Eigenvalue Decomposition

Theory of Eigenvalue Decomposition

Eigenvalue and Eigenvector

Definition: Let $A \in \mathbb{R}^{n\times n}$ be a square matrix. A nonzero vector x is an eigenvector of A with $\lambda \in \mathbb{C}$ being the corresponding eigenvalue if

$$Ax = \lambda x$$

Remarks:

- Even A is a real matrix, its eigenvalue and eigenvector can be complex.
- The set of eigenvalues of A is called the spectrum of A. The spectral radius $\rho(A)$ is the maximum value $|\lambda|$ over all eigenvalues of A.
- If (λ, x) is an eigenpair of A, then

  (λ^2, x) is an eigenpair of A^2,

  $(\lambda-\sigma, x)$ is an eigenpair of $A-\sigma I$

  $(\frac{1}{\lambda-\sigma}, x)$ is an eigenpair of $(A-\sigma I)^{-1}$.

  Proof. Since (λ, x) is an eigenpair of A, $Ax = \lambda x$.

  Multiply both sides by A from the left,

  $$A\cdot Ax = \lambda Ax \Rightarrow A^2 x = \lambda Ax = \lambda\cdot\lambda x = \lambda^2 x.$$

  Also, $Ax - \sigma x = \lambda x - \sigma x \Rightarrow (A-\sigma I)x = (\lambda-\sigma)x$

  $$\Rightarrow x = (\lambda-\sigma)(A-\sigma I)^{-1}x \Rightarrow \frac{1}{\lambda-\sigma}x = (A-\sigma I)^{-1}x \quad \boxtimes$$

Definition: Two matrices A and B are similar with each other if there exists

a nonsingular matrix T such that
$$B = TAT^{-1}.$$

Theorem: If A and B are similar, then A and B have the same eigenvalues.

proof. Since A, B are similar, $B = TAT^{-1}$, which implies $A = T^{-1}BT$

If (λ, x) is an eigenpair of A, then $Ax = \lambda x$, so that
$$T^{-1}BTx = \lambda x \Rightarrow B(Tx) = \lambda(Tx).$$
Thus, (λ, Tx) is an eigenpair of B.

i.e., any eigenvalue of A is an eigenvalue of B.

The reverse is shown similarly $\boxtimes$

Eigenvalue Decomposition:

$$\boxed{\begin{array}{l}\text{An eigenvalue decomposition of a square matrix } A \in \mathbb{R}^{n\times n} \text{ is a factorization} \\ \qquad\qquad A = X\Lambda X^{-1}, \\ \text{where } X \in \mathbb{C}^{n\times n} \text{ is non-singular and } \Lambda \in \mathbb{C}^{n\times n} \text{ is diagonal.}\end{array}}$$

- If $A \in \mathbb{R}^{n\times n}$ admits an eigenvalue decomposition, then
$$AX = X\Lambda$$

If we rewrite $X = [x_1 \; x_2 \; \cdots \; x_n]$ with $x_i \in \mathbb{C}^n$ the i-th column of X, and $\Lambda = \operatorname{diag}(\lambda_1, \lambda_2, \cdots, \lambda_n) \equiv \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix}$ with $\lambda_i \in \mathbb{C}$ being the i-th diagonal of Λ, then
$$A\,[x_1 \; x_2 \; \cdots \; x_n] = [x_1 \; x_2 \; \cdots \; x_n] \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix}$$
$$\Rightarrow \quad [Ax_1 \; Ax_2 \; \cdots \; Ax_n] = [\lambda_1 x_1 \; \lambda_2 x_2 \; \cdots \; \lambda_n x_n]$$
$$\Rightarrow \quad Ax_i = \lambda_i x_i, \quad i = 1, 2, \cdots, n.$$

In other words, (λ_i, x_i), $i = 1, \cdots, n$ are eigenpairs of A.

- Since X is nonsingular, x_i, $i = 1, \cdots, n$ are linearly independent. So, x_i, $i = 1, \cdots, n$, are n independent eigenvectors, which spans $\mathbb{C}^n$.

- Eigenvalue decomposition implies $X^{-1}AX = \Lambda$, so that we also says A is diagonalizable.
- Eigenvalue decomposition does not always exist, as a square matrix $A \in \mathbb{R}^{n\times n}$ does not always have n independent eigenvectors.
- Though $A \in \mathbb{R}^{n\times n}$ is real, the eigenvalue decomposition may be complex.

Characteristic Polynomial

The characteristic polynomial of $A \in \mathbb{R}^{n\times n}$, denoted P_A is a degree n polynomial defined by

$$P_A(z) = \det(zI - A) \quad , \quad \text{where} \quad z \in \mathbb{C}$$

Let (λ, x) be an eigenpair of A. Then $Ax = \lambda x$, which is equivalent to

$$(\lambda I - A)x = 0.$$

Since x is non-zero, $\lambda I - A$ has a non-zero solution. Therefore, $\lambda I - A$ is singular. That is, $\det(\lambda I - A) = P_A(\lambda) = 0$. Thus, λ is an eigenvalue of A if and only if

$$P_A(\lambda) = 0,$$

and the corresponding eigenvector x are non-zero solutions of

$$(\lambda I - A)x = 0.$$

Example 1: $A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$.

The characteristic polynomial is

$$P_A(z) = \det\left(zI - \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\right) = \det\left(\begin{bmatrix} z & -1 \\ 0 & z \end{bmatrix}\right) = z^2$$

Therefore, $P_A(\lambda) = \lambda^2 = 0 \Rightarrow \lambda_1 = \lambda_2 = 0$ are eigenvalues of A.

For eigenvectors, solve $(0I - A)x = 0$, i.e.,

$$\begin{bmatrix} 0 & -1 \\ 0 & 0 \end{bmatrix} x = 0$$

$$\Rightarrow x = \begin{pmatrix} a \\ 0 \end{pmatrix} \quad \forall a \in \mathbb{C}. \quad (\text{Only one independent eigenvector})$$

So, A is not diagonalizable (i.e., no eigenvalue decomposition).

Example 2: $A = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$

The characteristic polynomial is

$$p_A(z) = \det(zI - A) = \det\left(\begin{bmatrix} z & 1 \\ -1 & z \end{bmatrix}\right) = z^2+1$$

Therefore, $p_A(\lambda) = \lambda^2+1 = 0 \Rightarrow \lambda_1 = i, \ \lambda_2 = -i$ are eigenvalues.

For eigenvector of $\lambda_1 = i$, solve

$$(iI - A)x = 0, \quad \text{i.e.,} \quad \begin{bmatrix} i & 1 \\ -1 & i \end{bmatrix} x = 0 \Rightarrow x = \alpha \begin{bmatrix} i \\ 1 \end{bmatrix} \quad \forall \alpha \in \mathbb{C}$$

Therefore, a corresponding eigenvector is $x_1 = \begin{bmatrix} i \\ 1 \end{bmatrix}$.

For eigenvector of $\lambda_2 = -i$, solve

$$(-iI - A)x = 0, \quad \text{i.e,} \quad \begin{bmatrix} -i & 1 \\ -1 & -i \end{bmatrix} x = 0 \Rightarrow x = \beta \begin{bmatrix} i \\ -1 \end{bmatrix} \quad \forall \beta \in \mathbb{C}.$$

Therefore, a corresponding eigenvector is $x_2 = \begin{bmatrix} i \\ -1 \end{bmatrix}$.

Therefore, define $X = [x_1, x_2] = \begin{bmatrix} i & i \\ 1 & -1 \end{bmatrix}$, $\quad \Lambda = \begin{bmatrix} \lambda_1 & \\ & \lambda_2 \end{bmatrix} = \begin{bmatrix} i & \\ & -i \end{bmatrix}$

then $X^{-1} = \begin{bmatrix} i & i \\ 1 & -1 \end{bmatrix}^{-1} = \begin{bmatrix} -\frac{1}{2}i & \frac{1}{2} \\ -\frac{1}{2}i & -\frac{1}{2} \end{bmatrix}$

Therefore $\quad A = X \Lambda X^{-1}$

$$\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} i & i \\ 1 & -1 \end{bmatrix} \begin{bmatrix} i & \\ & -i \end{bmatrix} \begin{bmatrix} -\frac{1}{2}i & \frac{1}{2} \\ -\frac{1}{2}i & -\frac{1}{2} \end{bmatrix}$$

is an eigenvalue decomposition.

