

Gauss Elimination and LU factorization

Standard Gauss Elimination: $A = [a_{ij}]_{1 \leq i, j \leq n}$

$$\begin{matrix} -\frac{a_{21}}{a_{11}} \\ \vdots \\ \vdots \\ \vdots \end{matrix} \rightarrow \left(A \right)$$

$$\downarrow$$

$$\begin{pmatrix} a_{11} & a_{12} & \dots \\ 0 & \boxed{A'} & \\ \vdots & & \\ 0 & & \end{pmatrix} = \begin{pmatrix} 1 & & & 0 \\ -\frac{a_{21}}{a_{11}} & 1 & & \\ \vdots & & \ddots & \\ -\frac{a_{n1}}{a_{11}} & & & 1 \end{pmatrix} \cdot A$$

$$\downarrow$$

$$\begin{pmatrix} a_{11} & a_{12} & \dots \\ 0 & a_{22} & \dots \\ \vdots & \vdots & \boxed{A''} \\ 0 & 0 & \end{pmatrix} = \begin{pmatrix} 1 & 0 & & \\ 0 & 1 & & \\ \vdots & -\frac{a_{32}}{a_{22}} & \ddots & \\ 0 & -\frac{a_{n2}}{a_{22}} & & 1 \end{pmatrix} \begin{pmatrix} 1 & & & 0 \\ -\frac{a_{21}}{a_{11}} & 1 & & \\ \vdots & & \ddots & \\ -\frac{a_{n1}}{a_{11}} & & & 1 \end{pmatrix} A$$

$$\vdots$$

$$\downarrow$$

$$U = \begin{pmatrix} \triangle \\ 0 \end{pmatrix} = \begin{pmatrix} 1 & & & 0 \\ & \ddots & & \\ & & 1 & \\ 0 & & & x \end{pmatrix} \dots \begin{pmatrix} x & & & 0 \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{pmatrix} A$$

$$\Rightarrow A = \begin{pmatrix} 1 & & & \\ \frac{a_{21}}{a_{11}} & 1 & & \\ \vdots & & \ddots & \\ \frac{a_{n1}}{a_{11}} & & & 1 \end{pmatrix} \begin{pmatrix} 1 & & & \\ 0 & 1 & & \\ \vdots & -\frac{a_{32}}{a_{22}} & \ddots & \\ 0 & -\frac{a_{n2}}{a_{22}} & & 1 \end{pmatrix} \dots U$$

$$= \underbrace{\begin{pmatrix} 1 & & & \\ \frac{a_{21}}{a_{11}} & 1 & & \\ \vdots & -\frac{a_{32}}{a_{22}} & \ddots & \\ \frac{a_{n1}}{a_{11}} & -\frac{a_{n2}}{a_{22}} & & 1 \end{pmatrix}}_L \cdot U$$

Gauss-Elimination factors $A = LU$

L = lower triangular + diagonal elements = 1

U = upper triangular

Let $A = LU$. Then

$$Ax = b \iff LUx = b$$

can be solved in 2 steps:

$$\text{solve } Ly = b$$

$$\text{solve } Ux = y$$

Cost:

$$Ly = b \iff \begin{array}{|c} \triangle \\ \hline \end{array} \cdot y = b$$

$$y_1 = b_1$$

$$y_2 = b_2 - l_{21} b_1$$

$$y_3 = b_3 - l_{31} b_1 - l_{32} b_2$$

2 flops

4 flops

$$\text{flops: } \sum_{i=1}^{n-1} 2i = n^2 + O(n)$$

$Ux = y$: same flop count + n divisions

\Rightarrow Given $A = LU$, $Ax = b$ can be solved in $2n^2 + O(n)$ flops

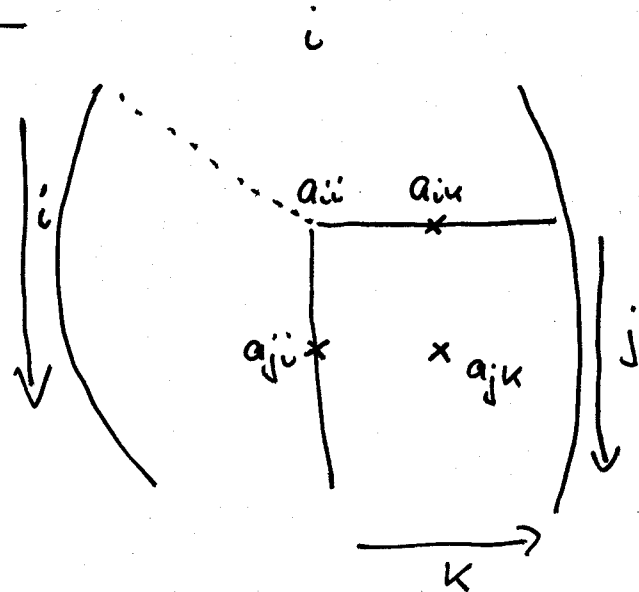
LU factorization: Algorithm

for $i = 1:n-1$

for $j = i+1:n$

for $k = i:n$

$$a_{jk} = a_{jk} - \frac{a_{ji}}{a_{ii}} a_{ik}$$



Let's optimize

1.) $k = i$ yields zero elements (we know that)

for $i = 1 : n-1$
 for $j = i+1 : n$ ← change
 for $k = i+1 : n$
 $a_{jk} = a_{jk} - \frac{a_{ji}}{a_{ii}} a_{ik}$

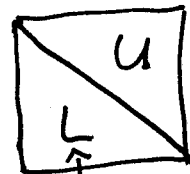
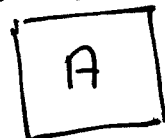
2.) $\frac{a_{ji}}{a_{ii}} = l_{ji}$ independent of k

for $i = 1 : n-1$
 for $j = i+1 : n$
 $l_{ji} = \frac{a_{ji}}{a_{ii}}$
 for $k = i+1 : n$
 $a_{jk} = a_{jk} - \frac{a_{ji}}{a_{ii}} a_{ik}$

3.) Store l_{ji} in a_{ji} , since a_{ji} never touched again after zeroed out. Also, avoid division by a_{ii}

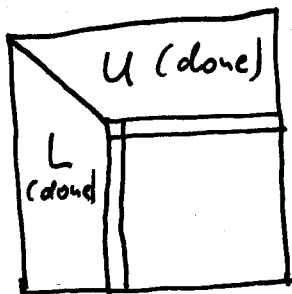
for $i = 1 : n-1$
 $c = 1/a_{ii}$
 for $j = i+1 : n$
 $a_{ji} = c \cdot a_{ji}$
 for $k = i+1 : n$
 $a_{jk} = a_{jk} - \frac{a_{ji}}{a_{ii}} a_{ik}$

computes:



without diagonal
(all 1's anyway)

intermediate step:



Cost analysis

- $(n-1)$ divisions

$$\begin{aligned} \text{- flops: } & \sum_{i=1}^{n-1} \sum_{j=i+1}^n \left(1 + \sum_{k=i+1}^n 2 \right) = \sum_{i=1}^{n-1} \sum_{j=i+1}^n (2(n-i) + 1) \\ & = \sum_{i=1}^{n-1} (2(n-i)^2 + (n-i)) = \sum_{i=1}^{n-1} (2i^2 + i) \end{aligned}$$

$$\text{Use: } \sum_{i=1}^{n-1} i^2 = \frac{(n-1)n(2n-1)}{6} = \frac{1}{3}n^3 + O(n^2) \quad \underline{1}$$

$$= \frac{2}{3}n^3 + O(n^2)$$

Result: - $A=LU$ can be computed using $\frac{2}{3}n^3 + O(n^2)$ flops
- $Ax=b$ can be solved using $\frac{2}{3}n^3 + O(n^2)$ flops
(provided $A=LU$ exists, more later)

Vector form of LU factorization

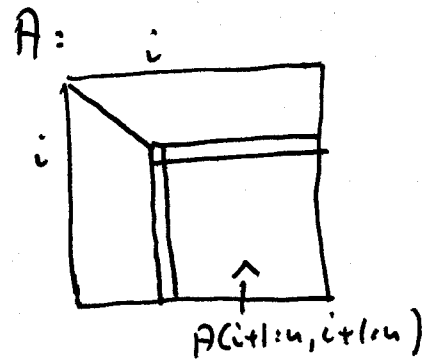
for $i=1, n-1$

$$\text{BLAS1 } A(i+1:n, i) = A(i+1:n, i) / A(i, i)$$

$$\text{BLAS2 } A(i+1:n, i+1:n) = A(i+1:n, i+1:n)$$

$$- A(i+1:n, i) \cdot A(i, i+1:n)$$

called
"rank-1 update"



Pivoting:

Theorem: $A=LU$ exists if in each step of Gauss elimination

$$a_{ii} \neq 0$$

\Leftrightarrow all matrices $A(1:j, 1:j)$ are invertible
(non-singular).

