Power Method

And other related eigenvalue computing methods

Outline

- Basic ideas of power method
- The power method
- The inverse power method
- The shifted power method
- Rayleigh quotient
- Rayleigh quotient iteration method

Basic Ideas

- The target matrices: matrix ***A*** is symmetric and its **eigenvalues** can be ordered as
  $|\lambda_0| > |\lambda_1| \geq |\lambda_2 \geq \cdots \geq |\lambda_{n-2}| > |\lambda_{n-1}|.$
  The correspondent **eigenvectors** are
  $\{x_o, x_1, \dots, x_{n-1}\}.$
- Basic ideas
  - The eigenvectors form a basis and any vector $y^{(0)}$ can be expressed as

$$y^{(0)} = c_0 x_0 + c_1 x_1 + \cdots + c_{n-1} x_{n-1} = \sum c_i x_i.$$

Basic Ideas

- Multiplying $y^{(0)}$ by $\boldsymbol{A}$, we have

  $Ay^{(0)} = \sum c_i \lambda_i x_i$,

  – we should select $y^{(0)}$ with care such that $c_0 \neq 0$.
- Repeatedly, do we multiply $y^{(0)}$ with $\boldsymbol{A}$,

  $y^{(k)} = A^k y^{(0)} = \sum c_i {\lambda_i}^k x_i$ //See the basic properties of eigenvalue.
- Factoring λ_0 from the equation,

  $y^{(k)} = {\lambda_0}^k \sum c_i (\frac{\lambda_i}{\lambda_0})^k x_i$.

  $|\lambda_0| > |\lambda_1| \geq |\lambda_2 \geq \cdots \geq |\lambda_{n-2}| > |\lambda_{n-1}|$.

  $k \to \infty, (\frac{\lambda_i}{\lambda_0})^k \approx 0, i \neq 0.$

  $y^{(k)} = c_0 {\lambda_0}^k x_0$.

- Thus, we obtain the **eigenvector** x_0.

Problems and Solutions

- **Two problems**:
  - We will encounter overflow or underflow in computing $y^{(k)}$ unless the eigenvalue is 1.

    $y^{(k)} = c_0 {\lambda_0}^k x_0.$
  - How to factor out λ_0 ?
- Solution 1: normalizing y after each iteration

  $y^{(k)} = \frac{Ay^{(k-1)}}{\|Ay^{(k-1)}\|}.$
  - In the next iteration::

  $y^{(k+1)} = Ay^{(k)},$

  $\|y^{(k+1)}\| \leq \|A\| \cdot \|y^{(k)}\| = \|A\| \leq |\lambda_0|.$

- As the computation converges, we treat $y^{(k)}$ as the eigenvector.
  - We will normalize it. (Why? See the following slides.)
- How can we compute the eigenvalue λ_0 ?

Problems and Solutions

- As the computation converges, $y^{(k+1)} = Ay^{(k)} = \lambda y^{(k)}$.

- We normalized $y^{(k)}$ in the previous iteration.
- Solution 2: Before normalizing $y^{(k+1)}$, we perform the following computation:
  $< y^{(k)}, y^{(k+1)} > = < y^{(k)}, Ay^{(k)} >$
  $= < y^{(k)}, \lambda y^{(k)} > = \lambda < y^{(k)}, y^{(k)} > = \lambda.$
  //as the computation is nearly converged.
- Thus, in each iteration, we improve not only the eigenvector but also the eigenvalue.

The Power Method

$select\ y \neq 0;$

$x = A * y;$

$repeat\{$

$\quad y = \frac{x}{\|x\|};$ //Normalization $y^{(k)}$

$\quad x = A * y;$ //Un-normalized $y^{(k+1)}$

$\quad \lambda = y^T x;$ //Approximation of the eigenvalue.

$\quad r = \lambda * y - x;$ //$r^{(k)} = \lambda^{(k)} y^{(k)} - A y^{(k)}$

$\}until(\|r\| \leq \varepsilon);$

Time Complexity

- In each iteration, we need $O(n^2)$ steps. //Matrix-vector multiplication
- If we can ignore the round-off errors, how many iterations are required?
  - It depends on the ratio of $t = \left|\frac{\lambda_1}{\lambda_0}\right| < 1$.

  Assume that we need k iterations to ensure $t^k \leq \varepsilon$

  Then $k \geq \frac{\log\frac{1}{\varepsilon}}{\log(|\frac{\lambda_0}{\lambda_1}|)} = \log_{|\frac{\lambda_0}{\lambda_1}|}\frac{1}{\varepsilon}$.

  - It may be slow if $t \sim 1.0$.

- Conclusion: the convergence rate is linearly related to the base, $t = |\frac{\lambda_1}{\lambda_0}|$.

