Solving Linear Systems

[Darve 3.1, 3.4, 3.5]

Triangular System :

$$\begin{bmatrix} l_{11} & & 0 \\ l_{21} & l_{22} & \\ \vdots & & \ddots \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \end{bmatrix}$$

$$\left.\begin{aligned} b_1 &= l_{11} x_1 \\ b_2 &= l_{21} x_1 + l_{22} x_2 \\ &\vdots \end{aligned}\right\} \Rightarrow \left\{\begin{aligned} x_1 &= \frac{b_1}{l_{11}} \\ x_2 &= \frac{1}{l_{22}}\left(b_2 - l_{21} x_1\right) \\ &\vdots \\ x_i &= \frac{1}{l_{ii}}\left(b_i - \sum_{j=1}^{i-1} l_{ij} x_j\right) \end{aligned}\right.$$

known after 1st step

Two implementations possible

(i) Inner product form (Slow) → L accessed row-wise

```
for i = 1:n
    z = 0.0
    for j = 1:i-1
        z = z + L[i,j] * x[j]
    end
    x[i] = (b[i] - z) / L(i,i)
end
```

(ii) Outer Product form (Fast) → Column access

```
x = copy(b)
for j = 1:n
    x[j] = x[j] / L[j,j]
    for i = j+1:n
        x[i] = x[i] - L[i,j] * x[j]
    end
end
```

Example comp:

$$\begin{bmatrix} l_{11} & & & \\ l_{21} & l_{22} & & \\ l_{31} & l_{32} & l_{33} & \\ & & & \ddots \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \end{bmatrix}$$

$j=1 \quad j=2 \quad j=3$

$$\begin{array}{ll} i=1 & l_{11}x_1 = b_1 \\ i=2 & l_{21}x_1 + l_{22}x_2 = b_2 \\ i=3 & l_{31}x_1 + l_{32}x_2 + l_{33}x_3 = b_3 \end{array}$$

$$x_1 = \frac{b_1}{l_{11}}$$

$$x_2 = \left[b_2 - \frac{l_{21} b_1}{l_{11}}\right]\frac{1}{l_{22}}$$

$$x_3 = \left[b_3 - l_{31}\cdot\frac{b_1}{l_{11}} - \frac{l_{32}}{l_{22}}\left[b_2 - \frac{l_{21}b_1}{l_{11}}\right]\right]\frac{1}{l_{33}}$$

$j=1 \longrightarrow x[1] = \dfrac{b[1]}{l_{11}}$

$i=2 \longrightarrow \underbrace{x(2)}_{} = \overset{x[2]}{b[2]} - l_{21}\cdot x[1]$

$i=3 \longrightarrow x(3) = \overset{x[3]}{b(3)} - l_{31}\, x[1]$

$j=2 \longrightarrow x[2] = \dfrac{x[2]}{l_{22}}$

$i=3 \longrightarrow x[3] = x[3] - l_{32}\,x[2]$

$= b[3] - l_{31}\,x[1] - l_{32}\left[\dfrac{b(2) - l_{21}x(1)}{l_{22}}\right]$

$j=3 \qquad x[3] = \dfrac{x[3]}{l_{33}}$

Computational complexity:

[Watkins 1.3]

```
for i = 1:n
    z = 0.0
    for j = 1:i-1
        z = z + L[i,j] * x[j]
    end
    x[i] = (b[i] - z) / L(i,i)
end
```

$\to 2(i-1)$ flops

$\hookrightarrow \boxed{i = 1, \cdots, n}$

$\to 2$

$$2(0 + 1 + \cdots + n-1) + 2n$$

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

$$\approx n^2$$

Solving General Systems $Ax = b$

Def: $A \in \mathbb{R}^{n \times n}$. An LU factorization is $A = LU$ where L is lower tr. & U is upper tr.

$A =$ 

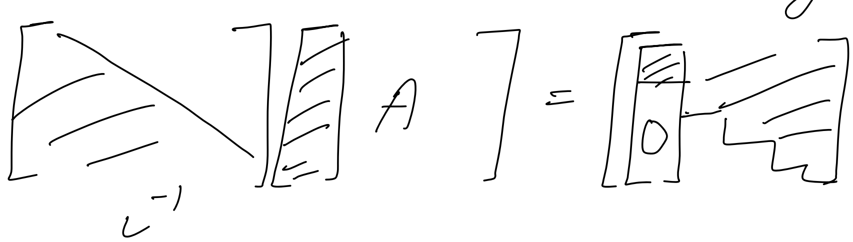
Q. When does it exist? – Postponed
Q. Is it unique? – Also postponed
Q. How does it help solve $Ax = b$?

If it exists: $Ax = b \Rightarrow L\underbrace{Ux}_{z} = b$

1) First solve $Lz = b$. Since L is lower tr. this is easy with previous algo.
2) Second: solve $Ux = z$. Also easy since U is upper tr.

Construction: First assume A is full rank s.t. unique solⁿ exists for $Ax = b$.

