Gaussian Elimination & LU Decompositions

Square System of Equations

Consider

$$Ax = b$$

where

$$A = [a_{ij}] \longleftarrow n \times n \text{ matrix}$$

$$b \longleftarrow n \times 1 \text{ vector.}$$

In expanded form,

$$\begin{aligned}
a_{11}x_1 + \cdots + a_{1n}x_n &= b_1 \\
a_{21}x_1 + \cdots + a_{2n}x_n &= b_2 \\
&\vdots \\
a_{n1}x_1 + \cdots + a_{nn}x_n &= b_n
\end{aligned}$$

Gaussian Elimination

Carl Friedrich Gauss
(1777-1855)

A Summary of the Evolution of Gaussian Elimination

Gaussian Elimination With No Pivoting (GENP)

$$A \longrightarrow A^{(1)} \longrightarrow \cdots \longrightarrow A^{(n-1)} =: U \text{ (upper triangular).}$$
$$b \longrightarrow b^{(1)} \longrightarrow \cdots \longrightarrow b^{(n-1)}.$$

Gaussian Elimination With No Pivoting (GENP)

$$A \longrightarrow A^{(1)} \longrightarrow \cdots \longrightarrow A^{(n-1)} =: U \text{ (upper triangular)}.$$
$$b \longrightarrow b^{(1)} \longrightarrow \cdots \longrightarrow b^{(n-1)}.$$

Step 1: Create zeros in the first column of A:

$$A \longrightarrow \underbrace{\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ 0 & a_{22}^{(1)} & \cdots & a_{2n}^{(1)} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n2}^{(1)} & \cdots & a_{nn}^{(1)} \end{bmatrix}}_{=:A^{(1)}}; \quad b \longrightarrow \underbrace{\begin{bmatrix} b_1 \\ b_2^{(1)} \\ \vdots \\ b_n^{(1)} \end{bmatrix}}_{=:b^{(1)}}$$

where

$$a_{ij}^{(1)} = a_{ij} - \underbrace{\frac{a_{i1}}{a_{11}}}_{=:m_{i1}} a_{1j}; \quad b_i^{(1)} = b_i - \frac{a_{i1}}{a_{11}} b_1; \quad i = 2:n, j = 2:n.$$

Here $a_{11} \longleftarrow$ pivot (assumed non zero); $m_{i1} \longleftarrow$ multipliers;

GENP

Step k: Create zeros in column k of $A^{(k-1)}$:

$$
A^{(k-1)} = \left[\begin{array}{cccc|ccc}
a_{11} & \cdots & \cdots & a_{1k} & a_{1,k+1} & \cdots & a_{1k} \\
 & a_{22}^{(1)} & \cdots & a_{2k}^{(1)} & a_{2,k+1}^{(1)} & \cdots & a_{2n}^{(1)} \\
 & & \ddots & \vdots & \vdots & \cdots & \vdots \\
 & & 0 \approx & a_{kk}^{(k-1)} & a_{k,k+1}^{(k-1)} & \cdots & a_{kn}^{(k-1)} \\
\hline
 & & & a_{k+1,k}^{(k-1)} & a_{k+1,k+1}^{(k-1)} & \cdots & a_{k+1,n}^{(k-1)} \\
 & & & \vdots & \vdots & & \vdots \\
 & & & a_{n,k}^{(k-1)} & a_{n,k+1}^{(k-1)} & \cdots & a_{nn}^{(k-1)}
\end{array}\right]
$$

$$
\longrightarrow \left[\begin{array}{cccc|ccc}
a_{11} & \cdots & \cdots & a_{1k} & a_{1,k+1} & \cdots & a_{1k} \\
 & a_{22}^{(1)} & \cdots & a_{2k}^{(1)} & a_{2,k+1}^{(1)} & \cdots & a_{2n}^{(1)} \\
 & & \ddots & \vdots & \vdots & \cdots & \vdots \\
 & & & a_{kk}^{(k-1)} & a_{k,k+1}^{(k-1)} & \cdots & a_{kn}^{(k-1)} \\
\hline
 & & & & a_{k+1,k+1}^{(k)} & \cdots & a_{k+1,n}^{(k)} \\
 & & & & \vdots & & \vdots \\
 & & & & a_{n,k+1}^{(k)} & \cdots & a_{nn}^{(k)}
\end{array}\right] =: A^{(k)};
$$

GENP

The same operations are performed on $b^{(k-1)}$:

$$b^{(k-1)} \longrightarrow \begin{bmatrix} b_1 \\ b_2^{(1)} \\ \vdots \\ b_k^{(k-1)} \\ b_{k+1}^{(k)} \\ \vdots \\ b_n^{(k)} \end{bmatrix} =: b^{(k)}$$

where for $i = k+1 : n, j = k+1 : n$,

$$a_{ij}^{(k)} = a_{ij}^{(k-1)} - \underbrace{\frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}}}_{=:m_{ik}} a_{kj}^{(k-1)}; \quad b_i^{(k)} = b_i^{(k-1)} - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} b_k^{(k-1)};$$

Here $a_{kk}^{(k-1)} \longleftarrow$ pivot (assumed non zero); $m_{ik} \longleftarrow$ multipliers;

GENP

Step $n-1$: Create a zero in the $(n, n-1)$ of $A^{(n-2)}$:

$$A^{(n-2)} \longrightarrow \underbrace{\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ & a_{22}^{(1)} & \cdots & a_{2n}^{(1)} \\ & & \ddots & \vdots \\ & & & a_{nn}^{(n-1)} \end{bmatrix}}_{=:A^{(n-1)} \text{ (also called } U)}; \quad b^{(n-2)} \longrightarrow b^{(n-1)};$$

where assuming pivot $a_{n-1,n-1}^{(n-2)} \neq 0$ and using multiplier $m_{n,n-1} := \frac{a_{n,n-1}^{(n-2)}}{a_{n-1,n-1}^{(n-2)}}$,

$$a_{nn}^{(n-1)} = a_{nn}^{(n-2)} - m_{n,n-1} a_{n-1,n}^{(n-2)}; \quad b_n^{(n-1)} = b_n^{(n-2)} - m_{n,n-1} b_{n-1}^{(n-2)}.$$

