Iterative Solvers

Last Time Eigenvalue calculations

1. Power method:

$$A^k\vec{w} \approx \lambda_1^k \vec{v}_1 \qquad , |\lambda_1| > |\lambda_2| > \ldots$$

converges at a rate $\sim O\left(\left|\frac{\lambda_2}{\lambda_1}\right|\right)$

2. To increase convergence rate: shift

$A - sI$ has eigenvalues equal to $\lambda_1 - s$, $\lambda_2 - s \ldots$

pick s to increase convergence rate.

Ex: $\lambda_j = 1, 2, \ldots 8, 9, 10$

$\lambda_1 = 10$, $\lambda_2 = 9$

Power method converges at a rate of $\left(\frac{9}{10}\right)^k$.

Shift by $s = 1$, $\lambda_1 = 9$, $\lambda_2 = 8 \ldots$ $\lambda_{10} = 0$

$$\text{rate} = \left(\frac{8}{9}\right)^k < \left(\frac{9}{10}\right)^k$$

Best shift: shift such that $|\lambda_2| = |\lambda_{10}|$

$s = 5 \Rightarrow$

$$\lambda_1 = 5$$
$$\lambda_2 = 4 \qquad \text{rate} = \left(\frac{4}{5}\right)^k = .8^k$$
$$\vdots$$
$$\lambda_{10} = -4$$

(Eigenvectors did not change)

3 To find interior eigenvalues, the <u>inverse iteration with shift</u> is required:

$$(A-sI)^{-1} \text{ has eigenvalues } \frac{1}{\lambda_1 - s}, \dots, \frac{1}{\lambda_{10} - s}.$$

<u>Ex:</u> for $s = 3.2$, $\tilde{\lambda}_j$ are

$$\tilde{\lambda}_1 = \frac{1}{10-3.2}, \quad \tilde{\lambda}_2 = \frac{1}{9-3.2} \quad \dots \quad \tilde{\lambda}_{10} = \frac{1}{1-3.2}$$

The <u>largest</u> absolute value $\tilde{\lambda}$ is $\tilde{\lambda}_8 = \frac{1}{3-3.2} = \frac{1}{-.2} = -5$.

$$\tilde{\lambda}_7 = \frac{1}{.8} = 1.25 \qquad \text{convergence rate: } O\left(\left(\frac{1.25}{5}\right)^k\right).$$

$$\tilde{\lambda}_9 = \frac{1}{-1.2} = -\frac{5}{6}$$

$\Rightarrow$ $\tilde{\lambda}_j$'s can be estimated with <u>power method</u> + <u>deflation</u> ...

How do we compute $\left((A-sI)^{-1}\right)^{k+1} \vec{w}^0$?

$\Rightarrow$ Solve $\quad (A - sI)\,\vec{w}^{(k+1)} = \vec{w}^{(k)}$

Intro: Iterative solver.

cost of Gaussian elimination : $O(n^3)$

" " matrix-vector-mult: $O(n^2)$ $\leftarrow$ maybe we can iterate <u>k</u> times $\Rightarrow O(kn^2)$.

Another reason we want iterative solvers: storage.

$$A = \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}$$

storage cost for A: $O(3n)$

A^{-1} is dense! (check in Matlab for yourself)

storage: $O(n^2)$.

If A is $n \times n$, and contains $O(n)$ or $O(n \log n)$ entries, we say that it is sparse.

Simple Iteration

Suppose that you have a matrix M such that $M^{-1}A \approx I$ and $M\vec{x} = \vec{y}$ is easy to solve.

Then: To solve $A\vec{x} = \vec{b}$:

Start with guess $\vec{x}_0$

Compute residual $\vec{r}_0 = \vec{b} - A\vec{x}$

Solve $M\vec{z}_0 = \vec{r}_0$, then $\vec{z}_0 = M^{-1}(\vec{b} - A\vec{x})$

$$\approx M^{-1}\vec{b} - \vec{x}$$

$$\approx \vec{e}_0 = A^{-1}\vec{b} - \vec{x}$$

set $\vec{x}_1 = \vec{x}_0 + \vec{z}_0$

And repeat:

$k = 1, 2, \ldots$

Set $\vec{x}_k = \vec{x}_{k-1} + \vec{z}_{k-1}$

Compute $\vec{r}_k = \vec{b} - A\vec{x}_k$

<u>If</u> $\|\vec{r}_k\|$ is small, stop

<u>Else</u> solve $M\vec{z}_k = \vec{r}_k$

M is referred to as a <u>left preconditioner</u>.

How do we choose M?