If $A = LU$ then $L^{-1}A = U$ and L^{-1} is also lower triangular



$$\underset{\to \mathbb{R}^{n\times n}}{A} = LU = \begin{bmatrix} * & & 0 \\ * & * & \\ * & * & * \end{bmatrix}\begin{bmatrix} * & * & * \\ & * & * \\ 0 & & * \end{bmatrix}$$

$$L^{-1}A = U$$

$$\begin{bmatrix} 1 & 0 & 0 \\ -\frac{a_{21}}{a_{11}} & 1 & \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} a_{11} & \cdot & \cdot \\ a_{21} & \cdot & \cdot \\ a_{31} & \cdot & \cdot \end{bmatrix} = \begin{bmatrix} a_{11} & * & * \\ 0 & * & * \\ a_{31} & * & * \end{bmatrix}$$

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 \\ -\frac{a_{21}}{a_{11}} & 1 & 0 \\ -\frac{a_{31}}{a_{11}} & 0 & 1 \end{bmatrix}}_{G_1}\begin{bmatrix} a_{11} & * & * \\ a_{21} & * & * \\ a_{31} & * & * \end{bmatrix} = \begin{bmatrix} a_{11} & * & * \\ 0 & * & * \\ 0 & * & * \end{bmatrix}$$

$$\underbrace{\left[\begin{array}{c|cc} 1 & 0 & 0 \\ \hline 0 & 1 & 0 \\ 0 & -\frac{a_{32}^*}{a_{22}^*} & 1 \end{array}\right]}_{G_2}\left[\begin{array}{c|cc} a_{11} & a_{12} & a_{13} \\ \hline 0 & a_{22}^* & a_{23}^* \\ 0 & a_{32}^* & a_{33}^* \end{array}\right] = \underbrace{\left[\begin{array}{c|cc} a_{11} & a_{12} & a_{13} \\ \hline 0 & a_{22}^* & a_{23}^* \\ 0 & 0 & a_{33}^{**} \end{array}\right]}_{U}$$

So $L^{-1} = G_2 G_1$; $[G_1]_{ii} = 1$; $[G_1]_{i1} = -\frac{a_{i1}}{a_{11}}$

Other entries of G_1 are zero

In general: $G_{n-1} G_{n-2} \cdots G_1 A = U$

$$A = \underbrace{G_1^{-1} G_2^{-1} \cdots G_{n-1}^{-1}}_{L \,\to\, \text{lower tr since each } G_i^{-1} \text{ is L.T.}} U = LU$$

$$G_i = \begin{array}{c} \begin{array}{ccccccc} & & & \text{i^{th} column} & & & \\ & & & \downarrow & & & \end{array} \\ \begin{bmatrix} 1 & & & & & & \\ 0 & \ddots & & & & \text{\huge 0} & \\ 0 & \cdots & 0 & 1 & & & \\ 0 & \cdots & \cdots & 0 & 1 & & \\ 0 & & & 0 & -(g_i)_{i+1} & 1 & \\ \vdots & & & \vdots & -(g_i)_{i+2} & 0 & \ddots \\ 0 & & & 0 & -(g_i)_n & 0 \;\cdots\; 0 & 1 \end{bmatrix} \end{array}$$

$(g_i)_j = \dfrac{a_{ji}^{(i)}}{a_{ii}^{(i)}}$ where $a_{ij}^{(k)}$ is the (i,j)-th entry in $[G_{k-1} \cdots G_1 A]$

G_i's are called Gauss transforms

$G_i = I + g_i e_i^T = I + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ -(g_i)_{i+1} \\ \vdots \\ -(g_i)_n \end{bmatrix} [0 \cdots 1 \; 0 \cdots 0]$

i^{th} position

<u>FACT</u>: If G_i, G_j are two Gauss transforms $(i<j)$

$G_i G_j = I + g_j e_j^T + g_i e_i^T$

Proof: Extra term $= g_i \underbrace{e_i^T g_j}\, e_j^T = 0$

$[\text{since } e_i^T g_j = (g_j)_i = 0 \text{ for } i \le j]$

FACT: $[G_i]^{-1} = I - g_i e_i^T$

Proof: $[I + g_i e_i^T][I - g_i e_i^T] = I$

Exercise: What is $[G_i]^{-1}[G_j]^{-1}$, for $i<j$?

$$\left[\; G_i^{-1} G_j^{-1} = I - g_i e_i^T - g_j e_j^T + \underbrace{g_i e_i^T g_j e_j^T}_{=0} \;\right]$$

Hence: $L = G_1^{-1} G_2^{-1} \cdots G_{n-1}^{-1}$

(**) $$= \begin{bmatrix} 1 & & & & \\ +(g_1)_2 & 1 & & & 0 \\ +(g_1)_3 & +(g_2)_3 & 1 & & \\ \vdots & \vdots & & \ddots & \\ \vdots & \vdots & & +(g_{n-1})_n & 1 \end{bmatrix}$$

Note: # 1's on diagonal → "Unit" lower triangular

Q. Construction seems to prove existence of LU fact? Is this correct or assumptions are req?

Implementation Strategy #1 (Outer product)

Let $A^{(k)} = G_{k-1} \cdots G_1 A$. Then $A^{(k+1)} = G_k A^{(k)}$

Then, by above calculations; for $i > k$

(*) $A^{(k+1)}[i,:] = A^{(k)}[i,:] - (g_k)_i A^{(k)}[k,:]$

for all $\begin{bmatrix} k+1 \\ \vdots \\ n \end{bmatrix}$ Recall: $\dfrac{a_{ik}^{(k)}}{a_{kk}^{(k)}}$

If we iterate through $k = 1, \cdots, n$ then
$A^{(n)} = U$ i.e. upper triangular

To save space, we can store L (see (**)) in the "strictly" lower triangular part of the matrix A. [Recall $[L]_{ii} = 1 \ \forall i$]