GENP

Step $n-1$**:** Create a zero in the $(n, n-1)$ of $A^{(n-2)}$:

$$A^{(n-2)} \longrightarrow \underbrace{\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ & a_{22}^{(1)} & \cdots & a_{2n}^{(1)} \\ & & \ddots & \vdots \\ & & & a_{nn}^{(n-1)} \end{bmatrix}}_{=:A^{(n-1)} \text{ (also called } U)}; \quad b^{(n-2)} \longrightarrow b^{(n-1)};$$

where assuming pivot $a_{n-1,n-1}^{(n-2)} \neq 0$ and using multiplier

$m_{n,n-1} := \frac{a_{n,n-1}^{(n-2)}}{a_{n-1,n-1}^{(n-2)}}$,

$$a_{nn}^{(n-1)} = a_{nn}^{(n-2)} - m_{n,n-1} a_{n-1,n}^{(n-2)}; \quad b_n^{(n-1)} = b_n^{(n-2)} - m_{n,n-1} b_{n-1}^{(n-2)}.$$

- The system is transformed to $Ux = b^{(n-1)}$.
- The pivots at each step are on the diagonal of U!
- All steps have to be repeated to solve any new system $Ax = c$ if the multipliers used in the GENP are not saved.

GENP $\equiv$ *LU* decomposition of *A*

Let

$$L = \begin{bmatrix} 1 & & & & & & & \\ m_{21} & 1 & & & & & & \\ m_{31} & m_{32} & \ddots & & & & & \\ \vdots & \vdots & & \ddots & & & & \\ m_{k1} & m_{k2} & \cdots & \cdots & 1 & & & \\ m_{k+1,1} & m_{k+1,2} & \cdots & \cdots & m_{k+1,k} & \ddots & & \\ \vdots & \vdots & & & \vdots & & 1 & \\ m_{n1} & m_{n2} & \cdots & \cdots & m_{nk} & \cdots & m_{n,n-1} & 1 \end{bmatrix}.$$

GENP $\equiv$ *LU* decomposition of *A*

Let

$$L = \begin{bmatrix} 1 & & & & & & & \\ m_{21} & 1 & & & & & & \\ m_{31} & m_{32} & \ddots & & & & & \\ \vdots & \vdots & & \ddots & & & & \\ m_{k1} & m_{k2} & \cdots & \cdots & 1 & & & \\ m_{k+1,1} & m_{k+1,2} & \cdots & \cdots & m_{k+1,k} & \ddots & & \\ \vdots & \vdots & & & \vdots & & 1 & \\ m_{n1} & m_{n2} & \cdots & \cdots & m_{nk} & \cdots & m_{n,n-1} & 1 \end{bmatrix}.$$

Then $A = LU$!

LU **decomposition:** A square matrix A is said to have an LU decomposition if there exists a unit lower triangular matrix L and an upper triangular matrix U such that $A = LU$.

GENP $\equiv$ *LU* decomposition of *A*

Let

$$L = \begin{bmatrix} 1 & & & & & & & \\ m_{21} & 1 & & & & & & \\ m_{31} & m_{32} & \ddots & & & & & \\ \vdots & \vdots & & \ddots & & & & \\ m_{k1} & m_{k2} & \cdots & \cdots & 1 & & & \\ m_{k+1,1} & m_{k+1,2} & \cdots & \cdots & m_{k+1,k} & \ddots & & \\ \vdots & \vdots & & & \vdots & & 1 & \\ m_{n1} & m_{n2} & \cdots & \cdots & m_{nk} & \cdots & m_{n,n-1} & 1 \end{bmatrix}.$$

Then $A = LU$! This needs a proof!

LU **decomposition:** A square matrix A is said to have an LU decomposition if there exists a unit lower triangular matrix L and an upper triangular matrix U such that $A = LU$.

GENP $\equiv$ *LU* decomposition of *A*

Proof: In step k of GENP

$$A^{(k)} = \underbrace{\begin{bmatrix} 1 & & & & & \\ 0 & \ddots & & & & \\ \vdots & \cdots & 1 & & & \\ 0 & & -m_{k+1,k} & \ddots & & \\ \vdots & & \vdots & & \ddots & \\ 0 & \cdots & -m_{nk} & \cdots & & 1 \end{bmatrix}}_{=:M_k} A^{(k-1)}$$

Then

$$U = A^{(n-1)} = M_{n-1} M_{n-2} \cdots M_k \cdots M_2 M_1 A$$

where M_k, $k = 1, \ldots, n-1$ are the *multiplier* matrices or *Gauss transforms* of Gaussian Elimination.

Exercise: $b^{(n-1)} = M_{n-1} M_{n-2} \cdots M_1 b.$

GENP $\equiv$ *LU* decomposition of A

Note that,

$$U = A^{(n-1)} = M_{n-1}\underbrace{\left(M_{n-2}\cdots\underbrace{\left(M_k\cdots\underbrace{\left(M_2\underbrace{(M_1 A)}_{=A^{(1)}}\right)}_{=A^{(2)}}\right)}_{=A^{(k)}}\right)}_{=A^{(n-2)}}$$

and

$$M_k = I_n - \begin{bmatrix} 0 \\ \vdots \\ 0 \\ m_{k+1,k} \\ \vdots \\ m_{nk} \end{bmatrix} e_k^T, \quad k = 1 : n-1,$$

are rank one updates of I_n. Here e_k is column k of I_n.