This shows that a real matrix may have a complex eigenvalue decomposition.

We will not use the method solve $p_A(\lambda) = 0$ and $(\lambda I - A)x = 0$ to compute the eigenvalue decomposition, because polynomial root-finding is not numerically stable in general.

Special Case: Symmetric matrix and SPD matrix.

Assume $A \in \mathbb{R}^{n \times n}$ is symmetric. Then

① The eigenvalues of A are real.

Proof. Let (λ, x) be an eigenpair of A. Then, $Ax = \lambda x$.

Multiply both sides by $x^* \equiv \overline{x^T}$ (conjugate transpose) from the left,

$$x^* A x = \lambda x^* x \Rightarrow \lambda = \frac{x^* A x}{x^* x}.$$

$x^* A x$ is real, because $\overline{x^* A x} = \overline{(x^* A x)^T} = \overline{x^T A^T \overline{x}} = \overline{x^T}\, \overline{A^T}\, \overline{\overline{x}}$

$$= x^* A x$$

$x^* x$ is also real, because $\overline{x^* x} = \overline{(x^* x)^T} = \overline{x^T \overline{x}} = x^* x$

So, $\lambda = \frac{x^* A x}{x^* x}$ is real. ⊠

② The eigenvectors corresponding to distinct eigenvalues of A are orthogonal to each other.

proof. Let (λ_1, x_1) and (λ_2, x_2) are eigenpairs of A with $\lambda_1 \neq \lambda_2$.

Then $A x_1 = \lambda_1 x_1$ and $A x_2 = \lambda_2 x_2$

Let us consider $x_2^T A x_1 \in \mathbb{R}$.

On the one hand, $x_2^T A x_1 = x_2^T(\lambda_1 x_1) = \lambda_1 x_2^T x_1 = \lambda_1 (x_1^T x_2)$.

On the other hand, $x_2^T A x_1 = (x_2^T A x_1)^T = x_1^T A^T x_2 = x_1^T A x_2$

$$= x_1^T(\lambda_2 x_2) = \lambda_2 (x_1^T x_2).$$

Therefore $\lambda_1 (x_1^T x_2) = \lambda_2 (x_1^T x_2)$. Since $\lambda_1 \neq \lambda_2$, we have

$$x_1^T x_2 = 0, \quad \text{i.e.,} \quad x_1 \perp x_2.$$ ⊠.

③ A is always diagonalizable, and the eigenvalue decomposition has a special form (due to the orthogonality of eigenvectors)

$$A = Q \Lambda Q^T,$$

where $Q \in \mathbb{R}^{n \times n}$ is orthogonal and $\Lambda \in \mathbb{R}^{n \times n}$ is diagonal. (Need some efforts to show this.)

④ If A is SPD, then all eigenvalues are positive.

If A is SPSD (symmetric positive semi-definite), then all eigenvalues are non-negative.

proof. Let (λ, x) be an eigenpair of A. Then $Ax = \lambda x$.

So, $x^T A x = \lambda x^T x \Rightarrow \lambda = \dfrac{x^T A x}{x^T x}$.

Since $x^T x = \|x\|_2^2 > 0$ $(x \neq 0)$,

If A is SPD, $x^T A x > 0 \Rightarrow \lambda = \dfrac{x^T A x}{x^T x} > 0$.

If A is SPSD, $x^T A x \geq 0 \Rightarrow \lambda = \dfrac{x^T A x}{x^T x} \geq 0$. $\boxtimes$

Computation of Eigenvalue Decomposition

Computing a single eigenvalue/eigenvector

Problem Setup

For simplicity, we restrict our attention to real symmetric matrices whose eigenvalues/eigenvectors are all real. Many of the ideas can be similarly extended to non-symmetric matrices.

Let $A \in \mathbb{R}^{n\times n}$ be symmetric. Let λ_i, $i=1,2,\cdots,n$ be its eigenvalues, which are sorted in magnitude, i.e.,

$$|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n|.$$

The corresponding eigenvectors are denoted by q_i, $i=1,2,\cdots,n$, which form an orthogonal matrix, i.e, $Q = [q_1 \; q_2 \; \cdots \; q_n] \in \mathbb{R}^{n\times n}$ with $Q^T Q = Q Q^T = I$.

Rayleigh Quotient

Let $A \in \mathbb{R}^{n\times n}$ be symmetric. For a vector $x \in \mathbb{R}^n$, the Rayleigh quotient is defined by

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

- Typically, Rayleigh quotient is used to compute an estimate to an eigenvalue, given an estimate to an eigenvector. In particular, if x is

an eigenvector of A with the corresponding eigenvalue λ, then it is easy to see that
$$r(x) = \frac{x^T A x}{x^T x} = \frac{\lambda x^T x}{x^T x} = \lambda.$$

- Actually, the eigenvalues of A are critical points of the optimizations of Rayleigh quotient, i.e.,
$$\max_{x \neq 0} r(x) \quad \text{and} \quad \min_{x \neq 0} r(x)$$
It can be proven that
$$\min_i \lambda_i = \min_{x \neq 0} r(x) \quad \text{and} \quad \max_i \lambda_i = \max_{x \neq 0} r(x)$$
and any eigenvalues that are not max or min are saddle points of $r(x)$.

Power Iteration

Purpose: Find λ_1 and its associated eigenvector x_1 with $\|x_1\|_2 = 1$.

Algorithm:

Choose $y^{(0)} \in \mathbb{R}^n$ s.t. $\|y^{(0)}\|_2 = 1$

for $k = 1, 2, \cdots$

$$z^{(k)} = A y^{(k-1)}$$
$$y^{(k)} = z^{(k)} / \|z^{(k)}\|_2$$
$$\mu^{(k)} = (y^{(k)})^T A y^{(k)}$$

end

$\mu^{(k)}$ is an estimation of eigenvalue, because $\mu^{(k)} = r(y^{(k)})$ as $\|y^{(k)}\|_2 = 1$.

Illustration by Examples:

- Assume $(2, x_1), (1, x_2)$ are two eigenpairs of $A \in \mathbb{R}^{2\times 2}$. (So $x_1 \perp x_2$)

Assume $y^{(0)} = \frac{1}{\sqrt{2}}(x_1 + x_2)$

$k=1$: $z^{(1)} = A y^{(0)} = A\left(\frac{1}{\sqrt{2}}(x_1 + x_2)\right) = \frac{1}{\sqrt{2}}(Ax_1 + Ax_2) = \frac{1}{\sqrt{2}}(2x_1 + x_2)$

$\|z^{(1)}\|_2 = \frac{1}{\sqrt{2}}\|2x_1 + x_2\|_2 = \frac{1}{\sqrt{2}}\sqrt{2^2 + 1^2} = \sqrt{\frac{5}{2}}$

