LU Decomposition

Let $\underline{A} \in M_{nn}$ & let $\underline{L}$ be a lower triangular matrix w/ $\underline{1}$ on the diagonal & let $\underline{U}$ be an upper triangular matrix such that

$$\underbrace{\underline{A} = \underline{L}\,\underline{U}}_{\text{LU Decomposition}}$$

Why do we care? let $\underline{A} \in M_{nn}$ w/ $\det(\underline{A}) \neq 0$

Look at $\underline{A}\underline{x} = \underline{b}$

$$\underline{L}\,\underline{U}\,\underline{x} = \underline{b}$$

$$\underline{U}\,\underline{x} = \underline{L}^{-1}\underline{b}$$

$$\underline{x} = \underline{U}^{-1}\underline{L}^{-1}\underline{b}$$

In practice solve $\underline{L}\,\underline{y} = \underline{b}$

$$\underline{U}\,\underline{x} = \underline{y}$$

Only useful if solving $\underline{L}\,\underline{y} = \underline{b}$ & $\underline{U}\,\underline{x} = \underline{y}$ is cheaper than solving $\underline{A}\,\underline{x} = \underline{b}$

$\underline{L}\,\underline{y} = \underline{b}$:

$$\begin{bmatrix} 1 & 0 & 0 \\ a & 1 & 0 \\ b & c & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

$$y_1 = b_1$$

$$y_2 = b_2 - a y_1$$

$$y_3 = b_3 - c y_1 - d y_2$$

Forward Substitution

$\underline{U}\,\underline{x} = \underline{y}$: $\begin{bmatrix} e & f & g \\ 0 & h & i \\ 0 & 0 & j \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$

$$x_3 = y_3 / j$$
$$x_2 = (y_2 - i x_3)/h$$
$$x_1 = (y_1 - f x_2 - g x_3)/e$$

Backward Substitution

If $\underline{A}^{-1}$ exits then solving $\underline{L}\,\underline{U}\,\underline{x} = \underline{b}$
Given $\underline{A}\,\underline{x} = \underline{b}$ & $\underline{A} = \underline{L}\,\underline{U}$ is cheap

Steps : ① Find $\underline{A} = \underline{L}\,\underline{U}$ ← expensive
② Solve $\underline{L}\,\underline{y} = \underline{b}$ ← cheap
③ Solve $\underline{U}\,\underline{x} = \underline{y}$ ← cheap

If $\underline{A}$ is fixed in time or if you have multiple $\underline{b}$ vectors find $\underline{A} = \underline{L}\,\underline{U}$ once isn't bad.

Gaussian Elimination to LU.

Look at $\underline{A} = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 6 & 10 \\ 3 & 14 & 28 \end{bmatrix}$

Find an elimination matrix $\underline{E}_1$ such that

$$\underline{E}_1 \underline{A} = \begin{bmatrix} 1 & 2 & 3 \\ 0 & X & X \\ X & X & X \end{bmatrix}$$

To remove the 2 in (2,1) multiply row-1 by $-(2/1)$ & add to row-2

$$\underline{E}_1 = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \Rightarrow \underline{E}_1 \underline{A} = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 2 & 4 \\ 3 & 14 & 15 \end{bmatrix}$$

$$\underline{E}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -(3/1) & 0 & 1 \end{bmatrix}$$

$$\underline{E}_2 \underline{E}_1 \underline{A} = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 2 & 4 \\ 0 & 8 & 19 \end{bmatrix} \Rightarrow \underline{E}_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -(8/2) & 1 \end{bmatrix}$$

$$\underline{E}_3 \underline{E}_2 \underline{E}_1 \underline{A} = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 2 & 4 \\ 0 & 0 & 3 \end{bmatrix} = \underline{U}$$

For this to be an LU-Decomposition is iff $\underline{L} = (\underline{E}_3 \underline{E}_2 \underline{E}_1)^{-1}$

If true then $\underline{A} = (\underline{E}_3 \underline{E}_2 \underline{E}_1)^{-1} \underline{U} = \underline{L}\,\underline{U}$

Rules about lower triangular matrices:

(1) Multiplication of lower triangular matrices remains lower triangular.

$$\underline{L}_1 \underline{L}_2 = \underline{L}_3$$

(2) Inverses of a lower triangular matrix remain lower triangular if they exist.