For all i, the (*) step looks like:

$$A^{(k)} - \begin{bmatrix} 0 \\ \vdots \\ 0 \\ (g_k)_{k+1} \\ \vdots \\ (g_k)_n \end{bmatrix} \begin{bmatrix} 0 \cdots 0 & a_{kk}^{(k)} & a_{k,k+1}^{(k)} & \cdots & a_{k,n}^{(k)} \end{bmatrix}$$

$$= \begin{bmatrix} a_{11} & \cdots & \cdots & \cdots \\ 0 \cdots & a_{k-1,k-1} & \cdots & \cdots \\ 0 \cdots 0 & a_{kk}^{(k)} & \cdots & \cdots \\ 0 & a_{k+1,k}^{(k)} & \cdots & \cdots \\ \vdots & \vdots & & \\ 0 \quad 0 & a_{nk}^{(k)} & \cdots & \cdots \end{bmatrix} - \begin{array}{c} \begin{bmatrix} 0 & \cdots & & & \cdots & 0 \\ 0 & \cdots & & & \cdots & 0 \\ 0 \cdots 0 & (g_k)_{k+1} a_{kk}^{(k)} & (g_k)_{k+1} a_{k,k+1}^{(k)} & \cdots & \cdots & \times \\ & \vdots & & & & \\ 0 \cdots 0 & (g_k)_n a_{kk}^{(k)} & (g_k)_n a_{k,k+1}^{(k)} & \cdots & \cdots & \times \end{bmatrix} \begin{array}{l} 1 \\ k \\ \leftarrow (k+1) \text{ row} \\ \\ \leftarrow n \end{array} \end{array}$$

k^{th} col will be zeroed out

$\uparrow k^{th}$ col

(*)

$$\boxed{(g_k)_{k+1} = \frac{a_{k+1,k}^{(k)}}{a_{kk}^{(k)}}}$$

$[\text{Result}]_{k+1,k} = 0$
$\vdots$
$[\text{Result}]_{n,k} = 0$

No point storing zeros. Instead we could store
$\begin{bmatrix} (g_k)_{k+1} \\ \vdots \\ (g_k)_n \end{bmatrix}$ in $\begin{bmatrix} \text{Result}_{k+1,k} \\ \\ \text{Result}_{n,k} \end{bmatrix}$.

Code:

```
for k = 1:n
    for i = k+1:n
        A[i,k] = A[i,k] / A[k,k]
    end
    for i = k+1:n
        for j = k+1:n
            A[i,j] = A[i,j] - A[i,k]*A[k,j]
        end
    end
end
```

Stores g_k in the LT part of A

Implements (*) without the kth column operation

Complexity

$(n-k)$

$2(n-k)\times(n-k)$

$$\sum_{k=1}^{n} 2(n-k)^2 + (n-k)$$

$$\approx \sum_{p=1}^{n} 2p^2 + p$$

$$= 2\left[\frac{1}{3}n^3 + \frac{1}{2}n^2 + \frac{1}{6}n\right] + \left[\frac{n(n+1)}{2}\right]$$

$$\approx \frac{2}{3}n^3$$ [Watkins 1.7]

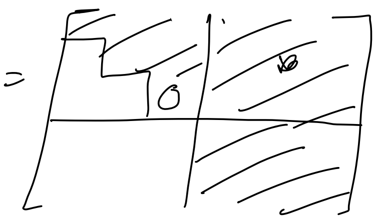
Faster (column access) code: (jki – order)

```
for j = 1:n
    for k = 1:j-1
        for i = k+1:n
            A[i,j] = A[i,j] - A[i,k]*A[k,j]
        end
    end
    for i = j+1:n
        A[i,j] = A[i,j] / A[j,j]
    end
end
```

☐ Show intermediate steps in code

Q. Does above algo. work always?

\# Notice the division by $A[k,k]$ at each iteration

Suppose $A^{(k)} = G_{k-1} \cdots G_1 A =$ 

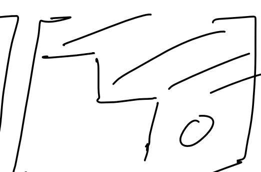
$\boxed{\square}$ Show example $A^{(k)}$ with $A = \begin{bmatrix} 1 & 6 & 1 & 0 \\ 0 & 1 & 9 & 3 \\ 1 & 6 & 1 & 1 \\ 0 & 0 & 1 & 9 \end{bmatrix}$

→ show the k^{th} step with $\left[A^{(k)}\right]_{kk} = 0$

clearly at this step:

$$\underbrace{\begin{bmatrix} 1 & & 0 \\ & 1 & \\ & & 1 \end{bmatrix}}_{G_{k-1} \cdots G_1} \begin{bmatrix} A[1:k, 1:k] \end{bmatrix} =$$ 

or

$$\begin{bmatrix} 1 & & \\ & 1 & \\ & & 1 \end{bmatrix}$$  $= A[1:k, 1:k]$

$\underbrace{\qquad\qquad}_{\det = 0} \Rightarrow \det\{A[1:k, 1:k]\} = 0$

\# We cannot allow this at any stage.

<u>Thm:</u> If $\det(A[1:k, 1:k]) \neq 0$ for all $1 \leq k \leq n-1$ then the LU factorization exists. and is unique

<u>Proof:</u> Exercise

Equivalent Statement: If the leading principal submatrices are non-singular, LU decomp. exists & is unique.

Uniqueness:

$$\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ a_{21} & \cdots & a_{2n} \\ \vdots & & \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} 1 & & 0 \\ l_{21} & \ddots & \\ \vdots & & \\ l_{n1} & & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ & \ddots & & \\ 0 & & \ddots & u_{nn} \end{bmatrix}$$

Clearly $a_{1j} = u_{1j} \Rightarrow [u_{11} \cdots u_{1n}]$ is uniquely determined.

Also $a_{i1} = l_{i1} u_{11}$ $i = 2, \dots, n$. ($u_{11} \neq 0$ since A is non-singular)

$\Rightarrow l_{i1} = \dfrac{a_{i1}}{u_{11}}$ $i = 2, \dots, n$.

$\Rightarrow$ 1st column of L is uniquely determined.

Remaining by induction: Exercise.

