Contents

Chapter 5

Solving matrix equations $A\mathbf{x} = \mathbf{b}$

5.1 Preliminaries

Definition 5.1.1. A $n \times n$ matrix A is it similar to B if there exists a nonsingular matrix P such that $PAP^{-1} = B$. It is called a **similarity transformation.** If B is a diagonal matrix then A is called **diagonalizable.**

Definition 5.1.2 (non-defective)**.** If A has n-linearly independent eigenvectors, then it is called **non-defective**. Otherwise, it is **defective**.

A nonsingular matrix can be defective.

Theorem 5.1.3. *Eigenvectors corresponding to distinct eigenvalues are linearly independent.*

Theorem 5.1.4. *If $PAP^{-1} = B$ then A and B have the same set of eigenvalues.*

We have a characterization of diagonalizable matrices:

Theorem 5.1.5. *A is diagonalizable if and only if there exists a positive definite Hermitian matrix P such that $PAP^{-1} = N$ is a normal matrix.*

How do we classify defective matrices? The best way is to study the multiplicity of the eigenvalues and the number of linearly independent eigenvectors associated to it.

Definition 5.1.6. The elementary divisors of a square matrix A are the polynomials $(\lambda - \alpha_1)^{e_1}, \cdots, (\lambda - \alpha_t)^{e_t}$ obtained from the characteristic equation $|A - \lambda I| = 0$.

Jordan canonical form

To each elementary divisor $(\lambda - \alpha)^e$ we associate an elementary Jordan block of the form

$$J_\alpha = \begin{bmatrix} \alpha & 1 & 0 & & 0 \\ & \alpha & 1 & & 0 \\ & & & \ddots & 1 \\ & & & & \alpha \end{bmatrix},$$

where J_α is order $e \times e$. If $e = 1$, J_α contains the single element α. Once we have elementary divisors of A, we can construct the Jordan canonical form.

Definition 5.1.7. Jordan canonical form is a block diagonal matrix whose diagonal blocks are elementary Jordan blocks.

The following shows the relation between elementary divisor and Jordan blocks.

Theorem 5.1.8. *Any $n \times n$ matrix A is similar to J, a Jordan canonical form. J is unique up to permutation of elementary Jordan blocks.*

An example

$$J = \left[\begin{array}{c|c|c|c|cc} 1 & 0 & & & & \\ \hline 0 & 1 & 0 & & & \\ \hline & 0 & \alpha_1 & 0 & & \\ \hline & & 0 & \alpha_1 & 0 & \\ \hline & & & 0 & \alpha_2 & 1 \\ & & & & 0 & \alpha_2 \end{array}\right]$$

Corollary 5.1.9. *If $(\lambda - \lambda_1)^{e_1}, (\lambda - \lambda_2)^{e_2}, \cdots, (\lambda - \lambda_t)^{e_t}$ are the elementary divisors of A, where $\lambda_1, \lambda_2, \cdots$ need not be distinct, then*

$$n = e_1 + \cdots + e_t.$$

If all elementary divisors are linear, then $e_1 = e_2 = \cdots = 1$ and $t = n$. Thus Jordan canonical form is diagonal.

Theorem 5.1.10. *If A has t elementary Jordan blocks in its Jordan canonical form, then A has a set of t linearly independent eigenvectors.*

Theorem 5.1.11. *If α is an eigenvalue of A which appears in r elementary divisors of A (or in r elementary Jordan blocks) then there exists r linearly independent eigenvectors of A associated with α.*

Definition 5.1.12. If S is a vector subspace of R^n such that $AS \subset S$, then S is called an invariant subspace of A.

Thus the subspace of eigenvectors corresponding to an eigenvalue α is an invariant subspace.

Definition 5.1.13. Let $(\lambda-\lambda_1)^{e_1},(\lambda-\lambda_2)^{e_2},\cdots,(\lambda-\lambda_t)^{e_t}$ be elementary divisors of A. If $\lambda_1,\cdots,\lambda_t$ are distinct, then A is called **non-derogatory.** On the other hand, if any two of $\lambda's$ are equal, we say A is **derogatory**. Stated again, A is derogatory if and only if the same eigenvalue appears in more than one elementary Jordan block in its Jordan canonical form.

Householder reflection

Given two vectors $\mathbf{u},\mathbf{v}$ of the same length, we want to find an orthogonal transform which maps $\mathbf{u}$ to $\mathbf{v}$.

The following form of $n\times n$ matrix $H_{\mathbf{w}}$ is called a **Householder reflection (elementary reflector)**, if

$$H_{\mathbf{w}} = I - 2\mathbf{w}\mathbf{w}^*, \quad \mathbf{w}^* := \bar{\mathbf{w}}^T$$

for some unit vector $\mathbf{w}\in\mathbb{R}^n$. Here $P_{\mathbf{w}} := \mathbf{w}\mathbf{w}^*$ is a projection along the vector $\mathbf{w}$. Clearly $H_{\mathbf{w}}$ is symmetric and orthogonal ($H_{\mathbf{w}}^* = H_{\mathbf{w}}$ and $H_{\mathbf{w}}^* H_{\mathbf{w}} = I$). We need to find $\mathbf{w}$ so that the Householder reflection satisfies $H_{\mathbf{w}}\mathbf{u} = \mathbf{v}$. From the figure 5.1 we can see $\mathbf{w}$ is given by

$$\mathbf{w} = \frac{\mathbf{u}-\mathbf{v}}{\|\mathbf{u}-\mathbf{v}\|}. \tag{5.1}$$

Theorem 5.1.14 (Schur)**.** *If $M\in\mathbb{C}^{n,n}$, then $\exists$ a unitary matrix U such that $U^H MU = T$, where T is upper triangular, displaying eigenvalues on the diagonal. If M is a real matrix, then it is possible to choose a real symmetric matrix U, and T is real, block upper triangular, each of the blocks are either 1×1 or 2×2 blocks of the form $\begin{bmatrix} a & -b \\ b & a \end{bmatrix}$.*

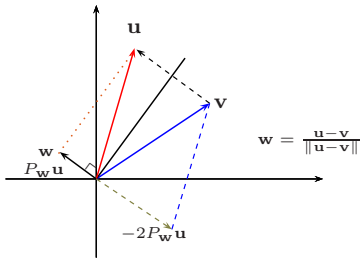


Figure 5.1: Action of an elementary reflector $H_{\mathbf{w}}$

A constructive proof. Let λ_1 be an eigenvalue of M and $\mathbf{u}$ be a corresponding eigenvector(there exists at least one) with $u_1\le 0$ and $\mathbf{u}^H\mathbf{u} = \|\mathbf{u}\|^2 = 1$ so that $M\mathbf{u} = \lambda_1\mathbf{u}$. There exists a vector $\mathbf{w}$ such that $\mathbf{w}^H\mathbf{w} = 1$ and

$$(I - 2\mathbf{w}\mathbf{w}^H)\mathbf{e}_1 = \mathbf{u}. \tag{5.2}$$

Let $U_1 = I - 2\mathbf{w}\mathbf{w}^H$. Then $U_1^H = U_1$ and we have

$$U_1^H M U_1 = U_1^H M(I - 2\mathbf{w}\mathbf{w}^H).$$

Its first column is

$$U_1^H M(I - 2\mathbf{w}\mathbf{w}^H)\mathbf{e}_1 = U_1^H M\mathbf{u} = U_1^H \lambda_1 \mathbf{u} = \lambda_1 \mathbf{e}_1.$$

Thus,

$$U_1^H M U_1 = \left[\begin{array}{c|ccc} \lambda_1 & b_2 & \dots & b_n \\ \hline 0 & & & \\ \vdots & & M_1 & \\ 0 & & & \end{array}\right]. \tag{5.3}$$

We repeat above process for M_1 to construct $U_2' \in \mathbb{R}^{(n-1)\times(n-1)}$ so that $(U_2')^H M_1 U_2'$ is a $(n-1)\times(n-1)$ matrix similar to (5.3). Let

$$U_2 = \left[\begin{array}{c|ccc} 1 & 0 & \dots & 0 \\ \hline 0 & & & \\ \vdots & & U_2' & \\ 0 & & & \end{array}\right].$$

Then

$$\begin{aligned} U_2^H M U_2 &= \left[\begin{array}{c|ccc} 1 & 0 & \dots & \\ \hline 0 & & & \\ \vdots & & (U_2')^H & \\ 0 & & & \end{array}\right] \left[\begin{array}{c|ccc} \lambda_1 & b_2 & \dots & b_n \\ \hline 0 & & & \\ \vdots & & M_1 & \\ 0 & & & \end{array}\right] \left[\begin{array}{c|ccc} 1 & 0 & \dots & \\ \hline 0 & & & \\ \vdots & & U_2' & \\ 0 & & & \end{array}\right] \\ &= \left[\begin{array}{c|ccc} \lambda_1 & * & \dots & * \\ \hline 0 & & & \\ \vdots & & (U_2')^H M U_2' & \\ 0 & & & \end{array}\right] = \left[\begin{array}{cc|ccc} \lambda_1 & * & * & \dots & * \\ 0 & \lambda_2 & * & \dots & * \\ \hline 0 & 0 & & & \\ \vdots & \vdots & & M_2 & \\ 0 & 0 & & & \end{array}\right]. \end{aligned}$$

Continue the same process to obtain $U_3, \dots, U_{n-1}$. Finally, $U = U_1 U_2 \cdots U_{n-1}$ is the desired unitary matrix. The eigenvalues of M are clearly the diagonal elements of T obtained in this process. □

A nonconstructive proof. Let $\mathbf{u}_1$ be an eigenvector of A, which has unit length. By the Gram-Schmidt process we may choose $\mathbf{u}_2', \cdots, \mathbf{u}_n'$ so that $\{\mathbf{u}_1, \mathbf{u}_2', \cdots, \mathbf{u}_n'\}$ is an orthonormal basis. Let $Q_0 = [\mathbf{u}_1, \mathbf{u}_2', \cdots, \mathbf{u}_n']$. Then Q_0 is unitary and $Q_0^* A Q_0 = \begin{bmatrix} \lambda_1 & * \\ 0 & A_1 \end{bmatrix}$, for some $(n-1)\times(n-1)$ matrix

A_1. Likewise, we may find a unitary $(n-1)\times(n-1)$ matrix Q_1 so that $Q_1^* A_1 Q_1 = \begin{bmatrix} \lambda_2 & * \\ 0 & A_2 \end{bmatrix}$. Then if $S_1 = Q_0 \begin{bmatrix} 1 & 0 \\ 0 & Q_1 \end{bmatrix}$, we have

$$S_1^* A S_1 = \begin{bmatrix} \lambda_1 & * & * \\ 0 & \lambda_2 & * \\ 0 & 0 & A_2 \end{bmatrix}$$

Note that S_1 is unitary by Theorem. Now continue in this fashion, letting $S_k = S_{k-1}\begin{bmatrix} I_k & 0 \\ 0 & Q_k \end{bmatrix}$, and we see that $U = S_n^* A S_n$ is upper triangular. Letting $S = S_n$ we see that $A = SUS^*$.

