Todays top method for EVP: Concluding, stability.

Last time: direct vs iterative complexity
convergence rate of error to zero vs k (iters, effort):
- direct: $K^{-T}$ often algebraic order $\mu, r = O(1)$
- iterative: geometric / exp / linear: $1/r_k$ decreases faster?

Recall: power iteration: $x^{(k)} \rightarrow v_1$, w/ exp. rate $r = |\lambda_2|/|\lambda_1|$

Coral's power iter. is catastrophically slow.

How inc. rate? get other evecs? $v_m$?

$A = V \Lambda V^T$

let $\mu \in \mathbb{R}$

why $$(A-\mu I)^{-1} = V \sum_j \frac{1}{\lambda_j - \mu} \Lambda_j^{-1} V^T$$

Inverse iteration:
Let $\mu$ be estimate for $\lambda_m$, $x^{(0)}$ and $e \in \mathbb{R}^n$

for $k = 1, 2, \ldots$

solve $(A-\mu I)w = x^{(k-1)}$
for $w \in \mathbb{R}^n$
$x^{(k)} = w / \|w\|_2$

End:

applies power iteration to $(A-\mu I)^{-1}$

Convergence: let $\lambda_m$ be closest to $\mu$, $A_k$ $2^{th}$-closest, then reuse them on power iter.,

Then: $\|x^{(k)} - (A_m - \mu) v_m \| = O\left(\frac{\lambda_m - \mu}{\lambda_3 - \mu}\right)$

Code: power iter. $\mu = \lambda_m$, $\lambda_3$, $\bar{\lambda}$

Estimating $\lambda_m$ from $x$, an estimate for $v_m$?

Rayleigh quotient: $R(x) = \frac{x^T Ax}{x^T x}$

why good?

$x = \sum_j \alpha_j v_j$

$x^T x = \sum_j \alpha_j^2 v_j v_j^T$

$\sum_j = \sum_i \alpha_i^2 \

x^T A x = \sum_j \alpha_j v_j (\sum_i \alpha_i v_i^T) v_j = \sum_j \alpha_j^2$

Rayleigh quotient: $R(x) = \frac{\sum \alpha_j^2}{\sum \alpha_j^2} \approx \lambda_m$ if $\alpha_j$ large

$\lambda_m$ small, $\alpha_j$
Thus, say \( \|x - v_m\| = \varepsilon \) then \( R(x) - \lambda_m = O(\varepsilon^2) \) as \( \varepsilon \to 0 \).

pf 2: \( R(x) \) stationary at \( x = v_m \) i.e. \( \nabla R = 0 \) (HW-check), and \( R(v_m) = \lambda_m \).

since \( R(x) \) smooth for \( x \neq 0 \), holds by calculus.

pf 1: \( \|z_{x_i} v_i - v_m\|^2 = \varepsilon^2 \)

since o.m.b: \( (\mu_i-1) + \sum_{j \neq i} \chi_j^2 \leq \varepsilon^2 \) so \( |\mu_i-1| \leq \varepsilon \)

\( R(\varepsilon) = \frac{\lambda_m + \sum_{j \neq i} \chi_j^2}{1 + \sum_{j \neq i} \chi_j^2} \leq \frac{\lambda_m + \mu_i^2}{1 + \mu_i^2} \) worst case

Sim for lower bound.

\[ \lambda_m, (1 + O(\varepsilon)), \text{ using eq. } \frac{1}{1 - \varepsilon} = 1 + O(\varepsilon), \text{ etc.} \]

asymptotics.

\( \lambda_{j+1} = R(j) \) in inverse iter code: show many more digits \( \lambda_{j+1} \) converge than \( x_{j+1} \)? finally.

Any ideas how to improve inverse iter? use \( p = R(x_{j+1}) = \lambda_{j+1} \) best approx eig.

Ray: Quick Iter: \[ \begin{array}{ccc} \text{inv. update } \lambda & \Rightarrow \text{Ray. update } \lambda \end{array} \]

Thus: \( \forall x_0 \) except set minus, e.g., conv. obey: \( z_{k+1} = O(\varepsilon^2) \)

where \( z_k := (1 \times (x_k) - (\pm) v_m, \|z_k\|_2 \)

\( \text{or: } |z_k - \lambda| = \frac{\|z_k\|_2}{\lambda} \) similarly as \( k \) const in \( O(\cdot) \) uniform for \( k \) suff. large.

Each \( x^2 \) converges: means \( \varepsilon_k \leq C \varepsilon_0^3 \leq C \varepsilon_0^q \).

Can ask faster than exp. \( e^{-k} \)?

If yes, then \( x \) converges faster: \( z_{k+1} = O(\varepsilon^2) \) e.g. Newton's method fast for roots.

If sketch, say \( (x^{(k)}) - \pm v_m \| = \varepsilon \),

then \( |\hat{\lambda}^{(k)} - \lambda_m| = O(\varepsilon^2) \)

so \( |\hat{\lambda}^{(k)} - \lambda| = O(\varepsilon^2) \) so dispersion w.r.t. initial error by this

\[ \|\lambda^{(k)} - \pm v_m \| = O(\varepsilon^2) \]

\( \lambda^{(k)}, x^{(k)} \) converge at same rate.

Code shows pt III: note: \( O(n^3) \) per step, but only \( n^2 \) needed. \( \Rightarrow QR \) iteration.

Next problem: builds on this.

\( \varepsilon \) small; \( \lambda_{j+1} \) large.
Condition # of problem

Numerical problem is map \( f: X \to Y \)

*Example* \( f(x) = \tan x \) eval. some fun.

*Example* \( f(x_0, \ldots, x_3) = \sum_{i=0}^{3} x_i \)

Poly coeff., root of poly.

If well-cond., if infinitesimal pert. \( \delta x \) in \( X \) causes "small" pert. \( \delta f = f(x+\delta x) - f(x) \)

Defn Abs. cond. # \( K = K(x) := \lim_{\delta x \to 0} \sup_{\|\delta x\| \leq \delta} \| f(\delta x) \| = \sup_{\|\delta x\| \leq \delta} \| f(\delta x) \| \)

"norm of Frechet derivative" "Lipschitz cond. at point x." eg 2-norms.

if \( x \in C^0 \), \( f(\delta x) \in C^0 \) vector, \( \delta f \frac{\partial f}{\partial x} = J_{\delta x}(x) \) matrix \( J \in C^{\infty} \), Jacobian.

If \( f \) smooth then \( \delta f = J \delta x \) as \( \| \delta x \| \to 0 \).

Multiply by gain matrix \( J \), what is largest factor by which length of vector can grow?

"max growth rate" = \( \| J \|_2 \) or \( \| J \| \) 2-norm of matrix = \( \sup_{0 \not= x \in C^n} \| Jx \| / \| x \| \)

Properties of \( A \): say \( A = \text{diag}(a_1, \ldots, a_n) \) then \( \| A \| = \max_{i \leq n} |a_i| \)

Say \( A = uv^T \) \( u, v \in C^n \), \( v \neq 0 \) rank-1 matrix \( \| A \| = \| v^T \| \| u \| \) \( \leq \| u \| \| v \| \) by what? 

So \( \| A \| = \| u \| \| v \| \)

More useful: Relative cond. # \( K := \sup_{\delta x} \| \frac{\delta f}{\delta x} / f(x) \| = \| J(x) \| / \| f(x) \| \)

Ref. changes in input/output. Important since computer introduces relative errors.

Say \( K \leq 10^{-3} \) well-cond.

otherwise ill-cond. \( \Rightarrow 10^3 \).

\( K \) is property of problem *not* alg. used to solve it.