Q. But what if the pivot $A[k,k]$ is very small (not zero)?

$$A = \begin{bmatrix} \varepsilon & 1 \\ 1 & \pi \end{bmatrix} \xrightarrow[\text{algo}]{\text{Above}} L = \begin{bmatrix} 1 & 0 \\ \frac{1}{\varepsilon} & 1 \end{bmatrix} \quad U = \begin{bmatrix} \varepsilon & 1 \\ 0 & \pi - \frac{1}{\varepsilon} \end{bmatrix}$$

For $\varepsilon = 10^{-14}$, $L * U = \begin{bmatrix} 10^{-14} & 1 \\ 1 & \underbrace{3.140625}_{\text{too much error}} \end{bmatrix}$

$\boxed{\square}$ Show code

$\pi \approx 3.14159$

Q. Why is this happening?

A. Postponed.

LU with pivoting

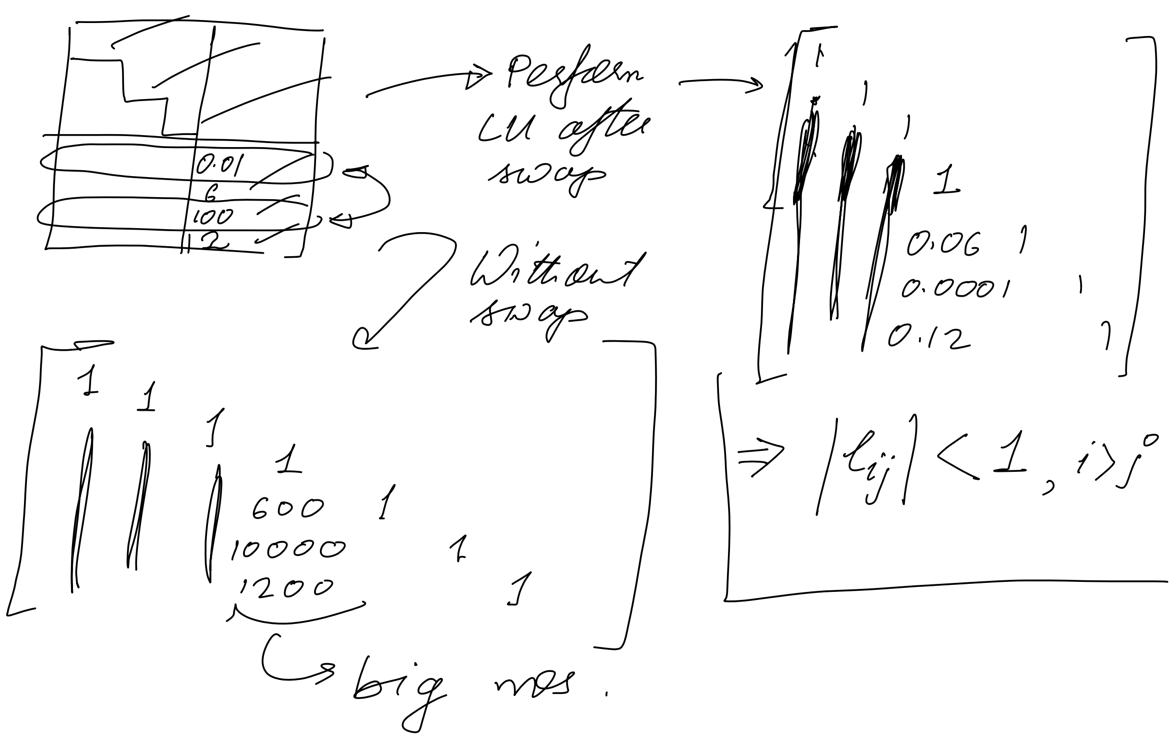
Instead of solving $\overbrace{\begin{bmatrix} \varepsilon & 1 \\ 1 & \pi \end{bmatrix}}^{A} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$

we can equivalently solve $\underbrace{\begin{bmatrix} 1 & \pi \\ \varepsilon & 1 \end{bmatrix}}_{\bar{A}} \begin{bmatrix} x_2 \\ x_1 \end{bmatrix} = \begin{bmatrix} b_2 \\ b_1 \end{bmatrix}$

The LU decomp. of $\bar{A}$ is perfectly accurate!

- [ ] Show code

At each step we first find the highest $A[k:n, k]$ & swap rows s.t. this entry appears in the (k,k) position



$\Rightarrow |l_{ij}| < 1, \ i > j$

Instead of $G_{n-1} \cdots G_1 A = U$

we get $G_{n-1} P_{n-1} \cdots G_2 P_2 G_1 P_1 A = U$

where $P_1, \ldots P_{n-1}$ are permutation matrices

$$P_1 = \begin{bmatrix} 1 & & & & \\ & & & 1 & \\ & & 1 & & \\ & 1 & & & \\ & & & & 1 \end{bmatrix} \Bigg\} \text{swaps rows } 2 \,\&\, 4$$

$\underbrace{\text{Permutation Matrix Properties}}$: # $P_1 P_2$ is a permutation

P is orthogonal $P^{-1} = P^T$;

PA permutes rows while AP permutes columns

For elementary permutation $P = P^T$

$$G_{n-1} P_{n-1} G_{n-2} P_{n-2} \cdots G_2 P_2 G_1 P_1 A$$

$$= G_{n-1} \left[P_{n-1} G_{n-2} P_{n-1}^{-1} \right] \underbrace{P_{n-1} P_{n-2}}_{\text{Permutation}} G_{n-2} \cdots G_2 P_2 G_1 P_1 A$$