Remark 5.1.15. Componentwise, the equation (5.2) becomes

$$1 - 2|w_1|^2 = u_1, -2w_i\bar{w}_1 = u_i, \text{ for } i = 2, \cdots, n.$$

Since $u_1 \le 0$, $w_1 = \pm\sqrt{\frac{1-u_1}{2}} \neq 0$ (real). So we have

$$w_i = -\frac{u_i}{2w_1}, \ i = 2, \cdots, n.$$

One can check that

$$\mathbf{w}^*\mathbf{w} = \sum_{i=1}^{n} |w_i|^2 = \frac{1-u_1}{2} + \sum_{i=2}^{n} \frac{|u_i|^2}{4|w_1|^2} = \frac{1-u_1}{2} + \frac{1-u_1^2}{2(1-u_1)} = 1.$$

Except a multiplicative constant, this process produces the same vector as (5.1).

Corollary 5.1.16. *If M is Hermitian, then T is diagonal.*

Proof.

$$(U^H M U)^H = U^H M^H U = U^H M U = T = T^H.$$

Moreover, since T has real diagonal elements, the eigenvalues of M are real. □

Vector norm and its subordinate matrix norm

Definition 5.1.17. A vector norm $\|\cdot\| : \mathbb{C}^n \to \mathbb{R}^1$ is a function satisfying

(1) $\|\mathbf{x}\| \ge 0$ and $\|\mathbf{x}\| = 0$ iff $\mathbf{x} = 0$

(2) $\|c\mathbf{x}\| = |c|\,\|\mathbf{x}\|$

(3) $\|\mathbf{x} + \mathbf{y}\| \le \|\mathbf{x}\| + \|\mathbf{y}\|$.

Example 5.1.18. (1) $\|\mathbf{x}\|_1 = \sum |x_i|$

(2) $\|\mathbf{x}\|_2 = (\sum |x_i|^2)^{\frac{1}{2}} = (\mathbf{x}^H \cdot \mathbf{x})^{\frac{1}{2}} \equiv (\mathbf{x}, \mathbf{x})^{\frac{1}{2}}$

(3) $\|\mathbf{x}\|_p = (\sum |x_i|^p)^{\frac{1}{p}}$ for $1 \le p < \infty$.

(4) $\|\mathbf{x}\|_\infty = \max_i |x_i|$.

Exer. prove that $\|\cdot\|_2$ satisfies (3).

$$\begin{aligned}\|\mathbf{x}+\mathbf{y}\|_2^2 &= (\mathbf{x}+\mathbf{y})^H(\mathbf{x}+\mathbf{y}) = \|\mathbf{x}\|_2^2 + \|\mathbf{y}\|_2^2 + \mathbf{y}^H\mathbf{x} + \mathbf{x}^H\mathbf{y} \quad \text{(real number)} \\ &\le \|\mathbf{x}\|_2^2 + \|\mathbf{y}\|_2^2 + 2(\sum |y_i|^2)^{1/2} \cdot (\sum |x_i|^2)^{1/2} \\ &= (\|\mathbf{x}\|_2 + \|\mathbf{y}\|_2)^2.\end{aligned}$$

Theorem 5.1.19. *We have*

(1) $\|\mathbf{x}\|_\infty \le \|\mathbf{x}\|_2 \le \|\mathbf{x}\|_1 \le n\|\mathbf{x}\|_\infty$

(2) $\|\mathbf{x}\|_2 \le n^{\frac{1}{2}}\|\mathbf{x}\|_\infty$

(3) $n^{-\frac{1}{2}}\|\mathbf{x}\|_1 \le \|\mathbf{x}\|_2$.

Exercise 5.1.20. Show

(1) $\|\mathbf{x}\|_2 \le n^{\frac{1}{2}}\|\mathbf{x}\|_\infty$

(2) $n^{-\frac{1}{2}}\|\mathbf{x}\|_1 \le \|\mathbf{x}\|_2$

(3) Show that $\|\mathbf{x}\|_p$ is a norm for $1 \le p < \infty$, but not a norm for $0 < p < 1$.

1982 MAA Ratio of Matrix Norms E 2847 [1980, 671]. Proposed by Emeric Deutsch, Polytechnic Institute of New York, Brooklyn. For the $n \times n$ real matrix $A = [a_{ij}]$, define

$$\|A\|_\infty = \max_i \sum_{j=1}^n |a_{ij}|; \|A\|^2 = \max \textit{ eigenvalue of } A*A; \|\!|A|\!\|^2 = \sum_{j=1}^n |a_{ij}|^2.$$

Set $Re\ A = \frac{1}{2}(A + A*)$. Find

$$(i) \max_{A\ge 0, \ne 0} \|A\|_\infty / \|\!|Re\ A|\!\|; \quad (ii) \max_{A\ge 0, \ne 0} \|A\|_\infty / \|\!|Re\ A|\!\|.$$

Here $A*$ denotes the transpose of A; $A > 0$ means A has no negative element.

The answer is

$$(i) 2\sqrt{(n-1)}\ if\ n \ge 2 (and\ 1\ if\ n = 1); \quad (ii)\sqrt{(2n-1)}.$$

Definition 5.1.21. We say $f : \mathbb{C}^n \to \mathbb{R}$ is uniformly continuous, if for each $\mathbf{x} \in \mathbb{C}^n$, and given $\varepsilon > 0$ there exists a $\delta > 0$ such that $|f(\mathbf{x}) - f(\mathbf{y})| < \varepsilon$, whenever $\|\mathbf{x} - \mathbf{y}\|_\infty < \delta$.

Theorem 5.1.22. *If $\|\cdot\|$ is any norm, then it is a continuous function.*

Proof.

$$\begin{aligned} |\,\|\mathbf{x}\| - \|\mathbf{y}\|\,| &\leq \|\mathbf{x}-\mathbf{y}\| = \left\|\sum_{i=1}^{n}(x_i-y_i)e_i\right\| \leq \max|x_i-y_i|\sum_{i=1}^{n}\|e_i\| \\ &= \|\mathbf{x}-\mathbf{y}\|_\infty \sum_{i=1}^{n}\|e_i\|. \end{aligned}$$

Thus if we choose $\delta < \frac{\epsilon}{\sum_{i=1}^{n}\|e_i\|}$, then we see

$$|\,\|\mathbf{x}\| - \|\mathbf{y}\|\,| < \epsilon.$$

□

Definition 5.1.23. We say two norms $\|\cdot\|_\alpha$ and $\|\cdot\|_\beta$ are equivalent if there exist positive constants C_0, C_1 such that

$$C_0\|\mathbf{x}\|_\alpha \leq \|\mathbf{x}\|_\beta \leq C_1\|\mathbf{x}\|_\alpha, \quad \text{for all } \mathbf{x} \in V.$$

Theorem 5.1.24. *In a finite dimensional normed linear space V, all norms are equivalent. In other words, if $\|\cdot\|_\alpha$ and $\|\cdot\|_\beta$ are any two norms, then there exist positive constants C_0, C_1 such that $C_0\|\mathbf{x}\|_\alpha \leq \|\mathbf{x}\|_\beta \leq C_1\|\mathbf{x}\|_\alpha$ for all $\mathbf{x} \in V$.*

Proof. We shall show every norm is equivalent to $\|\cdot\|_\infty$. First let $S = \{\mathbf{y} \in V : \|\mathbf{y}\|_\infty = 1\}$. Then S is closed and bounded. Let $\|\cdot\|$ be any other norm. Since any norm is a continuous function, it assumes a positive max and min on S, i.e., $0 < m \leq \|\mathbf{y}\| \leq M$ on S. Then for any $\mathbf{x} \neq 0 \in \mathbb{C}^n$, $\frac{\mathbf{x}}{\|\mathbf{x}\|_\infty} \in S$. Hence $m \leq \left\|\frac{\mathbf{x}}{\|\mathbf{x}\|_\infty}\right\| \leq M$. In other words,

$$m\|\mathbf{x}\|_\infty \leq \|\mathbf{x}\| \leq M\|\mathbf{x}\|_\infty.$$

This inequality obviously holds for $\mathbf{x} = 0$. Finally transitivity of equivalence relation proves the result. □

Definition 5.1.25. Let $\|\cdot\|$ be a vector norm, and let A be in $\mathbb{C}^{n,n}$. Then the norm $\|A\|$ of A defined by

$$\|A\| := \sup_{\|\mathbf{x}\|\neq 0} \frac{\|A\mathbf{x}\|}{\|\mathbf{x}\|} \equiv \sup_{\|\mathbf{x}\|=1} \|A\mathbf{x}\|$$

is called the **matrix norm subordinate** to the vector norm $\|\cdot\|$.