$y^{(1)} = z^{(1)} / \|z^{(1)}\|_2 = \frac{1}{\sqrt{5}}(2x_1 + x_2)$

$K=1$: $z^{(2)} = Ay^{(1)} = A\left(\frac{1}{\sqrt{5}}(2x_1 + x_2)\right) = \frac{1}{\sqrt{5}}(2Ax_1 + Ax_2) = \frac{1}{\sqrt{5}}(2^2 x_1 + x_2)$

$$\|z^{(2)}\|_2 = \frac{1}{\sqrt5}\|2^2x_1+x_2\|_2 = \frac{1}{\sqrt5}\cdot\sqrt{(2^2)^2+1} = \sqrt{\frac{17}{5}}$$

$$y^{(2)} = z^{(2)}/\|z^{(2)}\|_2 = \frac{1}{\sqrt{17}}(2^2x_1+x_2)$$

$$\vdots$$

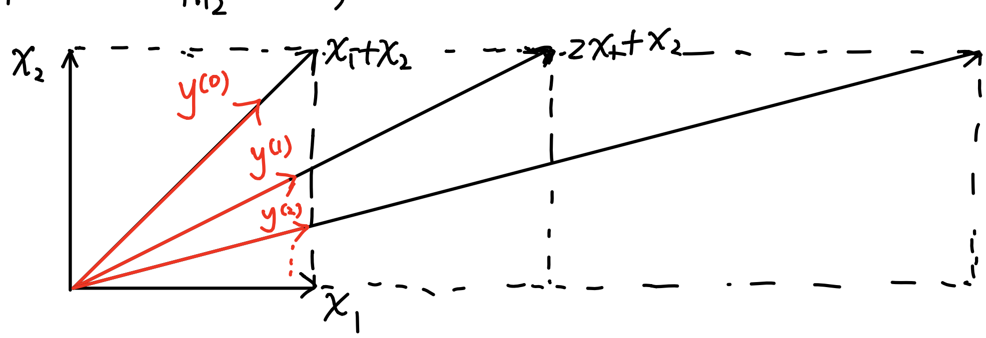
$k+1$: $$z^{(k+1)} = Ay^{(k)} = A\left(\frac{1}{\sqrt{2^{2k}+1}}(2^kx_1+x_2)\right)$$

$$= \frac{1}{\sqrt{2^{2k}+1}}(2^kAx_1+Ax_2) = \frac{1}{\sqrt{2^{2k}+1}}(2^{k+1}x_1+x_2)$$

$$y^{(k+1)} = z^{(k+1)}/\|z^{(k+1)}\|_2 = (2^{k+1}x_1+x_2)/\sqrt{2^{2k+2}+1}$$

Therefore, when k becomes larger and larger, x_1 becomes more and more dominant in $y^{(k)}$, i.e.,

$$\|y^{(k)}-x_1\|_2 \to 0, \quad \text{as } k\to+\infty$$



of

- Power iteration may not be convergent.

Assume $(1,x_1),(-1,x_2)$ are two eigen pairs of $A\in\mathbb{R}^{2\times2}$

Assume $y^{(0)} = \frac{1}{\sqrt2}(x_1+x_2)$.

$k=1$: $$z^{(1)} = Ay^{(0)} = A\left(\frac{1}{\sqrt2}(x_1+x_2)\right) = \frac{1}{\sqrt2}(Ax_1+Ax_2) = \frac{1}{\sqrt2}(x_1-x_2)$$

$$\|z^{(1)}\|_2 = \frac{1}{\sqrt2}\|x_1-x_2\|_2 = 1$$

$$y^{(1)} = z^{(1)}/\|z^{(1)}\|_2 = \frac{1}{\sqrt2}(x_1-x_2)$$

$k=1$: $$z^{(2)} = Ay^{(1)} = A\left(\frac{1}{\sqrt2}(x_1-x_2)\right) = \frac{1}{\sqrt2}(Ax_1-Ax_2) = \frac{1}{\sqrt2}(x_1+x_2)$$

$$\|z^{(2)}\|_2 = \frac{1}{\sqrt2}\|x_1+x_2\|_2 = 1$$

$$y^{(2)} = z^{(2)}/\|z^{(2)}\|_2 = \frac{1}{\sqrt2}(x_1+x_2)$$

$$\vdots$$

$k+1$: $$z^{(k+1)} = Ay^{(k)} = \frac{1}{\sqrt2}(x_1+(-1)^kx_2)$$

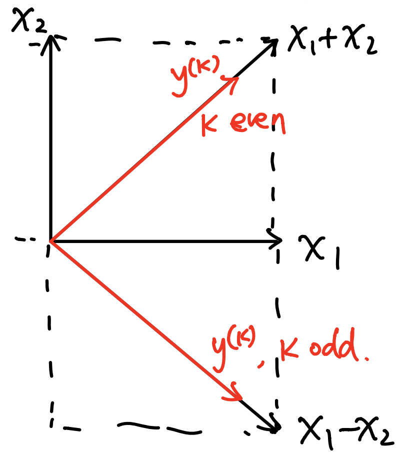
$$\|z^{(k+1)}\|_2 = 1$$

$$y^{(k+1)} = z^{(k+1)} / \|z^{(k+1)}\|_2 = \frac{1}{\sqrt{2}}(x_1 + (-1)^k x_2)$$

Therefore $y^{(k)}$ is $\frac{1}{\sqrt{2}}(x_1+x_2)$ if k even
$\frac{1}{\sqrt{2}}(x_1-x_2)$ if k odd

(Neither $\frac{1}{\sqrt{2}}(x_1+x_2)$ nor $\frac{1}{\sqrt{2}}(x_1-x_2)$ is an eigenvector)

$\therefore$



- Power iteration may not be convergent to (λ_1, x_1)

i) Assume $(-2, x_1)$, $(1, x_2)$ are two eigenpairs of $A \in \mathbb{R}^{2\times 2}$

$y^{(0)} = \frac{1}{\sqrt{2}}(x_1 + x_2)$

$$\Rightarrow \quad y^{(k)} = \frac{1}{\sqrt{2^{2k}+1}}\left((-2)^k x_1 + x_2\right).$$

So that $\|y^{(k)} - x_1\|_2$ is small when k is even
$\| y^{(k)} + x_1 \|_2$ is small when k is odd,

$\Rightarrow \quad y^{(k)} \rightarrow \pm x_1$ (only the direction is correct)

ii) Assume $(2, x_1)$ and $(1, x_2)$ are two eigenvectors of $A \in \mathbb{R}^{2\times 2}$

Assume $y^{(0)} = x_2$

$$\Rightarrow \quad z^{(1)} = Ax_2 = x_2 \Rightarrow z^{(1)} = x_2$$

$$\cdots \Rightarrow y^{(k)} = x_2 \quad \forall k.$$

i.e., $y^{(k)} \rightarrow x_2$.

Analysis of Power Iteration:

Since $y^{(k)}$ may converge to x_1 or $-x_1$, or alternatively, we can use $\min\{\|y^{(k)} - x_1\|_2^2, \|y^{(k)} + x_1\|_2^2\}$ to measure the convergence.

Note that $\|y^{(k)} - x_1\|_2^2 = \|y^{(k)}\|_2^2 + \|x_1\|_2^2 - 2\langle y^{(k)}, x_1\rangle = 2 - 2\langle y^{(k)}, x_1\rangle$

$$\|y^{(k)}+x_1\|_2^2 = \|y^{(k)}\|_2^2 + \|x_1\|_2^2 + 2\langle y^{(k)}, x_1\rangle = 2 + 2\langle y^{(k)}, x_1\rangle$$

Therefore, we use

$1-\langle y^{(k)}, x_1\rangle^2$ to measure the convergence.

we will show $1-\langle y^{(k)}, x_1\rangle^2 \to 0$ as $k \to +\infty$

Theorem: Assume $A \in \mathbb{R}^{n\times n}$ is symmetric and $|\lambda_1| > |\lambda_2|$.

If $\langle y^{(0)}, x_1\rangle \neq 0$, then $\exists$ C_0 depending on $y^{(0)}$ s.t.

$$(1-\langle y^{(k)}, x_1\rangle^2)^{1/2} \le C_0 \left|\frac{\lambda_2}{\lambda_1}\right|^k \to 0$$

Cosequently, ① $\min\{\|y^{(k)}-x_1\|_2, \|y^{(k)}+x_1\|_2\} \le \sqrt{2}\, C_0 \left|\frac{\lambda_2}{\lambda_1}\right|^k \to 0$

i.e., the limit of $y^{(k)}$ is $\pm x_1$

and ② $|\mu^{(k)}-\lambda_1| \le 2\sqrt{2}\, C_0 \left|\frac{\lambda_2}{\lambda_1}\right|^k \to 0$.

proof. By induction,

$$y^{(k)} = A^k y^{(0)} / \|A^k y^{(0)}\|_2 .$$

Let $A = X\Lambda X^T$ be an eigenvalue decomposition of A.

