

Mat609

L #2

Sept 1, 2011

Easy to solve systems. LU-factorization

We already discussed solution of the system  $Lx=b$  where  $L$  is lower triangular matrix. The same is true for  $Ux=b$ , where  $U$  is an upper triangular matrix. Now if we have  $A=LU$  (LU decomposition of  $A$ ), then solving  $Ax=b$  becomes an easy task

$$\underbrace{LUx=b}_{\text{Ly=b}} \quad Ly=b \Rightarrow \text{find } y \quad x=U^{-1}L^{-1}b$$

$$Ux=y \Rightarrow \text{find } x \text{ for } n^2 + O(n) \text{ operations}$$

Thus we can be benefited if we can factorize  $A=LU$ , called LU-factorization method.

There are several algorithms that deal with LU-factorization. Essentially, they depend on how we would like to handle the data ( $a_{ij}$  &  $b_i$ ).

If you look at the number of entries

$$L = \begin{vmatrix} l_{11} & 0 & \dots & 0 \\ l_{21} & l_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \dots & l_{nn} \end{vmatrix} \quad U = \begin{vmatrix} u_{11} & u_{12} & \dots & u_{1n} \\ 0 & u_{22} & \dots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & u_{nn} \end{vmatrix}$$

$n^2+n$  unknowns     $n^2$  equations

(2)

So usually we fix the diagonal elements either of L or of U. Say we are looking at the following representation of  $A = LU$

$$\begin{vmatrix} a_{11} & \dots & a_{1n} \\ a_{21} & \dots & a_{2n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nn} \end{vmatrix} = \begin{vmatrix} l_{11} & 0 & \dots & 0 \\ l_{21} & l_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \dots & l_{nn} \end{vmatrix} \begin{vmatrix} 1 & u_{12} & \dots & u_{1n} \\ 0 & 1 & \dots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{vmatrix}$$

$$i=j=1 \quad a_{11}=l_{11} \cdot 1 \quad \Rightarrow \quad l_{11}=a_{11} \quad \left\{ \begin{array}{l} \text{if we do columns we set} \\ \text{do the method by rows} \end{array} \right. \quad \left\{ \begin{array}{l} a_{j1}=l_{j1} \\ \dots \end{array} \right.$$

$$i=1 \quad j=1, 2, \dots, n \quad l_{11} u_{1j} = a_{1j} \rightarrow u_{1j} = a_{1j}/l_{11}$$

So we have computed the first row of L & U.

Assume we have computed the rows up to  $i-1$  of L & U and we want to advance one step further.

(3)

$$\begin{array}{c}
 L \\
 \left| \begin{array}{cccc}
 l_{11} & 0 & \dots & 0 \\
 l_{21} & l_{22} & \dots & \\
 \vdots & & & \\
 \hline
 l_{i-1,1} & l_{i-1,2} & \dots & l_{i-1,i-1} & 0 \\
 \hline
 l_{ii} & l_{i2} & \dots & l_{ii-1} & l_{ii}
 \end{array} \right| \quad U \\
 \left| \begin{array}{ccccc}
 1 & u_{12} & u_{13} & & \\
 0 & 1 & u_{23} & & \\
 \hline
 0 & 0 & \dots & 1 & u_{i-1,i} \dots \\
 \hline
 0 & 0 & & 0 & 1 \quad u_{ii+1} \dots
 \end{array} \right|^{i-1}
 \end{array}$$

Assume we have computed all rows  $1, \dots, i-1$  (e.g. 1)  
 Now we would like to compute the elements of row  $i$  in  
 both  $L$  &  $U$ .

Step 1. Multiply row  $i$  of  $L$  by column  $j \leq i$  of  $U$  to get

$$l_{i1}u_{1j} + l_{i2}u_{2j} + \dots + l_{ij} \cdot 1 = a_{ij} \quad j=1, 2, \dots, i$$

$$l_{ij} = a_{ij} - \sum_{s=1}^{j-1} l_{is}u_{sj} \quad j=1, 2, \dots, i \quad l_{ii} \text{ computed}$$

Step 2 Multiply row  $i$  of  $L$  by column  $j > i$  of  $U$  to get

$$l_{i1}u_{1j} + l_{i2}u_{2j} + \dots + l_{ii}u_{ij} = a_{ij} \quad j=i+1, i+2, \dots, n$$

$$u_{ij} = (a_{ij} - \sum_{s=1}^{i-1} l_{is}u_{sj}) / l_{ii} \quad j=i+1, \dots, n$$

Operation count

S1:  $j-1$  for  $j=1, 2, \dots, i$   $\boxed{\frac{i(i-1)}{2}}$  computing  $l_{ij}$  }  $n(i+1) - \frac{i^2}{2} - \frac{3i}{2}$

S2:  $(i+1)(n-i)$   $n(i+1) - i^2 - i$  } for  $i=1, \dots, n$

$$\sum n(i+1) - \frac{1}{2} \sum i^2 = \frac{n(n+2)n}{2} - \frac{n(2n+1)(n+1)}{12} \sim \frac{n^3}{2} - \frac{n^3}{6} + \frac{3}{4}n^2 \sim \boxed{\frac{n^3}{3}} + O(n^2)$$

(4)

Main theoretical question: When LU-factorization is possible?