Properties of a matrix norm:

(1) $\|A\| \geq 0$ and $\|A\| = 0$ iff $A \equiv 0$

(2) $\|cA\| = |c|\,\|A\|$

(3) $\|A + B\| \leq \|A\| + \|B\|$

(4) $\|A \cdot B\| \leq \|A\| \cdot \|B\|$

(5) $\|A\mathbf{x}\| \leq \|A\| \cdot \|\mathbf{x}\|$

Definition 5.1.26. Let $A \in \mathbb{C}^{n,n}$ have eigenvalues $\{\lambda_i\}$ ordered in such a way that $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n|$. Then $\rho(A) = |\lambda_1|$ is called the **spectral radius** of A.

Theorem 5.1.27. *Let $A = [a_{ij}] \in \mathbb{C}^{n,n}$. Then we have*

(1) $\|A\|_1 = \max_j \sum_i |a_{ij}|$*,(maximum column sum),*

(2) $\|A\|_2 = \sqrt{\rho(A^H A)}$*,* $\|A\|_2 = \rho(A)$ *if* $A = A^H$*,*

(3) $\|A\|_\infty = \max_i \sum_j |a_{ij}|$*, (maximum row sum).*

Proof. (2) We note that

$$\|A\|_2 = \sup_{\|\mathbf{x}\|_2=1} \|A\mathbf{x}\|_2 = \sup \sqrt{(A\mathbf{x}, A\mathbf{x})} = \sup \sqrt{\mathbf{x}^H A^H A\mathbf{x}}.$$

Since $A^H A$ is Hermitian, it has a complete set of orthonormal eigenvectors, $\mathbf{v}_1, \ldots, \mathbf{v}_n$ corresponding to eigenvalues $\{\mu_j\}_{j=1}^n$. Moreover, the eigenvalues are non-negative since $A^H A$ positive semi-definite. We arrange them so that $\mu_1 \geq \mu_2 \cdots \geq \mu_n \geq 0$. Since any $\mathbf{x}$ can be written as $\mathbf{x} = \sum_{j=1}^n c_j \mathbf{v}_j$, we have

$$0 \leq (\sum_{j=1}^n c_j \mathbf{v}_j)^H A^H A (\sum_{j=1}^n c_j \mathbf{v}_j) = (\sum_{j=1}^n c_j \mathbf{v}_j)^H (\sum_{j=1}^n \mu_j c_j \mathbf{v}_j) = \sum_{j=1}^n \mu_j |c_j|^2.$$

But since $\|\mathbf{x}\|_2 = \sum_{j=1}^n |c_j|^2 = 1$, we see

$$\sup_{\|\mathbf{x}\|_2=1} \sqrt{\sum_{j=1}^n \mu_j |c_j|^2} = \sqrt{\mu_1}.$$

□

H.W prove (1) and (3).

Theorem 5.1.28. *If λ is any eigenvalue of A and $\|\cdot\|$ is any matrix norm, then $|\lambda| \leq \|A\|$. Furthermore, if $\varepsilon > 0$ is given, then there exists a norm $\|\cdot\|_\alpha$ such that $\|A\|_\alpha \leq \rho(A) + \varepsilon$.*

Proof. First statement is left to the reader. For the second assertion, we recall Schur's lemma. There exists a unitary matrix U such that $U^H A U = T$ is upper triangular whose diagonal elements are eigenvalues of A. Let $D = \text{Diag}(\alpha^1, \alpha^2, \ldots, \alpha^n)$, with $\alpha > 0$. Define $\|\cdot\|_\alpha$ by $\|\mathbf{x}\|_\alpha = \|D^{-1} U^H \mathbf{x}\|_\infty$. Then

we see $(D^{-1}TD)_{ij} = t_{ij}\alpha^{j-i}$ and we compute the norm of A subordinate to the vector norm $\|\cdot\|_\alpha$.

$$\begin{aligned}
\|A\|_\alpha &= \sup_{\|\mathbf{x}\|_\alpha=1} \|A\mathbf{x}\|_\alpha = \sup_{\|D^{-1}U^H\mathbf{x}\|_\infty=1} \|D^{-1}U^HA\mathbf{x}\|_\infty \\
&= \sup_{\|D^{-1}U^H\mathbf{x}\|_\infty=1} \|D^{-1}U^HAUDD^{-1}U^H\mathbf{x}\|_\infty = \sup_{\|\mathbf{y}\|_\infty=1} \|D^{-1}U^HAUD\mathbf{y}\|_\infty \\
&= \sup_{\|\mathbf{y}\|_\infty=1} \|D^{-1}TD\mathbf{y}\|_\infty = \sup_{\|\mathbf{y}\|_\infty=1} \max_i \left(\sum_{j\geq i} t_{ij}\alpha^{j-i}y_j\right) \\
&\leq \sup_{\|\mathbf{y}\|_\infty=1} \max_i \left(\sum_{j\geq i} |t_{ij}\alpha^{j-i}|\right) \\
&= \max_i \left(|t_{ii}| + \sum_{j>i} |t_{ij}|\alpha^{j-i}\right) \\
&\leq \max_i |t_{ii}| + \varepsilon = \rho(A) + \varepsilon,
\end{aligned}$$

if α is sufficiently small. □

Definition 5.1.29 (Convergence of a matrix)**.** We say A^k **converges to zero** if the entries of A^k converges to zero.

Corollary 5.1.30. *For $A \in \mathbb{C}^{n,n}$, we have $\lim_k A^k = 0$ if and only if $\rho(A) < 1$.*

Proof. Suppose that $\rho(A) < 1$. Then by above theorem, there exists an $\alpha > 0$ such that $\|A\|_\alpha \leq \rho(A) + \frac{1-\rho(A)}{2} < 1$. So

$$\|A^k\|_\alpha \leq \|A\|_\alpha^k \leq (\frac{\rho(A)+1}{2})^k \longrightarrow 0.$$

By the norm equivalences theorem for matrices, we conclude that

$$\lim_k \|A^k\|_\infty = 0,$$

i.e., the maximum row sum approaches 0. Thus, each entry approaches zero. Conversely, assume $\lim_k A^k = 0$. Then $\lim_k \|A^k\|_\infty = 0$. But then $\|A^k\| \geq \rho(A^k) = \rho^k(A) \to 0$. Hence $\rho(A) < 1$.
(Alternative proof) We transform A into a Jordan canonical form $S^{-1}AS = J$. Then $S^{-1}A^kS = J^k \to 0$ if the eigenvalues are < 1. Hence $A^k \to 0$. □

Banach Fixed Point Theorem

Definition 5.1.31. Let S be a subset of a normed linear space X. If every Cauchy sequence in S converges in S, we say S is *complete*.

Definition 5.1.32. Let S be a subset of a normed linear space X. An operator $A : S \to X$ is called a *contraction operator* if there is a constant $\lambda \in [0, 1)$ such that

$$\|Ax - Ay\| \leq \lambda \|x - y\|$$

for all $x, y \in S$.

Theorem 5.1.33 (Banach)**.** *Let S be a complete subset of a normed space X and $A : S \to A$ be a contraction. Then A has a unique fixed point. Furthermore, the sequence generated by*

$$x_{n+1} := Ax_n, \ n = 0, 1, 2, \cdots,$$

with any starting point x_0 converges to the unique fixed point x and we have the a priori error estimate

$$\|x_n - x\| \leq \frac{\lambda^n}{1 - \lambda} \|x_1 - x_0\|.$$

Proof. Left as an exercise. □

Exercise 5.1.34. (1) Let A be a $n \times n$ symmetric, positive definite matrix. Show that $\sqrt{\langle A\cdot, \cdot \rangle}$ is a norm on $\mathbb{R}^n$.

(2) Let X, Y, Z be three normed linear spaces and let $B : X \to Y$ and $A : Y \to Z$ are bounded linear operators. Show that $\|AB\| \leq \|A\|\|B\|$.

(3) Show the following:

Let $f : [0, \infty) \to [0, \infty)$ be given by

$$f(x) = x + \frac{1}{1 + x}$$

satisfy $|f(x) - f(y)| < |x - y|$ for all $x \geq 0$. However, it does not have a fixed point.

(4) Can you find another nontrivial example in the above problem ? especially vector valued case?

$$\mathbf{f}(\mathbf{x}) = (x_1 + \frac{1}{x_1 + x_2}, x_2 + \frac{1}{x_1 + x_2})$$

(5) Let X, Y be normed linear spaces and denote by $L(X, Y)$ the linear space of all bounded linear operators $A : X \to Y$. Show that the space $L(X, Y)$ equipped with the norm

$$\|A\| := \sup_{\|x\|=1} \|Ax\|$$

is again a normed space and $L(X, Y)$ is a Banach space if Y is a Banach space.

(6) (a) Show that for any $n \times n$ matrix A with $\|A\| < 1$, the series

$$A + \frac{A^2}{2} + \frac{A^3}{3} + \cdots$$

converges and denote its limit by $-\ln(I - A)$.

(b) Show that if λ is an eigenvalue of A then $-\ln(1-\lambda)$ the eigenvalue of $-\ln(I - A)$.

5.2 System of linear equations

$$\sum_{j=1}^{n} a_{ij}x_j = b_i, \qquad i = 1, \ldots, m(m \geq n)$$

$$A\mathbf{x} = \mathbf{b}, \qquad A \in \mathbb{C}^{m,n}, \quad b \in \mathbb{C}^m$$

We say $A\mathbf{x} = \mathbf{b}$ is **consistent** if it has at least one solution.

Definition 5.2.1. $[A : \mathbf{b}] : m \times (n+1)$ is called augmented matrix.

Theorem 5.2.2. *$A\mathbf{x} = \mathbf{b}$ is consistent iff*

$$\operatorname{rank}[A : \mathbf{b}] = \operatorname{rank}[A],$$

i.e, if $\mathbf{b}$ can be obtained by a linear combination of columns of A. In this case, the coefficient is a solution $\mathbf{x}$.