Claim: PGP^{-1} is a Gauss transform

$$\overset{P}{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}} \overset{G}{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & g_1 & 1 & 0 \\ 0 & g_2 & 0 & 1 \end{bmatrix}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & g_2 & 0 & 1 \\ 0 & g_1 & 1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & g_2 & 0 & 1 \\ 0 & g_1 & 1 & 0 \end{bmatrix} \underset{P^T}{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & g_2 & 1 & 0 \\ 0 & g_1 & 0 & 1 \end{bmatrix}}_{\hat{G}}$$

another $\leftarrow \hat{G}$
Gauss transform

Hence $G_{n-1}P_{n-1}G_{n-2}P_{n-2}\cdots G_2P_2G_1P_1A$

$$= \underbrace{\hat{G}_{n-1}\hat{G}_{n-2}\cdots\hat{G}_1}_{L^{-1}}\;\underbrace{P_{n-1}P_{n-2}\cdots P_1}_{\hat{P}}\,A$$

$$\Rightarrow L^{-1}\hat{P}A = U \quad\Leftrightarrow\quad \hat{P}A = LU$$

Q. Is $\hat{G}_k$ a Gauss transform? (in general)

[Golub 3.4]

$$\hat{G}_k = [P_{n-1}P_{n-2}\cdots P_{k+1}]\,G_k\,[P_{k+1}\cdots P_{n-1}]$$

$$= \quad {''} \quad [I + g_k e_k^T]\,[\quad {''} \quad]$$

$$= [P_{n-1}P_{n-2}\cdots P_{k+2}]\,[I + P_{k+1} g_k e_k^T P_{k+1}]\,[P_{k+2}\cdots P_{n-1}]$$

$$= [P_{n-1}P_{n-2}\cdots P_{k+2}]\,[I + \tilde{g}_k e_k^T]\,[P_{k+2}\cdots P_{n-1}]$$

$$P_{k+1}\begin{bmatrix}0\\ \vdots\\ 0\\ g_{k+1}\\ \vdots\\ \vdots\\ g_n\end{bmatrix}[0\cdots 1\; 0\cdots]\,P_{k+1} = \begin{bmatrix}0\\ \vdots\\ 0\\ \tilde{g}_{k+1}\\ \vdots\\ \tilde{g}_n\end{bmatrix}[0\cdots \underset{\substack{\uparrow\\ k^{th}}}{1}\; 0\cdots 0] = \tilde{g}_k e_k^T$$

$\therefore$ after all the multiplications

$$= [I + \hat{g}_k e_k^T]$$

Advantages / Observations

1) $|l_{ij}| < 1 \quad i > j$

So the problem with l_{ij} growing is solved

$$\begin{bmatrix} \varepsilon & 1 \\ 1 & \pi \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 \\ \frac{1}{\varepsilon} & 1 \end{bmatrix}}_{\text{will not happen}} \begin{bmatrix} \varepsilon & 1 \\ 0 & \pi - \frac{1}{\varepsilon} \end{bmatrix}$$

2) Algo (partial pivoting) always runs to completion.

If $A^{(k)}[k:n, k] = 0$, then $\hat{G}_k = I_n$

Thm: LU with partial pivoting, applied to any $n \times n$ matrix A produces a unit lower triangular matrix L with $|l_{ij}| < 1$, an upper triangular U, and a permutation matrix P s.t.

$$A = P^T L U$$

```
for k = 1:n
    imx = k-1 + argmax(abs.(A[k:end, k]))
    for j = 1:n
        A[k,j], A[imx,j] = A[imx,j], A[k,j]
    end
    P[[k,imx]] = P[[imx,k]]
    ⋮
    usual LU
    ⋮
end
```

Complexity

Remains same as $\frac{2}{3}n^3$ because the complexity of $(n-k)$ comparisons at each step is $\sum_{k=1}^{n-1} (n-k) = \frac{n(n-1)}{2} \approx \frac{n^2}{2}$

$\boxed{\text{D}}$ Show code + example

Q. What can go wrong? [Darve 3.4]

Problem 1 / Example:

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ -1 & 1 & 0 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & -1 & -1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ -1 & -1 & 1 & 0 \\ -1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 4 \\ 0 & 0 & 0 & 8 \end{bmatrix}$$

$n \times n \quad (n = 4)$ $\qquad 2^{n-1}$

Elements of U can still grow! → $\boxed{\text{Very Rare}}$

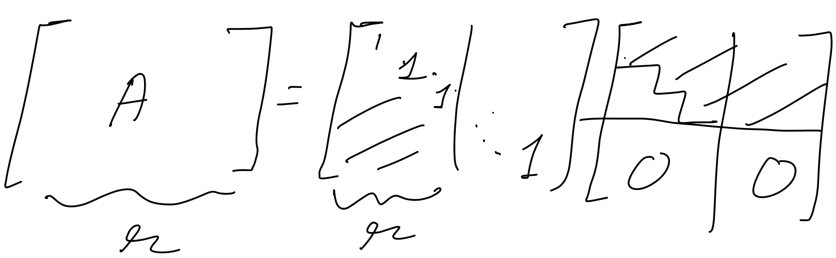
Problem 2

2) Cannot be used to reveal rank!

If $A \in \mathbb{R}^{n\times n}$ is of rank $r < n$, then a factorization of the form $A = WV^T$ where W & V have r-columns is "rank-revealing".

