- Rayleigh Quotient.
- Power Iteration.
- Inverse Iteration.
- Rayleigh Quotient Iteration.
- Arnoldi Iteration.
- Lanczos Iteration.

Rayleigh Quotient

- For a given complex Hermitian matrix M and nonzero vector x, the Rayleigh quotient $r(x)$, is defined as:

$$r(x) = \frac{x^* M x}{x^* x}.$$

Rayleigh Quotient

- For a given complex Hermitian matrix M and nonzero vector x, the Rayleigh quotient $r(x)$, is defined as:

$$r(x) = \frac{x^* M x}{x^* x}.$$

- It addresses the question, given an x, what scalar, α, "acts most like the eigenvalue" for x in the sense of minimizing $||Mx - \alpha x||$?

Rayleigh Quotient

- For a given complex Hermitian matrix M and nonzero vector x, the Rayleigh quotient $r(x)$, is defined as:

$$r(x) = \frac{x^* M x}{x^* x}.$$

- It addresses the question, given an x, what scalar, α, "acts most like the eigenvalue" for x in the sense of minimizing $||Mx - \alpha x||$?
- Calculating the gradient:

$$\nabla r(x) = \frac{2}{x^* x}(Mx - r(x)x)$$

Rayleigh Quotient

- For a given complex Hermitian matrix M and nonzero vector x, the Rayleigh quotient $r(x)$, is defined as:

$$r(x) = \frac{x^* M x}{x^* x}.$$

- It addresses the question, given an x, what scalar, α, "acts most like the eigenvalue" for x in the sense of minimizing $||Mx - \alpha x||$?
- Calculating the gradient:

$$\nabla r(x) = \frac{2}{x^* x}(Mx - r(x)x)$$

- This shows that the eigenvectors of M are the stationary points of the function $r(x)$ and the eigenvalues of M are the values of $r(x)$ at these stationary points.

Rayleigh Quotient

- For a given complex Hermitian matrix M and nonzero vector x, the Rayleigh quotient $r(x)$, is defined as:

$$r(x) = \frac{x^* M x}{x^* x}.$$

- It addresses the question, given an x, what scalar, α, "acts most like the eigenvalue" for x in the sense of minimizing $||Mx - \alpha x||$?
- Calculating the gradient:

$$\nabla r(x) = \frac{2}{x^* x}(Mx - r(x)x)$$

- This shows that the eigenvectors of M are the stationary points of the function $r(x)$ and the eigenvalues of M are the values of $r(x)$ at these stationary points.
- The Rayleigh quotient is a quadratically accurate estimate of the eigenvalue!

Power Iteration

- Suppose $v^{(0)}$ is a normalized vector. The following is expected to produce a sequence $v^{(i)}$ that converges to an eigenvector corresponding to the largest eigenvalue of A.

Power Iteration

- Suppose $v^{(0)}$ is a normalized vector. The following is expected to produce a sequence $v^{(i)}$ that converges to an eigenvector corresponding to the largest eigenvalue of A.

Algorithm 2 Power Iteration

$v^{(0)} =$ some vector with $||v^{(0)}|| = 1$
for $k = 1, 2, \ldots$ **do**
    $w = Av^{(k-1)}$
    $v^{(k)} = w/||w||$
    $\lambda_k = (v^{(k)})^T A v^{(k)}$
end for

Power Iteration

- We can analyze the power iteration as:

$$v^{(0)} = a_1 q_1 + a_2 q_2 + \ldots + a_m q_m$$

where q_i are orthonormal eigenvectors with corresponding eigenvalues satisfying $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda|_m \geq 0$.

Power Iteration

- We can analyze the power iteration as:

$$v^{(0)} = a_1 q_1 + a_2 q_2 + \ldots + a_m q_m$$

  where q_i are orthonormal eigenvectors with corresponding eigenvalues satisfying $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda|_m \geq 0$.

- Then one can write:

$$\begin{aligned} v^{(k)} = & c_k A^k v^{(0)} \\ = & c_k \lambda_1^k (a_1 q_1 + a_2 (\lambda_2/\lambda_1)^k q_2 + \cdots + a_m (\lambda_m/\lambda_1)^k q_k \end{aligned}$$

Power Iteration

- We can analyze the power iteration as:

$$v^{(0)} = a_1 q_1 + a_2 q_2 + \ldots + a_m q_m$$

  where q_i are orthonormal eigenvectors with corresponding eigenvalues satisfying $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda|_m \geq 0$.

- Then one can write:

$$\begin{aligned} v^{(k)} =& c_k A^k v^{(0)} \\ =& c_k \lambda_1^k (a_1 q_1 + a_2 (\lambda_2/\lambda_1)^k q_2 + \cdots + a_m (\lambda_m/\lambda_1)^k q_k \end{aligned}$$

- In the limit $k \to \infty$,

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

Inverse Iteration

- For any μ that is not an eigenvalue of A, the eigenvectors of $(A - \mu I)^{-1}$ has the same eigenvectors as A and the corresponding eigenvalues are $(\lambda_j - \mu)^{-1}$.