GENP $\equiv$ *LU* decomposition of *A*

Observe that

- $$M_k^{-1} = I_n + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ m_{k+1,k} \\ \vdots \\ m_{kn} \end{bmatrix} e_k^T, \quad k = 1 : n-1,$$ (Prove this!)
- For $i_1 < \cdots < i_p$,

$$M_{i_1}^{-1} \cdots M_{i_p}^{-1} = I_n + \sum_{i=i_1}^{i_p} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ m_{i+1,i} \\ \vdots \\ m_{ni} \end{bmatrix} e_i^T,$$ (Prove this!)

GENP $\equiv$ *LU* decomposition of *A*

So $U = M_{n-1}M_{n-2}\cdots M_2M_1A$ implies,

$$A = M_1^{-1}M_2^{-1}\cdots M_{n-1}^{-1}U = \left(I_n + \sum_{k=1}^{n-1}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \\ m_{k+1,k} \\ \vdots \\ m_{nk} \end{bmatrix} e_k^T\right)\right) U$$

$$= \begin{bmatrix} 1 & & & & & & & \\ m_{21} & 1 & & & & & & \\ m_{31} & m_{32} & \ddots & & & & & \\ \vdots & \vdots & & \ddots & & & & \\ m_{k1} & m_{k2} & \cdots & \cdots & 1 & & & \\ m_{k+1,1} & m_{k+1,2} & \cdots & \cdots & m_{k+1,k} & \ddots & & \\ \vdots & \vdots & & & \vdots & & 1 & \\ m_{n1} & m_{n2} & \cdots & \cdots & m_{nk} & \cdots & m_{n,n-1} & 1 \end{bmatrix} U = LU.$$

Algorithm for GENP/LU

```
for k = 1 : n − 1
    if a_kk ≠ 0          (multiplier computation begins)
        for i = k + 1 : n
            a_ik = a_ik / a_kk;
        end
    else
            exit {‘zero pivot encountered’}
    end       (multiplier computation ends)
    for i = k + 1 : n (matrix update begins)
        for j = k + 1 : n
            a_ij = a_ij − a_ik a_kj
        end
    end       (matrix update ends)
end
```

Algorithm for GENP/LU

```
for k = 1 : n − 1
    if a_kk ≠ 0          (multiplier computation begins)
        for i = k + 1 : n
            a_ik = a_ik / a_kk;
        end
    else
            exit {'zero pivot encountered'}
    end        (multiplier computation ends)
    for i = k + 1 : n (matrix update begins)
        for j = k + 1 : n
            a_ij = a_ij − a_ik a_kj
        end
    end        (matrix update ends)
end
```

$L \longrightarrow I_n +$ strictly lower triangular part of output A.
$U \longrightarrow$ upper triangular part of output A.

Algorithm for GENP/LU

```
for k = 1 : n − 1
    if a_kk ≠ 0          (multiplier computation begins)
        for i = k + 1 : n
            a_ik = a_ik / a_kk;
        end
    else
            exit {'zero pivot encountered'}
    end       (multiplier computation ends)
    for i = k + 1 : n (matrix update begins)
        for j = k + 1 : n
            a_ij = a_ij − a_ik a_kj
        end
    end       (matrix update ends)
end
```

$L \longrightarrow I_n +$ strictly lower triangular part of output A.
$U \longrightarrow$ upper triangular part of output A.

Exercise: Show that the flop count of LU decomposition of an $n \times n$ matrix is $\frac{2}{3}n^3 + O(n^2)$ flops.

Rank one updates in GENP

In Step k,

$$
\begin{aligned}
A^{(k)} &= M_k A^{(k-1)} \\
&= \left(I_n - \begin{bmatrix} 0 \\ \vdots \\ 0 \\ m_{k+1,k} \\ \vdots \\ m_{nk} \end{bmatrix} e_k^T \right) A^{(k-1)} = A^{(k-1)} - \begin{bmatrix} 0 \\ \vdots \\ 0 \\ m_{k+1,k} \\ \vdots \\ m_{nk} \end{bmatrix} \underbrace{e_k^T A^{(k-1)}}_{\text{row } k \text{ of } A^{(k-1)}} \\
&= \underbrace{A^{(k-1)} - \begin{bmatrix} 0 \\ \vdots \\ 0 \\ m_{k+1,k} \\ \vdots \\ m_{nk} \end{bmatrix} \begin{bmatrix} 0 & \cdots & 0 & a_{kk}^{(k-1)} & \cdots & a_{kn}^{(k-1)} \end{bmatrix}}_{\text{rank one update of } A^{(k-1)}}
\end{aligned}
$$

Rank one updates in GENP

Now

$$\begin{bmatrix} 0 \\ \vdots \\ 0 \\ m_{k+1,k} \\ \vdots \\ m_{nk} \end{bmatrix} \begin{bmatrix} 0 & \cdots & 0 & a_{kk}^{(k-1)} & \cdots & a_{kn}^{(k-1)} \end{bmatrix} = \left[\begin{array}{c|c} \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \widehat{M}_k \end{array}\right],$$

where $\widehat{M}_k = \begin{bmatrix} m_{k+1,k} \\ \vdots \\ m_{nk} \end{bmatrix} \begin{bmatrix} a_{kk}^{(k-1)} & \cdots & a_{kn}^{(k-1)} \end{bmatrix}.$

Rank one updates in GENP

Now

$$\begin{bmatrix} 0 \\ \vdots \\ 0 \\ m_{k+1,k} \\ \vdots \\ m_{nk} \end{bmatrix} \begin{bmatrix} 0 & \cdots & 0 & a_{kk}^{(k-1)} & \cdots & a_{kn}^{(k-1)} \end{bmatrix} = \left[\begin{array}{c|c} \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \widehat{M}_k \end{array}\right],$$

where $\widehat{M}_k = \begin{bmatrix} m_{k+1,k} \\ \vdots \\ m_{nk} \end{bmatrix} \begin{bmatrix} a_{kk}^{(k-1)} & \cdots & a_{kn}^{(k-1)} \end{bmatrix}$.