Then $A^k = X\Lambda X^T X\Lambda X^T \cdots X\Lambda X^T = X\Lambda^k X^T$ (Because X is orthogonal)

Thus, $A^k y^{(0)} = X\Lambda^k X^T y^{(0)}$

Let $v = X^T y^{(0)}$

$$A^k y^{(0)} = X\Lambda^k v = \sum_{i=1}^n \lambda_i^k v_i x_i , \quad \text{where } v_i \in \mathbb{R},\ x_i \in \mathbb{R}^n.$$

$$\|A^k y^{(0)}\|_2^2 = \left(\sum_{i=1}^n |\lambda_i|^{2k} |v_i|^2\right)$$

$$= |\lambda_1|^{2k} |v_1|^2 \left(1 + \left(\frac{|\lambda_2|}{|\lambda_1|}\right)^{2k} \left(\frac{|v_2|}{|v_1|}\right)^2 + \cdots + \left(\frac{|\lambda_n|}{|\lambda_1|}\right)^{2k} \left(\frac{|v_n|}{|v_1|}\right)\right)$$

$$\ge (|\lambda_1|^k |v_1|)^2$$

and

$$\langle y^{(k)}, x_1\rangle^2 = \frac{1}{\|A^k y^{(0)}\|_2^2} \left\langle \sum_{i=1}^n \lambda_i^k v_i x_i, x_1\right\rangle^2 = \frac{1}{\|A^k y^{(0)}\|_2^2} |\lambda_1|^{2k} |v_1|^2$$

(Because $\langle x_i, x_1\rangle = \begin{cases} 1 & \text{if } i=1 \\ 0 & \text{otherwise} \end{cases}$)

There fore,

$$1-\langle y^{(k)}, x_1\rangle^2 = \frac{1}{\|A^k y^{(0)}\|_2^2} \left(\|A^k y^{(0)}\|_2^2 - \left\langle \sum_{i=1}^n \lambda_i^k v_i x_i, x_1\right\rangle^2\right)$$

$$= \frac{|\lambda_1|^{2K}|v_1|^2}{\|A^{(K)}y^{(0)}\|_2^2}\left(\left(\frac{|\lambda_2|}{|\lambda_1|}\right)^{2K}\left(\frac{|v_2|}{|v_1|}\right)^2 + \cdots + \left(\frac{|\lambda_n|}{|\lambda_1|}\right)^{2K}\left|\frac{v_n}{v_1}\right|\right)$$

$$\leq \left|\frac{\lambda_2}{\lambda_1}\right|^{2K}\left|\frac{v_2}{v_1}\right|^2 + \left|\frac{\lambda_3}{\lambda_1}\right|^{2K}\left|\frac{v_3}{v_1}\right| + \cdots + \left|\frac{\lambda_n}{\lambda_1}\right|^{2K}\left|\frac{v_n}{v_1}\right|$$

$$\leq \left|\frac{\lambda_2}{\lambda_1}\right|^{2K}\cdot\left(\sum_{i=2}^{n}\left|\frac{v_i}{v_1}\right|^2\right) \equiv C_0 < +\infty \ (\text{because } v_1 \neq 0)$$

(Because $|\lambda_2| \geq |\lambda_3| \geq \cdots \geq |\lambda_n|$)

Thus,

$$\sqrt{1-\langle y^{(K)}, x_1\rangle^2} \leq C_0\left|\frac{\lambda_2}{\lambda_1}\right|^K \to 0 \quad \left(\text{since } \frac{|\lambda_2|}{|\lambda_1|} < 1\right)$$

Consequently,

① $\langle y^{(K)}, x_1\rangle^2 \geq 1 - C_0^2\left|\frac{\lambda_2}{\lambda_1}\right|^{2K}$.

Also, by Cauchy-Schwartz, $\langle y^{(K)}, x_1\rangle^2 \leq \|y^{(K)}\|_2^2\,\|x_1\|_2^2 = 1$

So $1 - C_0^2\left|\frac{\lambda_2}{\lambda_1}\right|^{2K} \leq \langle y^{(K)}, x_1\rangle^2 \leq 1$

If $\langle y^{(K)}, x_1\rangle \geq 0$, then

$$\|y^{(K)} - x_1\|_2 = \left(2 - 2\langle y^{(K)}, x_1\rangle\right)^{1/2} \leq \sqrt{2C_0^2}\left|\frac{\lambda_2}{\lambda_1}\right|^K$$

If $\langle y^{(K)}, x_1\rangle \leq 0$, then

$$\|y^{(K)} + x_1\|_2 = \left(2 + 2\langle y^{(K)}, x_1\rangle\right)^{1/2} \leq \sqrt{2C_0^2}\left|\frac{\lambda_2}{\lambda_1}\right|^K$$

So $y^{(K)}$ is close to either x_1 or $-x_1$ when K is sufficiently large.

② $\mu^{(K)} = (y^{(K)})^T A y^{(K)}$

If $\langle y^{(K)}, x_1\rangle \geq 0$, then,

$$|\mu^{(K)} - \lambda_1| = |(y^{(K)})^T A y^{(K)} - x_1^T A x_1|$$

$$= |(y^{(K)})^T A (y^{(K)} - x_1) - (x_1 - y^{(K)})^T A x_1|$$

$$\leq |(y^{(K)})^T A (y^{(K)} - x_1)| + |(x_1 - y^{(K)})^T A x_1|$$

$$= |\langle y^{(K)}, A(y^{(K)} - x_1)\rangle| + |\langle x_1, A^T(x_1 - y^{(K)})\rangle|$$

$$\leq \underbrace{\|y^{(K)}\|_2}_{=1}\|A(y^{(K)} - x_1)\|_2 + \underbrace{\|x_1\|_2}_{=1}\|A^T(x_1 - y^{(K)})\|_2$$

$$\leq \|A\|_2\|y^{(K)} - x_1\|_2 + \underbrace{\|A^T\|_2}_{=\|A\|_2}\|x_1 - y^{(K)}\|_2$$

$$\leq 2\|A\|_2\|y^{(K)} - x_1\|_2 \leq 2\sqrt{2C_0^2}\left|\frac{\lambda_2}{\lambda_1}\right|^K$$

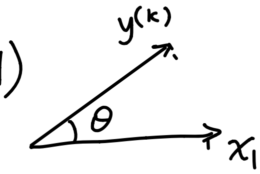
If $\langle y^{(k)}, x_1\rangle \le 0$, similarly,

$$|\mu^{(k)} - \lambda_1| \le |(y^{(k)})^T A (y^{(k)} + x_1)| + |(x_1 + y^{(k)})^T A x_1|$$

$$\cdots \le 2\sqrt{2C_0^2}\left|\frac{\lambda_2}{\lambda_1}\right|^k.$$

In either case,

$$|\mu^{(k)} - \lambda_1| \le 2\sqrt{2C_0^2}\left|\frac{\lambda_2}{\lambda_1}\right|^k. \qquad \boxtimes$$



Remark: ① $\langle y^{(k)}, x_1\rangle = \cos\angle(y^{(k)}, x_1)$ (since $\|y^{(k)}\|_2 = \|x_1\|_2 = 1$)

So $(1 - \langle y^{(k)}, x_1\rangle^2)^{1/2} = \sin\angle(y^{(k)}, x_1)$

② The convergence rate depends on $\frac{|\lambda_2|}{|\lambda_1|} < 1$

The smaller $\left|\frac{\lambda_2}{\lambda_1}\right|$, the faster convergence.

When $|\lambda_2| = |\lambda_1|$, the power iteration may not converge to x_1.

③ When $\langle y^{(0)}, x_1\rangle = 0$, C_0 may be infinity,

so that $|\langle y^{(k)}, x_1\rangle| \not\to 0$. Consequently, $y^{(k)} \not\to \pm x_1$.