Inverse Iteration

- For any μ that is not an eigenvalue of A, the eigenvectors of $(A - \mu I)^{-1}$ has the same eigenvectors as A and the corresponding eigenvalues are $(\lambda_j - \mu)^{-1}$.
- Then by a judicious choice of μ close to λ_J, $(\lambda_J - \mu)^{-1}$ can be made much larger for $j \neq J$.

Inverse Iteration

- For any μ that is not an eigenvalue of A, the eigenvectors of $(A - \mu I)^{-1}$ has the same eigenvectors as A and the corresponding eigenvalues are $(\lambda_j - \mu)^{-1}$.
- Then by a judicious choice of μ close to λ_J, $(\lambda_J - \mu)^{-1}$ can be made much larger for $j \neq J$.
- Applying the power iteration should converge to q_J.

- For any μ that is not an eigenvalue of A, the eigenvectors of $(A - \mu I)^{-1}$ has the same eigenvectors as A and the corresponding eigenvalues are $(\lambda_j - \mu)^{-1}$.
- Then by a judicious choice of μ close to λ_J, $(\lambda_J - \mu)^{-1}$ can be made much larger for $j \neq J$.
- Applying the power iteration should converge to q_J.

Algorithm 6 Inverse Iteration

1: $v^{(0)} =$ some vector with $||v^{(0)}|| = 1$
2: **for** $k = 1, 2, \dots$ **do**
3: Solve $(A - \mu I)w = v^{(k-1)}$ for w
4: $v^{(k)} = w/||w||$
5: $\lambda_k = (v^{(k)})^T A v^{(k)}$
6: **end for**

Rayleigh Quotient Iteration

- The Rayleigh Quotient is a method for obtaining eigenvalue from and eigenvector estimate while the Inverse iteration obtains an eigenvector from estimate of the eigenvalue.

- The Rayleigh Quotient is a method for obtaining eigenvalue from and eigenvector estimate while the Inverse iteration obtains an eigenvector from estimate of the eigenvalue.
- In this algorithm, one combines the two:

Algorithm 8 Rayleigh quotient Iteration

1: $v^{(0)} =$ some vector with $||v^{(0)}|| = 1$
2: $\lambda^{(0)} = (v^{(0)})^T A v^{(0)}$
3: **for** $k = 1, 2, \ldots$ **do**
4: Solve $(A - \lambda^{(k-1)} I) w = v^{(k-1)}$ for w
5: $v^{(k)} = w/||w||$
6: $\lambda^{(k)} = (v^{(k)})^T A v^{(k)}$
7: **end for**

Projection into Krylov Subspace

- Iterative methods are based on projecting an $m-$dimensional problem into a lower dimensional Krylov subspace.

Projection into Krylov Subspace

- Iterative methods are based on projecting an $m-$dimensional problem into a lower dimensional Krylov subspace.
- Given a matrix A and a vector b, the associated sequence of vectors:

$$b, Ab, A^2b, A^3b \ldots$$

  is called Krylov sequence or Krylov subspace.

- For a matrix A, a complete reduction into Hessenberg form by an orthogonality transformation:

$$AQ = QH$$

is out of question if m is large.

Arnoldi Iteration

- For a matrix A, a complete reduction into Hessenberg form by an orthogonality transformation:

$$AQ = QH$$

  is out of question if m is large.
- Instead consider the first n columns of the above factorization.

Arnoldi Iteration

- For a matrix A, a complete reduction into Hessenberg form by an orthogonality transformation:

$$AQ = QH$$

  is out of question if m is large.
- Instead consider the first n columns of the above factorization.
- Let Q_n be a $m \times n$ matrix whose columns are the first columns of Q:

$$\begin{bmatrix} \Bigg| & \Bigg| & & \Bigg| \\ q_1 & q_2 & \cdots & q_n \\ \Bigg| & \Bigg| & & \Bigg| \end{bmatrix}$$

Arnoldi Iteration – modified Gram-Schmidt

Algorithm 9 Arnoldi Iteration

$b =$ some vector with $q_1 = b/||b$
for $n = 1, 2, 3, \ldots$ **do**
  $v = Aq_n$
  for $j = 1$ to n **do**
    $h_{jn} = q_j^* v$
    $v = v - h_{nj} q_j$
  end for
  $h_{n+1,n} = ||v||$
  $q_{n+1} = v/h_{n+1,m}$
end for

The Arnoldi process can be described as a systematic construction of orthonormal bases using successive Krylov subspaces – done using the Gram-Schmidt orthogonalization procedure.

$$K_n = Q_n R_n$$

Arnoldi Iteration

Let H_n be the (upper Hessenberg) matrix formed by the numbers $h_{j,k}$ computed by the algorithm:

$$H_n = \begin{bmatrix} h_{1,1} & h_{1,2} & h_{1,3} & \cdots & h_{1,n} \\ h_{2,1} & h_{2,2} & h_{2,3} & \cdots & h_{2,n} \\ 0 & h_{3,2} & h_{3,3} & \cdots & h_{3,n} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & h_{n,n-1} & h_{n,n} \end{bmatrix}.$$

We then have

$$H_n = Q_n^* A Q_n.$$

This yields an alternative interpretation of the Arnoldi iteration as a (partial) orthogonal reduction of A to Hessenberg form. The matrix H_n can be viewed as the representation in the basis formed by the Arnoldi vectors of the orthogonal projection of A onto the Krylov subspace.