As

$$m_{ik} a_{kk}^{(k-1)} = \left(a_{ik}^{(k-1)} / a_{kk}^{(k-1)} \right) a_{kk}^{(k-1)} = a_{ik}^{(k-1)}, \quad i = k+1 : n,$$

the first column of $\widehat{M}_k$ is

$$\begin{bmatrix} a_{k+1,k}^{(k-1)} \\ \vdots \\ a_{nk}^{(k-1)} \end{bmatrix}.$$

Rank one updates in GENP

Therefore,

$$A^{(k)} = A^{(k-1)} - \left[\begin{array}{c|c} \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \widehat{M}_k \end{array}\right]$$

$$= \left[\begin{array}{c|c} A_{11}^{(k-1)} & A_{12}^{(k-1)} \\ \hline A_{21}^{(k-1)} & A_{22}^{(k-1)} \end{array}\right]$$

$$- \left[\begin{array}{cccc|ccc} 0 & & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \cdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & & 0 & 0 & 0 & \cdots & 0 \\ \hline 0 & & 0 & a_{k+1,k}^{(k-1)} & & & \\ \vdots & \cdots & \vdots & \vdots & \multicolumn{3}{c}{\begin{bmatrix} m_{k+1,k} \\ \vdots \\ m_{nk} \end{bmatrix} \begin{bmatrix} a_{k,k+1}^{(k-1)} & \cdots & a_{kn}^{(k-1)} \end{bmatrix}} \\ 0 & & 0 & a_{nk}^{(k-1)} & & & \end{array}\right]$$

where

$$A_{11}^{(k-1)} \to k \times k; \quad A_{21}^{(k-1)} \to k \times (n-k);$$
$$A_{21}^{(k-1)} \to (n-k) \times k; \quad A_{22}^{(k-1)} \to (n-k) \times (n-k).$$

Rank one updates in GENP

$$\text{As } A_{21}^{(k-1)} = \begin{bmatrix} 0 & & 0 & a_{k+1,k}^{(k-1)} \\ \vdots & \cdots & \vdots & \vdots \\ 0 & & 0 & a_{nk}^{(k-1)} \end{bmatrix}, \text{ therefore,}$$

$$A^{(k)} = \left[\begin{array}{c|c} A_{11}^{(k-1)} & A_{12}^{(k-1)} \\ \hline & \color{red}{A_{22}^{(k)}} \end{array}\right],$$

where

$$A_{22}^{(k)} = \underbrace{A_{22}^{(k-1)} - \begin{bmatrix} m_{k+1,k} \\ \vdots \\ m_{nk} \end{bmatrix} \begin{bmatrix} a_{k,k+1}^{(k-1)} & \cdots & a_{kn}^{(k-1)} \end{bmatrix}}_{\color{red}{\text{rank one update of } A_{22}^{(k-1)}}}$$

$$= \begin{bmatrix} a_{k+1,k+1}^{(k-1)} & \cdots & a_{k+1,n}^{(k-1)} \\ \vdots & \ddots & \vdots \\ a_{n,k+1}^{(k-1)} & \cdots & a_{nn}^{(k-1)} \end{bmatrix} - \begin{bmatrix} \frac{a_{k+1,k}^{(k-1)}}{a_{kk}^{(k-1)}} \\ \vdots \\ \frac{a_{nk}^{(k-1)}}{a_{kk}^{(k-1)}} \end{bmatrix} \begin{bmatrix} a_{k,k+1}^{(k-1)} & \cdots & a_{kn}^{(k-1)} \end{bmatrix}$$

$$\approx - [\;] [\;]$$

Algorithm for GENP/LU with higher level BLAS

```
for k = 1:n-1
    if A(k,k) ≠ 0        (multiplier computation begins)
        A(k+1:n,k) = A(k+1:n,k)/A(k,k);
    else
            exit {'zero pivot encountered'}
    end        (multiplier computation ends)

    (matrix update)
    A(k+1:n,k+1:n) = A(k+1:n,k+1:n) - A(k+1:n,k)∗A(k,k+1:n);
end
```

GENP for solving systems of equations

Pseudocode for solving $n \times n$ system $Ax = b$:

1. Find LU decomposition of A. $\left(\frac{2}{3}n^3 + O(n^2)\text{ flops}\right)$
2. Solve $Ly = b$ for y. (n^2 flops)
3. Solve $Ux = y$ for x. (n^2 flops)

Total flops: $\frac{2}{3}n^3 + O(n^2)$ flops.

it takes 1+2(k-1) flops to
convert a k*k to (k-1)*(k-1) matrix

also after its a 1*1 matrix,we will need 1 flop
to divide b by l11

GENP for solving systems of equations

Pseudocode for solving $n \times n$ system $Ax = b$:

1. Find LU decomposition of A. $\left(\frac{2}{3}n^3 + O(n^2)\right.$ flops$\left.\right)$
2. Solve $Ly = b$ for y. (n^2 flops)
3. Solve $Ux = y$ for x. (n^2 flops)

Total flops: $\frac{2}{3}n^3 + O(n^2)$ flops.

First step need NOT be repeated for solving other systems with same A.

GENP for solving systems of equations

Pseudocode for solving $n \times n$ system $Ax = b$:

1. Find LU decomposition of A. $\left(\frac{2}{3}n^3 + O(n^2)\text{ flops}\right)$
2. Solve $Ly = b$ for y. $(n^2\text{ flops})$
3. Solve $Ux = y$ for x. $(n^2\text{ flops})$

Total flops: $\frac{2}{3}n^3 + O(n^2)$ flops.

First step need NOT be repeated for solving other systems with same A.

But the algorithm does not always work!

GENP for solving systems of equations

Pseudocode for solving $n \times n$ system $Ax = b$:

1. Find LU decomposition of A. $\left(\frac{2}{3}n^3 + O(n^2)\text{ flops}\right)$
2. Solve $Ly = b$ for y. $(n^2$ flops)
3. Solve $Ux = y$ for x. $(n^2$ flops)

Total flops: $\frac{2}{3}n^3 + O(n^2)$ flops.