④ Since the main computation in power iteration is matrix-vector product, the computational cost is $O(n^2)$ per iteration. Further, the error (of both eigenvector and eigenvalue) is $C\left|\frac{\lambda_2}{\lambda_1}\right|^k$. Thus, to get an eigenvalue and eigenvector of error ε, we need to choose k such that

$$C\cdot\left|\frac{\lambda_2}{\lambda_1}\right|^k \le \varepsilon \implies \left|\frac{\lambda_1}{\lambda_2}\right|^k \ge \frac{C}{\varepsilon} \implies k \ge \frac{\log(\frac{1}{\varepsilon}) + \log C}{\log(|\frac{\lambda_1}{\lambda_2}|)}$$

In other words, we need at least $\frac{\log(\frac{1}{\varepsilon}) + \log C}{\log|\frac{\lambda_1}{\lambda_2}|} = O(\log\frac{1}{\varepsilon})$ iterations to get an eigenvalue and eigenvector with precision ε. Therefore, the total computational complexity is

$$O\left(\log\frac{1}{\varepsilon}\cdot n^2\right)$$

Inverse Power Iteration

Since λ_i, $i=1,2,\cdots,n$ with $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n|$ are eigenvalues of A, the eigenvalues of A^{-1} are $\frac{1}{\lambda_i}$ with associated eigenvector x_i, because

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

Also, $\frac{1}{|\lambda_n|} \geq \frac{1}{|\lambda_{n-1}|} \geq \cdots \geq \frac{1}{|\lambda_1|}$.

Therefore, we can apply power iteration to A^{-1} to get λ_n (hence x_n).

Algorithm:

Choose $y^{(0)} \in \mathbb{R}^n$ s.t. $\|y^{(0)}\|_2 = 1$

for $k = 1, 2, \cdots$

$z^{(k)} = A^{-1} y^{(k-1)}$

$y^{(k)} = z^{(k)} / \|z^{(k)}\|_2$

$\mu^{(k)} = (y^{(k)})^T A y^{(k)}$

end

This is done by solving $Az^{(k)} = y^{(k-1)}$

Because $A^{-1} y^{(k)} \approx \frac{1}{\lambda_n} y^{(k)}$, i.e., $A y^{(k)} \approx \lambda_n y^{(k)}$, multiplying both sides by $(y^{(k)})^T$, we obtain $\lambda_n \approx (y^{(k)})^T A y^{(k)}$. Therefore, $\mu^{(k)} \approx \lambda_n$.

Remarks:

1. The iteration is called inverse power iteration.
2. From the convergence analysis of power iteration, we need the following conditions for the convergence of the inverse power iteration.

   a). $\langle y^{(0)}, x_n \rangle \neq 0$

   b) Since the largest two eigenvalues in magnitude of A^{-1} are $\frac{1}{\lambda_n}$ and $\frac{1}{\lambda_{n-1}}$ respectively, the convergence rate depends on

   $$\frac{1}{|\lambda_{n-1}|} \Big/ \frac{1}{|\lambda_n|} = \frac{|\lambda_n|}{|\lambda_{n-1}|}.$$

   When $\frac{|\lambda_n|}{|\lambda_{n-1}|} < 1$, the iteration will converge.

   The smaller $\frac{|\lambda_n|}{|\lambda_{n-1}|}$, the faster convergence.

   The limit of the iteration is (λ_n, x_n).
3. We need to solve $Az^{(k)} = y^{(k-1)}$ in each iteration. We don't need to compute $A = LU$ in each iteration: We can compute $A = LU$ before the iteration, so that, in each iteration, we apply only the

forward and back substitutions to solve $Az^{(k)} = y^{(k-1)}$. Therefore, the computational cost per iteration is $O(n^2)$, and the total cost for an ε precision solution is

$$O(n^3) + O(\log\tfrac{1}{\varepsilon} \cdot n^2)$$

for LU decomposition.

4. If $|\lambda_n|$ is very close to 0, then A is very close to singular, and the solution of linear equation $Az^{(k)} = y^{(k-1)}$ may have a large error. However, this is not an issue of the iteration. We still can get quite accurate eigenvalue and eigenvector. (The discussion is beyond the scope of the lecture).

Shifted Inverse Power Iteration

How to compute an arbitrary eigenpair (λ_j, x_j) of A?

Let $\mu \in \mathbb{R}$ be an estimate of λ_j (i.e., $|\mu - \lambda_j|$ is small but not 0)

Recall that eigenpairs of $(A-\mu I)^{-1}$ are $\left(\frac{1}{\lambda_i - \mu}, x_i\right)$, $i = 1, 2, \cdots, n$.

(Because $Ax_i = \lambda_i x_i \Rightarrow (A-\mu I)x_i = (\lambda_i - \mu)x_i \Rightarrow (A-\mu I)^{-1}x_i = \frac{1}{\lambda_i-\mu}x_i$).

Therefore, if $|\mu - \lambda_j|$ is small enough such that $\frac{1}{|\lambda_j - \mu|}$ is the largest among $\frac{1}{|\lambda_i - \mu|}$ for $i \neq j$, then $\left(\frac{1}{\lambda_j-\mu}, x_j\right)$ is the largest eigenpair of $(A-\mu I)^{-1}$.

To get $\left(\frac{1}{\lambda_j - \mu}, x_j\right)$ (hence (λ_j, x_j)), we apply power iteration to $(A-\mu I)^{-1}$.

Algorithm: Choose $y^{(0)} \in \mathbb{R}^n$ s.t. $\|y^{(0)}\|_2 = 1$

for $k = 1, 2, \cdots$

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

$y^{(k)} = z^{(k)} / \|z^{(k)}\|_2$

$\mu^{(k)} = (y^{(k)})^T A y^{(k)}$

end

This is done by solving $(A-\mu I) z^{(k)} = y^{(k-1)}$

Because $(A-\mu I)^{-1} y^{(k)} \approx \frac{1}{\lambda_j - \mu} y^{(k)}$, $Ay^{(k)} \approx \lambda_j y^{(k)}$, multiplying both sides by $(y^{(k)})^T$, we obtain $\lambda_j \approx (y^{(k)})^T A y^{(k)}$. Therefore, $\mu^{(k)} \approx \lambda_j$.

Remarks: 1. The iteration is called shifted inverse power iteration.

2. To make the iteration converge to (λ_j, x_j), it has to be satisfied:

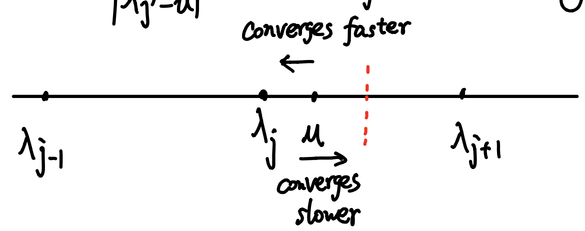
a). μ is chosen s.t. $\frac{1}{|\lambda_j-\mu|}$ is the largest among $\frac{1}{|\lambda_i-\mu|}$, $i=1,2,\cdots,n$.

b). $\langle y^{(0)}, x_j \rangle \neq 0$.

c). Let $j' = \arg\max \left\{ \frac{1}{|\lambda_i-\mu|},\ i=1,2,\cdots,n \text{ and } i \neq j \right\}$. Then

$$\frac{1}{|\lambda_{j'}-\mu|} \Big/ \frac{1}{|\lambda_j-\mu|} = \frac{|\lambda_j-\mu|}{|\lambda_{j'}-\mu|} < 1$$

The smaller $\frac{|\lambda_j-\mu|}{|\lambda_{j'}-\mu|}$, the faster convergence.



3. For an ε-precision eigenpair (λ_j, x_j), the computational complexity is

$$O(n^3) + O(\log\tfrac{1}{\varepsilon} \cdot n^2),$$

because we only need to compute the LU of $A-\mu I$ once.

Rayleigh Quotient Iteration

Instead of using a fixed μ, we can choose an adaptive shift using the current estimate of eigenvalue (i.e., the Rayleigh Quotient).

Algorithm:

$$\boxed{\begin{aligned}
&\text{Choose } y^{(0)} \in \mathbb{R}^n \text{ s.t. } \|y^{(0)}\|_2 = 1 \\
&\mu^{(0)} = (y^{(0)})^T A\, y^{(0)} \\
&\text{for } k = 1, 2, \cdots \\
&\qquad z^{(k)} = (A - \mu^{(k-1)} I)^{-1} y^{(k-1)} \\
&\qquad y^{(k)} = z^{(k)} / \|z^{(k)}\|_2 \\
&\qquad \mu^{(k)} = (y^{(k)})^T A\, y^{(k)} \\
&\text{end}
\end{aligned}}$$

$\mu^{(k)}$ are both an estimate of the eigenvalue and the shift

Remarks: 1. The iteration is known as Rayleigh quotient iteration.