LU can "almost" be used for such a factorization

FACT: If $\det(A[1:r, 1:r]) \neq 0$, then

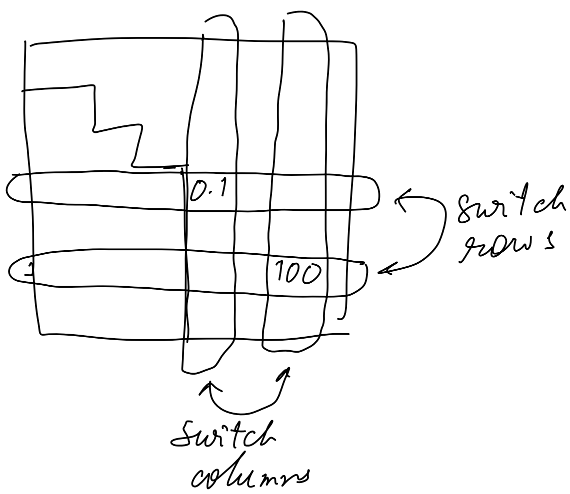


$=$ 

This will not work if the above condition does not hold.

LU with Complete (row + column) pivoting

[solves both P1 & P2 above)



Swap rows & columns at each step to get the largest entry in $A[k:n, k:n]$ to the $(k,k)^{th}$ position

The steps result in:

$$G_{n-1}P_{n-1}\cdots G_1 P_1 A Q_1^T Q_2^T \cdots Q_{n-1}^T = U$$

Following same arguments as above

$$\hat{G}_{n-1}\cdots \hat{G}_1 P_{n-1}\cdots P_1 A = Q_1^T \cdots Q_{n-1}^T = U$$

or $PAQ^T = L\cdot U$ $[L^{-1} = \hat{G}_{n-1}\cdots \hat{G}_1]$

If we were trying to solve $Ax = b$ & $PAQ^T = LU$ then

i) solve $Lz = Pb$ for z

2) Solve $Uy = z$ for y
3) Set $x = Q^T y$

$A = P^T L U Q$
$P^T L U Q x = b$
$L U y = Pb =: \tilde{b}$
$L U y = \tilde{b}$

\# This method is rank-revealing (ideally) or rank-approximating (in practice)

$$PAQ^T = \begin{bmatrix} L_{11} & 0 \\ l_{21} & I_{n-1} \end{bmatrix} \begin{bmatrix} U_{11} & U_{12} \\ 0 & 0 \end{bmatrix} \leftarrow \text{ideally}$$

$$= \begin{bmatrix} L_{11} \\ l_{21} \end{bmatrix} \begin{bmatrix} U_{11} & U_{12} \end{bmatrix}$$

In practice elements might be small.

\# Elements do not grow (How do we prove?)
→ But complete pivoting is slow (How to quantify?)

Complexity: Cost of comparision: $\sum_{k=1}^{n-1} (n-k)^2$
(Since at each step $(n-k)^2$ nos need to be compared)

$$\sum_{k=1}^{n-1} (n-k)^2 \approx \frac{1}{3} n^3$$

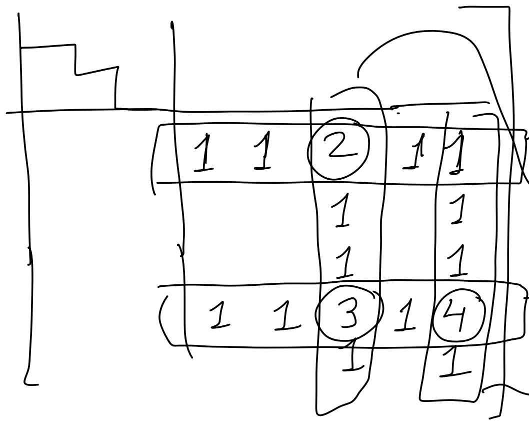
So total cost $\approx \underbrace{\frac{2}{3} n^3}_{\text{from usual } LU} + \frac{1}{3} n^3 \approx \underline{n^3}$

significantly more than partial pivoting.

[Watkins 1.8]

Rook Pivoting –

[Darve 3.4]



Step 1: Search 1^{st} row $\Rightarrow \boxed{2}$

Step 2: Search 3^{rd} col $\Rightarrow \boxed{3}$

Step 3: Search 4^{th} row $\Rightarrow \boxed{4}$

Step 4: Search 5^{th} col $\Rightarrow \boxed{4}$

stop

Summery: Find any element in $A^{(k)}[k:n, k:n]$ which is maximal in both its row & col.

→ Then use row & col swap to get that element at $a^{(k)}[k,k]$.

[D] Show code

Q. Calculate the complexity (worst case?)

Open Issues: We need a language to describe
1> Error in computation
2> Growth of elements } Next chapter

Remaining Material in Gaussian Elimination

LU Fact. of SPD matrices

[Golub 4.1 & 4.2]

FACT: If $A \in \mathbb{R}^{n\times n}$ is symmetric with non-singular leading principal minors, then $\exists$ a unique unit lower tr. L & a unique diag. D s.t. $A = LDL^T$

Proof: We know $A = LU$ (unique L & U)

Since $\underbrace{L^{-1}AL^{-T}}_{\text{Symmetric}} = \underbrace{UL^{-T}}_{\text{Upper tr.}}$, then both sides must be diagonal

Let $D = UL^{-T}$, then $A = LDL^T$.

FACT: If A is SPD, then all principal submatrices are positive definite. In particular, all diagonal entries are +ve.

Proof: Exercise

Thm (Cholesky Fact): If $A \in \mathbb{R}^{n\times n}$ is SPD, then $\exists$ a unique L.T. $G \in \mathbb{R}^{n\times n}$ with positive diagonal entries s.t. $A = GG^T$

Proof: From above facts, $\exists$ unit L.T. L

and diagonal D, s.t. $A = LDL^T$

Clearly, $D = L^{-1} A L^{-T}$ is SPD. (since A is SPD & L^{-1} is full rank)

$\Rightarrow D =: \text{diag}(d_1, \cdots, d_n)$, d_i's are +ve.

& $G := L \, \text{diag}(\sqrt{d_1}, \cdots, \sqrt{d_n})$ is real & lower triangular with +ve diagonals.