Theorem 5.2.3. *If $A\mathbf{x}_p = \mathbf{b}$, then any solution of $A\mathbf{x} = \mathbf{b}$ has the form $\mathbf{y} = \mathbf{x}_p + \mathbf{w}$, where $\mathbf{w} \in \ker(A)$.*

Theorem 5.2.4. *If $A \in \mathbb{C}^{m,n}$ has a submatrix of order r such that the submatrix has nonzero determinant and r is the largest such number, then*

$$\operatorname{rank} A = r.$$

Corollary 5.2.5. *$A\mathbf{x} = \mathbf{b}$ has a unique solution iff $\operatorname{rank}(A) = n$ iff there exists some $n \times n$ submatrix with nonzero determinant.*

Cramer's rule

If $m = n$ and A is nonsingular, then

$$\begin{aligned} A\mathbf{x} &= \mathbf{b} \\ \mathbf{x} &= A^{-1}\mathbf{b}, \qquad A^{-1} = \frac{1}{\det A}[c_{ij}], \end{aligned}$$

where $c_{ij} = (-1)^{i+j} \det A_{ji}$ is the cofactor obtained by deleting j-th row and i-th column. In other words,

$$A^{-1} = \frac{1}{\det A}(Adj(A)).$$

Cramer's rule is useful for theoretical analysis. However, it's computational cost is high and hardly used for $n > 3$.

5.2.1 Gaussian elimination-LU decomposition

Let $A \in \mathbb{C}^{n,n}$ be nonsingular. We would like to transform the equation $A\mathbf{x} = \mathbf{b}$ into $LU\mathbf{x} = \mathbf{b}$ or $U\mathbf{x} = \mathbf{b}'$, where $U(L)$ is upper(lower) triangular. The resulting upper triangular system of equation

$$\sum_{j \geq i} u_{ij}^{(1)} x_j = b_i', \qquad i = 1, \ldots, n$$

is easily solved by back(forward) substitution.

Step 1; Create zeros on the first column

We first **create** 0 **at** $(2,1)$ **position while adjusting the entire 2nd row**. To do so we assume $a_{11} \neq 0$ and let $m_{21} = \frac{a_{21}^{(1)}}{a_{11}}$. Then do

$$a_{2j}^{(2)} = a_{2j}^{(1)} - m_{21} a_{1j}^{(1)} \quad j = 2, \ldots, n; \tag{5.4}$$

Repeat procedure (5.4) for $i = 3, \cdots, n$: **Create** 0 **at** $(i,1)$ **position while adjusting the entire** i**-th row**.
Let $m_{i1} = \frac{a_{i1}^{(1)}}{a_{11}}$. Set

$$a_{ij}^{(2)} = a_{ij}^{(1)} - m_{i1} a_{1j}^{(1)}, \quad j = 2, \ldots, n.$$

Resulting matrix looks like:

$$(A' : \mathbf{b}') = \left[\begin{array}{c|ccc|c} a_{11}^{(1)} & a_{12}^{(1)} & \ldots & a_{1n}^{(1)} & b_1^{(1)} \\ \hline 0 & a_{22}^{(2)} & \ldots & a_{2n}^{(2)} & b_2^{(2)} \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & a_{n2}^{(2)} & \ldots & a_{nn}^{(2)} & b_n^{(2)} \end{array}\right]$$

operation count

of divisions for m_{i1}, $(2 \leq i \leq n)$ is $n-1$ and # of multiplication for adjusting $(n-1) \times (n-1)$ submatrix is $(n-1)^2$. (Same for addition)

The matrix (elementary) involved is $E_{21}(m_{21}), \ldots, E_{n1}(m_{n1})$, where

$$E_{ij}(r) \equiv \begin{matrix} \\ \\ i-th \\ \\ \end{matrix} \begin{bmatrix} 1 & & & 0 \\ & \ddots & & \\ & & \ddots & \\ & r & & 1 \end{bmatrix}$$

is an elementary row matrix whose action, when multiplied on the left, is to add r-times of j-th row to i-th row. Let $M_1 = E_{n1}(m_{n1}) \ldots E_{21}(m_{21})$. Then

$$[A' : \mathbf{b}'] = E_{n1}(m_{n1}) \ldots E_{21}(m_{21})[A : \mathbf{b}] = M_1[A : \mathbf{b}].$$

Observe

(1) M_1 is nonsingular,

(2) E_{ij} $(i > j)$ are elementary row operations whose entries are m_{ij}.

(3) $\det M_1 = 1$,

(4) M_1 is lower triangular.

Now M_1 and its inverse are given by the following form.

Lemma 5.2.6.

$$M_1 = \left[\begin{array}{c|c} 1 & 0 \\ \hline -\mathbf{m}_1 & I_{n-1} \end{array}\right], \quad M_1^{-1} = \left[\begin{array}{c|c} 1 & 0 \\ \hline \mathbf{m}_1 & I_{n-1} \end{array}\right], \tag{5.5}$$

where $\mathbf{m}_1 = [m_{21}, m_{31}, \cdots, m_{n1}]^T$ *and* I_{n-1} *is* $(n-1)\times(n-1)$ *identity matrix.*

Since

$$M_1 A = \left[\begin{array}{c|c} a_{11} & A_{12}^{(1)} \\ \hline 0 & A_{22}^{(2)} \end{array}\right] \equiv U_1 = \left[\begin{array}{c|c} I_1 & 0 \\ \hline 0 & A_{22}^{(2)} \end{array}\right] \left[\begin{array}{c|c} a_{11} & A_{12}^{(1)} \\ \hline 0 & I_{n-1} \end{array}\right], \tag{5.6}$$

we have

$$A = \left[\begin{array}{c|c} I_1 & 0 \\ \hline \mathbf{m}_1 & I_{n-1} \end{array}\right] \left[\begin{array}{c|c} I_1 & 0 \\ \hline 0 & A_{22}^{(2)} \end{array}\right] \left[\begin{array}{c|c} a_{11} & A_{12}^{(1)} \\ \hline 0 & I_{n-1} \end{array}\right]. \tag{5.7}$$

Step 2: Create zeros on the second column

Assume $a_{22}^{(2)} \neq 0$ and let $m_{32} = \frac{a_{32}^{(2)}}{a_{22}^{(2)}}$, and set

$$a_{3j}^{(3)} = a_{3j}^{(2)} - m_{32} a_{2j}^{(2)}, \quad j = 3, \ldots, n. \tag{5.8}$$

Repeat procedure (5.8) for for $i = 4, \cdots, n$: Let $m_{i2} = \frac{a_{i2}^{(2)}}{a_{22}^{(2)}}$ and set

$$a_{ij}^{(3)} = a_{ij}^{(2)} - m_{i2} a_{2j}^{(2)}, \quad j = 3, \ldots, n.$$

Then we see

$$E_{n2}(m_{n2}) \cdots E_{i2}(m_{i2}) \cdots E_{32}(m_{32})[A^1 : \mathbf{b}^1]$$

$$= \left[\begin{array}{cc|ccc} a_{11}^{(1)} & & \cdots\cdots\cdots & a_{1n}^{(1)} & b_1^{(1)} \\ 0 & a_{22}^{(2)} & \cdots\cdots & a_{2n}^{(2)} & b_2^{(2)} \\ \hline \vdots & 0 & a_{33}^{(3)} \quad \cdots & a_{3n}^{(3)} & b_3^{(3)} \\ \vdots & \vdots & & & \vdots \\ 0 & 0 & a_{n3}^{(3)} \quad \cdots & a_{nn}^{(3)} & b_n^{(3)} \end{array}\right]$$

# of divisions is $n-2$ for m_{i2}, for $3 \leq i \leq n$
# of multiplication is $(n-2)^2$

Step 3

Continuing above process, we get(as long as no zero elements appear in the pivot) a sequence of matrices

$$M_r = \begin{bmatrix} I_r & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \vdots & -\mathbf{m}_r & \ddots & \\ 0 & & 0 & 1 \end{bmatrix},$$

where $\mathbf{m}_r = [m_{r+1,r}, \cdots, m_{nr}]^T$.

$$A = \begin{bmatrix} I_k & 0 \\ L_{21}^{(k)} & I_{n-k} \end{bmatrix} \begin{bmatrix} I_k & 0 \\ 0 & A_{22}^{(k)} \end{bmatrix} \begin{bmatrix} A_{11}^{(k)} & A_{12}^{(k)} \\ 0 & I_{n-k} \end{bmatrix}. \tag{5.9}$$

This process is called **Gaussian elimination**.

Now the pseudo algorithm for Gaussian elimination(LU-decomposition) is

input n, a_{ij}, (b_i)
for $k = 1, \cdots, n-1$ do
  for $i = k+1, \cdots, n$ do
    $m_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}$;
    for $j = k+1, \cdots, n$ do
      $a_{ij}^{(k+1)} \leftarrow a_{ij}^{(k)} - m_{ik} a_{kj}^{(k)}$
    end
  end
end

It involves a construction of lower triangular matrix L such that

(1) $L^{-1}A$ is upper triangular

(2) L is nonsingular

(3) $\det L = 1$

(4) L^{-1} is also triangular and $L^{-1} = M_{n-1} \cdots M_1$.

$$L^{-1}A = U \qquad \therefore A = LU.$$

Here direct computation of $M_{n-1} \cdots M_1$ is not so neat, but $L = M_1^{-1} M_2^{-1} \cdots M_{n-1}^{-1}$ is!

Lemma 5.2.7. $\{E_{21}(r)\}^{-1} = E_{21}(-r)$.

In fact, we have

$$L = M_1^{-1} M_2^{-1} \cdots M_{n-1}^{-1},$$

where

$$\begin{aligned} M_1^{-1} &= E_{n1}^{-1}(-m_{n1}) \cdots E_{21}^{-1}(-m_{21}) = E_{n1}(m_{n1}) \cdots E_{21}(m_{21}) \\ &= \cdots \\ M_k^{-1} &= E_{nk}^{-1}(-m_{nk}) \cdots E_{k+1,k}^{-1}(-m_{k+1,k}) = E_{nk}(m_{nk}) \cdots E_{k+1,k}(m_{k+1,k}). \end{aligned}$$

$$M_k^{-1} = \begin{bmatrix} I_k & 0 & & \\ 0 & 1 & 0 & \\ & \mathbf{m}_k & \ddots & \\ 0 & & 0 & 1 \end{bmatrix}.$$

Incidently we have obtained a LU-decomposition.