2. The iteration is convergent to an eigen pair (λ_i, x_i), for which λ_i is close to $\mu^{(0)}$.

3. We need to compute the LU decomposition in each iteration, because the coefficient matrix $A-\mu^{(k)}I$ varies in each iteration.

Computing Multiple Eigenvalue / Eigenvectors

Simultaneous Iterations

To compute the first r ($r \le n$, ordered in magnitude) eigenvalues and eigenvectors of A, we can apply power method with r vectors (i.e., a matrix of size $n \times r$): At each iteration,

apply A to the estimation of eigenvectors, and then normalize them.

In particular, let $Y^{(k-1)} \in \mathbb{R}^{n \times r}$ be the current estimation of r eigenvectors, we first compute $Z^{(k)} = AY^{(k-1)}$. Then, we "normalize" $Z^{(k)}$. If we just set each column has a unit 2-norm, the corelation of independent eigenvectors is ignored. Due to the eigenvalue decomposition of A, independent eigenvectors are orthogonal. Therefore, we normalize $Z^{(k)}$ to an orthogonal matrix. This can be done by QR decomposition.

We get the following Simutaneous Power Iteration:

Choose $Y^{(0)} \in \mathbb{R}^{n \times r}$ satisfying $(Y^{(0)})^T Y^{(0)} = I$

for $k = 1, 2, \cdots$.

$\quad Z^{(k)} = AY^{(k-1)}$

$\quad$ Set $Y^{(k)}$ the Q factor in QR decomposition of $Z^{(k)}$

$\quad \mu_i^{(k)} = (y_i^{(k)})^T A (y_i^{(k)})$, $i = 1, 2, \cdots, r$

end

Under mild conditions, one has

$$\|y_i^{(k)} - \pm x_i\|_2 \le C\rho^k, \qquad i=1,2,\cdots,r,$$

where $\rho = \max_{i=1,\cdots,r} \frac{|\lambda_{i+1}|}{|\lambda_i|}$. Consequently,

$$|\mu_i^{(k)} - \lambda_i| \le C'\rho^k, \qquad i=1,2,\cdots,r.$$

QR algorithm for eigenvalue decomposition

If we set $r=n$ in the simultaneous power iteration, then we get

$$\begin{cases} Z^{(k)} = A\,Y^{(k-1)} \\ Y^{(k)}R^{(k)} = Z^{(k)} \end{cases} \qquad (\text{Let } Z^{(k)} = Y^{(k)}R^{(k)} \text{ be the QR decomposition})$$

Eliminating $Z^{(k)}$, we obtain $Y^{(k)}R^{(k)} = A\,Y^{(k-1)}$.

Let $A^{(k)} = (Y^{(k)})^T A\, Y^{(k)}$. Then

$$A^{(k-1)} = (Y^{(k-1)})^T A\, Y^{(k-1)} = (Y^{(k-1)})^T Y^{(k)} R^{(k)}$$

This is the QR decomposition of $A^{(k-1)}$, where Q is $(Y^{(k-1)})^T Y^{(k)}$ and R is $R^{(k)}$

$$A^{(k)} = (Y^{(k)})^T A\, Y^{(k)} = (Y^{(k)})^T A\, Y^{(k-1)} (Y^{(k-1)})^T Y^{(k)} = R^{(k)} (Y^{(k-1)})^T Y^{(k)}$$

By setting $Q^{(k)} = (Y^{(k-1)})^T Y^{(k)}$. Then the iteration becomes

Choose initial guess $A^{(0)}$ (e.g., $A^{(0)} = A$)

for $k=1,2,\cdots$

  Compute QR decomposition: $A^{(k-1)} = Q^{(k)} R^{(k)}$

  Set $A^{(k)} = R^{(k)} Q^{(k)}$

end

This algorithm is known as QR algorithm for eigenvalues. (Note that it is not QR decomposition.)

Remarks: 1. Since $A^{(k-1)} = Q^{(k)}R^{(k)}$, we have $R^{(k)} = (Q^{(k)})^T A^{(k-1)}$, which implies

$$A^{(k)} = R^{(k)}Q^{(k)} = (Q^{(k)})^T A^{(k-1)} Q^{(k)}$$

By induction,

$$A^{(k)} = (Q^{(k)})^T \cdots (Q^{(1)})^T A\, Q^{(1)} \cdots Q^{(k)}$$

$$= (Q^{(1)}\cdots Q^{(k)})^T A (Q^{(1)}\cdots Q^{(k)})$$

Since $Q^{(1)}\cdots Q^{(k)}$ is orthogonal, $A^{(k)}$ is similar to A. Thus, eigenvalues of $A^{(k)}$ are the same as eigenvalues of A.

2. Since $Y^{(k)}$ is expected to converge to eigenvectors of A, $A^{(k)} = (Y^{(k)})^T A Y^{(k)}$ is expected to converge to $\Lambda = \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix}$, the diagonal matrix of eigenvalues of A. Indeed, it is shown that:

   if the eigenvalues of A are well separated, then $A^{(k)}$ converges to Λ, and $(Q^{(1)}\cdots Q^{(k)})$ converges to eigenvectors of A.

4. Since QR decomposition costs $O(n^3)$ and matrix-matrix multiplication costs $O(n^3)$ as well, the computational cost of the standard QR algorithm is: $\quad O(Kn^3)$, where K is the number of iterations needed.

Practical Implementation of QR algorithm.

The complexity of QR algorithm can be reduced to $O(n^2)$ per iteration, and this strategy is used in practice. This practical implementation is divided into two phases:

Phase I: Reduction of A to tridiagonal $A^{(0)}$

$$(Q^{(0)})^T A Q^{(0)} = A^{(0)} \equiv \begin{bmatrix} \times & \times & & & \\ \times & \times & \times & & \\ & \times & \ddots & \ddots & \\ & & \ddots & \ddots & \times \\ & & & \times & \times \end{bmatrix},$$

where $Q^{(0)} \in \mathbb{R}^{n\times n}$ is orthogonal.

Phase II: Apply QR algorithm with initial guess $A^{(0)}$.

for $k = 1, 2, \ldots$

  Compute QR decomposition: $A^{(k-1)} = Q^{(k)} R^{(k)}$

  Set $A^{(k)} = R^{(k)} Q^{(k)}$

end

Since $A^{(0)}$ is tridiagonal, its QR can be done in $O(n^2)$ by Givens rotation, and $A^{(1)}$ will keep tridiagonal, and so on

Therefore, in each iteration, $A^{(k-1)}$ is tridiagonal, and the computation can be done in $O(n^2)$ operations.

Therefore, it can be derived that

$$A^{(k-1)} = Q^{(k)} R^{(k)} \Rightarrow R^{(k)} = (Q^{(k)})^T A^{(k-1)} \Rightarrow$$

$$A^{(k)} = R^{(k)} Q^{(k)} = (Q^{(k)})^T A^{(k-1)} Q^{(k)} = (Q^{(k)})^T (Q^{(k-1)})^T A^{(k-2)} Q^{(k-1)} Q^{(k)}$$

$$= \cdots = (Q^{(k)})^T (Q^{(k-1)})^T \cdots (Q^{(1)})^T A^{(0)} Q^{(1)} \cdots Q^{(k-1)} Q^{(k)}$$

$$= (Q^{(k)})^T \cdots (Q^{(1)})^T (Q^{(0)})^T A \, Q^{(0)} Q^{(1)} \cdots Q^{(k)}$$

Thus, $A^{(k)}$ and A are similar. By the convergence, we know that $A^{(k)}$ converges to a diagonal matrix Λ, whose diagonals are eigenvalues of A. Also, $Q^{(0)} Q^{(1)} \cdots Q^{(k)}$ converges to eigenvectors of A.