Example: $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ has no LU factorization

Modify Gauss elimination:

GEPP (Gauss elimination with partial pivoting):

for $i = 1:n-1$
 permute rows s.t. $a_{ii} \neq 0$ and $|a_{ii}|$ largest in column
 for numerical stability

[as before

- computes $A = PLU$, P a permutation matrix
- adds $O(n^2)$ comparisons

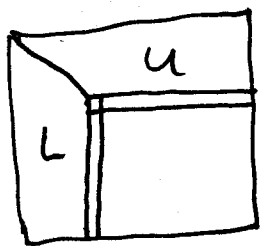
Blocking (as implemented in LAPACK)

Reminder: Why blocking?

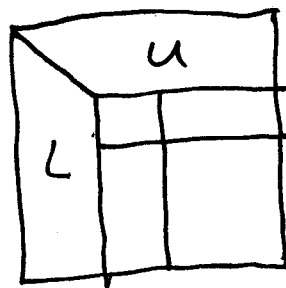
BLAS 1 $y = x^T x_2$
 BLAS 2 $y = Ax$
 BLAS 3 $A = B \cdot C$
 [FFT

memory ops	flops	ratio
$3n$	$2n-1$	$\sim \frac{2}{3}$
$n^2 + 2n$	$2n^2 - n$	~ 2
$3n^2$	$2n^3 - n^3$	$\sim \frac{2}{3}n$
$2n(3n)$	$\sim 4n \log n$	$O(\log n)$

Basic idea:



standard



} $b = \text{block size}$

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$$

step 1: LU factorization of

$$\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} = \begin{pmatrix} L_{11} \\ L_{21} \end{pmatrix} \cdot U_{11}$$

using standard algorithm

then:

$$A = \begin{pmatrix} L_{11} & U_{11} & A_{12} \\ L_{21} & U_{11} & A_{22} \end{pmatrix}$$

$$= \begin{pmatrix} L_{11} & 0 \\ L_{21} & I \end{pmatrix} \begin{pmatrix} U_{11} & L_{11}^{-1} A_{12} \\ 0 & A_{22} - L_{21} L_{11}^{-1} A_{12} \end{pmatrix}$$

$$= \begin{pmatrix} L_{11} & 0 \\ L_{21} & I \end{pmatrix} \begin{pmatrix} U_{11} & U_{12} \\ 0 & A_{22} - L_{21} U_{12} \end{pmatrix}$$

$$U_{12} = L_{11}^{-1} A_{12}$$

$$U_{22} = A_{22} - L_{21} U_{12}$$

thus:

step 2: $U_{12} = L_{11}^{-1} A_{12}$

step 3: $U_{22} = A_{22} - L_{21} U_{12}$

final algorithm:

for $i = 1 : b : n - 1$

use standard algorithm to factorize $A(i:n, i:i+b-1) = \begin{pmatrix} L_{11} \\ L_{21} \end{pmatrix} U_{11}$

$$A(i:i+b-1, i+b:n) = L_{21}^{-1} A(i:i+b-1, i+b:n)$$

$$A(i+b:n, i+b:n) = A(i+b:n, i+b:n)$$

$$- A(i+b:n, i:i+b-1) \cdot A(i:i+b-1, i+b:n)$$

BLAS3

BLAS3

(mm)

Matrix inversion

Given A , find M : $AM = I_n \Leftrightarrow$

solving n systems of linear equations

$$A m_i = e_i$$

↑
ith column of M

step 1: compute $A = PLU$
($\frac{2}{3}n^3 + O(n^2)$)

step 2: solve n systems of l.e.
($2n^2 + O(n)$ each)

$$\Rightarrow \text{cost } \frac{8}{3}n^3 + O(n^2)$$

Determinant

step 1: compute $A = PLU$

step 2: $\det(A) = \prod_{i=1}^n u_{ii} \cdot \underbrace{\det(P)}_{\pm 1}$

$$\Rightarrow \text{cost } \frac{2}{3}n^3 + O(n^2)$$