Then $A = GG^T$ (uniqueness follows from LDL^T)

Computing Cholesky: a modification of LU

$\underline{A = GG^T}$ Normally if $A = GH$, $A[i,j] = \sum_{k=1}^{n} G[i,k] H[k,j]$

Here $H = G^T \Rightarrow H[k,j] = G[j,k]$

$$\Rightarrow A[i,j] = \sum_{k=1}^{n} G[i,k] . G[j,k]$$

$$\Rightarrow A[:,j] = \sum_{k=1}^{n} G[:,k] . G[j,k]$$

$$\Rightarrow G[j,j] G[:,j] = \underbrace{A[:,j] - \sum_{k=1}^{j-1} G[j,k] . G[:,k]}_{=: v} \quad (\circledast)$$

Then $v[j] = G[j,j] G[j,j]$

But G is lower triangular. Hence from $\circledast$

$$G[j:n, j] = \frac{v[j:n]}{G[j,j]} = \frac{v[j:n]}{\sqrt{v[j]}}$$

Hence the following algo computes G:

```
for j = 1:n
    V[j:n] = A[j:n, j]
    for k = 1:j-1
        V[j:n] = V[j:n] - G[j,k] . G[j:n, k]
    end
    G[j:n, j] = V[j:n] / sqrt(V[j])
end
```

The above algo can be re-arranged so that G overwrites the lower-tr. part of A

```
for j = 1:n
    for k = 1:j-1
        for i = j:n
            A[i,j] = A[i,j] - A[i,k] * A[j,k]
        end
    end
    ajj = sqrt(A[j,j])
    for i = j:n
        A[i,j] = A[i,j] / ajj
    end
end
```

$\rightarrow \dfrac{n^3}{3}$ flops

$$2\sum_{j=1}^{n}(n-j)(j-1)$$

$$\approx 2\left[n\sum j - \sum j^2\right]$$

$$\approx 2\left[\frac{n^3}{2} - \frac{n^3}{3}\right] \approx \frac{1}{3}n^3$$

Q. How to compute L, D ? [Darve 3.5]

Let A be SPD & we blindly use LU without any pivoting.

Let $A = \underbrace{\begin{bmatrix} \overset{1\times 1}{a} & c^T \\ c & B \end{bmatrix}}_{n\times n} = \begin{bmatrix} 1 & 0 \\ c/a & I \end{bmatrix} \begin{bmatrix} a & c^T \\ 0 & B - \frac{1}{a}cc^T \end{bmatrix}$

Note: $B - \frac{1}{a}cc^T$ is still symmetric $\Rightarrow$ we can compute/store only half the entries

Q. Is $\left(B - \frac{1}{a}cc^T\right)$ SPD ? $\longrightarrow$ Yes

---

Schur Complement: Let $A = \begin{bmatrix} \overset{k\times k}{A_{11}} & A_{12} \\ A_{21} & \underbrace{A_{22}}_{(n-k)\times(n-k)} \end{bmatrix}$ be SPD.

Then clearly,

$$A = \begin{bmatrix} I & 0 \\ A_{21}A_{11}^{-1} & I \end{bmatrix} \begin{bmatrix} A_{11} & 0 \\ 0 & \underbrace{A_{22} - A_{21}A_{11}^{-1}A_{12}}_{\text{Schur Complement of } A_{11}} \end{bmatrix} \begin{bmatrix} I & A_{11}^{-1}A_{12} \\ 0 & I \end{bmatrix}$$

Clearly, $\begin{bmatrix} a & c^T \\ c & B \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \frac{c}{a} & I \end{bmatrix} \begin{bmatrix} a & 0 \\ 0 & B - \frac{1}{a}cc^T \end{bmatrix} \begin{bmatrix} 1 & c^T/a \\ 0 & I \end{bmatrix}$ — ($\ast$)

Since $A = A^T > 0$, $\begin{bmatrix} a & 0 \\ 0 & B - \frac{1}{a}cc^T \end{bmatrix} > 0 \Rightarrow B - \frac{1}{a}cc^T > 0$

Hence we can perform LU steps (as in ($\ast$)) recursively. ending in $A = LDL^T$

$\check{=}$ $\check{}$

\# Then define $G = LD^{1/2} \Rightarrow A = GG^T$

\# This idea can also be used for pivoting.

<u>LDL^T with symmetric Pivoting</u> $A = A^T > 0$

\# Find P_1 s.t. $P_1 A P_1^T = \begin{bmatrix} a & c^T \\ c & B \end{bmatrix}$ and

$a = \max\{\operatorname{diag}(A)\}$ —— (*)

But we have seen $\begin{bmatrix} a & c^T \\ c & B \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \frac{c}{a} & I_{n-1} \end{bmatrix} \begin{bmatrix} a & 0 \\ 0 & A_1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \frac{c}{a} & I_{n-1} \end{bmatrix}^T$

\# Use this strategy recursively to A_1 & compute

$P_2 A_1 P_2^T = L_2 D_2 L_2^T$

Then

$$\underbrace{\begin{bmatrix} 1 & 0 \\ 0 & P_2 \end{bmatrix} \begin{bmatrix} & & \\ & P_1 & \\ & & \end{bmatrix}}_{P} A P^T = \begin{bmatrix} 1 & 0 \\ c/a & L_2 \end{bmatrix} \underbrace{\begin{bmatrix} a & 0 \\ 0 & D_2 \end{bmatrix}} L^T$$

Because of (*), $d_1 \geq d_2 \geq \cdots \geq d_n > 0$

Q. How does complexity compare with previous method.

$\boxed{\text{m}}$ show Cholesky code.

