Linear Systems of Eqn's

Part 2: Orthogonalization & Triangularization

Recap: Singular Value Decomposition (SVD)

$$\begin{array}{ccccccc} & n & & m & n & n & \\ m & \begin{bmatrix} & & \\ & & \\ & & \end{bmatrix} & = & \begin{bmatrix} | & & | \\ u_1 & \cdots & u_m \\ | & & | \end{bmatrix} & \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \\ & 0 & \end{bmatrix} & \begin{bmatrix} - & v_1^* & - \\ & \vdots & \\ - & v_n^* & - \end{bmatrix} & n \\ & A & = & U & \Sigma & V^* & \end{array}$$

$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n \geq 0$ singular values

$u_1, \dots, u_m \in \mathbb{C}^m$ left singular vectors

$v_1, \dots, v_n \in \mathbb{C}^n$ right singular vectors

U, V are unitary: This means that

$\Rightarrow$ orthonormal columns $\quad u_i^* u_j = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$

$\Rightarrow U^* U = I$

$\Rightarrow U^{-1} = U^*$

$\Rightarrow \|Ux\|_2 = \|x\|_2$ for any $x \in \mathbb{C}^m$

Any one of these implies the others.

Thm: Every matrix $A \in \mathbb{C}^{m \times n}$ has an SVD.

Last time we diagonalized $Ax = b$

with the SVD by rotating inputs & outputs.

$$Ax = b \iff U\Sigma V^* x = b \iff \Sigma(\underbrace{V^* x}_{y}) = (\underbrace{U^* b}_{d})$$

diagonal matrix decouples linear eqns $\leftarrow$ $\Sigma y = d$

$\uparrow$ $\uparrow$ inputs' & outputs' coordinates w.r.t. singular vectors.

Although conceptually powerful, direct diagonalization is usually not the most efficient way to solve $Ax = b$. We'll compromise and triangularize A instead.

LU factorization

Let's focus on the case $n = m$ (square A)

In 18.06, we learn Gaussian elimination:

$$\begin{bmatrix} x & x & x \\ x & x & x \\ x & x & x \end{bmatrix} \xrightarrow{L_1} \begin{bmatrix} x & x & x \\ 0 & x & x \\ 0 & x & x \end{bmatrix} \xrightarrow{L_2} \begin{bmatrix} x & x & x \\ 0 & x & x \\ 0 & 0 & x \end{bmatrix}$$

"Convert A to upper triangular with elementary row operations."

Elementary row op L_k = lower triangular matrix

<u>Illustrate w/example</u>

$$A = \begin{bmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ 6 & 7 & 9 & 8 \end{bmatrix} \xrightarrow{L_1} \begin{bmatrix} 2 & 1 & 1 & 0 \\ & 1 & 1 & 1 \\ & 3 & 5 & 5 \\ & 4 & 6 & 8 \end{bmatrix}$$

$$\underset{L_1}{\begin{bmatrix} 1 & & & \\ -2 & 1 & & \\ -4 & & 1 & \\ -3 & & & 1 \end{bmatrix}} \underset{A}{\begin{bmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ 6 & 7 & 9 & 8 \end{bmatrix}} = \begin{bmatrix} 2 & 1 & 1 & 0 \\ & 1 & 1 & 1 \\ & 3 & 5 & 5 \\ & 4 & 6 & 8 \end{bmatrix}$$

$$\underset{L_2}{\begin{bmatrix} 1 & & & \\ & 1 & & \\ & -3 & 1 & \\ & -4 & & 1 \end{bmatrix}} \underset{L_1 A}{\begin{bmatrix} 2 & 1 & 1 & 0 \\ & 1 & 1 & 1 \\ & 3 & 5 & 5 \\ & 4 & 6 & 8 \end{bmatrix}} = \begin{bmatrix} 2 & 1 & 1 & 0 \\ & 1 & 1 & 1 \\ & & 2 & 2 \\ & & 2 & 4 \end{bmatrix}$$

$$\underset{L_3}{\begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & -1 & 1 \end{bmatrix}} \underset{L_2 L_1 A}{\begin{bmatrix} 2 & 1 & 1 & 0 \\ & 1 & 1 & 1 \\ & & 2 & 2 \\ & & 2 & 4 \end{bmatrix}} = \underset{U}{\begin{bmatrix} 2 & 1 & 1 & 0 \\ & 1 & 1 & 1 \\ & & 2 & 2 \\ & & & 2 \end{bmatrix}}$$