First step need NOT be repeated for solving other systems with same A.

But the algorithm does not always work!

Theorem: A nonsingular square matrix has an LU decomposition if and only if all its leading principal submatrices are nonsingular.

Additionally for such matrices, the LU decomposition is unique.

$\boxed{A \to \text{nonsingular.}}$

Suppose $A = LU \Rightarrow$

$$\begin{array}{c} \\ k \\ n-k \end{array}\begin{array}{c} \begin{array}{cc} k & n-k \end{array} \\ \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array}\right] \end{array} = \begin{array}{c} \\ k \\ n-k \end{array}\begin{array}{c} \begin{array}{cc} k & n-k \end{array} \\ \left[\begin{array}{c|c} L_{11} & \\ \hline L_{12} & L_{22} \end{array}\right] \end{array} \begin{array}{c} \begin{array}{cc} k & n-k \end{array} \\ \left[\begin{array}{c|c} U_{11} & U_{12} \\ \hline & U_{22} \end{array}\right] \end{array}$$

$$\Rightarrow A_{11} = L_{11} U_{11}$$

$\therefore \det A_{11} = \det(L_{11} U_{11}) = \underbrace{\det L_{11}}_{=1} \det U_{11} = \det U_{11}$ — (1)

But $0 \neq \det A = \det(LU) = \underbrace{\det L}_{=1} \det U = \det U_{11} \det U_{22}$

$\Rightarrow \det U_{11} \neq 0$

$\therefore$ (1) $\Rightarrow \det A_{11} \neq 0 \Rightarrow$ leading principal submatrices of A are nonsingular.

Conversely let all leading principal submatrices of A be nonsingular. For $n = 1$, $A = [a]$, $a \neq 0$.

& $A = [a] = \underbrace{[1]}_{=:L} \underbrace{[a]}_{=:U}$.

Suppose that A has an LU decomposition for size $< n$,

$$A = \begin{array}{c} \\ n-1 \\ 1 \end{array}\!\!\begin{array}{c} \begin{array}{cc} n-1 & 1 \end{array} \\ \left[\begin{array}{c|c} \hat{A} & b \\ \hline c^T & a_{nn} \end{array}\right] \end{array}.$$

So $\hat{A}$ has an LU decomposition say $\hat{A} = \hat{L}\hat{U}$.

$$= \left[\begin{array}{c|c} \hat{L}\hat{U} & b \\ \hline c^T & a_{nn} \end{array}\right] = \left[\begin{array}{c|c} \hat{L}\hat{U} & \hat{L}\hat{L}^{-1}b \\ \hline c^T & a_{nn} \end{array}\right]$$

$$= \left[\begin{array}{c|c} \hat{L} & \\ \hline l^T & 1 \end{array}\right] \left[\begin{array}{c|c} \hat{U} & \hat{L}^{-1}b \\ \hline & u_{nn} \end{array}\right]$$

where $l^T\hat{U} = c^T$ and $u_{nn} + l^T\hat{L}^{-1}b = a_{nn}$

i.e. $l = (\hat{U}^T)^{-1}c$ and $u_{nn} = a_{nn} - l^T(\hat{L}^{-1}b)$

Gaussian Elimination With Partial Pivoting (GEPP)

1. Checking A for existence of LU decomposition is not possible in practice.
   (i) Numerically it is only possible to ascertain how close A and its leading principal submatrices are to being singular.
   (ii) Ascertaining the proximity of A and its leading principal submatrices to a singular matrix will cost more flops than finding the LU factors.

Gaussian Elimination With Partial Pivoting (GEPP)

1. Checking A for existence of LU decomposition is not possible in practice.
2. Even if A has an LU decomposition, computing it is a numerically unstable process.
   - Small pivots can lead to large multipliers and result in instability in finite precision arithmetic.

Gaussian Elimination With Partial Pivoting (GEPP)

1. Checking A for existence of LU decomposition is not possible in practice.
2. Even if A has an LU decomposition, computing it is a numerically unstable process.

Any remedies?

Gaussian Elimination With Partial Pivoting (GEPP)

1. Checking A for existence of LU decomposition is not possible in practice.
2. Even if A has an LU decomposition, computing it is a numerically unstable process.

Any remedies?

Try Gaussian Elimination with row exchanges also called **Gaussian Elimination with Partial Pivoting (GEPP)**!

Gaussian Elimination With Partial Pivoting (GEPP)

1. Checking A for existence of LU decomposition is not possible in practice.
2. Even if A has an LU decomposition, computing it is a numerically unstable process.

Any remedies?

Try Gaussian Elimination with row exchanges also called **Gaussian Elimination with Partial Pivoting (GEPP)**!

What is this?

Gaussian Elimination With Partial Pivoting (GEPP)

1. Checking A for existence of LU decomposition is not possible in practice.
2. Even if A has an LU decomposition, computing it is a numerically unstable process.

$PA = A(p,:) = LU$

$P = I(p,:)$

Any remedies?

Try Gaussian Elimination with row exchanges also called **Gaussian Elimination with Partial Pivoting (GEPP)**!

What is this?