$$\underline{L}_1^{-1} = \underline{L}_4$$

Our elimination matrices will have $\det(\underline{E}_i) \neq 0$

$$\begin{bmatrix} 1 & & & & 0 \\ 0 & \ddots & & & \\ a & 0 & - & \ddots & \\ 0 & 0 & - & & 1 \end{bmatrix}$$

$$\underline{E}_1 = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \det(\underline{E}_1) = 1$$

$$\underline{E}_1^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ +2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\underline{E}_2^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ +3 & 0 & 1 \end{bmatrix} \qquad \underline{E}_3^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & +4 & 1 \end{bmatrix}$$

$$\Rightarrow \underline{E}_1^{-1} \underline{E}_2^{-1} \underline{E}_3^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & 4 & 1 \end{bmatrix} = \underline{L}$$

$$\Rightarrow \underline{A} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & 4 & 1 \end{bmatrix}}_{\underline{L}} \underbrace{\begin{bmatrix} 1 & 2 & 3 \\ 0 & 2 & 4 \\ 0 & 0 & 3 \end{bmatrix}}_{\underline{U}}$$

Generic (Simple) LU Algorithm

let $\underline{A} \in M_{mn}$

Initialize $\underline{L} = \underline{I}_{m,\min(m,n)}$
$\underline{U} = \underline{A}$

```
for i=1: min(m-1,n)          % Iterate over columns
   for j= i+1 : m            % All rows below current one
      L(j,i) = U(j,i)/U(i,i)   % factor
      U(j,i) = 0
      for k= i+1:n             % future columns
         U(j,k)= U(j,k) - L(j,i)*U(i,k)
      end
   end
end

if m>n
   U= U(1:n, 1:n)
end
```

In Matlab: `[L, U]=lu(A)`

LU Op Count

look at $\begin{bmatrix} 1 & 2 & 3 & 4 \\ 0 & 5 & 6 & 7 \\ 0 & 8 & 9 & 10 \\ 0 & 11 & 12 & 13 \end{bmatrix}$

eliminate   update

Op count: all $+, -, *, \div$

1op : For each row below a pivot, compute a factor ($8/5$, $11/5$)

2op : For each column & row past a pivot, multiply by the factor & add

In this case: 10 total operations
1(2) for elimination
2(4) for update

let $\underline{A} \in M_{mn}$

Initialize $\underline{L} = \underline{I}_{m,\min(m,n)}$
$\underline{U} = \underline{A}$

```
for i=1: min(m-1,n)              % Iterate over columns
    for j= i+1 : m               % All rows below current one
1op     L(j,i) = U(j,i)/U(i,i)   % factor
        U(j,i) = 0
        for k= i+1:n             % future columns
2op        U(j,k)= U(j,k) - L(j,i)*U(i,k)
        end
    end
end

if m>n
    U= U(1:n, 1:n)
end
```

For simplicity let $m=n$

$T =$ OP count

$$= \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left(1 + \sum_{k=i+1}^{n} 2 \right)$$

Relationships:

$$\sum_{i=1}^{n-1} i = \frac{n(n-1)}{2} \qquad \sum_{i=1}^{n-1} i^2 = \frac{n(n-1)(n-2)}{6}$$

$$\sum_{j=i+1}^{n} 1 = n-i \qquad \sum_{j=i+1}^{n} i = i(n-i)$$

$$T = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left(1 + \sum_{k=i+1}^{n} 2\right)$$

$$= \sum_{i=1}^{n-1} \underbrace{\sum_{j=i+1}^{n} 1} + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \underbrace{\sum_{k=i+1}^{n} 2}$$

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

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

$$= \sum_{i=1}^{n-1} \left[(n-i) + 2(n-i)(n-i)\right]$$

$$= \sum_{i=1}^{n-1} \left(2n^2 - 4ni + n + 2i^2 - i\right)$$

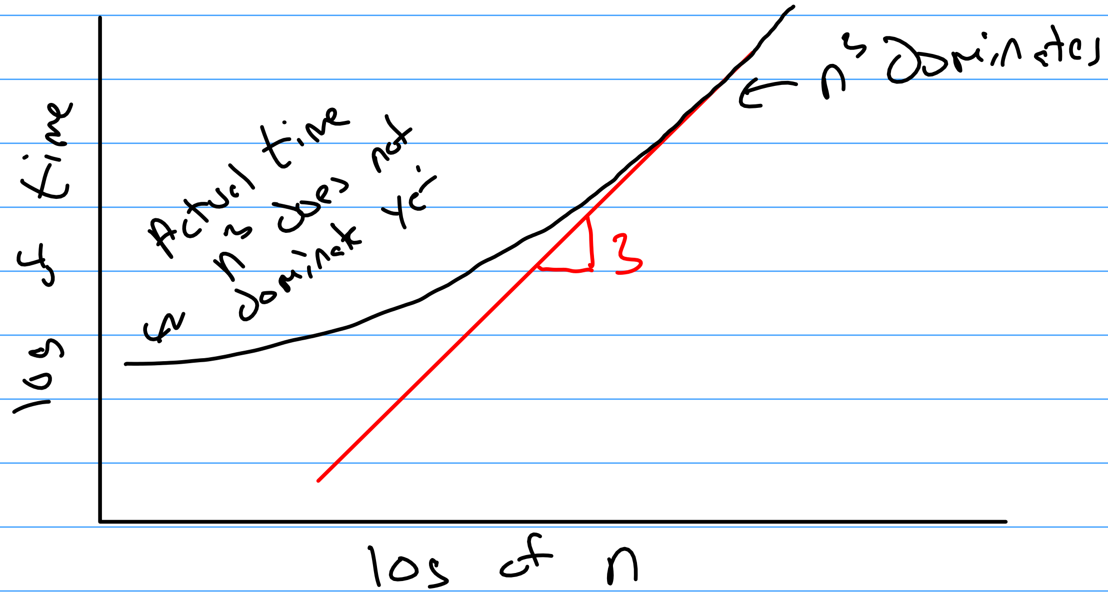
$$T = \frac{2n^3}{6} - \frac{n^2}{6} - \frac{n}{6}$$