Theorem: If all principal minors of  $A$  are nonsingular then  $A$  has an LU-decomposition

$$A_k = \begin{vmatrix} a_{11} & \cdots & a_{1k} \\ \vdots & & \vdots \\ a_{kk} & \cdots & a_{kk} \end{vmatrix} \quad k=1, \dots, n \text{ principle minors}$$

Proof. Done by induction in  $n$ , for  $n=1$ , if  $a_{11} \neq 0$  then LU decomposition can be performed

$$a_{11} = a_{11} \cdot 1$$

Assume we have done the first  $k-1$  steps i.e.

$$\rightarrow \left| \begin{array}{ccc|c} l_{11} & 0 & 0 & 0 \\ l_{21} & l_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{k-1,1} & l_{k-1,2} & \cdots & l_{k-1,k-1} \\ \hline l_{k1} & l_{k2} & \cdots & l_{kk} \end{array} \right| \left| \begin{array}{ccc} 1 & u_{12} & u_{1k-1} \\ 0 & 1 & u_{2k-1} \\ \vdots & \vdots & \vdots \\ 0 & 0 & 1 \\ \hline 1 & u_{kk+1} & \cdots \end{array} \right| \text{ Computed}$$

$$A = \left[ \begin{array}{c|c} A_{k-1} & \boxed{\phantom{0}} \\ \hline \boxed{\phantom{0}} & \boxed{\phantom{0}} \end{array} \right] \quad L = \left[ \begin{array}{cc|c} l_{k-1} & 0 & \\ \hline \boxed{\phantom{0}} & \boxed{\phantom{0}} & \end{array} \right] \quad U = \left[ \begin{array}{c|c} u_{k-1} & \cdots \\ \hline 0 & \cdots \end{array} \right]$$

$$A_{k-1} = L_{k-1} U_{k-1} \text{ is done}$$

Since  $A_{k-1}$  is nonsingular then  $L_{k-1}$  is nonsingular as well.

(5)

Now we find  $l_{k1} l_{k2} \dots l_{kk}$  by our formulas. Is  $l_{kk} \neq 0$ , i.e. is  $L_k$  nonsingular

But

$$A_n = L_k U_k$$

$\downarrow$                      $\downarrow$   
nonsingular        nonsingular

If  $L_k$  singular  $\Rightarrow A_n$  singular and  
we have contradiction

Therefore  $L_k$  cannot be singular  $\square$

Question: Is there a simple way to find out when a given matrix  $A$  has all leading minors nonsingular? The answer is No!

Question: Are there classes of matrices so that the LU decomposition exists?

Def. The matrix  $A$  is diagonally dominant if

$$|a_{ii}| > \sum_{j \neq i} |a_{ij}| \quad i=1, \dots, n$$

Theorem: If  $A$  is diagonally dominant then it is nonsingular and all leading minors are nonsingular.

(6)

Proof:  $|a_{11}| > \sum_{j=1}^k |a_{1j}| \geq 0 \quad a_{11} \neq 0$

$A_{1,-}$ -minor is nonsingular

$A_K$ -nonsingular !. If singular

$$A_n x = 0 \quad x = \begin{pmatrix} x_1 \\ \vdots \\ x_K \end{pmatrix} \text{ nonzero vector}$$

$$1 = x_{j_0} \neq 0 \quad \max |x_j| \leq 1$$

$j_0$ -equation

$$a_{j_01} x_1 + a_{j_02} x_2 + \dots + \overset{1}{a_{j_0j_0}} \overset{1}{x_{j_0}} + \dots + a_{j_0K} x_K = 0$$

$$a_{j_0j_0} = - \sum_{\substack{s=1 \\ s \neq j_0}}^K a_{j_0s} x_s$$

$$|a_{j_0j_0}| \leq \sum |a_{j_0s}| |x_s| \leq \sum_{\substack{s=1 \\ s \neq j_0}}^K |a_{j_0s}| \leq \sum_{s=1}^n |a_{j_0s}|$$

But our assumption was

$$|a_{j_0j_0}| > \sum_{s \neq j_0} |a_{j_0s}| \quad \text{obvious contradiction}$$

Remark: If  $A$  is nonsingular then there is a permutation matrix  $P$  (interchanging the rows of  $A$ ) so that  $PA = LU$

## Gauss elimination:

$$\left| \begin{array}{l} x_2 + x_3 = 5 \\ 3x_1 - x_2 + x_3 = 6 \\ x_1 + 5x_3 = 1 \end{array} \right.$$

### (A) Gauss

S1: Eliminate  $x_j$  from all equations and reduce the system to  $n-1$  equations

S2: Repeat S1 until one gets just one equation.

### (A) Gauss

To perform S1 one needs to choose one of the equations so that  $x_j$  can be expressed through the other unknowns

The best results produce the strategy when we choose equation  $j$  so that  $a_{jj}$  has maximum absolute value compared to the other elements in column 1. This is called choice of a pivot, or Gauss elimination with pivoting.

If by some reasons we believe that  $x_j$  could be eliminated by its  $j$ -th equation starting with  $j=1, 2, \dots, n$  then this is called Gauss elimination without pivoting

### (B) Jordan

on the  $j$ -th step we eliminate  $x_j$  from all equations

## Gauss elimination without pivoting

$$\left| \begin{array}{ccc|c} a_{11} & a_{12} & a_{1n-1} & b \\ 0 & a_{22} & a_{2n-1} & \\ \hline 0 & 0 & a_{k-1, k-1} & \\ & & & a_{kk} \quad a_{kk+1} \\ 0 & & & a_{kn} \quad a_{kn+1} \\ & & a_{ik} \quad a_{i, n+1} & a_{in} \quad a_{in+1} \\ & & a_{nk} \quad a_{n, n+1} & a_{nn} \quad a_{nn+1} \end{array} \right|$$

for  $k=1, 2, \dots, n-1$

for  $i=k+1, \dots, n$

$$\xi = \frac{a_{ik}}{a_{kk}}$$

for  $j=k+1, \dots, n, n+1$

end  $j$

$$a_{ij} \leftarrow a_{ij} - a_{kj} \xi \quad ]_{n-k}$$

end  $i$

end  $K$

operation count  
 $\times, /$

$]_1$

$(n-k+1)(n-k)$

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