Inverse Power Method

- Matrix ***A*** is symmetric and its eigen pairs can be ordered as
  - Eigenvalues: $|\lambda_0| > |\lambda_1| \geq |\lambda_2 \geq \cdots \geq |\lambda_{n-2}| > |\lambda_{n-1}|$.
  - Eigenvectors: $\{x_o, x_1, \ldots, x_{n-1}\}$.

- Considering ***A***$^{-1}$, its eigenvalues are:
  $|\frac{1}{\lambda_{n-1}}| > \left|\frac{1}{\lambda_{n-2}}\right| \geq |\frac{1}{\lambda_{n-3}} \geq \cdots \geq \left|\frac{1}{\lambda_1}\right| > |\frac{1}{\lambda_0}|$.

- If we compute the largest eigenvalue of ***A***$^{-1}$, then the minimum eigenvalue of ***A*** is computed too.
  - Just taking the reciprocal.

Basic Algorithm

- Naïve method:
  - Compute $B = A^{-1}$;
  - Calculate the largest eigenvalue of ***B*** by using the power method;
  - Invert this eigenvalue;

- Problem
  - The matrix inverse ***B*** need O(n^3) steps to obtain.
  - The round-off errors cause loss-of-significant digits as *n* becomes larger. → Incorrect ***B***.

Improvement (1)

- Instead of computing the inverse matrix, the computation is replaced as follows.

  Instead of computing $y^{(k+1)} = A^{-1} * y^{(k)}$, we solve $Ay^{(k+1)} = y^{(k)}$.

- Problem:
  - Solving the linear system requires $O(n^3)$ steps in each iteration.
  - Too slow.
  - Too many round-off errors.

Improvement (2)

- Decomposing $\boldsymbol{A}$ into $A = L*U$ by using an LU decomposition method.
  - Perform Doolittle's method on a very accurate computer.
- Replace the linear system $Ay^{(k+1)} = y^{(k)}$ by $L\left(Uy^{(k+1)}\right) = y^{(k)}$.
- Let $h = Uy^{(k+1)}$, take the following steps to compute $y^{(k+1)}$:
  - Solve $L * h = y^{(k)}$; //Forward substitution
  - Solve $U * y^{(k+1)} = h$; //Backward substitution

The Inverse Power Method

$decompose\ A\ into\ L\ and\ U$; //Doolittle's method
$select\ y \neq 0$; //Initial $y^{(0)}$
$solve\ L * h = y$; //Using forward substitution.
$solve\ U * x = h$; //Using backward substitution, un-normalized $y^{(1)}$.
$repeat\{$
$\quad y = \frac{x}{\|x\|}$; //Normalization $y^{(k)}$
$\quad solve\ L * h = y$; //Using forward substitution.
$\quad solve\ U * x = h$; //Using backward substitution, un-normalized $y^{(k+1)}$
$\quad \lambda = \frac{y^T x}{y^T y}$; //Approximation of the eigenvalue.
$\quad r = \lambda * y - x$; //$r^{(k)} = \lambda^{(k)} y^{(k)} - A y^{(k)}$
$\}until(\|r\| \leq \varepsilon)$;
$return \left(\frac{1}{\lambda}\right)$;

Shift Inverse Power Method

- The inverse power method can be used to compute other eigenvalues.

- Theorem 1: $B = (A - \mu I)$, then the eigenvalues of $\boldsymbol{B}$ are $\{(\lambda_0 - \mu), (\lambda_1 - \mu), \dots, (\lambda_{n-1} - \mu)\}$

- Theorem 2: The eigenvectors of ***A*** and ***B*** are the same.

- Proof of Theorems 1 & 2:
  - Let x and λ be an eigenvalue and an eigenvector of $\boldsymbol{A}$.
  - Then,

  $Bx = (A - \mu I)x = Ax - \mu Ix$
  $= \lambda x - \mu x = (\lambda - \mu)x.$

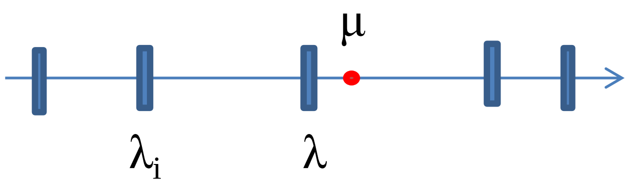
  - By definition, x and $(\lambda\text{-}\mu)$ are an eigenvector and an eigenvalue of $\boldsymbol{B}$.