Lemma 5.2.8.

$$M_1^{-1} M_2^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ m_{21} & 1 & 0 & 0 & 0 \\ \vdots & m_{32} & 1 & 0 & 0 \\ \cdot & \cdot & 0 & \ddots & \\ m_{n1} & m_{n2} & 0 & \cdots & 1 \end{bmatrix}.$$

Apply similar idea repeatedly, we can find L.

Lemma 5.2.9. *The following block matrix multiplication formula holds:*

$$\begin{bmatrix} I_k & 0 \\ M_{21} & I_{n-k} \end{bmatrix} \begin{bmatrix} I_k & 0 \\ 0 & M_{22} \end{bmatrix} = \begin{bmatrix} I_k & 0 \\ M_{21} & M_{22} \end{bmatrix}. \tag{5.10}$$

Thus

$$L = M_1^{-1} M_2^{-1} \cdots M_{n-1}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ \mathbf{m_1} & 1 & 0 & 0 & 0 \\ & \mathbf{m_2} & 1 & 0 & 0 \\ \vdots & \vdots & \mathbf{m_3} & \ddots & \vdots \\ \cdot & \cdot & \cdot & \cdots & 1 \end{bmatrix}. \tag{5.11}$$

Corollary 5.2.10. *The entries of $L = (\ell_{ij})$ are*

$$\begin{aligned} \ell_{ii} &= 1, \quad i = 1, \cdots n \\ \ell_{ik} &= m_{ik}, \quad i = k+1, \cdots n, k = 1, \cdots, n \\ \ell_{ik} &= 0, \quad i < k. \end{aligned}$$

Back substitution

From the above process we have obtained $U\mathbf{x} = \mathbf{b}'$, which can be easily solved by back substitution:

$$\begin{array}{rrrrcl} u_{11}x_1+ & u_{12}x_2+ & \cdots+ & u_{1n}x_n & = & b'_1 \\ & u_{22}x_2+ & \cdots+ & u_{2n}x_n & = & b'_2 \\ & & \cdots & & & \\ & & & u_{nn}x_n & = & b'_n. \end{array}$$

This can be solved easily ba back substitution:

$$x_n = \frac{b'_n}{u_{nn}}, \cdots, \quad x_i = \frac{(b'_i - \sum_{j>i} u_{i,j}x_j)}{u_{ii}}, \cdots, \quad x_1 = \frac{(b'_1 - \sum_{j>1} u_{1j}x_j)}{u_{11}}.$$

Back substitution algorithm in pseudo code is

```
input n, a_ij, (b_i)
for i = n, ··· ,1 do
    sum = 0
    for j = i + 1, ··· , n do
        sum ← sum + a_ij x_j
    x_i = (b_i − sum)/a_ii
    end
end
```

Total computational complexity

Now we count the number of operations for LU decomposition(only count multiplications and divisions ignoring additions and subtractions) : For $k = 1, 2, \cdots, n-1$, each step involves $n-k$ operations for the multiplier vector $\mathbf{m}_k$ a total of $\sum_1^{n-1}(n-k) = \frac{n(n-1)}{2}$ operations. Next, elimination of k-th column involves $(n-k)^2$ a total of $\sum_{k=1}^{n-1}(n-k)^2$ operations.

$$\begin{array}{rl} & n-k, \qquad k = 1, \ldots, n-1 \quad \text{for } m_{ik} \\ \therefore & \sum_1^{n-1}(n-k) = \frac{n(n-1)}{2} \\ & (n-k)^2, \qquad k = 1, \ldots, n-1 \quad \text{for multiplication.} \\ \therefore & \sum_{k=1}^{n-1}(n-k)^2 = \sum_{s=1}^{n-1} s^2 = \frac{(n-1)n(2n-1)}{6} = O(\frac{n^3}{3}). \end{array}$$

The number of operations for right hand side and back substitution is

$$\begin{array}{rl} & n-k, \qquad k = 1, \ldots, n-1 \quad \text{for the 1-right hand side} \\ \therefore & \sum_{k=1}^{n-1}(n-k) = \frac{n(n-1)}{2} \\ & k, \qquad k = 1, \cdots, n \quad \text{for the back substitution} \\ \therefore & \sum_1^n k = \frac{n(n+1)}{2}. \end{array}$$

Here, we ignored addition.

Total: $\frac{n(n^2-1)}{3}$ for LU decomposition, $\frac{mn(n-1)}{2}$ for the m right hand sides and $\frac{mn(n+1)}{2}$ for the back substitution.

Remark 5.2.11. Gaussian elimination without pivoting is mathematically equivalent to the unique triangular decomposition LU with L unit lower triangular matrix. But LU decomposition has computational advantage when there are several right hand sides. In general, $A\mathbf{x} = \mathbf{b}$ is equivalent to $L\mathbf{y} = \mathbf{b}$ and $U\mathbf{x} = \mathbf{y}$ and solve two systems by forward and backward substitution.

Numerical stability and partial pivoting

If $a_{rr}^{(r)}$ is either zero or very small during the LU decomposition, the rounding error is likely to be large. One way to correct this phenomenon is to interchange the rows during the elimination so that the pivoting element is largest, called partial pivoting.

Example 5.2.12. Consider solving

$$10^{-6}x_1 + x_2 = 0.501 \tag{5.12}$$
$$x_1 - x_2 = 999.5 \tag{5.13}$$

The exact solution is $x_1 = 1,000, x_2 = 0.5$. We use 6 digit decimal arithmetic. Eliminating x_1 from the first row, we get

$$-(10^6 + 1)x_2 = -500,000.5$$

Since $1/(10^6 + 1) = 10^{-6}$ with 6 digit decimal arithmetic, we have $x_2 = 500,000.5 * 10^{-6} = 0.5000005 = 0.500001$ (rounded). Substituting into (5.12), we obtain

$$10^{-6}x_1 = 0.501 - 0.500001 = 0.000999$$

and hence $x_1 = 999$.

Partial pivoting

If some $a_{kk}^{(k)} = 0$ (zero pivot) or very small, we choose a p such that $|a_{pk}| = \max_{i \geq k} |a_{ik}|$. Such p exists since $\det A \neq 0$.) We interchange p-th row with k-th row including the right hand side vector $\mathbf{b}$. (In the actual coding, we do not interchange rows. Instead, we merely permute the index using a permutation vector $\mathbf{p}$.)

Now the pseudo algorithm for Gaussian elimination with partial pivoting is

input n, a_{ij}, (b_i)
for $i = 1, \cdots, n$ do
  $p_i \leftarrow i$
end
For $k = 1, \cdots, n-1$ do
  Determine $j \geq k$ so $|a_{p_j k}| = \max_{i \geq k} |a_{p_i k}|$

Swap $p_k \leftrightarrow p_j$
for $i = k+1, \cdots, n$ do
$m_{p_i k} = \frac{a^{(k)}_{p_i k}}{a^{(k)}_{p_k k}}$;
$a^{(k)}_{p_i k} = m_{p_i k}$ (this is to overwrite A by L)
for $j = k+1, \cdots, n$ do
$a^{(k+1)}_{p_i j} \leftarrow a^{(k)}_{p_i j} - m_{p_i k} a^{(k)}_{p_k j}$
end
end
end

The result of partial pivoting is of the form

$$L_{n-1} P_{n-1} \cdots L_1 P_1 A = U, \tag{5.14}$$

where P_i are permutations. Is this LU decomposition ? No! Clearly, the following matrix

$$(L_{n-1} P_{n-1} \cdots L_1 P_1)^{-1}$$

is not a lower triangular matrix. But by multiplying $P = P_{n-1} \cdots P_1$, we can show it is an LU decomposition.

Theorem 5.2.13. *Let $P = P_{n-1} \cdots P_1$. Then we have*

$$PA = LU,$$

where

$$\begin{aligned} L &= P(L_{n-1} P_{n-1} \cdots L_1 P_1)^{-1} \\ &= P_{n-1} \cdots P_2 P_1 P_1^{-1} L_1^{-1} P_2^{-1} L_2^{-1} \cdots P_{n-1}^{-1} L_{n-1}^{-1} \\ &= (P_{n-1} \cdots (P_2 L_1^{-1} P_2^{-1}) L_2^{-1} \cdots P_{n-1}^{-1}) L_{n-1}^{-1} \end{aligned}$$

is a lower triangular matrix.