$p = [1:n]$

For each $k = 1 : n-1$

$[r, m] = \max(\text{abs}(A(k:n,k)))$

1. Find $a_{pk}^{(k-1)}$ such that $|a_{pk}^{(k-1)}| = \max_{k \le j \le n} |a_{jk}^{(k-1)}|$.

$r \to$ largest abs. entry
$m \to$ position where r occurs

2. If $p \neq k$ interchange rows k and p.
3. Perform the usual GE steps to create zeros in column k.

if $m \neq k$ $A([k,m],:) = A([m,k],:)$ end

$p([k,m],:) = p([m,k],:)$

GEPP

$$
A^{(k-1)} = \left[\begin{array}{cccc|ccc}
a_{11} & \cdots & \cdots & a_{1k} & a_{1,k+1} & \cdots & a_{1k} \\
m_{21} & a_{22}^{(1)} & \cdots & a_{2k}^{(1)} & a_{2,k+1}^{(1)} & \cdots & a_{2n}^{(1)} \\
 & m_{32} & \ddots & \vdots & \vdots & \cdots & \vdots \\
\hline
\vdots & \vdots & & a_{kk}^{(k-1)} & a_{k,k+1}^{(k-1)} & \cdots & a_{kn}^{(k-1)} \\
 & & & a_{k+1,k}^{(k-1)} & a_{k+1,k+1}^{(k-1)} & \cdots & a_{k+1,n}^{(k-1)} \\
 & & & \vdots & \vdots & & \vdots \\
m_{n1} & m_{n2} & & a_{n,k}^{(k-1)} & a_{n,k+1}^{(k-1)} & \cdots & a_{nn}^{(k-1)}
\end{array}\right]
$$

GEPP $\equiv$ LU decomposition of row permuted A

Permutation Matrices: An $n \times n$ permutation matrix P is obtained by interchanging rows and/or columns of I_n.

GEPP $\equiv$ *LU* decomposition of row permuted *A*

Permutation Matrices: An $n \times n$ permutation matrix P is obtained by interchanging rows and/or columns of I_n.

Examples: $\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$

GEPP $\equiv$ *LU* decomposition of row permuted *A*

Permutation Matrices: An $n \times n$ permutation matrix P is obtained by interchanging rows and/or columns of I_n.

Examples: $\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$

Transposition: A transposition is a permutation matrix obtained by interchanging only two rows or two columns of an identity matrix.

GEPP $\equiv$ LU decomposition of row permuted A

Permutation Matrices: An $n \times n$ permutation matrix P is obtained by interchanging rows and/or columns of I_n.

Examples: $\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$

Transposition: A transposition is a permutation matrix obtained by interchanging only two rows or two columns of an identity matrix.

1. Permutations are orthogonal matrices. $P^T = P^{-1}$

GEPP $\equiv$ *LU* decomposition of row permuted *A*

Permutation Matrices: An $n \times n$ permutation matrix P is obtained by interchanging rows and/or columns of I_n.

Examples: $\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$

Transposition: A transposition is a permutation matrix obtained by interchanging only two rows or two columns of an identity matrix.

1. Permutations are orthogonal matrices.
2. Transpositions are symmetric matrices.

GEPP $\equiv$ *LU* decomposition of row permuted *A*

Permutation Matrices: An $n \times n$ permutation matrix P is obtained by interchanging rows and/or columns of I_n.

Examples: $\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$

Transposition: A transposition is a permutation matrix obtained by interchanging only two rows or two columns of an identity matrix.

1. Permutations are orthogonal matrices. $P^T = P^{-1}$
2. Transpositions are symmetric matrices. $P^T = P$
3. Transpositions are there own inverses. $P^T = P = P^{-1}$

GEPP $\equiv$ *LU* decomposition of row permuted *A*

Permutation Matrices: An $n \times n$ permutation matrix P is obtained by interchanging rows and/or columns of I_n.

Examples: $$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$$

Transposition: A transposition is a permutation matrix obtained by interchanging only two rows or two columns of an identity matrix.

1. Permutations are orthogonal matrices.
2. Transpositions are symmetric matrices.
3. Transpositions are there own inverses.
4. Every permutation is a finite product of transpositions.
5. A product of permutation matrices is a permutation matrix.

GEPP $\equiv$ *LU* decomposition of row permuted *A*

Permutation Matrices: An $n \times n$ permutation matrix P is obtained by interchanging rows and/or columns of I_n.

Examples: $$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$$

Transposition: A transposition is a permutation matrix obtained by interchanging only two rows or two columns of an identity matrix.

1. Permutations are orthogonal matrices.
2. Transpositions are symmetric matrices.
3. Transpositions are there own inverses.
4. Every permutation is a finite product of transpositions.
5. A product of permutation matrices is a permutation matrix.

Theorem Given any $n \times n$ matrix A, there exists a permutation P such that PA has an LU decomposition.

GEPP $\equiv$ LU decomposition of row permuted A

Recall that GENP requires multiplier matrices $M_1, \ldots, M_{n-1}$ such that

$$U = M_{n-1} M_{n-2} \cdots M_1 A.$$

GEPP $\equiv$ LU decomposition of row permuted A

Recall that GENP requires multiplier matrices $M_1, \ldots, M_{n-1}$ such that

$$U = M_{n-1}M_{n-2}\cdots M_1 A.$$

Now GEPP requires finding transpositions $P_1, \ldots P_{n-1}$ and multipliers matrices $M_1, \ldots, M_{n-1}$, such that

$$U = \underbrace{\left\{ M_{n-1}P_{n-1} \underbrace{\left(M_{n-2}P_{n-2}\cdots \underbrace{\left(M_kP_k\cdots \underbrace{\left(M_2P_2 \underbrace{(M_1P_1A)}_{=A^{(1)}} \right)}_{=A^{(2)}} \right)}_{=A^{(k)}} \right)}_{=A^{(n-2)}} \right\}}_{=A^{(n-1)}}$$

GEPP $\equiv$ *LU* decomposition of row permuted *A*

Here for $k = 1, \ldots, n-1$,

1. $P_k = \left[\begin{array}{c|c} I_{k-1} & \\ \hline & \widehat{P}_k \end{array}\right]$, $\widehat{P}_k$ being a $(n-k+1) \times (n-k+1)$ transposition.

GEPP $\equiv$ *LU* decomposition of row permuted *A*

Here for $k = 1, \ldots, n-1$,

1. $P_k = \left[\begin{array}{c|c} I_{k-1} & \\ \hline & \widehat{P}_k \end{array}\right]$, $\widehat{P}_k$ being a $(n-k+1) \times (n-k+1)$ transposition.
2. $M_k = I_n - \begin{bmatrix} 0 \\ \vdots \\ 0 \\ m_{k+1,k} \\ \vdots \\ m_{nk} \end{bmatrix} e_k^T$, with $m_{jk} = a_{jk}^{(k-1)} / a_{kk}^{(k-1)}$, $j = k+1 : n$.