Shift Inverse Power Method

- Compute $\boldsymbol{B} = \boldsymbol{A} - \mu \boldsymbol{I}$.
- Compute $\boldsymbol{C} = \boldsymbol{B}^{-1}$.
- If λ is an eigenvalue of $\boldsymbol{A}$, then $\frac{1}{\lambda-\mu}$ is an eigenvalue of $\boldsymbol{C}$.
- If μ is very lose to λ, then $\left|\frac{1}{\lambda-\mu}\right|$ will be large, compared with other eigenvalues of $\boldsymbol{C}$.
- If we have a good approximation of λ, say μ, then we can use this shifted inverse power method to compute λ.

$$t = \frac{1}{\lambda-\mu}, \lambda = \frac{1}{t} + \mu.$$

- Principle of the shift inverse power method



$$\frac{1}{|\lambda - \mu|} \gg \frac{1}{|\lambda_i - \mu|}$$

Shift Inverse Power Method

$B = A - \mu I;$ //μ is given by the user.

$decompose\ B\ into\ L\ and\ U$; //Doolittle's method

$select\ y \neq 0$; //Initial $y^{(0)}$

$solve\ L * h = y$; //Using forward substitution.

$solve\ U * x = h$; //Using backward substitution, un-normalized $y^{(1)}$.

$repeat\{$

$y = \frac{x}{\|x\|}$; //Normalization $y^{(k)}$

$solve\ L * h = y$; //Using forward substitution.

$solve\ U * x = h$; //Using backward substitution, un-normalized $y^{(k+1)}$

$\rho = \frac{y^T x}{y^T y}$; //Approximation of the eigenvalue.

$r = \rho * y - x$; //$r^{(k)} = \rho^{(k)} y^{(k)} - B^{-1} y^{(k)}$

$\}until(\|r\| \leq \varepsilon)$;

$return\left(\frac{1}{\rho} + \mu, y\right)$; //$\rho = \frac{1}{\lambda - \mu}, \lambda = \frac{1}{\rho} + \mu$

Rayleigh Quotient

- Definition: Assume x is a vector and $\boldsymbol{A}$ is a symmetric matrix, the Rayleigh quotient is defined as:

$$R(A,x) = \frac{x^T A x}{x^T x}.$$

[note] if x is a unit vector, then $R(A,x) = x^T A x$.

Rayleigh Quotient

- P1. $R(A,x) = \frac{x^T Ax}{x^T x}$. $\lambda_{min} \le R(A,x) \le \lambda_{max}$. //***A*** is symmetric.

Proof:

- Let $\{v_0, v_1, \dots, v_{n-1}\}$ and $\{\lambda_0, \lambda_1, \dots, \lambda_{n-1}\}$ be the eigenvectors and the eigenvalues of $\boldsymbol{A}$.
- Assume the eigenvectors have been normalized.
- These eigenvectors form a basis in R^n space. (They are mutual orthogonal.)

$x = \sum c_i v_i$,
$x^T x = \langle \sum c_i v_i, \sum c_i v_i \rangle = \sum c_i{}^2$, // Matrix ***A*** is symmetric, and $\vec{v}_i \perp \vec{v}_j$
$x^T Ax = \langle \sum c_i v_i, \sum c_i \lambda_i v_i \rangle = \sum \lambda_i c_i{}^2$.

$R(A,x) = \frac{\sum \lambda_i c_i{}^2}{\sum c_i{}^2}$, *R(A,x)* is a convex combination of the eigenvalues.
$|\lambda|_{min} \le R(A,x) \le |\lambda|_{max}$.

Usage of Rayleigh Quotient

- Rayleigh quotient is a good selection of μ in the shift inverse power method.

  $Select\ q, \|q\| = 1$;

  $\mu = \frac{q^T A q}{q^T q} = q^T A q.$

- Since $\lambda_{min} \leq R(A, x) \leq \lambda_{max}$, we can compute *m* Rayleigh quotients by using *m* random unit vectors and apply the shift inverse power method to compute other eigenvalues.
  - The max *R(A, x)* is used for computing λ_{max}.
  - The min *R(A, x)* is used for computing λ_{min}.
  - Using bisection method, after computing $\lambda_{min}\ and\ \lambda_{max}$.

- [Note] ***A*** must be symmetric.

Rayleigh Quotient Iteration

- Property: Rayleigh quotient is minimized at the eigenvector x_{n-1} .
  - Its associated eigenvalue is the minimum λ_i.

- Review: in the shift inverse power method
  - We replace the multiplication of inverse matrix $y^{(k+1)} = (A - \mu I)^{-1} y^{(k)}$ by $(A - \mu I) y^{(k+1)} = y^{(k)}$.

$$\boxed{\begin{aligned} &Select\ q, \|q\| = 1\ ; \\ &\mu = \frac{q^T A q}{q^T q} = q^T A q. \end{aligned}}$$

Rayleigh Quotient Iteration

- The shift inverse power method

  $$(A-\mu I)y^{(k+1)} = y^{(k)}.$$

  $$Select\ q, \|q\| = 1\ ;$$
  $$\mu = \frac{q^T Aq}{q^T q} = q^T Aq.$$

- Modification of the shift inverse matrix method:
  - Using the Rayleigh quotient to obtain a new shift matrix in each iteration:

  Different q vectors produce different μ values.