Proof. We see from (5.14) that

$$A = P_1^{-1} L_1^{-1} P_2^{-1} L_2^{-1} P_3^{-1} \cdots P_{n-1}^{-1} L_{n-1}^{-1} U.$$

Thus

$$P_2 P_1 A = (P_2 L_1^{-1} P_2^{-1}) L_2^{-1} P_3^{-1} \cdots P_{n-1}^{-1} L_{n-1}^{-1} U.$$

Observe that

$$L_1^{-1} = \begin{bmatrix} 1 & 0 \\ \mathbf{m}_1 & I_{n-1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \mathbf{0} \\ m_1 & 1 & \mathbf{0} \\ \mathbf{m}^* & 0 & I_{n-2} \end{bmatrix} \tag{5.15}$$

and P_2 is a permutation of rows not involving the first row. Hence we see $L_1' := P_2 L_1^{-1} P_2^{-1}$ has exactly the same form as L_1^{-1}.(By permuting rows and

columns of $(n-1)$ identity matrix twice, I_{n-1} remain unchanged. Meanwhile we exchange the corresponding entries of $\mathbf{m_1}$). Now

$$P_3P_2P_1A = (\underbrace{P_3L_1'L_2^{-1}P_3^{-1}}_{L_2'})L_3^{-1}\cdots P_{n-1}^{-1}L_{n-1}^{-1}U.$$

By Lemma 5.2.8, $L_1'L_2^{-1}$ is of the form :

$$L_1'L_2^{-1} = \begin{bmatrix} 1 & 0 & \mathbf{0} \\ m_1 & 1 & \mathbf{0} \\ \bar{\mathbf{m}}^* & 0 & I_{n-2} \end{bmatrix} \begin{bmatrix} 1 & 0 & \mathbf{0} \\ 0 & 1 & \mathbf{0} \\ 0 & \mathbf{m_2^*} & I_{n-2} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \mathbf{0} \\ m_1 & 1 & \mathbf{0} \\ \bar{\mathbf{m}}^* & \mathbf{m_2^*} & I_{n-2} \end{bmatrix}. \quad (5.16)$$

Again observe that P_3 is a permutation of rows (not involving the first two rows). Hence $L_2' := P_3L_1'L_2^{-1}P_3^{-1}$ is a form similar to $L_1'L_2^{-1}$ in (5.16). Thus

$$P_4P_3P_2P_1A = (\underbrace{P_4L_2'L_3^{-1}P_4^{-1}}_{L_3'})L_4^{-1}\cdots P_{n-1}^{-1}L_{n-1}^{-1}U = L_3'L_4^{-1}\cdots P_{n-1}^{-1}L_{n-1}^{-1}U,$$

where $L_3' := P_4L_2'L_3^{-1}P_4^{-1}$ is a lower triangular matrix having a similar block structure as $L_2'L_3^{-1}$. Repeating this process, we see

$$\begin{aligned} L &= (P_{n-1}\cdots(P_2L_1^{-1}P_2^{-1})L_2^{-1}\cdots P_{n-1}^{-1})L_{n-1}^{-1} && (5.17)\\ &= (P_{n-1}L_{n-3}'L_{n-2}^{-1}P_{n-1}^{-1})L_{n-1}^{-1} && (5.18)\\ &= L_{n-2}'L_{n-1}^{-1}. && (5.19) \end{aligned}$$

Here the product of last matrices is a lower triangular matrix.

Remark 5.2.14. The result was known in Wendroff(1966), Forsythe-Moler (1967). This result says that the partial pivoting is equiv. to LU decomposition of PA without pivoting. But we do not know P in advance. The U is computed in (5.14) during the elimination using partial pivoting and L is obtained by (5.17) (row exchange to L_k^{-1} for $k = 1, \cdots, n-1$).

5.2.2 Keeping track of Permutation

Note $P = P_{n-1}\cdots P_1$ is the product of permutation matrices involved in the pivoting process. We do not save the matrices $P_k's$. Instead, we start with a pivot vector $\mathbf{p}_0 = [1, 2, 3, \cdots, n]$ and let $\mathbf{p}_k = P_k\mathbf{p}_0$ permute it according to P.Since the system $A\mathbf{x} = \mathbf{b}$ is changed to $PA\mathbf{x} = P\mathbf{b}$, we need to permute $\mathbf{b}$ accordingly to form $P\mathbf{b}$.

Example 5.2.15. Assume $n = 4$. If the final $\mathbf{p} = \mathbf{p}_3$ is $[2, 4, 3, 1]$, then we see that the rows of PA are the rows of A in the order of $2, 4, 3, 1$. So the right hand side $\mathbf{b}$ can be permuted in the same way, i.e., $P\mathbf{p} = (b_2, b_4, b_3, b_1)$.

□

There are cases when partial pivoting is not necessary.

Proposition 5.2.16. *If A is symmetric positive definite, then it has the following properties:*

(1) A is nonsingular; in fact, $det(A) > 0$.

(2) Let A_k be the leading principal minors of A. It can be shown that A is positive definite if and only if $det(A_k) > 0$ for $k = 1, 2, \cdots, n$.

(3) All of the diagonal elements of A are positive.

(4) The largest element of the matrix lies on the diagonal.

(5) All of the eigenvalues of A are positive.

Theorem 5.2.17. *If A is symmetric positive definite, then*

(1) $a_{ii} > 0$,

(2) $\max_i a_{ii} = \max_{i,j} |a_{ij}|$,

(3) The submatrices $(a_{ij}^{(k)})$, $1 \le k \le n$ appearing during the Gaussian eliminations are also symmetric positive definite.

(4) $a_{ii}^{(k)} \le a_{ii}^{(k-1)}$, for $k \le i \le n$.

(5) If A is, in addition, diagonally dominant, then so are the submatrices appearing in the G.E.

Proof. (1) $\mathbf{x}^* A\mathbf{x} \ge 0$, equality when (only) $\mathbf{x} = 0$. Put $\mathbf{x} = \mathbf{e}_i$

(2) Put $\mathbf{x} = \mathbf{e}_i + \mathbf{e}_j = (0, \dots, 1, 0, \dots, 1, 0, \dots)$, then $\mathbf{x}^* A\mathbf{x} = a_{ii} + a_{ij} + a_{ji} + a_{jj} > 0$, so that $a_{ii} + a_{jj} \ge -2a_{ij}$. Put $\mathbf{y} = \mathbf{e}_i - \mathbf{e}_j = (0, \dots, 1, 0, \dots, 0, -1, 0 \dots)$, we obtain $a_{ii} + a_{jj} \ge 2a_{ij}$. Thus $a_{ii} + a_{jj} \ge 2|a_{ij}|$ and $\max_i a_{ii} \ge \frac{a_{ii}+a_{jj}}{2} \ge |a_{ij}|$.

(3) **symmetry**: $a_{ij}^{(2)} = a_{ij}^{(1)} - \frac{a_{i1}^{(1)}}{a_{11}^{(1)}} a_{1j}^{(1)} \Rightarrow a_{ji}^{(2)} = a_{ij}^{(2)}$.

positive definiteness: Let $\mathbf{x} = (x_1, x_2, \cdots, x_n)$ and $\bar{\mathbf{x}} = (x_2, \cdots, x_n)$.

$$\begin{aligned}
\bar{\mathbf{x}} A^{(2)} \mathbf{x} &= \sum_{i,j=2}^{n} a_{ij}^{(2)} x_i x_j \\
&= \sum_{i,j=2} a_{ij}^{(1)} x_i x_j - \sum_{i,j=2}^{n} \frac{a_{i1}^{(1)}}{a_{11}^{(1)}} a_{1j}^{(1)} x_i x_j \quad \text{(Caution with index } i, j\text{)} \\
&= \sum_{i,j=1}^{n} a_{ij}^{(1)} x_i x_j - 2\sum_{j=2}^{n} a_{1j}^{(1)} x_1 x_j - a_{11}^{(1)} x_1^2 - a_{11}^{(1)} \left(\sum_{i=1}^{1} \frac{a_{i1}^{(1)}}{a_{11}^{(1)}} x_i \right)^2 \\
&= \mathbf{x}^T A\mathbf{x} - a_{11}^{(1)} \left[x_1 + \sum_{i=2}^{n} \frac{a_{i1}^{(1)}}{a_{11}^{(1)}} x_i \right]^2
\end{aligned}$$

Set $x_1 = -\sum_{i=2}^n \frac{a_{i1}^{(1)}}{a_{11}^{(1)}} x_i$. Then

$$\bar{\mathbf{x}} A^{(2)} \mathbf{x} = \sum_{i,j=2}^n a_{ij}^{(2)} x_i x_j \quad = \mathbf{x}^T A \mathbf{x} \geq 0.$$

□

Proof (3):

$$A^{(k)} \quad = \quad P^T A, \quad P \text{ is lower triangular.}$$

Then

$$B := A^{(k)} P = P^T A P = \begin{bmatrix} D & 0 \\ 0 & A_{22}^{(k)} \end{bmatrix},$$

where D is diagonal and $A_{22}^{(k)}$ is symmetric. Let $\mathbf{x}$ be any vector in $\mathbb{R}^n$ of the form $\mathbf{x} = [\mathbf{0}, \mathbf{z}]$, $\mathbf{z} \in \mathbb{R}^{n-k}$. Then

$$\mathbf{x}^T B \mathbf{x} = (P\mathbf{x})^T A (P\mathbf{x}) = \mathbf{y}^T A \mathbf{y} = \mathbf{z}^T A_{22}^{(k)} \mathbf{z} > 0,$$

since $\mathbf{y} = P\mathbf{x}$ is nonzero vector.

Theorem 5.2.18. *The GE. without pivoting preserves the diagonally dominance of A. Thus if A is a symmetric, diagonally dominant then partial pivoting is not necessary.*

Remark 5.2.19. See p 189 of Kincaid-Cheney and corollary 2. They give reasoning to row scaled pivot. Question: If A is SPD, is it true that A is diagonally dominant ?