Let us elaborate on how to do Phases I and II.

Phase I: Reduction of A to tridiagonal by Householder transform

① Let P_1 be the householder matrix that reduce $A(2:n, 1)$ to a multiple of e_1, so that

$$P_1 A = \begin{bmatrix} \Delta & \Delta & \Delta & \cdots & \Delta \\ \times & \times & \times & \cdots & \times \\ 0 & \times & \times & \cdots & \times \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & \times & \times & \cdots & \times \end{bmatrix}$$

Δ — unchanged entry from A (i.e., $A(1, 1:n)$ is unchanged)

Then,

$P_1 A P_1^T$ will not change the first column of $P_1 A$.

(the first row has $n-2$ zeros due to symmetry of A)

Therefore, $(3:n, 1)$ entries of $P_1 A P_1^T$ are zeros.

Also, $P_1 A P_1^T$ is symmetric

$\Longrightarrow$ Both $(3:n, 1)$ and $(1, 3:n)$ entries of $P_1 A P_1^T$ are zeros.

So,

$$P_1 A P_1^T = \begin{bmatrix} \Delta & \times & 0 & \cdots & 0 \\ \times & \times & \times & \cdots & \times \\ 0 & \times & \times & \cdots & \times \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \times & \times & \cdots & \times \end{bmatrix} \equiv T^{(1)}$$

Efficient implementation:

Let $x = A(2:n, 1)$ and u_1 be the householder vector s.t.

$$(I - 2u_1u_1^T)x = \underbrace{-\operatorname{sgn}(x_1)\|x\|_2}_{\text{denoted by } \beta} e_1 .$$

Then $P_1 = \begin{bmatrix} 1 & \\ & I - 2u_1u_1^T \end{bmatrix}$ and

$$P_1 A P_1^T = \begin{bmatrix} 1 & \\ & I-2u_1u_1^T \end{bmatrix} \begin{bmatrix} A(1,1) & x^T \\ x & A(2:n,2:n) \end{bmatrix} \begin{bmatrix} 1 & \\ & I-2u_1u_1^T \end{bmatrix}$$

$$= \begin{bmatrix} A(1,1) & \beta e_1^T \\ \beta e_1 & (I-2u_1u_1^T)A(2:n,2:n)(I-2u_1u_1^T) \end{bmatrix}$$

$$= \begin{bmatrix} a_{11} & \beta \ 0 \cdots 0 \\ \begin{matrix} \beta \\ 0 \\ \vdots \\ 0 \end{matrix} & \boxed{} \end{bmatrix}$$

By simple calculation,

$$(I-2u_1u_1^T)A(2:n,2:n)(I-2u_1u_1^T) = A(2:n,2:n) - 2u_1w^T - 2wu_1^T, \text{ where}$$

$$w = (I - u_1u_1^T)A(2:n,2:n)u_1$$

② Apply the same procedure to $T^{(1)}(2:n,2:n)$ to get $A^{(2)}$

$$P_2 T^{(1)} P_2^T = \begin{bmatrix} \Delta & \Delta & 0 & 0 & \cdots & 0 \\ \Delta & \Delta & \times & 0 & \cdots & 0 \\ 0 & \times & \times & \times & \cdots & \times \\ 0 & 0 & \times & \times & \cdots & \times \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \times & \times & \cdots & \times \end{bmatrix} \equiv T^{(2)}$$

③ Apply the same procedure to $T^{(2)}(3:n,3:n)$, and so on.

$\vdots$

After $n-2$ steps, we get $A^{(0)} = T^{(n-2)}$, which is tridiagonal.

So
$$A^{(0)} = (P_{n-2}P_{n-1}\cdots P_2P_1)A(P_1^TP_2^T\cdots P_{n-2}^T) = (P_{n-2}P_{n-1}\cdots P_2P_1)A(P_1P_2\cdots P_{n-2})$$
$$= \underbrace{(P_1P_2\cdots P_{n-2})^T}_{Q_0^T} A \underbrace{(P_1P_2\cdots P_{n-2})}_{Q_0}$$

$$A^{(0)} = Q_0^T A Q_0$$

Tridiagonal.

Code:

```
for k=1:n-2
    x=A(k+1:n,k)
    u_k = x + sgn(x(1)) ||x||_2 e_1
    u_k = u_k / ||u_k||_2
    A(k,k+1) = -sgn(x(1)) ||x||_2
    A(k+1,k) = A(k,k+1)
    A(k+2:n,k) = 0
    A(k,k+2:n) = 0
    w = A(k+1:n,k+1:n) u_k
    w = w - (u_k^T w) u_k
    A(k+1:n,k+1:n) = A(k+1:n,k+1:n) - 2 u_k w^T - 2 w u_k^T
end
```

Computational cost: $O(n^3)$

Phase II: QR iteration

for $k = 1, 2, \cdots$

Compute QR decomposition $A^{(k-1)} = Q^{(k)}R^{(k)}$

$A^{(k)} = R_kQ_k$

end

The QR decomposition of $A^{(k-1)}$ can be done by Givens rotation

$$A^{(k-1)} = \begin{bmatrix} \boxed{x} & x & x & & \\ x & x & x & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & x \\ & & & x & x \end{bmatrix} \xrightarrow[\text{Givens rotation}]{G_1^{(k)}} \begin{bmatrix} \boxed{\begin{matrix} x \\ x \end{matrix}} & \begin{matrix} x \\ x \end{matrix} & \begin{matrix} \\ x \end{matrix} & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & x \\ & & & x & x \end{bmatrix} \xrightarrow{G_2^{(k)}} \begin{bmatrix} x & x & + & & \\ & x & x & & \\ & x & x & x & \\ & & \ddots & \ddots & x \\ & & & x & x \end{bmatrix}$$

$$\cdots \xrightarrow{G_{n-1}^{(k)}} \begin{bmatrix} x & x & + & & \\ & x & x & + & \\ & & x & \ddots & + \\ & & & \ddots & x \\ & & & & x \end{bmatrix} \equiv R^{(k)} \text{ — upper triangular}$$

$+$ — fill-in of nonzeros.

So $\quad G_{n-1}^{(k)} G_{n-2}^{(k)} \cdots G_1^{(k)} A^{(k-1)} = R^{(k)}$

$$\Rightarrow \quad A^{(k-1)} = \underbrace{(G_1^{(k)})^T (G_2^{(k)})^T \cdots (G_{n-1}^{(k)})^T}_{Q^{(k)}} R^{(k)}$$

By this way, we get QR decomposition of $A^{(k-1)}$.

In the application of $G_1^{(k)}$, only $(1:2, 1:3)$ entries are changed, and the operations needed are $O(1)$. Similarly, at each substep, the computational cost is $O(1)$. Since we need $n-1$ substeps, the total computational complexity for the $R^{(k)}$ in the QR decomposition of $A^{(k-1)}$ is $O(n)$. To get $Q^{(k)}$, we need to multiply all $G_i^{(k)}$ together, which needs $O(n^2)$ operations.

Now, $\quad A^{(k)} = R^{(k)} Q^{(k)}$

$$= R^{(k)} (G_1^{(k)})^T (G_2^{(k)})^T \cdots (G_{n-1}^{(k)})^T$$

Left-multiplying $G_1^{(k)}$ rotates 1st & 2nd rows.

$\Rightarrow$ Right-multiplying $(G_1^{(k)})^T$ rotates 1st & 2nd columns.

$$R^{(k)} (G_1^{(k)})^T = \begin{bmatrix} x & x & + & & \\ + & x & x & + & \\ & & x & \ddots & + \\ & & & \ddots & x \\ & & & & x \end{bmatrix} .$$

$$R^{(k)}(G_1^{(k)})^T(G_2^{(k)})^T = \begin{bmatrix} \times & \times & + & & & \\ + & \times & \times & + & & \\ & + & \times & \ddots & & \\ & & \ddots & \ddots & + & \\ & & & \ddots & \times & \\ & & & & \times & \end{bmatrix}, \quad \cdots\cdots$$

Finally,

$$A^{(k)} = R^{(k)}Q^{(k)} = \begin{bmatrix} \times & \times & + & & & \\ + & \times & \times & + & & \\ & + & \times & \ddots & & \\ & & \ddots & \ddots & + & \\ & & & \ddots & \ddots & \times \\ & & & & + & \times \end{bmatrix}$$

The computational complexity for $A^{(k)}$ is $O(n)$, since for each substep, the computational cost is $O(1)$ (we change only 6 entries).