Now invert $L_3 L_2 L_1$ to get $A = LU$

$$L_1^{-1} = \begin{bmatrix} 1 & & & \\ -2 & 1 & & \\ -4 & & 1 & \\ -3 & & & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & & & \\ 2 & 1 & & \\ 4 & & 1 & \\ 3 & & & 1 \end{bmatrix}$$

$$L_2^{-1} = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & -3 & 1 & \\ & -4 & & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & 3 & 1 & \\ & 4 & & 1 \end{bmatrix}$$

$$L^{-1} = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & -1 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & 1 & 1 \end{bmatrix}$$

Easy to invert by forward substitution

$$L_1^{-1} L_2^{-1} L_3^{-1} = \begin{bmatrix} 1 & & & \\ 2 & 1 & & \\ 4 & 3 & 1 & \\ 3 & 4 & 1 & 1 \end{bmatrix}$$

(Check it yourself!)

$$\underbrace{\begin{bmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ 6 & 7 & 9 & 8 \end{bmatrix}}_{A} = \underbrace{\begin{bmatrix} 1 & & & \\ 2 & 1 & & \\ 4 & 3 & 1 & \\ 3 & 4 & 1 & 1 \end{bmatrix}}_{L = L_1^{-1} L_2^{-1} L_3^{-1}} \underbrace{\begin{bmatrix} 2 & 1 & 1 & 0 \\ & 1 & 1 & 1 \\ & & 2 & 2 \\ & & & 2 \end{bmatrix}}_{U}$$

<u>In general</u>

$$a_k = \begin{bmatrix} a_{1k} \\ \vdots \\ a_{kk} \\ a_{k+1,k} \\ \vdots \\ a_{mk} \end{bmatrix} \xrightarrow{L_k} L_k a_k = \begin{bmatrix} x_{1k} \\ \vdots \\ x_{kk} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

L_k = "(row j) $-$ $l_{jk} \cdot$ (row k)" for each $k < j \leq m$

with multipliers $l_{jk} = \dfrac{a_{jk}}{a_{kk}}$

$$L_k = \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & -l_{k+1,k} & 1 & \\ & & \vdots & & \ddots \\ & & -l_{m,k} & & & 1 \end{bmatrix}$$

$k \longrightarrow j$

"(row j) $-$ $l_{jk} \cdot$ (row k)"

Fact 1

$$L_k^{-1} = \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & l_{k+1,k} & 1 & \\ & & \vdots & & \ddots \\ & & l_{mk} & & & 1 \end{bmatrix}$$

Fact 2

$$L = L_1^{-1} \cdots L_{m-1}^{-1} = \begin{bmatrix} 1 & & & & \\ l_{21} & 1 & & & \\ l_{31} & l_{32} & 1 & & \\ \vdots & \vdots & & \ddots & \\ l_{m1} & l_{m2} & \cdots & l_{m,m-1} & 1 \end{bmatrix}$$

In practice L is not usually formed & multiplied explicitly, but multipliers are stored and transformations applied implicitly

Gaussian Elimination (no pivoting)

$U = A$, $L = I$

for $k = 1$ to $m-1$ (each column)

for $j = k+1$ to m (subdiagonal entries)

$l_{jk} = u_{jk} / u_{kk}$

$u_{j,k:m} = u_{j,k:m} - l_{jk} u_{k,k:m}$

end

end

Solve $Ax = b$ with LU ($m \times m$ invertible A)

$Ax = b$ breaks into two triangular solves

$$L(\underbrace{Ux}_{y}) = b$$

1) $Ly = b$ (forward substitution) and 2) $Ux = y$ (backward substitution)

Cost/Complexity

How many floating point operations required?
"flops"

Inside the loop, the work is dominated by multiplying entries of row k on and above the diagonal by multipliers and then subtracting them from corresponding entries of row j: 2 flops $\cdot (m-k-1)^2$

This is repeated for each diagonal entry $k=1$ to $m-1$ so

$$\# \text{flops} \approx \sum_{k=1}^{m-1} 2(m-k-1)^2$$

$$\sim \frac{2}{3} m^3 \quad \text{as} \quad m \to \infty$$

<u>Unfortunately, GE is unstable</u> (without permutations)

$$\hat{A} = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \xrightarrow{\text{Perturb}} A = \begin{bmatrix} 10^{-20} & 1 \\ 1 & 1 \end{bmatrix}$$

GE fails even though A is invertible (divide by zero)

$$L = \begin{bmatrix} 1 & 0 \\ 10^{20} & 1 \end{bmatrix}$$

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

rounded L and U

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

rounded

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

$$A - \tilde{L}\tilde{U} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

not even close!

$Ax = b$

$\tilde{L}\tilde{U} = b$

$b = (1,0)^T$

$x \approx (-1,1)^*$

$\tilde{x} = (0,1)^*$

Solution of $Ax = b$ is not close to solution of $\tilde{L}\tilde{U}x = b$, even though $Ax = b$ is well-conditioned in this instance.