Look at "Big-O" notation $\rightarrow$
what happens when n is large,

$n = 10^2$: $n^3 = 10^6$ $\quad n^2 = 10^4$ $\quad n^1 = 10^2$

$n = 10^6$: $n^3 = 10^{18}$ $\quad n^2 = 10^{12}$ $\quad n^1 = 10^6$

Cost of LU is $O(n^3)$
Double the size of A, cost goes up by 8x.



Failure of Simple LU

Simple Gaussian Elimination is not numerically stable

ex, 1) Look at $\underline{A} = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ $\quad$ $\text{rank}(\underline{A}) = 2$, $\kappa(\underline{A}) \sim 2.618$

$$\underline{A}^{-1} = \begin{bmatrix} -1 & 1 \\ 1 & 0 \end{bmatrix}$$

Issues!

1) How to eliminate the $\underline{1}$ in $(2,1)$?

2) let $\underline{A}$ be peturbed

$\hat{\underline{A}} = \begin{bmatrix} 10^{-20} & 1 \\ 1 & 1 \end{bmatrix}$ $\quad$ Exact LU of $\hat{\underline{A}}$

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

In double precision $1 - 10^{20} = -10^{20}$

$$\Rightarrow \underline{L} = \begin{bmatrix} 1 & 0 \\ 10^{20} & 1 \end{bmatrix} \qquad \underline{\hat{U}} = \begin{pmatrix} 10^{-20} & 1 \\ 0 & -10^{20} \end{pmatrix}$$

$$\underline{L}\,\underline{\hat{U}} = \begin{bmatrix} 10^{-20} & 1 \\ 1 & 0 \end{bmatrix} \neq \underline{\hat{A}}$$

Using LU to solve $\underline{A}\,\underline{x} = \underline{b}$ will give incorrect results.

& $\underline{\hat{A}}\,\underline{x} = \underline{b}$ will also be incorrect.

3) What if rows are dependent?

Fix: Introduce (Partial) Pivoting

Swap rows in the method to avoid issues.

Find a matrix $\underline{P}$ such that $\underline{P}\,\underline{A}$ swaps 2 rows of $\underline{A}$.

$\underline{P}$ is simply $\underline{I}$ w/ swapped rows.

ex.) let $\underline{A} \in M_{4,4}$ Swap rows 2 & 4.

$$\underline{I} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \Rightarrow \underline{P} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

ex.) $\underline{A} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 0 \end{bmatrix}$ Swap rows if:
1) Zero in pivot or
2) Larger value below pivot.

$$\underline{P}_1 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \Rightarrow \underline{P}_1 \underline{A} = \begin{bmatrix} 7 & 8 & 0 \\ 4 & 5 & 6 \\ 1 & 2 & 3 \end{bmatrix}$$

$$\underline{E}_1 = \begin{bmatrix} 1 & 0 & 0 \\ -4/7 & 1 & 0 \\ -1/7 & 0 & 1 \end{bmatrix} \Rightarrow \underline{E}_1 \underline{P}_1 \underline{A} = \begin{bmatrix} 7 & 8 & 0 \\ 0 & 3/7 & 6 \\ 0 & 6/7 & 3 \end{bmatrix}$$

$$\underline{P}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \Rightarrow \underline{P}_2 \underline{E}_1 \underline{P}_1 \underline{A} = \begin{bmatrix} 7 & 8 & 0 \\ 0 & 6/7 & 3 \\ 0 & 3/7 & 6 \end{bmatrix}$$

$$\underline{E}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1/2 & 1 \end{bmatrix} \Rightarrow \underline{E}_2 \underline{P}_2 \underline{E}_1 \underline{P}_1 \underline{A} = \begin{bmatrix} 7 & 8 & 0 \\ 0 & 6/7 & 3 \\ 0 & 0 & 9/2 \end{bmatrix} = \underline{U}$$