<u>Ex:</u> $\underbrace{M = \text{Lower triangle of } A.}_{\text{Easy to solve, } O(n^2) \text{ flops by FWD substitution}}$ (Gauss-Seidel iterations)

<u>Convergence of Iterative Methods</u>

Consider the matrix splitting: $A = M - N$.

$$A\vec{x} = \vec{b}$$

$$(M-N)\vec{x} = \vec{b}$$

$$\vec{x} = M^{-1}N\vec{x} + M^{-1}\vec{b}$$ ← similar to fixed point iteration with $\vec{g}(\vec{x}) = M^{-1}N\vec{x} + M^{-1}\vec{b}$

This is, in fact, equivalent to the earlier iteration method:

$$\vec{x}_k = \vec{x}_{k-1} + \vec{z}_{k-1} \qquad \left(M\vec{z}_k = \vec{r}_k = \vec{b} - A\vec{x}_k \right)$$

$$= \vec{x}_{k-1} + M^{-1}(\vec{b} - A\vec{x}_{k-1})$$

$$= (I - M^{-1}A)\vec{x}_{k-1} + M^{-1}\vec{b}.$$

and since $A = M - N$

$$\underbrace{M^{-1}N}_{\text{same as before}} = I - M^{-1}A = M^{-1}\underbrace{(M - A)}_{N}.$$

So when does $\vec{x}_k = (I - M^{-1}A)\vec{x}_{k-1}$ converge?

At step k, the error is $\vec{e}_k = A^{-1}\vec{b} - \vec{x}_k$

$$\vec{x}_k = \vec{x}_{k-1} + \vec{z}_{k-1}$$

$$\underbrace{-\vec{x}_k + A^{-1}\vec{b}}_{\vec{e}_k} = \underbrace{-\vec{x}_{k-1} + A^{-1}\vec{b}}_{\vec{e}_{k-1}} - \underbrace{\vec{z}_{k-1}}_{M^{-1}\vec{r}_{k-1}}$$

$$\vec{r}_k = \vec{b} - A\vec{x}_k = A(A^{-1}\vec{b} - \vec{x}_k) = A\vec{e}_k$$

$$\vec{e}_k = \vec{e}_{k-1} - M^{-1}\vec{r}_{k-1}$$

$$= \vec{e}_{k-1} - M^{-1}A\vec{e}_{k-1}$$

$$= (I - M^{-1}A)\vec{e}_{k-1}$$

This implies that

$$\vec{e}_k = (I - M^{-1}A)^k \, \vec{e}_0$$

$$\Rightarrow \|\vec{e}_k\| \le \|(I - M^{-1}A)^k\| \, \|\vec{e}_0\|$$

any vector norm and the induced matrix norm.

<u>Theorem:</u> $\|\vec{\epsilon}_k\| \to 0$ and $\vec{x}_k \to \vec{x}$ if and only if

$$\lim_{k \to \infty} \|(I - M^{-1}A)^k\| = 0$$

for <u>every</u> initial $\vec{x}_0$

(Unsurprising)

<u>Example</u> Recall $\|A\|_2$.

$$\|A\|_2 = \sqrt{\max_\lambda (A^TA)}$$

$$\Rightarrow \|A\|_2 = \max |\lambda_j|$$

Therefor we need the eigenvalues of $I - M^{-1}A$ to be < 1 in absolute value.

<u>Definition</u> Spectral radius:

For an $n \times n$ matrix G,

$$\rho(G) = \max_j |\lambda_j|$$

<u>Theorem</u>: Let G be $n \times n$ matrix. Then $\lim_{k\to\infty} G^k = 0$ if and only if $\rho(G) < 1$.

<u>Idea</u>: If $G = VDV^{-1}$, then $G^k = V D^k V^{-1}$

$D^k \to 0$ iff elements < 1.

$\Rightarrow$ for convergence, we need $\rho(I - M^{-1}A) < 1$.

(and if $\|(I - M^{-1}A)^k\| \to 0$, then $\rho(I - M^{-1}A) < 1$).

<u>Special case: symmetric positive definite matrix</u>

No preconditioner, consider only multiply of A, and the simple iteration:

$$\vec{x}_0, \quad \vec{r}_0 = \vec{b} - A\vec{x}_0 \quad (= A\vec{e}_0)$$

$$\vec{x}_1 = \vec{x}_0 + c_0 \vec{r}_0$$

$$= \vec{x}_0 + c_0(\vec{b} - A\vec{x}_0), \quad \vec{r}_1 = \vec{b} - A\vec{x}_1$$

$$\vec{x}_2 = \vec{x}_1 + c_1 \vec{r}_1$$

$$= \vec{x}_1 + c_1(\vec{b} - A\vec{x}_1)$$

$$= \vec{x}_1 + c_1(\vec{b} - A\vec{x}_0 - c_0 A\vec{r}_0)$$

Conjugate Gradient for RSPD matrices

Def:

$\vec{u}, \vec{v}$ are conjugate w.r.t. to A if

$$(\vec{u}, A\vec{v}) = 0 \qquad (\text{the } A\text{-inner product})$$

Note $(\vec{u}, A\vec{v})$ is only an inner product if A is symmetric and positive definite.

