Welcome to Ma221, Lecture 11, Sep 18

Gaussian Elimination

Additional Goals, on top of
avoid communication, run in $O(n^\omega)$ ops

Backward Stability: exact solution
of $(A+E)\hat{x} = b+f$
$\frac{\|E\|}{\|A\|} = O(\varepsilon)$ $\frac{\|f\|}{\|b\|} = O(\varepsilon)$

Exploit Structure of A
A: symmetric, positive definite
"sparse" = depends on $\ll n^2$
parameters, so could have
mostly zero entries in A,
or depend on few parameters
but be dense.

Ex: Vandermonde Matrix
$V_{ij} = x_i^{j-1}$ given $[x_1, \dots, x_n]$

multiplication V by c: $V \cdot c = b$
= polynomial evaluation at x_i,
coefficients c_i

solve $Vc=b$ for c:
= polynomial interpolation
can be done much faster than $O(n^3)$

Seek Matrix Factorizations

A = product of simpler matrices

SVD: $A = U \Sigma V^T$

= orthog · diag · orthog

Gaussian Elim:

$A = P \cdot L \cdot U$

P = permutation

L = "unit" Lower triangular

$L_{ii} = 1$

U = upper triangular

Least Squares $A = Q \cdot R$

= orthog · upper triang

Eigen $A = Q T Q^T$

= orthog · triang · orthog

Def: Permutation matrix:

identity matrix with permuted rows

Facts: Let P, P_1, P_2 be permutation matrices

P has exactly one 1 in each row and column

$P \cdot X$ = X with permuted rows

$X \cdot P$ = X with permuted columns

$P_1 \cdot P_2$ = permutation

$P^{-1} = P^T$ i.e. P orthogonal

proof: note that $(P^T P)_{ii} = 1 \Rightarrow P^T P = I$

$\det(P) = \pm 1$

storing and multiplying by P cheap:
store indices of 1's, copy rows

Thm: (LU decomposition)

Given any $m \times n$ full rank A $m \geq n$

$\exists$ $m \times m$ perm P

$m \times n$ unit lower triangular $L =$ $L_{ii} = 1$

$n \times n$ upper triangular nonsingular U

such that $A = P \cdot L \cdot U$

Cor: A $n \times n$, nonsingular:

$\exists$ $n \times n$ perm P

$n \times n$ unit Lower triang L:

$n \times n$ upper triang nonsing U:

$A = P \cdot L \cdot U$

To Solve $Ax = b$

(1) Factor $A = P \cdot L \cdot U$

expensive part: costs $\frac{2}{3} n^3 + O(n^2)$

(2) Solve $P \cdot L \cdot U \cdot x = b$ for $L \cdot U \cdot x = P^T b$

by permuting entries of b cost $= O(n)$

(3) Solve $L \cdot U \cdot x = P^T b$ for $U \cdot x = L^{-1} \cdot P^T \cdot b$

by forward substitution, cost $= n^2$

(4) Solve $U \cdot x = L^{-1} \cdot P^T \cdot b$ for $x = U^{-1} \cdot L^{-1} \cdot P^T \cdot b$

by back substitution, cost $= n^2$

Given another b', can solve $Ax' = b'$ in just $O(n^2)$ additional work

Note: We do not compute A^{-1} and multiply $A^{-1} \cdot b$
(1) $3\times$ more expensive in dense case
(can be $O(n^2)\times$ more expensive in sparse case)
(2) Not numerically stable

Proof of: $A = P \cdot L \cdot U$ (Gaussian Elimination)

If A full rank, $\Rightarrow$ first column nonzero
$\Rightarrow \exists \underset{\text{perm}}{P}$ s.t. $(PA)(1,1) \neq 0$

$$PA = \begin{array}{c} \\ 1 \\ m-1 \end{array} \begin{array}{c} \begin{array}{cc} 1 & n-1 \end{array} \\ \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array}\right] \end{array} = \begin{array}{c} \\ 1 \\ m-1 \end{array} \begin{array}{c} \begin{array}{cc} 1 & m-1 \end{array} \\ \left[\begin{array}{c|c} 1 & 0 \\ \hline \frac{A_{21}}{A_{11}} & I \end{array}\right] \end{array} \cdot \begin{array}{c} \\ 1 \\ m-1 \end{array} \begin{array}{c} \begin{array}{cc} 1 & n-1 \end{array} \\ \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline 0 & \underbrace{A_{22} - A_{21} \cdot A_{11}^{-1} \cdot A_{12}}_{S = \text{Schur Complement}} \end{array}\right] \end{array}$$