Q. How does this extend to PD but not symmetric matrices?

$$\left[\begin{array}{c|cc} a_{11} & a_{12} & a_{13} \\ \hline a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{array}\right]$$

Step 1

$j=1,\ k=0,\ i=1:3$

$A[1,1]=A[1,1]$
$A[2,1]=A[2,1]$
$A[3,1]=A[3,1]$

$a11=\sqrt{a_{11}}$

$A[1,1]=\dfrac{A[1,1]}{\sqrt{a_{11}}}$

$A[2,1]=\dfrac{A[2,1]}{\sqrt{a_{11}}}$

$A[3,1]=\dfrac{A[3,1]}{\sqrt{a_{11}}}$

$$\begin{bmatrix} \dfrac{a_{11}}{\sqrt{a_{11}}} & & \\ \dfrac{a_{21}}{\sqrt{a_{11}}} & & \\ \dfrac{a_{31}}{\sqrt{a_{11}}} & & \end{bmatrix}$$

Step 2: $j=2,\ k=1,\ i=2:3$

$i=j:n$
$(k=1:j-1)$

$$\left[\begin{array}{c|c} & \text{Don't care} \\ \hline & a_{22}-\frac{1}{\sqrt{a_{11}}}a_{21}\times\frac{1}{\sqrt{a_{11}}}a_{21} \\ & a_{32}-\frac{1}{\sqrt{a_{11}}}a_{31}\times\frac{1}{\sqrt{a_{11}}}a_{21} \end{array}\right]$$

$\underset{i\ \ j}{A[2,2]}=A[2,2]-\underset{i,k}{A[2,1]}*\underset{j,k}{A[2,1]}$

$A[3,2]=A[3,2]-A[3,1]*A[2,1]$

$a22=\sqrt{A[2,2]}$

$A[2,2]=\dfrac{A[2,2]}{\sqrt{a_{22}}}$

$A[3,2]=\dfrac{A[3,2]}{\sqrt{a_{22}}}$

Step 3: $j=3,\ k=1,2,\ i=3$

$A[3,3]=A[3,3]-\underset{i\ \ k}{A[3,1]}\times\underset{j\ \ k}{A[3,1]}$

$A[3,3]=\underset{i\ \ j}{A[3,3]}-A[3,2]*A[3,2]$

$a_{33}=a_{33}-\frac{1}{a}a_{31}\times a_{31}-\frac{1}{a}a_{32}\times a_{32}$

$$B - \frac{1}{a}CC^T = \begin{bmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{bmatrix} - \frac{1}{a}\begin{bmatrix} a_{21} \\ a_{31} \end{bmatrix}\begin{bmatrix} a_{21} & a_{31} \end{bmatrix}$$

$$= \begin{bmatrix} a_{22} - \frac{1}{a} a_{21} * a_{21} & a_{32} - \frac{1}{a} a_{21} * a_{31} \\ a_{32} - \frac{1}{a} a_{31} * a_{21} & a_{33} - \frac{1}{a} a_{31} * a_{31} \end{bmatrix}$$

$$\begin{bmatrix} 4 & -2 & 4 \\ -2 & 10 & -2 \\ 4 & -2 & 8 \end{bmatrix} \rightsquigarrow \begin{bmatrix} \frac{4}{\sqrt{4}} & \text{same} \\ \frac{-2}{\sqrt{4}} & \\ \frac{4}{\sqrt{4}} & \end{bmatrix} = \begin{bmatrix} \overset{\text{new}}{\begin{matrix} 2 \\ -1 \\ 2 \end{matrix}} & \text{same} \end{bmatrix}$$

$$\begin{bmatrix} 2 & \boxed{-2 \quad 4} \\ -1 & 10-(-1)(-1) & -2 \\ 2 & -2-(-1)(2) & 8 \end{bmatrix} \rightsquigarrow \begin{bmatrix} 2 & \boxed{\text{same}} \\ -1 & \frac{9}{\sqrt{9}} & \boxed{\text{same}} \\ 2 & \frac{0}{\sqrt{9}} & \end{bmatrix} = \begin{bmatrix} 2 & -2 & 4 \\ -1 & 3 & -2 \\ 2 & 0 & 8 \end{bmatrix}$$

new

$$\begin{bmatrix} 2 & \overset{\text{same}}{-2} & 4 \\ -1 & 3 & -2 \\ 2 & 0 & 8-(2\times 2) \\ & & -(0)\times(0) \end{bmatrix} \rightsquigarrow \begin{bmatrix} 2 & & \\ -1 & 3 & \\ 2 & 0 & \frac{4}{\sqrt{4}} \end{bmatrix} = \begin{bmatrix} 2 & & \\ -1 & 3 & \\ 2 & 0 & 2 \end{bmatrix}$$

$$8 - 2\times 2 - \frac{1}{3}\times\frac{1}{3} = 8 - 4 - \frac{1}{9} = 4 - \frac{1}{9}$$

$$G = \begin{bmatrix} 2 & & 0 \\ -1 & 3 & \\ 2 & 0 & 2 \end{bmatrix}$$

$\sqrt{a_{ii}}$ is stored here

$A = G G^T$

$= L U = L D L^T$

In original L:
$\frac{a_{ij}}{a_{ii}}$ is stored.

Here $\frac{a_{ij}}{\sqrt{a_{ii}}}$ is stored.

$= \begin{bmatrix} & & \\ & & \\ & & \end{bmatrix}$

So to get original L, we need to divide further by $\sqrt{a_{ii}}$.

$$L = \begin{bmatrix} \cancel{2}/2 & & 0 \\ -1/2 & \cancel{3}/3 & \\ 2/2 & 0/3 & 2/2 \end{bmatrix} \qquad D = \begin{bmatrix} 2 & & 0 \\ & 3 & \\ 0 & & 2 \end{bmatrix}$$