Furthermore,

Since $A^{(k)} = R^{(k)}Q^{(k)} = (Q^{(k)})^T A^{(k-1)} Q^{(k)}$ and $A^{(k-1)}$ is symmetric

$A^{(k)}$ is symmetric.

This, together with the fact that $A^{(k)}$ has only one sub-diagonal, implies $A^{(k)}$ is also tridiagonal

In summary, the computational cost per iteration in Phase II is $O(n^2)$, and the iteration matrix $A^{(k)}$ is at most tridiagonal.

Non-Symmetric Eigenvalue Problem and Schur Decomposition

All the power iteration and its variants can be extended straightforwardly to eigenvalues and eigenvectors of a non-symmetric matrix $A \in \mathbb{R}^{n\times n}$.

1. one eigenvalue and the corresponding eigenvector

Power iteration

Choose $y^{(0)}$ s.t. $\|y^{(0)}\|_2 = 1$

for $k = 1, 2, \cdots,$

$z^{(k)} = A y^{(k-1)}$

The same as symmetric case, the algorithm will converge to (λ_1, x_1) if

① $|\lambda_1| > |\lambda_2|$ (In this case, λ_1 must be real, and x_1 is real also)

$$y^{(k)} = z^{(k)}/\|z^{(k)}\|_2$$

$$\mu^{(k)} = (y^{(k)})^T A (y^{(k)})$$

end

② $y^{(0)} \not\perp x_1$ (why?

The convergence speed depends on the ratio $\frac{|\lambda_2|}{|\lambda_1|}$.

Inverse iteration, shifted inverse iteration, are the same.

The convergence is also the same.

2. for all eigenvalues

- QR algorithm.

① Phase I:

We cannot reduce A to a tridiagonal $A^{(0)} = (Q^{(0)})^T A Q^{(0)}$, because A is non-symmetric.

Instead, we can reduce A to an upper Hessenberg form $A^{(0)}$.

$$\begin{bmatrix} 1 & 0 \\ 0 & P_1 \end{bmatrix} A = \begin{bmatrix} x & x & \cdots & x \\ x & x & \cdots & x \\ 0 & x & \cdots & x \\ \vdots & \vdots & \ddots & \vdots \\ 0 & x & \cdots & x \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 0 \\ 0 & P_1 \end{bmatrix} A \begin{bmatrix} 1 & 0 \\ 0 & P_1 \end{bmatrix}^T = \begin{bmatrix} x & x & \cdots & x \\ x & x & \cdots & x \\ 0 & x & \cdots & x \\ \vdots & \vdots & \ddots & \vdots \\ 0 & x & \cdots & x \end{bmatrix}$$

(Due to the non-symmetry, elements of the first row are generally non-zeros)

$$(Q^{(0)})^T A Q^{(0)} = \begin{bmatrix} x & x & x & \cdots & x \\ x & x & x & \cdots & x \\ 0 & x & x & \cdots & x \\ \vdots & 0 & x & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots 0 & x & x \end{bmatrix} \equiv A^{(0)}$$

← upper Hessenberg (i.e., $A^{(0)}(i,j) = 0$ if $i-j > 1$)

② Phase II:

for $k = 1, 2, \cdots$

Compute QR decomposition $A^{(k-1)} = Q^{(k)} R^{(k)}$

$A^{(k)} = R^{(k)} Q^{(k)}$

end

The QR decomposition of $A^{(k-1)}$ can be done in $O(n^2)$ by Givens Rotation

$$\begin{bmatrix} x & x & \cdots & & x \\ 0 & x & \cdots & & x \\ \vdots & 0 & \ddots & & \vdots \\ 0 & 0 & \cdots & 0\ x & x \end{bmatrix} \xrightarrow{G_K} \begin{bmatrix} 0 & x & \cdots & & x \\ 0 & x & \cdots & & x \\ \vdots & & \ddots & & \vdots \\ 0 & \cdots & 0 & x & x \end{bmatrix} \xrightarrow{G_K} \begin{bmatrix} 0 & x & x & \cdots & \vdots \\ 0 & 0 & x & \cdots & \vdots \\ \vdots & \vdots & \cancel{x}\ 0 & \ddots & \vdots \\ 0 & 0 & 0 & \cdots\ 0\ x & x \end{bmatrix} \rightarrow \cdots$$

↑ Change two rows.

So the cost is Cn for some C.

cost $C(n-1)$

$$\xrightarrow{G_K^{(n-1)}} \begin{bmatrix} x & x & \cdots & x \\ & x & \ddots & \vdots \\ & & \ddots & \vdots \\ & & & x \end{bmatrix}$$

Total computational cost: $C(n+(n-1)+\cdots+2) = O(n^2)$

The 2nd step $A^{(k)} = R^{(k)}Q^{(k)}$ also done in $O(n^2)$.

- Convergence of QR algorithm for non-symmetric A.

We don't expect $A^{(k)} \to \Lambda$ and $Q^{(1)}Q^{(2)}\cdots Q^{(k)}$ converge to eigenvector of A, because the eigenvectors of A might not be orthogonal and A might not exist an eigenvalue decomposition.

Also, all the computations are done in real numbers. In case A has a complex eigenvalue, it is impossible to obtain it by using the QR algorithm in real numbers.

What is the limit?

Schur Decomposition: For any matrix $A \in \mathbb{R}^{n\times n}$, there exists $Q, U \in \mathbb{R}^{n\times n}$ such that: $A = QSQ^T$,

where $Q \in \mathbb{R}^{n\times n}$ is orthogonal, and

$S \in \mathbb{R}^{n\times n}$ is block upper triangular with 1×1 or 2×2 $\overbrace{\text{blocks}}^{\text{diagonal}}$.

(i.e., $\exists$ a partition of

s.t. $S = \begin{bmatrix} S_{11} & S_{12} & \cdots & S_{1p} \\ & S_{22} & \ddots & \vdots \\ & & \ddots & \\ & & & S_{pp} \end{bmatrix}$ with S_{ii} either 1×1 or 2×2)

Furthermore, if $S_{ii} \in \mathbb{R}^{1\times1}$, then it is an eigenvalue of A;

if $S_{ii} \in \mathbb{R}^{2\times 2}$, then $S_{ii} = \begin{bmatrix} a & -b \\ b & a \end{bmatrix}$ with $a \pm bi$ eigenvalues of A.

The blocks S_{ii} can be sorted s.t.

$|eig(S_{11})| \geq |eig(S_{22})| \geq \dots |eig(S_{pp})|$.

Remarks: ① Schur decomposition is in real numbers.

② It gives all eigenvalues of A.

③ The independent eigenvectors can be calculated easily from Schur decomposition

④ If $A \in \mathbb{R}^{n\times n}$ is symmetric, then the Schur Decomposition is the same as the eigenvalue decomposition.

Convergence of QR algorithm for non-symmetric $A = QSQ^T$:

$$\left.\begin{aligned} A^{(k)} &\to S \\ Q^{(1)}Q^{(2)}\cdots Q^{(k)} &\to Q \end{aligned}\right\} \Longrightarrow \text{we can get eigenvalues and eigenvectors of } A.$$

Example of Schur decomposition

$$A = \begin{bmatrix} 1 & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{3}{2} & -\frac{1}{2} \\ \frac{1}{\sqrt{2}} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

Then

$$A = \underbrace{\begin{bmatrix} 1 & & \\ & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}}_{Q} \underbrace{\begin{bmatrix} 1 & -1 & 0 \\ 1 & 1 & 1 \\ & & 1 \end{bmatrix}}_{S} \underbrace{\begin{bmatrix} 1 & & \\ & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}}_{Q^T}$$

So the eigenvalues are $1 \pm i$, 1.