Tues. ① 3/29/16

M116 lec 1

Names & intros.

Syll.

→ Shirley Zhao Latex course.

- syll.
- Latex flyer.
- papers & books.

proj: website, converge.m (56)

1 wk to pick papers; discuss in OH if want; all are good.

Types & Complexity of algs.

Direct vs **Iterative**

in predictable fixed # operations gives exact ans if ops done exactly.

"ops" = $+, -, \times, \div, \sqrt{\ }, \sin$, etc.

Eg given $A \in \mathbb{R}^{n\times n}$ square, $b \in \mathbb{R}^n$: $\boxed{A}\,x = b$

solve $Ax=b$ by Gaussian elim: add lin. combos of rows to each other to set $a_{ij}=0$ below diagonal.

How long take? #els below diagonal $\approx \frac{n^2}{2}$

to add a multiple of row to another $= n$ adds, n mults $= 2n$ ops.

Total $\propto n^3$ ops. (back-substitution? $\approx 2n^2$)

Can we do better? (ask) yes: later rows have more zeros, don't do them

cost $\sim \frac{4}{3}n^3$ where $f(n) \sim g(n)$ if $\lim_{n\to\infty}\frac{f(n)}{g(n)} = 1$ asymptotic

HW: prove.

Often prefactor not crucial; we say cost $= O(n^3)$

'big O' notation: cost $\leq Cn^3$ (*) for some C for all $n \geq n_0$

'complexity' of alg. vs problem size n

Defn: $f(n) = O(g(n))$ as $n\to\infty$ if $\exists C, n_0$ st. $|f(n)| \leq Cg(n)\ \forall n \geq n_0$

Q: which stronger 'O' or '~'? $\sim \Rightarrow O$.

What about back-subst? $cn^3 + dn^2 \overset{?}{=} O(n^3)$ no matter how big const d is.

prove it: exhibit a C & n_0 st $cn^3 + dn^2 \leq Cn^3\ \forall n \geq n_0$

ie $c(1 + \frac{d}{c}n^{-1}) \leq C$

eg pick $n_0 = \frac{d}{c}$ & $C = 2c$.

The biggest power wins. Q: is $n^3 = O(n^4)$? yes

gets closer to answer but never quite reaches it. even if ops exact.

$O(n^p)$ polynomial time. vs. $O(C^n)$ $C>1$ exponential complexity (NP).

Any egs? sum a func. on $\{0,1\}^n$: 2^n work.

- Eg. Krylov space methods for $Ax=b$.

Start w/ $x_0 \in \mathbb{R}^n$, eg a random guess for x.

Alg gives $x_0 \to x_1 \to x_2 \cdots$ Each step needs mult a vector by A.

where residuals $r_k = Ax_k - b$ get smaller. $\|r_k\|_2 \to 0$ as $k\to\infty$.

defn $\|x\|_2 := \sqrt{\sum_{i=1}^{n} x_i^2}$ ℓ_2, Euclidean

You run alg. until $\|r_k\| < \varepsilon$ where ε is desired error level.

(Q: does this control error $\|x_k - x\|$? A: see later)

Cost dep. on n & ε, exact ans never reached.

Why better than direct?

Each step $O(?)$ ask: $O(n^2)$

& if A well-behaved, resid. decays fast indep. of n.

eg say $\|r_{k+1}\| \leq \frac{1}{2}\|r_k\|$

then $\|r_k\| \leq C2^{-k}$

say $C \approx 1$ & want 10^{-10} error, $k \approx 30$.

Cost $= O(n^2k) = O(n^2 \log_2 \frac{C}{\varepsilon}) = O(n^2 \log \frac{C}{\varepsilon})$

A often sparse (most els zero), so can apply to vector faster, eg $O(n)$ per step.

- Much faster than direct n^3.
- High acc. ε not always needed.

Picture: 

Thes. ② 3/25/14.

[Direct vs Iterative].

Roots of polynomial order p:
eg $p=2$ quadratic formula (exact)
3 cubic "
4 quartic "
5 ?

$p \geq 5$ no exact seq of rationals, $+, \times, -, \div, \sqrt[n]{}$ giving root (Abel 1824) Galois.
$\Rightarrow$ algorithms must be iterative.
eg. root finding.

- $Ax = \lambda x$ eigenvalues λ_j $j=1..n$.

$n=2$: $\underbrace{\begin{bmatrix} a_{11}-\lambda & a_{12} \\ a_{21} & a_{22}-\lambda \end{bmatrix}}_{\text{singular} \Rightarrow \det = 0} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$

$(a_{11}-\lambda)(a_{22}-\lambda) - a_{12}a_{21} = 0$.
2^{nd} order poly for λ roots.
general n: n^{th} order. poly
$\Rightarrow$ if can solve poly roots can solve EVP
(actually terrible way to do it).
Could there be direct alg. for $n \geq 5$ general EVP?
(that didn't go via poly roots)

Ans: no.
$\Rightarrow$ all EV. algs ($n \geq 5$) iterative.
Main method: QR iteration (1961, paper).

converse: Companion matrix actually great way to find poly roots in $\mathbb{C}$. (J.P. Boyd).

- Ans examples of direct vs iter algs? put on board.

Direct vs Iter.

- direct not always slower: conf. on fast direct solvers June 23-29.

