} & 0 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \end{bmatrix} \Rightarrow \begin{aligned} d_1 &= \frac{b_1}{l_{11}} \\ d_2 &= \frac{b_2 - l_{21}d_1}{l_{22}} \end{aligned}$$

$\Rightarrow$ this is fast too!

Then, we have L, U, d, are now known, and the last step of the algorithm is back-substitution: $Ux = d \longrightarrow x$ solved

this is fast too!

So, the most expensive step by far is the generation of U.

Algorithm: for $Ax = b \longrightarrow Ux = d$

(1) Row operations to generate U (slow)

(1a) As part of this, we get L as a by-product (fast)

(2) Use forward-substitution to get d from $Ld = b$ (fast)

(3) Use back-substitution to get x from $Ux = d$ (fast)

* Only (2) & (3) change if we are testing multiple b's

What is L?

$$
\underbrace{\begin{bmatrix}
1 & 0 & & & & & \\
\frac{a_{21}}{a_{11}} & 1 & 0 & & & & \\
\frac{a_{31}}{a_{11}} & \frac{a'_{32}}{a'_{22}} & 1 & 0 & & & \\
\frac{a_{41}}{a_{11}} & \frac{a'_{42}}{a'_{22}} & \frac{a''_{43}}{a''_{33}} & 1 & 0 & & \\
\frac{a_{51}}{a_{11}} & \frac{a'_{52}}{a'_{22}} & \frac{a''_{53}}{a''_{33}} & \frac{a'''_{54}}{a'''_{44}} & 1 & 0 & \cdots \\
\vdots & \vdots & & & \vdots & &
\end{bmatrix}}_{L}
\underbrace{\begin{bmatrix}
a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & \cdots \\
0 & a'_{22} & a'_{23} & a'_{24} & a'_{25} & \\
 & 0 & a''_{33} & a''_{34} & a''_{35} & \\
 & & 0 & a'''_{44} & a'''_{45} & \\
 & & & 0 & a''''_{55} & \cdots \\
 & & & & 0 &
\end{bmatrix}}_{U}
$$

these are the multipliers used in the row change operations, e.g. the first batch includes...

(second row) becomes (second row) $- \frac{a_{21}}{a_{11}}$ (first row)

(third row) becomes (third row) $- \frac{a_{31}}{a_{11}}$ (first row)

$$
\begin{bmatrix}
1 & 0 & 0 & \cdots \\
\frac{a_{21}}{a_{11}} & 1 & 0 & \cdots \\
\frac{a_{31}}{a_{11}} & \cdots & & \cdots
\end{bmatrix}
$$