Arnoldi Iteration

- Let $\tilde{H}_n$ be the $(n+1) \times n$ upper-left section of H,

$$\tilde{H}_n = \begin{bmatrix} h_{11} & & \cdots & & h_{1n} \\ h_{21} & h_{22} & & & \\ & \ddots & \ddots & & \vdots \\ & & & h_{n,n-1} & h_{n,n} \\ & & & & h_{n+1,n} \end{bmatrix}$$

Arnoldi Iteration

- Let $\tilde{H}_n$ be the $(n+1) \times n$ upper-left section of H,

$$\tilde{H}_n = \begin{bmatrix} h_{11} & & \cdots & & h_{1n} \\ h_{21} & h_{22} & & & \\ & \ddots & \ddots & & \vdots \\ & & & h_{n,n-1} & h_{n,n} \\ & & & & h_{n+1,n} \end{bmatrix}$$

- Then we have

$$AQ_n = Q_{n+1}\tilde{H}_n$$

Arnoldi Iteration

- Let $\tilde{H}_n$ be the $(n+1) \times n$ upper-left section of H,

$$\tilde{H}_n = \begin{bmatrix} h_{11} & & \cdots & & h_{1n} \\ h_{21} & h_{22} & & & \\ & \ddots & \ddots & & \vdots \\ & & & h_{n,n-1} & h_{n,n} \\ & & & & h_{n+1,n} \end{bmatrix}$$

- Then we have

$$AQ_n = Q_{n+1}\tilde{H}_n$$

- The n^{th} column of this equation:

$$Aq_n = h_{1n}q_1 + \cdots + h_{nn}q_n + h_{n+1,n}q_{n+1}$$

This indicates that $q_{(n+1)}$, satisfies an $(n+1)$ term recurrence relation involving itself and previous Krylov subspace vectors.

- Let $\tilde{H}_n$ be the $(n+1) \times n$ upper-left section of H,

$$\tilde{H}_n = \begin{bmatrix} h_{11} & & \cdots & & h_{1n} \\ h_{21} & h_{22} & & & \\ & \ddots & \ddots & & \vdots \\ & & & h_{n,n-1} & h_{n,n} \\ & & & & h_{n+1,n} \end{bmatrix}$$

- Then we have

$$AQ_n = Q_{n+1}\tilde{H}_n$$

- The n^{th} column of this equation:

$$Aq_n = h_{1n}q_1 + \cdots + h_{nn}q_n + h_{n+1,n}q_{n+1}$$

  This indicates that $q_{(n+1)}$, satisfies an $(n+1)$ term recurrence relation involving itself and previous Krylov subspace vectors.
- Arnoldi iteration is simply the modified Gram-Schmidt iteration that implements the above equation.

Arnoldi method

- To find the eigenvalues of A, one expects that since H_n is a projection of A, eigenvalues of H_n must be related to that of A.

Arnoldi method

- To find the eigenvalues of A, one expects that since H_n is a projection of A, eigenvalues of H_n must be related to that of A.
- These eigenvalues are called the Ritz value or Arnoldi eigenvalue estimates.

Lanczos Iteration

- For a symmetric matrix, since the matrix, H_n is both Hessenberg and symmetric – It just becomes a triangular matrix.

Lanczos Iteration

- For a symmetric matrix, since the matrix, H_n is both Hessenberg and symmetric – It just becomes a triangular matrix.
- Again just like Arnoldi, we find the eigenvalues of this triangular matrix.

Lanczos Iteration

- For a symmetric matrix, since the matrix, H_n is both Hessenberg and symmetric – It just becomes a triangular matrix.
- Again just like Arnoldi, we find the eigenvalues of this triangular matrix.
- This leads to a three term recurrence relation.

Lanczos Iteration

- For a symmetric matrix, since the matrix, H_n is both Hessenberg and symmetric – It just becomes a triangular matrix.
- Again just like Arnoldi, we find the eigenvalues of this triangular matrix.
- This leads to a three term recurrence relation.

Algorithm 13 Lanczos Iteration

$\beta_0 = 0$, $q_0 = 0$, $b =$ some vector with $q_1 = b/||b$
for $n = 1, 2, 3, \ldots$ **do**
    $v = Aq_n$
    $\alpha_n = q_n^T v$
    $v = v - \beta_{n-1}q_{n-1} - \alpha_n q_n$
    $\beta_n = ||v||$
    $q_{n+1} = v/\beta_n$
end for