(break, 50 mins)

Rates of convergence:

A) Eg say want calculate $f := \frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{3^2} + \cdots = \sum k^{-2}$ series.

in real life you must stop, eg truncate @ n terms: $f_k := \sum_{n=1}^{k} n^{-2}$

show convergence

k	f_k
10^3	1.643
10^6	1.64493.

* problem: exactly same (d. transistor). 10^8, 10^9.

frozen digits. — trust answer up to where frozen.
called checking convergence — in real apps not always done: hard / lazy.

Ask: $k = 10^3 \longrightarrow$ # digits = ? 3.

error $\varepsilon_k := f_k - f \overset{\text{obs.}}{\approx} n^{-1}$ "algebraic conv, order 1". — extremely slow eg $\varepsilon \approx 10^{-15}$ needs 10^{15} ops!

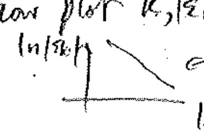
Can prove this? $|\varepsilon_k| = \sum_{n>k} n^{-2}$ ← bnd? ratio test?

$\leq \int_k^\infty n^{-2}\,dn = k^{-1}$ Lemma: $|\varepsilon_k| \leq k^{-1}$ we proved observed order & asymptotic prefactor; not usually this lucky.

Thu ③ 3/28/14

Usually enough to prove $\varepsilon_k = O(k^{-p})$ ← order

Qu: How plot $k, |\varepsilon_k|$ to best extract p?



loglog(k, eks, '+')

B) Basic iterative methods for EVP:

$A \in \mathbb{R}^{n\times n}$, $A^T = A$ ⇒ diagonalizable $A = V\Lambda V^T$ ← diag $\{\lambda_j\}_{j=1}^n$

V's cols are o.n.b. of eigenvectors $v_1, v_2, \dots$ unique up to ±.

$|\lambda_1| \geq |\lambda_2| \geq \dots$

every $\mathbb{R}^n \ni x = \sum \alpha_j v_j$

then $Ax = \sum \alpha_j A v_j = \sum (\alpha_j \lambda_j) v_j$ ← idea: coeffs of biggest λ's grow most ⇒ v_1 dominates.

Power iteration: who's heard of?

pick $x = x^{(0)}$ randomly.
for $k = 1, 2, \dots$
  $x^{(k)} = A x^{(k-1)}$
  $x^{(k)} \leftarrow x^{(k)}/\|x^{(k)}\|$ normalize.
end

each iter $O(n^2)$ if A dense

Note $x^{(k)} = c A^k x^{(0)} = c' \left(\alpha_1 \lambda_1^k v_1 + \alpha_2 \lambda_2^k v_2 + \dots\right)$

$\alpha_1 \neq 0$ ↳ $= c'' \left(v_1 + \underbrace{\frac{\alpha_2}{\alpha_1}\left(\frac{\lambda_2}{\lambda_1}\right)^k v_2 + \dots \frac{\alpha_n}{\alpha_1}\left(\frac{\lambda_n}{\lambda_1}\right)^k v_n}_{\text{bnd. in } l_2 \text{ norm. by } c\left|\frac{\lambda_2}{\lambda_1}\right|^k} \right)$

⇒ Thm let $v_1^T x^{(0)} \neq 0$, & $|\lambda_1| > |\lambda_2| \geq \dots$

then $\| x^{(k)} - (\pm v_1)\|_2 = O\left(\left|\frac{\lambda_2}{\lambda_1}\right|^k\right)$

↑ error of $x^{(k)}$ from an eigvec.

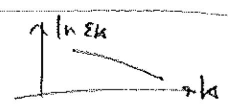
Pf: $\alpha_1 = v_1^T x^{(0)} \neq 0$.

Type of Convergence, $\varepsilon_k = O(r^k)$ for some $0 < r < 1$ called 'geometric', 'exponential', 'linear'. all used.

since $r^k = e^{-\alpha k}$ some $\alpha > 0$

Better or worse than algebraic? $r^k \stackrel{?}{=} O(k^{-p})$

· Plot?



· Note: Power Iter is a 'randomized alg' — why?
$x^{(0)}$ random — can fail but v. small prob.
see MCMC, randSVD papers

Try $p=1$ take ratio: $\lim_{k\to\infty} \frac{r^k}{k^{-1}} = \lim_{k\to\infty} \frac{k}{r^{-k}} \stackrel{\text{l'H.}}{=} \frac{1}{-\ln r}\frac{1}{r^{-k}} = c r^k \to 0$

$\frac{d}{dk} r^{-k} = \frac{d}{dk} e^{-k\ln r} = (-\ln r) r^{-k}$.

⇒ yes $r^k = O(k^{-1})$.

Also stronger $r^k = o(k^{-1})$

little-o: $f(k) = o(g(k))$ if $\lim_{k\to\infty} \frac{f(k)}{g(k)} = 0$.

Try $p=2$: $\frac{k^2}{r^{-k}} \xrightarrow{\text{l'H.}} \frac{1}{-\ln r}\frac{2k}{r^{-k}} \xrightarrow{\text{l'H.}} \frac{2!}{(-\ln r)^2} r^k \to 0$.

For each $p = 1, 2, \dots$, $r^k = o(k^{-p})$.