Proof. Let $A_{22}^{(1)}$ be the submtrix appearing in the G.E. after elim. the first row. Let $k = 1$.

$$\left| a_{22}^{(1)} - \frac{a_{21}^{(1)}}{a_{11}^{(1)}} a_{12}^{(1)} \right| \quad > \quad \sum_{i=3}^n \left| a_{2i}^{(1)} - \frac{a_{21}^{(1)}}{a_{11}^{(1)}} a_{1i}^{(1)} \right|$$

which is equiv. to

$$\left| a_{22}^{(1)} a_{11}^{(1)} - a_{21}^{(1)} a_{12}^{(1)} \right| \quad > \quad \sum_{i=3}^n \left| a_{2i}^{(1)} a_{11}^{(1)} - a_{21}^{(1)} a_{1i}^{(1)} \right|$$

This is to see

$$\begin{aligned} \sum_{i=3}^n \left| a_{2i}^{(1)} a_{11}^{(1)} - a_{21}^{(1)} a_{1i}^{(1)} \right| \quad &\leq \quad |a_{11}^{(1)}| \sum_{i=3}^n |a_{2i}^{(1)}| + |a_{21}^{(1)}| \sum_{i=3}^n |a_{1i}^{(1)}| \\ &\leq \quad |a_{11}^{(1)}|(|a_{22}^{(1)}| - |a_{21}^{(1)}|) + |a_{21}^{(1)}|(|a_{11}^{(1)}| - |a_{12}^{(1)}|) \\ &= \quad |a_{11}^{(1)}||a_{22}^{(1)}| - |a_{21}^{(1)}||a_{21}^{(1)}| \\ &\leq \quad |a_{11}^{(1)} a_{22}^{(1)} - a_{21}^{(1)} a_{21}^{(1)}|. \end{aligned}$$

□

Corollary 5.2.20. *If A is symmetric and diagonally dominant, then partial pivoting is not necessary. Just note that GE. plus columns wise elim. do not change the submatrices $A_{22}^{(k)}$ appearing in the G.E. after elim.*

LDM^T decomposition

Theorem 5.2.21. *If the leading principal submatrices A_k of $A \in \mathbb{R}^{n\times n}$ are nonsingular, then there exist unit lower triangular matrices L and M and diagonal matrix D such that $A = LDM^T$.*

Proof. Comparing the $(1,1)$ blocks of (5.9) we see that the leading principal submatrices $A_k = L_{11}^{(k)} A_{11}^{(k)}$. Since

$$\det\ [A_{11}^{(k)}] = a_{11}^{(k)}\ \det\ [A_{11}^{(k-1)}].$$

It follows that

$$\det\ A_k = a_{11}^{(1)} \cdots a_{kk}^{(k)}.$$

Thus the elimination can proceed.

Now let $A = LU$ be its LU-decomposition. Define $D = \mathrm{diag}(d_1, \cdots, d_n)$, where $d_i = u_{ii}$ for $i = 1, \cdots, n$. Note that D is nonsingular and $M^T = D^{-1}U$ is unit upper triangular. Then $A = LU = LD(D^{-1}U) = LDM^T$. □

Theorem 5.2.22. *If $A \in \mathbb{R}^{n\times n}$ is positive definite, then A has an LDM^T decomposition and diagonal entries of D are positive.*

Proof. All principle submatrices of a positive definite are positive definite and therefore nonsingular. By theorem above, $A = LDM^T$ exists. Let $S = DM^T L^{-T} = L^{-1}AL^{-T}$ is positive definite and upper triangular with $s_{ii} = d_i$. Hence d_i are positive. □

Positive definite matrix can be bad conditioned!

Example 5.2.23. The matrix

$$A = \begin{bmatrix} \epsilon & m \\ -m & \epsilon \end{bmatrix}, \quad m \gg \epsilon$$

is positive definite, and its LDM^T decomposition is

$$A = \begin{bmatrix} 1 & 0 \\ -\frac{m}{\epsilon} & 1 \end{bmatrix} \begin{bmatrix} \epsilon & 0 \\ 0 & \epsilon + \frac{m^2}{\epsilon} \end{bmatrix} \begin{bmatrix} 1 & \frac{m}{\epsilon} \\ 0 & 1 \end{bmatrix}.$$

Pivoting is necessary for this example.

Cholesky factorization

If A is real symmetric and positive definite, we may factor alternatively as

$$A = BB^T,$$

where $B = LD$, D is a diagonal matrix with $d_{kk} = \sqrt{a_{kk}^{(k)}}$, $k = 1, \cdots, n$. The details are as follows:

Let $A = LU = A^T = U^T L^T$ so that

$$U(L^T)^{-1} = L^{-1}U^T.$$

Since left hand side is upper triangular and the right hand side is lower triangular, we see this matrix is diagonal, say $D = L^{-1}U^T$. Hence $A = LDL^T$. Since D is positive definite, $\sqrt{D}$ exists and hence

$$A = BB^T, \quad B = LD^{1/2}.$$

The following algorithm is called Cholesky factorization.

$$\begin{aligned} b_{11} &= \sqrt{a_{11}} \\ b_{i1} &= \frac{a_{i1}}{b_{11}}, \quad i = 2, \cdots, n \end{aligned}$$

and for $j = 2, \cdots, n$

$$\begin{aligned} b_{jj} &= \left(a_{jj} - \sum_{k=1}^{j-1} b_{jk}^2 \right)^{1/2} \\ b_{ij} &= (a_{ij} - \sum_{k=1}^{j-1} b_{ik} b_{jk})/b_{jj}, \quad i = j+1, \cdots, n. \end{aligned}$$

In pseudo code, it is

input n, a_{ij}
for $j = 1, \cdots, n$ do
    $b_{jj} = \left(a_{jj} - \sum_{k=1}^{j-1} b_{jk}^2 \right)^{1/2}$
    for $i = j+1, \cdots, n$ do
        $b_{ij} = (a_{ij} - \sum_{k=1}^{j-1} b_{ik} b_{jk})/b_{jj}$
    end
end

Ill-conditioned system and condition number

When we numerically solve a system of equations

$$A\mathbf{x} = \mathbf{b},$$

we usually get approximate solution $\mathbf{x}+\mathbf{h}$ where $\mathbf{h}$ is the error. We would like analyze the relative error $\frac{\|\mathbf{h}\|}{\|\mathbf{x}\|}$ and find its relation with matrix A. Since

$$A(\mathbf{x}+\mathbf{h}) \quad \neq \mathbf{b},$$

we can view $\mathbf{x}+\mathbf{h}$ as the solution of a perturbed problem

$$(A+E)(\mathbf{x}+\mathbf{h}) = \mathbf{b}+\mathbf{k}.$$

Theorem 5.2.24. *If* $\|A^{-1}\| \cdot \|E\| < 1$, *then*

$$\frac{\|\mathbf{h}\|}{\|\mathbf{x}\|} \leq C\|A\| \cdot \|A^{-1}\| \left[\frac{\|\mathbf{k}\|}{\|\mathbf{b}\|} + \frac{\|E\|}{\|A\|}\right],$$

where $C = [1 - \|A^{-1}E\|]^{-1}$.

Proof. Note that

$$(A+E)\mathbf{h} = \mathbf{b}+\mathbf{k}-A\mathbf{x}-E\mathbf{x} = \mathbf{k}-E\mathbf{x}.$$

Write $A+E = A(I+A^{-1}E)$. Since

$$\rho(A^{-1}E) \leq \|A^{-1}E\| \leq \|A^{-1}\|\,\|E\| < 1,$$

the matrix $I+A^{-1}E$ is nonsingular and so is $A+E$.(see the Neumann lemma below)

$$\therefore \quad \mathbf{h} = (A+E)^{-1}(\mathbf{k}-E\mathbf{x}) = (I+A^{-1}E)^{-1}A^{-1}(\mathbf{k}-E\mathbf{x}).$$

Now we estimate the norm

$$\begin{aligned}
\|\mathbf{h}\| &\leq \|(I+A^{-1}E)^{-1}\| \cdot \|A^{-1}\| \cdot \|\mathbf{k}-E\mathbf{x}\| \\
&\leq \frac{\|A^{-1}\|}{1-\|A^{-1}E\|}[\|\mathbf{k}\| + \|E\mathbf{x}\|] \\
\frac{\|\mathbf{h}\|}{\|\mathbf{x}\|} &\leq \frac{\|A^{-1}\|}{1-\|A^{-1}E\|}\left[\frac{\|\mathbf{k}\|}{\|\mathbf{x}\|} + \|E\|\right] \\
&= \frac{\|A^{-1}\| \cdot \|A\|}{1-\|A^{-1}E\|}\left[\frac{\|\mathbf{k}\|}{\|A\|\,\|\mathbf{x}\|} + \frac{\|E\|}{\|A\|}\right].
\end{aligned}$$

Use the relation $\|\mathbf{b}\| = \|A\mathbf{x}\| \leq \|A\| \cdot \|\mathbf{x}\|$ to get

$$\frac{\|\mathbf{h}\|}{\|\mathbf{x}\|} \leq C\kappa(A)\left[\frac{\|\mathbf{k}\|}{\|\mathbf{b}\|} + \frac{\|E\|}{\|A\|}\right],$$

where $C = 1/(1-\|A^{-1}E\|)$ and $\kappa(A) = \|A^{-1}\| \cdot \|A\|$. □

The number $\kappa(A) = \|A^{-1}\| \cdot \|A\|$ depends on the particular norm $\|\cdot\|_\beta$ and such number $\kappa(A)_\beta$ are called the *condition number* of A. For example spectral condition number is defined as

$$\kappa_2(A) = \|A\|_2 \cdot \|A^{-1}\|_2 = \sqrt{\rho(A^*A)\rho(A^{-1}A^{*-1})}.$$

Noting that the eigenvalues of A^*A are the reciprocal of eigenvalue values of $(A^*A)^{-1}$(unless they are zero), $\kappa_2 = \sqrt{\frac{\mu_1}{\mu_n}}$, where $\mu_1 \geq \cdots > \mu_n$ are eigenvalues of A^*A. Furthermore, if A is Hermitian($A^* = A$), then $\|A\|_2 = \rho(A)$. Hence

$$\kappa_2 = \left|\frac{\lambda_1}{\lambda_n}\right|,$$

when eigenvalues are arranged in the order of magnitude $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n| > 0$.

If A is symmetric, positive definite, then $\kappa_2 = \frac{\lambda_1}{\lambda_n}$.

