University of Idaho

Lecture 39

Last time we showed:

$$\nabla R_A(x) = \frac{2}{x^T x}\left((Ax)^T - R_A(x)\cdot x^T\right)$$

Then

$$\nabla R_A(q_j) = \frac{2}{\underbrace{q_j^T q_j}_{=1}}\left((A q_j)^T - R_A(q_j)\cdot q_j^T\right) = 2\left((\lambda_j q_j)^T - \lambda_j \cdot q_j^T\right) = 0$$

1 since $\{q_j\}$ are orthonormal

$$\Rightarrow R_A(x) = \lambda_j + O(\|x - q_j\|^2):$$ <u>quadratic approximation</u>

Section 4.1: <u>Power method</u>

idea: $v,\ Av,\ A^2v^2,\ \dots$

algorithm

1. $v^{(0)}$: given, $\|v^{(0)}\|_2 = 1$
2. for $k = 1, 2, \ldots$
3. $w = A v^{(k-1)}$ % apply matrix A
   % if A is sparse, this can be
   % done efficiently
4. $v^{(k)} = w / \|w\|_2$ % normalize
   % this is done to avoid
   % overflow/underflow
5. $\lambda^{(k)} = (v^{(k)})^T A v^{(k)}$ % apply Rayleigh quotient
   % $\lambda^{(k)} = \lambda_1 + O(\|v^{(k)} - (\pm q_1)\|^2)$:
   more soon

Note Suppose that $A = A_h$, where

$$(A_h v)_i = -D_+ D_- v_i = \frac{1}{h^2}(-v_{i-1} + 2v_i - v_{i+1})$$

assuming $h = \frac{1}{n+1}$, $v_0 = 0 = v_{n+1}$. Then line 3, $w = Av$, can be coded as a loop:

```
for i=1:n;
    w_i = (-v_{i-1} + 2 v_i - v_{i+1})
end
```

This is more efficient than forming A_h and computing $w = A_h v$ by matrix-vector multiplication.

<u>Thm</u> Assume that $|\lambda_1| > |\lambda_2| > \ldots > |\lambda_n|$ and $q_1^T v_{\cancel{0}}^{(0)} \neq 0$.

Then $\| v^{(k)} - (\pm q_1) \| = O\left(\left| \frac{\lambda_2}{\lambda_1} \right|^k \right)$, $\quad |\lambda^{(k)} - \lambda_1| = O\left(\left| \frac{\lambda_2}{\lambda_1} \right|^{2k} \right)$

The $\pm$ depends on the sign of λ_1.

<u>Pf</u>

$v^{(0)} = \alpha_1 q_1 + \alpha_2 q_2 + \ldots + \alpha_n q_n$, where $\alpha_i = q_i^T v^{(0)}$

$v^{(k)} = \beta_k A^k v^{(0)} = \beta_k \left(\alpha_1 A^k q_1 + \alpha_2 A^k q_2 + \ldots + \alpha_n A^k q_n \right)$

$$= \beta_k \left(\alpha_1 \lambda_1^k q_1 + \alpha_2 \lambda_2^k q_2 + \dots + \alpha_n \lambda_n^k q_n \right) =$$

$$= \beta_k \lambda_1^k \left(\alpha_1 q_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1}\right)^k q_2 + \dots + \alpha_n \left(\frac{\lambda_n}{\lambda_1}\right)^k q_n \right)$$

$\Rightarrow v^{(k)} \sim \pm q_1$ as $k \to \infty$

if $\lambda_1 > 0 \Rightarrow$ all signs are + or all −

if $\lambda_1 < 0 \Rightarrow$ signs alternate

If $\underbrace{q_1^T v^{(0)}}_{\alpha_1} = 0$, then the scheme converges to $\lambda_2, \pm q_2$.

or

<u>Note</u> The power method has some limitations.

1. it only gives the larges e'value λ_1
2. $v^{(k)}, \lambda^{(k)}$ converges linearly and the convergence factor $\left|\frac{\lambda_2}{\lambda_1}\right|$ may not be small

Recall: linear convergence means

$$\| v^{(k)} - (\pm q_1) \| \leq C \, \| v^{(k-1)} - (\pm q_1) \|$$

Section 4.2 : inverse iteration

idea : apply power method to A^{-1}, $(A-\mu I)^{-1}$, μ: shift

1. $A q_i = \lambda_i q_i \Rightarrow A^{-1} q_i = \lambda_i^{-1} q_i$

The largest e'value of A^{-1} is λ_n^{-1}, so the vectors $v^{(k)}$ converge to $\pm q_n$.

2. $(A - \mu I) q_i = (\lambda_i - \mu) q_i \Rightarrow (A-\mu I)^{-1} q_i = (\lambda_i - \mu)^{-1} q_i$

The largest e'value of $(A-\mu I)^{-1}$ is $|\lambda_J - \mu|^{-1}$, where λ_J is the e'value of A closest to μ, so the vectors $v^{(k)}$ converge to $\pm q_J$.

3. $w = A^{-1} v \Rightarrow A w = v$

$$w = (A-\mu I)^{-1} v \Rightarrow (A-\mu I) w = v$$

<u>algorithm</u>

1. $v^{(0)}$: given, $\|v^{(0)}\|_2 = 1$
2. for $k = 1, 2, \ldots$
3. solve $(A-\mu I) w = v^{(k-1)}$; % apply $(A-\mu I)^{-1}$
   % e.g. LU factorization
   % etc.
4. $v^{(k)} = w / \|w\|_2$; % normalize
5. $\lambda^{(k)} = (v^{(k)})^T A v^{(k)}$; % Rayleigh quotient
   % why not $(A-\mu I)^{-1}$?

Thm Assume that λ_J is the e'value of A closest to μ and λ_K is the next closest, i.e.

$$|\lambda_J - \mu| < |\lambda_K - \mu| < |\lambda_i - \mu| \quad \text{for } i \neq K, J \text{ and}$$
$$q_J^T v^{(0)} \neq 0$$

Then

$$\| v^{(k)} - (\pm q_J) \| = O\left(\left| \frac{\lambda_J - \mu}{\lambda_K - \mu} \right|^k \right), \quad | \lambda^{(k)} - \lambda_J | = O\left(\left| \frac{\lambda_J - \mu}{\lambda_K - \mu} \right|^{2k} \right)$$

<u>Note</u> convergence is linear as in power method.

<u>Pf</u> as before, $\lambda_1 \to \frac{1}{\lambda_J - \mu}$, $\lambda_2 \to \frac{1}{\lambda_K - \mu}$

$$\Rightarrow \left| \frac{\lambda_2}{\lambda_1} \right| \to \left| \frac{\lambda_J - \mu}{\lambda_K - \mu} \right| \quad \text{ok}$$