  - Modifying the eigenvector using the shift matrix.

Rayleigh Quotient Iteration

$select\ y \neq 0;$ //Initial $y^{(0)}$

$solve\ (A - \mu I)x = y;$ // Using an initial μ

$\rho = \frac{y^T x}{y^T y};$ //Rayleigh quotient, approximation of the eigenvalue

$repeat\{$

$\quad y = \frac{x}{\|x\|};$ //Normalization $y^{(k)}$

$\quad solve\ (A - \rho I)x = y;$ //un-normalized $y^{(k+1)},$ new linear system.

$\quad \mu = \rho;$

$\quad \rho = \frac{y^T x}{y^T y};$ // modifying the Rayleigh quotient

$\quad r = \rho * y - x;$ //$r^{(k)} = \rho^{(k)} y^{(k)} - B y^{(k)}$

$\}until(\|r\| \leq \varepsilon);$

$return\left(\frac{1}{\rho} + \mu, y\right);$ //$\rho = \frac{1}{\lambda - \mu}, \lambda = \frac{1}{\rho} + \mu$

Comparison: Shift Inverse Power Method

$B = A - \mu I;$ //μ is given by the user.

$decompose\ B\ into\ L\ and\ U$; //Doolittle's method

$select\ y \neq 0$; //Initial $y^{(0)}$

$solve\ L * h = y$; //Using forward substitution.

$solve\ U * x = h$; //Using backward substitution, un-normalized $y^{(1)}$.

$repeat\{$

$\quad y = \frac{x}{\|x\|}$; //Normalization $y^{(k)}$

$\quad solve\ L * h = y$; //Using forward substitution.

$\quad solve\ U * x = h$; //Using backward substitution, un-normalized $y^{(k+1)}$

$\quad \rho = \frac{y^T x}{y^T y}$; //Approximation of the eigenvalue.

$\quad r = \rho * y - x$; //$r^{(k)} = \rho^{(k)} y^{(k)} - B^{-1} y^{(k)}$

$\}until(\|r\| \leq \varepsilon)$;

$return\left(\frac{1}{\rho} + \mu, y\right)$; //$\rho = \frac{1}{\lambda - \mu}, \lambda = \frac{1}{\rho} + \mu$

Discussion

- In each iteration, the shift matrix is re-computed.
  - For comparison: The inverse shift matrix method uses a fixed shift matrix ***B***.
- The linear system has to be solved by using $O(n^3)$ steps.
  - Each iteration needs $O(n^3)$ steps.
  - For comparison: The inverse shift matrix method needs $O(n^2)$ steps in each iteration.
- However, it had been proved that the Rayleigh quotient iteration enjoys a cubic convergence rate.

$$e_{k+1} = O({e_k}^3)$$

  - Thus very few iterations are required.

Deflection Method

- Theorem: ***A*** is SPD. If λ and ***x*** are an eigenvalue and the corresponding eigenvector of ***A***, then the following matrix shares at least one eigenvalue with ***A***.

  $B = A - \lambda x x^T$

- Proof
  - Select a unit vector ***y***.

    $B * y = (A - \lambda x x^T)y = Ay - \lambda x(x^T y).$
  - If y=x, $Bx = Ax - \lambda x = 0$.
    - The eigenvalue of ***B*** is 0 and the corresponding eigenvector is ***x***.
  - If ***y*** is another eigenvector of ***A***, ***y*** ≠ ***x***, we have

    $B * y = Ay = \lambda_j y$. //($x \perp y, A\ is\ SPD.$)
  - Thus we can use the power method to compute the max eigenvalue of ***B***.
  - The resultant eigenvalue is λ_1.

Conclusion

- The power method computes the max eigenvalue.
- The inverse power method computes the min eigenvalue.
- The shift matrix method computes arbitrary eigenvalues.
- The Rayleigh quotient shows the range of the eigenvalues.
- The shift inverse matrix method speeds-up the shift matrix method.
- The Rayleigh quotient iteration needs higher time complexity to complete an iteration.
  – But, it needs less iterations.

Take Home Exercise

- Use Rayleigh quotients to estimate the condition number of a symmetric matrix.
  - Randomly select k unit vectors;
  - Compute the Rayleigh quotients of these vectors;
  - Find the max and min Rayleigh quotients;
  - Compute $cond(A) \approx \frac{\rho_{max}}{\rho_{min}}$.