GEPP $\equiv LU$ decomposition of row permuted A

1. Let $\mathcal{P}_k = P_{k+1} \cdots P_{n-1}$, $k = 1, \ldots, n-2$. Then,

$$\mathcal{P}_k = \left[\begin{array}{c|c} I_k & \\ \hline & \widetilde{P}_{k+1} \cdots \widetilde{P}_{n-1} \end{array}\right]$$

where for all $j = k+1, \ldots, n-1$, $\widetilde{P}_j$ are transpositions of size $n-k \times n-k$.

GEPP $\equiv$ *LU* decomposition of row permuted *A*

1. Let $\mathcal{P}_k = P_{k+1} \cdots P_{n-1}$, $k = 1, \ldots, n-2$. Then,
$$\mathcal{P}_k = \left[\begin{array}{c|c} I_k & \\ \hline & \widetilde{P}_{k+1} \cdots \widetilde{P}_{n-1} \end{array}\right]$$
where for all $j = k+1, \ldots, n-1$, $\widetilde{P}_j$ are transpositions of size $n-k \times n-k$.
2. Let $\widetilde{M}_k = \mathcal{P}_k^T M_k \mathcal{P}_k$, $k = 1, \ldots, n-2$. Then,
$$\widetilde{M}_k = I_n - \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \widetilde{m}_{k+1,k} \\ \vdots \\ \widetilde{m}_{nk} \end{bmatrix} e_k^T,$$
where
$$\begin{bmatrix} \widetilde{m}_{k+1,k} \\ \vdots \\ \widetilde{m}_{nk} \end{bmatrix} = \widetilde{P}_{n-1} \cdots \widetilde{P}_{k+1} \begin{bmatrix} m_{k+1,k} \\ \vdots \\ m_{nk} \end{bmatrix}.$$

GEPP $\equiv$ LU decomposition of row permuted A

1. Let $\mathcal{P}_k = P_{k+1} \cdots P_{n-1}$, $k = 1, \ldots, n-2$. Then,
$$\mathcal{P}_k = \left[\begin{array}{c|c} I_k & \\ \hline & \widetilde{P}_{k+1} \cdots \widetilde{P}_{n-1} \end{array}\right]$$
where for all $j = k+1, \ldots, n-1$, $\widetilde{P}_j$ are transpositions of size $n-k \times n-k$.
2. Let $\widetilde{M}_k = \mathcal{P}_k^T M_k \mathcal{P}_k$, $k = 1, \ldots, n-2$. Then,
$$\widetilde{M}_k = I_n - \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \widetilde{m}_{k+1,k} \\ \vdots \\ \widetilde{m}_{nk} \end{bmatrix} e_k^T,$$
where
$$\begin{bmatrix} \widetilde{m}_{k+1,k} \\ \vdots \\ \widetilde{m}_{nk} \end{bmatrix} = \widetilde{P}_{n-1} \cdots \widetilde{P}_{k+1} \begin{bmatrix} m_{k+1,k} \\ \vdots \\ m_{nk} \end{bmatrix}.$$
3. $U = M_{n-1} \widetilde{M}_{n-2} \cdots \widetilde{M}_1 P_{n-1} P_{n-2} \cdots P_1 A$.

GEPP $\equiv$ *LU* decomposition of row permuted *A*

1. Let $\mathcal{P}_k = P_{k+1}\cdots P_{n-1}$, $k = 1, \ldots, n-2$. Then,

$$\Rightarrow \mathcal{P}_k^T = (P_{k+1}\cdots P_{n-1})^T = P_{n-1}\cdots P_{k+1}$$

$$\mathcal{P}_k = \left[\begin{array}{c|c} I_k & \\ \hline & \widetilde{P}_{k+1}\cdots\widetilde{P}_{n-1} \end{array}\right]$$

   where for all $j = k+1, \ldots, n-1$, $\widetilde{P}_j$ are transpositions of size $n-k \times n-k$. Prove this!

2. Let $\widetilde{M}_k = \mathcal{P}_k^T M_k \mathcal{P}_k$, $k = 1, \ldots, n-2$. Then,

$$\Rightarrow \widetilde{M}_k \mathcal{P}_k^T = \mathcal{P}_k^T M_k$$

$$\Rightarrow P_{n-1}\cdots P_{k+1} M_k = \widetilde{M}_k P_{n-1}\cdots P_{k+1}$$

$$P_{n-1} M_{n-2} = \widetilde{M}_{n-2} P_{n-1} \quad (k = n-2)$$

$$P_{n-1} P_{n-2} M_{n-3} = \widetilde{M}_{n-3} P_{n-1} P_{n-2} \quad (k = n-3)$$

$$\widetilde{M}_k = I_n - \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \widetilde{m}_{k+1,k} \\ \vdots \\ \widetilde{m}_{nk} \end{bmatrix} e_k^T,$$

   where $\begin{bmatrix} \widetilde{m}_{k+1,k} \\ \vdots \\ \widetilde{m}_{nk} \end{bmatrix} = \widetilde{P}_{n-1}\cdots\widetilde{P}_{k+1}\begin{bmatrix} m_{k+1,k} \\ \vdots \\ m_{nk} \end{bmatrix}$. Prove this!

3. $U = M_{n-1}\widetilde{M}_{n-2}\cdots\widetilde{M}_1 P_{n-1}P_{n-2}\cdots P_1 A$. Prove this!