Let $\{\vec{p}_1, \dots, \vec{p}_n\}$ be a set of n mutually A-conjugate vectors. Then they form a basis for $\mathbb{R}^n$ (if A is invertible).

Any $\vec{x}$ can be written as

$$\vec{x} = \sum \alpha_j \vec{p}_j .$$

to solve $A\vec{x} = \vec{b}$:

Therefore:

$$A\vec{x} = \sum \alpha_j A\vec{p}_j =$$

$$(\vec{p}_\ell, A\vec{x}) = \sum \alpha_j (p_\ell, A p_j) = (\vec{p}_\ell, \vec{b})$$

$$= \alpha_\ell (\vec{p}_\ell, A\vec{p}_\ell) = (\vec{p}_\ell, \vec{b})$$

$$\Rightarrow \alpha_\ell = \frac{(p_\ell, \vec{b})}{\|\vec{p}_\ell\|_A} .$$

Direct solution.

To do this iteratively, is there a way to select $\vec{p}_1, \dots, \vec{p}_n$ so that only a few inner products need to be calculated? (as an approximation to $\vec{x}$ s.t. $\vec{r} = \vec{b} - A\vec{x}$ is small).

<u>Observation</u>: The solution to $A\vec{x} = \vec{b}$ is <u>also</u> the <u>unique</u> minimizer of the function:

$$f(\vec{x}) = \frac{1}{2}(\vec{x}, A\vec{x}) - (\vec{x}, \vec{b})$$

(I.e. $\frac{\partial f}{\partial x_j} = 0$ when $A\vec{x} = \vec{b}$).

<u>Newton</u> would suggest moving in the direction of $-\nabla f(\vec{x}_0)$.

minimize

$$\nabla f(\vec{x}) = A\vec{x} - \vec{b}.$$

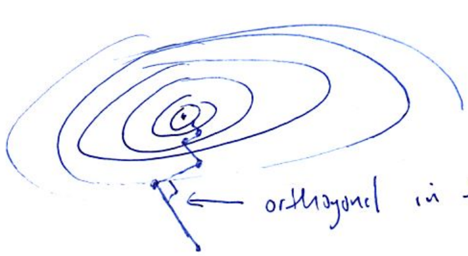
Let $\vec{p}_0 = \vec{b} - A\vec{x} = \vec{r}_0 = -\nabla f(\vec{x}_0)$.

We want the following vectors $\vec{p}_1, \vec{p}_2, \dots$ to be A-conjugate:

Let $\vec{r}_k = b - A\vec{x}_k = -\nabla f(\vec{x}_k)$

$$\Rightarrow \vec{p}_k = \vec{r}_k - \sum_{l<k} \frac{(\vec{p}_l, A\vec{r}_k)}{(\vec{p}_l, A\vec{p}_l)}$$

← "A-inner product Gram-Schmidt"



← orthogonal in the A-inner product.

In the direction of $\vec{p}_k$, the next iterate is:

$$\vec{x}_{k+1} = \vec{x}_k + \alpha_k \vec{p}_k$$

with $$\alpha_k = \frac{(p_k, \vec{b})}{(\vec{p}_k, A\vec{p}_k)} = \frac{(p_k, \vec{r}_{k-1} + A\vec{x}_{k-1})}{(\vec{p}_k, A\vec{p}_k)} = \frac{(\vec{p}_k, \vec{r}_{k-1})}{(\vec{p}_k, A\vec{p}_k)}$$

since $(\vec{p}_k, A\vec{x}_{k-1})$ are A-conjugate.

Convergence Rate

Thm Let $\vec{e}_k$ be the error on step k of CG. Then

$$\frac{\|\vec{e}_k\|_A}{\|\vec{e}_0\|_A} \le 2\left(\frac{\sqrt{K}-1}{\sqrt{K}+1}\right)^k \qquad K = \frac{\lambda_{max}}{\lambda_{min}} = \text{condition number}$$

(for RSPD matrices)

Conjugate Gradients is a Krylov Method

It minimizes some norm of $\|\vec{r}_k\|$ ← in fact the A-norm.
using $\{\vec{r}_0, A\vec{r}_0, A^2\vec{r}_0, \dots\}$

For nonsymmetric, arbitrary systems $A\vec{x} = \vec{b}$, the GMRES algorithm minimizes the 2 norm of $\|\vec{r}_k\|$.

Idea: solve $A\vec{x} = \vec{b}$ by choosing

an approximate iterate $\vec{x}_k$ that is of

the form $\vec{x}_0 + \{\vec{r}_0, A\vec{r}_0, \dots A^{k-1}\vec{r}_0\}$ to

minimize

$$\vec{r}_k = \vec{r}_0 - \sum_{1}^{k} c_0 A^j \vec{r}_0$$ in the <u>2-norm</u>.

(Use Gram-Schmidt!) and an orthogonal basis

for $\{\vec{r}_0, A\vec{r}_0, A^2\vec{r}_0, \dots\}$