Welcome to Ma 221!

people.eecs.berkeley.edu/~demmel/ma221_Fall23

see webpage for
- class notes
  - last semester's notes
  - latex version on bcourses
- office hours
- GSI - Lewis Pan
- Grading: weekly HW (#1 posted)
  - group projects
    - submit preproposal
  - no exams

Class survey - ChatGPT?

Notation:

$\|x\|_2 = \sqrt{\sum_i |x_i|^2}$ 2-norm

$\arg\min_x f(x)$ = value of x that minimizes f(x)

$f(n) = O(g(n))$ means $|f(n)| \le C \cdot |g(n)|$ for some $C > 0$, n large enough

$f(n) = \Omega(g(n))$ means $|f(n)| \ge C \cdot |g(n)|$ for $C > 0$, n large enough

$f(n) = \Theta(g(n))$ means $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$

Syllabus - 4 "axes" of design space of linear algebra algorithms

1) math problem:

solve $Ax = b$

least squares $\arg\min_x \|Ax - b\|_2$

eigenproblems: $Ax = \lambda x$

many generalizations...

2) structure of A

dense, symmetric ($A = A^T$)
positive definite

sparse
"structured"
eg Toeplitz: $A(i,j) = x_{i-j}$
matrix-vector-multiply:
convolution
solve $Ax=b$: $O(n^2)$ not $O(n^3)$

3) desired accuracy

guaranteed correct (too expensive
eg rational arithmetic)

"guaranteed correct" except for
"rare cases"

eg: iterative refinement for
$Ax=b$, using mixed precision

"backward stable"
exact answer for a
slighty wrong problem

residual as small as desired
eg $\|Ax-b\|_2$

"probably OK" (randomized algorithms)

---

error bounds
eg using condition numbers

4) as fast as possible on your target computer. eg ("A\b")
laptop (with GPU)
big parallel computer
cloud, cell-phone, ..

---

one "problem": choose from 1), 2), 3), 4)

answer could be
"type A\b"
"download standard SW from URL"
"project available to implement a proposed alg"
"open problem"

all possible points in 4D design space too big for 1 semester.
We will discuss important subset.

Axis 1: Math problem

Solve $Ax=b$: well defined when A square, full rank, otherwise (or if A close to low rank) then least squares may be better

Least Squares:

- Overdetermined: $A^{m\times n}$

  $\underset{x}{\mathrm{argmin}}\ \|Ax-b\|_2$ when A full column rank ($m \geq n$)

- A not full rank (eg: $m<n$)

  $\Rightarrow x$ not unique, so can pick x that also minimizes $\|x\|_2$, to make x unique

Ridge Regression:

$$\underset{x}{\mathrm{argmin}}\ \|Ax-b\|_2^2 + \lambda\|x\|_2^2$$

$\lambda > 0$

(also called Tikhonov regularization)

x unique if $\lambda > 0$

Constrained LS: $\underset{x:Cx=d}{\operatorname{argmin}} \|Ax-b\|_2$

Ex: x: fractions of population
$\sum_{i=1}^{n} x_i = 1$ (seems natural to also ask for $x_i \geq 0$, harder)

Weighted LS: $\underset{x}{\operatorname{argmin}} \|W^{+}(Ax-b)\|_2$

where W "weight matrix"

(Gauss-Markov linear model)

Total Least Squares

$$\underset{x:(A+E)x=b+r}{\operatorname{argmin}} \quad \| \underbrace{[E, r]}_{\text{matrix}} \|_2$$

Eigenproblems: Notation: $A^{n \times n}$

$A x_i = \lambda_i x_i \quad x_i \neq 0$, for $i=1, \dots n$

$X^{n \times n} = [x_1, \dots x_n]$, $\Lambda = \operatorname{diag}(\lambda_1, \dots, \lambda_n)$

$A X = X \Lambda$, if X nonsingular

$A = X \Lambda X^{-1}$ "eigendecomposition"

Recall A may not have n independent evecs. eg $A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$

Earlier LA class: Jordan Form
but this is numerically unstable
$A' = \begin{bmatrix} \varepsilon & 1 \\ 0 & 0 \end{bmatrix}$, Jordan form is $\begin{bmatrix} \varepsilon & 0 \\ 0 & 0 \end{bmatrix}$
for all $\varepsilon \neq 0$
We will use Schur Form instead

SVD: Singular Value Decomposition

$$A^{m\times n} = U \Sigma V^T \qquad m \geq n$$

$U^{m\times m}$ orthogonal $UU^T = I$

$V^{n\times n}$ " $VV^T = I$

$\Sigma^{m\times n}$ diagonal $\begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_n \\ & & 0 & \end{bmatrix}$

σ_i "singular values
$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n$
columns of U, V are left, right singular vectors:

$$AA^T = (U\Sigma V^T)(U\Sigma V^T)^T$$
$$= (U\Sigma V^T)(V\Sigma^T U^T)$$
$$= U\Sigma \underbrace{(V^T V)}_{= I} \Sigma^T U^T$$
$$= U \underbrace{\Sigma\Sigma^T}_{} U^T$$

$$\underbrace{\qquad}_{}\text{diagonal, square}$$
$= \text{eigendecomp of } AA^T$

$$A^TA = (U\Sigma V^T)^T(U\Sigma V^T)$$
$$= V\Sigma^T\underbrace{(U^TU)}_{=I}\Sigma V^T$$
$$= V\underbrace{\Sigma^T\Sigma}_{\text{square diagonal}} V^T$$

SVD: most "reliable" method for LS, also most expensive

Invariant subspaces: $x'(t) = Ax(t)$

$x(0)$ given. Suppose $Ax(0) = \lambda x(0)$

then $x(t) = e^{\lambda t} x(0)$

Easy to tell if $x(t) \to 0$ as $t \to \infty$

depends on $\text{Real}(\lambda) < 0$

$x(0) = \sum \beta_i x_i$ where $Ax_i = \lambda_i x_i$

$\Rightarrow x(t) = \sum_i e^{\lambda_i t} \beta_i x_i$

whether $x(t) \to 0$ as $t \to \infty$

depends on whether $\text{Real}(\lambda_i) < 0$

for all $\beta_i \neq 0$

i.e. whether $x(0)$ in subspace

spanned by all evecs with $\mathrm{Real}(\lambda_i) < 0$

– called "invariant subspace"

Often possible to compute invariant subspaces more cheaply than whole eigendecomposition