GEPP $\equiv$ LU decomposition of row permuted A

Theorem Gaussian Elimination with Partial Pivoting (GEPP) on an $n \times n$ matrix A that transforms it to an upper triangular matrix U also finds a permutation matrix P and a lower triangular matrix L such that $PA = LU$. Moreover if P_k be the transposition used in step k, $1 \le k \le n-1$, then $P = P_{n-1} \cdots P_1$ and

$$L = \begin{bmatrix} 1 & & & & & & & \\ \widetilde{m}_{21} & 1 & & & & & & \\ \widetilde{m}_{31} & \widetilde{m}_{32} & \ddots & & & & & \\ \vdots & \vdots & & \ddots & & & & \\ \widetilde{m}_{k1} & \widetilde{m}_{k2} & \cdots & \cdots & 1 & & & \\ \widetilde{m}_{k+1,1} & \widetilde{m}_{k+1,2} & \cdots & \cdots & \widetilde{m}_{k+1,k} & \ddots & & \\ \vdots & \vdots & & & \vdots & & 1 & \\ \widetilde{m}_{n1} & \widetilde{m}_{n2} & \cdots & \cdots & \widetilde{m}_{nk} & \cdots & m_{n,n-1} & 1 \end{bmatrix},$$

where $\widetilde{m}_{ik}$, $k+1 \le i \le n$, $1 \le k \le n-2$ and $m_{n,n-1}$ are as described earlier.

GEPP

Exercise: Prove the theorem in the previous slide.

Use it to write a Matlab program `[L,U,P] = gepp(A)` that execute GEPP on A to find a permutation P, a unit lower triangular matrix L and an upper triangular matrix U such that $PA = LU$.

Your program should make only the most essential modifications to `[L,U] = genp(A)` and retain all major features essential for efficiency.

Exercise: The flop count of GEPP on an $n \times n$ matrix A, or equivalently the flop count of finding the permutation P such that $PA = LU$ is $\frac{2}{3}n^3 + O(n^2)$ flops.

Solving a system of equations via GEPP

Pseudocode for solving $Ax = b$ via GEPP:

1. Find a permutation P a unit lower triangular matrix L and an upper triangular matrix U via GEPP such that $PA = LU$. ($\frac{2}{3}n^3 + O(n^2)$ flops)
2. Solve $Ly = Pb$ for y. (n^2 flops)
3. Solve $Ux = y$ for x. (n^2 flops)

Total flop count: $\frac{2}{3}n^3 + O(n^2)$.

$PAx = Pb$

$\Leftrightarrow LUx = Pb$

Let $Ux = y$

Then $Ly = Pb$

& $Ux = y$

Gaussian Elimination with Complete pivoting (GECP)

The following alternative strategy may be used to find a largest possible pivot:

For each $k = 1 : n - 1$

1. Find $a_{pm}^{(k-1)}$ such that

$$|a_{pm}^{(k-1)}| = \max_{k \leq j \leq n} \max_{k \leq i \leq n} |a_{ij}^{(k-1)}|.$$

2. If $p \neq k$ interchange rows p and k and if $m \neq k$ interchange columns m and k.
3. Perform the usual GE steps to create zeros in column k.

Gaussian Elimination with Complete pivoting (GECP)

The following alternative strategy may be used to find a largest possible pivot:

For each $k = 1 : n-1$

1. Find $a_{pm}^{(k-1)}$ such that

$$|a_{pm}^{(k-1)}| = \max_{k \le j \le n} \max_{k \le i \le n} |a_{ij}^{(k-1)}|.$$

2. If $p \neq k$ interchange rows p and k and if $m \neq k$ interchange columns m and k.
3. Perform the usual GE steps to create zeros in column k.

This is **Gaussian Elimination with Complete Pivoting (GECP).**

Gaussian Elimination with Complete pivoting (GECP)

The following alternative strategy may be used to find a largest possible pivot:

For each $k = 1 : n-1$

1. Find $a_{pm}^{(k-1)}$ such that

$$|a_{pm}^{(k-1)}| = \max_{k \le j \le n} \max_{k \le i \le n} |a_{ij}^{(k-1)}|.$$

2. If $p \neq k$ interchange rows p and k and if $m \neq k$ interchange columns m and k.
3. Perform the usual GE steps to create zeros in column k.

This is **Gaussian Elimination with Complete Pivoting (GECP).**

Theorem GECP is equivalent to finding permutation matrices P and Q, a unit lower triangular matrix L and an upper triangular matrix U such that $PAQ = LU$.

GECP

Flop Count: Pivoting costs an additional $(n-k+1)^2-1$ comparisons in step k. This raises the total flop count by $n^3/3$. Thus GECP (or equivalently) finding $PAQ = LU$ costs $n^3 + O(n^2)$ flops.

Exercise: Find a pseudocode for solving an $n \times n$ system of equations $Ax = b$ via GECP.

Decompositions related to $A = LU$.

Exercise: Let A be an $n \times n$ nonsingular matrix with nonsingular leading principal submatrices. Prove the following:

1. There exists a unique unit lower triangular matrix L, a unique unit upper triangular matrix V and a unique diagonal matrix D such that $A = LDV$.

Decompositions related to $A = LU$.

Exercise: Let A be an $n \times n$ nonsingular matrix with nonsingular leading principal submatrices. Prove the following:

1. There exists a unique unit lower triangular matrix L, a unique unit upper triangular matrix V and a unique diagonal matrix D such that $A = LDV$.
2. If A is symmetric, then there exists a unique unit lower triangular matrix L, and a unique diagonal matrix D such that $A = LDL^T$.

$A = A^T = (LDV)^T = V^T D L^T$

$\therefore V^T = L$

Decompositions related to $A = LU$.

Exercise: Let A be an $n \times n$ nonsingular matrix with nonsingular leading principal submatrices. Prove the following:

1. There exists a unique unit lower triangular matrix L, a unique unit upper triangular matrix V and a unique diagonal matrix D such that $A = LDV$.
2. If A is symmetric, then there exists a unique unit lower triangular matrix L, and a unique diagonal matrix D such that $A = LDL^T$.
3. Additionally the decomposition $A = LDL^T$ in part 2 has the property that $x^T Ax > 0$ for all nonzero $x \in \mathbb{R}^n$ if and only if D has positive diagonal entries.