Goal: Get $\underline{P}\,\underline{A} = \underline{L}\,\underline{U}$

Use the fact that if $\underline{P}_i$ is a permutation matrix $\underline{P}_i^T$ is it's inverse.

$$\underline{E}_2 \underline{P}_2 \underline{E}_1 \underline{P}_1 \underline{A} = \underline{U}$$

$$(\underline{E}_2 \underline{P}_2 \underline{E}_1 \underline{P}_2^T)(\underline{P}_2 \underline{P}_1)\underline{A} = \underline{U}$$

Define $\underline{L} = (\underline{E}_2 \underline{P}_2 \underline{E}_1 \underline{P}_2^{-1})^{-1}$

$\underline{P} = \underline{P}_2 \underline{P}_1$

$\Rightarrow \underline{P}\,\underline{A} = \underline{L}\,\underline{U}$

In this example:

$$\underline{P} = \underline{P}_2 \underline{P}_1 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$\underline{L} = \begin{bmatrix} 1 & 0 & 0 \\ 1/7 & 1 & 0 \\ 4/7 & 1/2 & 1 \end{bmatrix} \qquad \underline{U} = \begin{bmatrix} 7 & 8 & 0 \\ 0 & 6/7 & 3 \\ 0 & 0 & 9/2 \end{bmatrix}$$

To use $\underline{P}\,\underline{A} = \underline{L}\,\underline{U}$ to solve $\underline{A}\,\underline{x} = \underline{b}$

$$\underline{A}\,\underline{x} = \underline{b}$$
$$\underline{P}\,\underline{A}\,\underline{x} = \underline{P}\,\underline{b}$$
$$\underline{L}\,\underline{U}\,\underline{x} = \underline{P}\,\underline{b}$$
$$\underline{U}\,\underline{x} = \underline{L}^{-1}\underline{P}\,\underline{b}$$
$$\underline{x} = \underline{U}^{-1}\underline{L}^{-1}\underline{P}\,\underline{b}$$

In Matlab: [L, U, P] = lu(A)

LU Decomp w/ Pivoting

let $\underline{A} \in M_{nn}$

Initialize $\underline{L} = \underline{I}$, $\underline{U} = \underline{A}$, $\underline{P} = \underline{I}$

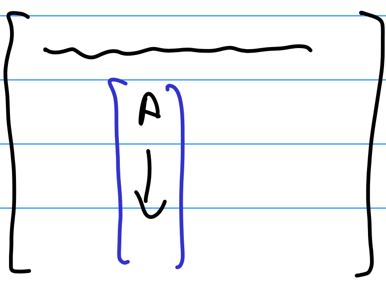
```
for k = 1:n-1
    select i ≥ k that maximizes |u_ik|
    if |u_ik| > 10^-8          % make sure it's not zero
        U(k, k:n) <-> U(i, k:n)        % swap rows
        L(k, 1:k-1) <-> L(i, 1:k-1)    % swap rows
        P(k, :) <-> P(i, :)            % swap rows
        for j = k+1:n
            L(j,k) = u(j,k)/u(k,k)
inner loop ->  U(j,k:n) = u(j,k:n) - L(j,k)*u(k,k:n)
        end
    end
end
```

OP count still $O(n^3)$

Matlab specific:

1) $A([i\ j], :) = A([j\ i], :)$
   swaps rows i & j

2) $[\sim, i] = \max(\text{abs}(A(k:n, k)))$ (?)



$i-1$ will be the # of rows below k that has the max value.

$$\begin{array}{c c}
\begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 0 & 6 & 7 & 8 & 9 \\ 0 & 0 & 2 & 3 & 4 \\ 0 & 10 & 9 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 \end{bmatrix} & \begin{array}{l} \\ r \\ \\ v \\ \\ \end{array}
\end{array}$$

$$\hookrightarrow \begin{bmatrix} 6 \\ 1 \\ 10 \\ 1 \end{bmatrix} \begin{array}{l} \\ \\ \leftarrow i=3 \\ \\ \end{array}$$

used directly it will swap rows 2 & 3

$$\begin{bmatrix} 1 & 0 & 0 \\ a & 1 & 0 \\ b & c & 1 \end{bmatrix} \begin{bmatrix} d & e & f \\ 0 & g & h \\ 0 & 0 & i \end{bmatrix} = \begin{bmatrix} d & e & f \\ ad & ae+g & af+h \\ bd & be+cg & bf+ch+i \end{bmatrix}$$