A full rank $\Rightarrow$ PA full rank $\Rightarrow$ S full rank

Otherwise $\exists x \neq 0$ s.t. $Sx = 0$

and then $A \begin{bmatrix} -A_{12} \cdot x / A_{11} \\ x \end{bmatrix} = 0 \Rightarrow A$ not full rank
contradiction

Simpler in square case:

$0 \neq \det(A) = \pm \det(PA) = \pm \det(1^{st}\text{ factor}) \cdot \det(2^{nd}\text{ factor})$

$= \pm 1 \cdot 1 \cdot A_{11} \cdot \det(S) \Rightarrow \det(S) \neq 0$

Apply induction to $S = P' \cdot L' \cdot U'$

$$PA = \left[\begin{array}{c|c} 1 & 0 \\ \hline \frac{A_{21}}{A_{11}} & I \end{array}\right] \cdot \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline 0 & P' \cdot L' \cdot U' \end{array}\right]$$

$$= \left[\begin{array}{c|c} 1 & 0 \\ \hline \frac{A_{21}}{A_{11}} & P' \cdot L' \end{array}\right] \cdot \underbrace{\left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline 0 & U' \end{array}\right]}_{\text{upper triangular, non sing}}$$

$$= \underbrace{\left[\begin{array}{c|c} 1 & 0 \\ \hline 0 & P' \end{array}\right]}_{\text{perm}} \cdot \underbrace{\left[\begin{array}{c|c} 1 & 0 \\ \hline P'^T \frac{A_{21}}{A_{11}} & L' \end{array}\right]}_{\text{unit lower triangular}} \cdot \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline 0 & U' \end{array}\right]$$

$$A = \underbrace{P^T \left[\begin{array}{c|c} 1 & 0 \\ \hline 0 & P' \end{array}\right]}_{\text{perm}} \cdot \left[\begin{array}{c|c} 1 & 0 \\ \hline P'^T \frac{A_{21}}{A_{11}} & L' \end{array}\right] \cdot \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline 0 & U' \end{array}\right]$$

$$= P \cdot L \cdot U \qquad \text{QED}$$

Express induction proof as Gaussian Elimination (GE)

for $i = 1$ to n
... when $i = 1$ perform alg in proof
... when $i > 1$ apply same alg. to S

for $i = 1$ to n
$L(i,i) = 1$, $L(i+1:n, 1) = A(i+1:n, 1) / A(1,1)$
... ignore permutations for now
$U(i, i:n) = A(i, i:n)$
if $(i < n)$

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

Add permutations: after "for i = 1 to n" add

if $A(i,i) = 0$ and some $A(j,i) \neq 0$ for some $j > i$

(else error)

swap rows i and j of L and of A

record swap in P

... how to choose $A(j,i)$ called "pivoting"

details later

Don't waste space: Let L and U overwrite A

row i of U overwrites row i of A:

omit $U(i, i:n) = A(i, i:n)$

col i of L (below diagonal) overwrites

same entries of A, they are available

because they get zeroed out:

change 1st line to

$$A(i+1:n, i) = A(i+1:n, i) / A(i,i)$$

iterate from i to n-1: change last line to

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

Summary:

for i = 1 to n-1

if $A(i,i) = 0$, and $A(j,i) \neq 0$ for $j > i$

swap rows i and j of A

record swap in P

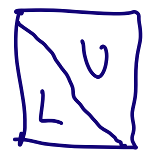
$A(i+1:n, i) = A(i+1:n, i) / A(i,i)$

... BLAS1 scal

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

... BLAS2 ger

when done



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

Same as "traditional GE":

for $i = 1$ to $n-1$

... add a multiple of row i to

... row $j > i$ to zero out entry (j, i)

... i.e. zero out below diagonal

$m = A(j,i) / A(i,i)$

$A(j, i:n) = A(j, i:n) - m \cdot A(i, i:n)$

"Optimize" by

(1) not bothering to compute 0s below diagonal:

change to $A(j, i+1:n) = A(j, i+1:n) - m \cdot A(i, i+1:n)$

(2) compute all multipliers m first, store in "zeroed out" entries

for $i = 1$ to $n-1$

for $j = i+1 : n$

$A(j,i) = A(j,i) / A(i,i)$

for $j = i+1 : n$

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

(3) Combine 2 inner loops

for $i = 1$ to $n-1$

$$A(i+1:n, i) = A(i+1:n, i) / A(i,i)$$

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

Same cost as above: $\frac{2}{3}n^3 + O(n^2)$