Scaled row pivoting

Another efficient way pivoting is to scale each row by its maximum norm during Gaussian elimination. LU-decomposition Algorithm with scaled row pivoting is as follows:

input n, a_{ij}, (b_i)
For $i = 1, \cdots, n$ do
    $p_i \leftarrow i$
    Let $s_i = |a_{i1}|$
    for $j = 2, \cdots, n$ do
        if $|a_{ij}| > s_i$, then set $s_i = |a_{ij}|$
    end
end

for $k = 1, \cdots, n-1$ do
    Select $j \geq k$ so that
        $|a_{p_j k}|/s_{p_j} \geq |a_{p_i k}|/s_{p_i}$ for $i = k, k+1, \cdots, n$
    $p_k \leftrightarrow p_j$
    for $i = k+1, \cdots, n$ do
        $m_{p_i k} = \frac{a^{(k)}_{p_i k}}{a^{(k)}_{p_k k}}$ (*);
        $a^{(k)}_{p_i k} = m_{p_i k}$ (this is for L)
        for $j = k+1, \cdots, n$ do
            $a^{(k+1)}_{p_i j} \leftarrow a^{(k)}_{p_i j} - m_{p_i k} a^{(k)}_{p_k j}$
        end
    end
end

Solving $A\mathbf{x} = \mathbf{b}$ from $LU\mathbf{x} = P\mathbf{b}$. First step is to solve for $\mathbf{y}$ from $L\mathbf{y} = P\mathbf{b}$.(Elimination of right hand side-this could have been done during the Elimination step (*) above)

for $k = 1, 2, \cdots, n-1$ do
    for $i = k+1, \cdots, n$ do
        $b_{p_i} \leftarrow b_{p_i} - a_{p_i k} b_{p_k}$;
    end
end

Now get $\mathbf{x}$ from $U\mathbf{x} = \mathbf{y}$(Backward substitution)

for $i = n, n-1, \cdots, 1$ do
    for $j = i+1, \cdots, n$ do
        $temp = b_{p_i} - a_{p_i j} x_j$;
        $x_i \leftarrow temp/a_{p_i i}$
    end
end

Symmetric indefinite matrix can be bad conditioned

Example 5.2.25. The matrix

$$A = \begin{bmatrix} \epsilon & 1 \\ 1 & \epsilon \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1/\epsilon & 1 \end{bmatrix} \begin{bmatrix} \epsilon & 0 \\ 0 & \epsilon - 1/\epsilon \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1/\epsilon & 1 \end{bmatrix}^T, \quad 1 \gg \epsilon.$$

Pivoting is necessary for this example.

Partial pivoting breaks the symmetry. Hence consider interchange of rows and columns together, i.e,

$$PAP^T = LDL^T.$$

But this may not be possible as the following example shows:

$$P \begin{bmatrix} \epsilon & 1 \\ 1 & \epsilon \end{bmatrix} P^T = \begin{bmatrix} \epsilon & 1 \\ 1 & \epsilon \end{bmatrix}$$

for any permutation matrix P. There is a decomposition by Aasen computing

$$PAP^t = LTL^t, \tag{5.20}$$

where L is lower triangular and T is tridiagonal. See section 5.4 of G. Golub for details.

Ill-conditioned system: Least square approximation

Given $f(x) \in L^2(0,1)$, find a polynomial of degree n such that

$$\int_0^1 (f(x) - p_n(x))^2 dx$$

is minimum. Let $p_n(x) = \sum_{i=0}^n c_i x^i$. Then the minimum is attained when

$$\frac{\partial}{\partial c_j} \int_0^1 (f(x) - p_n(x))^2 dx = 0, \quad \text{for } j = 0, 1, \cdots, n.$$

Hence, we get

$$\sum_{i=0}^n c_i \int_0^1 x^i x^j \, dx = \int_0^1 f(x) x^j \, dx, \quad j = 0, 1, \cdots, n$$

which can be written as

$$A\mathbf{c} = \mathbf{f}, \tag{5.21}$$

where $A_{ij} = \frac{1}{i+j+1}$, $i, j = 0, \cdots, n$, $\mathbf{c} = (c_0, \cdots, c_n)$ and

$$\mathbf{f} = \left(\int_0^1 f(x) x^0 dx, \cdots, \int_0^1 f(x) x^n dx \right).$$

This matrix is called Hilbert matrix and is extremely ill-conditioned! If we choose $f(x) = Q_n(x) = \sum_{i=0}^n q_i x^i$, then $c_i = q_i$, so that one can compare the numerical solution.(See exercise below. For $n \leq 10$, it seems OK. In general, it strongly depends on data.)

Example 5.2.26 (2D)**.** A typical 2nd order elliptic differential equation is

$$\begin{aligned} -\Delta u &= f \text{ on } \Omega \\ u &= g \text{ on } \partial\Omega. \end{aligned} \tag{5.22}$$

We take $\Omega = [0,1]^2$. With $h = 1/n$, the derivatives are approximated by

$$\begin{aligned} u_{xx}(x,y) &\doteq [u(x+h,y) - 2u(x,y) + u(x-h,y)]/h^2 \\ u_{yy}(x,y) &\doteq [u(x,y+h) - 2u(x,y) + u(x,y-h)]/h^2. \end{aligned}$$

The figure is called Molecule, Stencil, etc. For each point (interior mesh pt), approximate $\nabla^2 u = \Delta u$ by 5-point stencil. By Gershgorin disc theorem, the matrix is nonsingular.

When the unit square is divided by $n = 1/h$ equal intervals along x-axis and y-axis, then the corresponding matrix A is $(n-1) \times (n-1)$ block-diagonal matrix of the form:

$$A = \frac{1}{h^2} \begin{bmatrix} B & -I & 0 & \cdots & \\ -I & B & -I & 0 & \\ & -I & \ddots & \ddots & \\ & & \ddots & B & -I \\ & \cdots & 0 & -I & B \end{bmatrix} \tag{5.23}$$

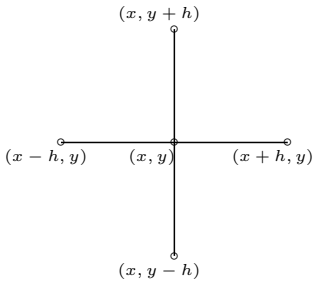


Figure 5.2: 5-point Stencil

where

$$B = \begin{bmatrix} 4 & -1 & 0 & \cdots & \\ -1 & 4 & -1 & 0 & \\ & -1 & \ddots & \ddots & \\ & & \ddots & 4 & -1 \\ & \cdots & 0 & -1 & 4 \end{bmatrix}$$

is $(n-1) \times (n-1)$ matrix. When $g = 0$ in (5.22), the eigenvectors of $(n-1)^2 \times (n-1)^2$ matrix $h^2 A$ are

$$x^h_{\mu\nu}(x_i, y_j) = \sin(\mu\pi x_i)\sin(\nu\pi y_j), \quad (x_i, y_j) = (hi, hj) \in \Omega_h \tag{5.24}$$

with the corresponding eigenvalues

$$\lambda^h_{\mu\nu} = 4h^{-2}(\sin^2(\mu\pi h/2) + \sin^2(\nu\pi h/2)), \quad 1 \le \mu, \nu \le n-1. \tag{5.25}$$

Note that the eigenvalues of Laplace equation for the domain $[0, a] \times [0, b]$ are

$$\lambda_{\mu\nu} = (\frac{\mu^2}{a^2} + \frac{\nu^2}{b^2})\pi^2, \ \mu, \nu = 1, \cdots, . \tag{5.26}$$

Tridiagonal matrix

Theorem 5.2.27. *For tridiagonal matrix we have the following simple LU decomposition:*

$$A = \begin{bmatrix} a_1 & c_1 & & \\ b_2 & a_2 & \ddots & \\ & \ddots & \ddots & c_{n-1} \\ & & b_n & a_n \end{bmatrix} = \begin{bmatrix} \alpha_1 & 0 & & \\ b_2 & \alpha_2 & \ddots & \\ & \ddots & \ddots & 0 \\ & & b_n & \alpha_n \end{bmatrix} \begin{bmatrix} 1 & \gamma_1 & & \\ & 1 & \ddots & \\ & & \ddots & \gamma_{n-1} \\ & & & 1 \end{bmatrix}$$

$$\begin{array}{llll} \alpha_1 = a_1, & \alpha_1 \cdot \gamma_1 = c_1 & \therefore\ \gamma_1 = \frac{c_1}{\alpha_1}, & \alpha_i \neq 0 \\ & b_2\gamma_1 + \alpha_2 = a_2 & \therefore\ \alpha_2 = a_2 - b_2\gamma_1 & \\ & \alpha_2\gamma_2 = c_2 & \therefore\ \gamma_2 = c_2/\alpha_2 & \end{array}$$

In general, let dummy the variables $b_0 = 0$, $\gamma_0 = 0, \gamma_n = 0$. *Then we have for* $i = 1, \ldots, n$

$$\begin{aligned} \alpha_i &= a_i - b_i\gamma_{i-1}, \\ \gamma_i &= c_i/\alpha_i. \end{aligned}$$

To solve the system $A\mathbf{x} = \mathbf{b}$, we only need back substitution where the computational counts are:

$$\begin{array}{lll} L\mathbf{y} = \mathbf{b} \cdots \text{forward-elimination} & & 2(n-1)+1 \\ U\mathbf{x} = \mathbf{y} \cdots \text{back-substitution} & & n-1 \end{array}$$

$$\therefore \quad \text{Total} \quad 5n - 4.$$

Block matrix or banded matrix

One can also consider the case of banded matrix or block tri-diagonal matrix for which similar Gauss Elimination can be derived. For example, Block Gaussian Elim. looks like:

$$\begin{bmatrix} A_{11} & A_{12} & \ldots \\ A_{21} & A_{22} & \ldots \\ & & \end{bmatrix}, \quad A_{2j}^{(2)} = A_{2j}^{(1)} - A_{21}^{(1)} A_{11}^{(1)-1} A_{1j}^{(1)}, \quad j = 2, \ldots, n.$$

The matrix A is called **banded** if there exists a natural number d such that

$$a_{ij} = 0, \quad |i - j| > d.$$

To factor a $n \times n$ matrix with bandwidth d, we need $nd^2/2$ operations asymptotically. To our application with $a_{ij} = a(\phi_i, \phi_j)$ where ϕ_i are basis functions, we have

$$d = \max\{|i - j| : a(\phi_i, \phi_j) \neq 0\},$$

where we assumed ϕ_i, ϕ_j are associated with d.o.f belonging to the same element. Clearly d depends on the chosen enumeration of nodes. Thus if we want to use Gaussian elimination, we want to enumerate the nodes so as to make band width as small as possible. Consider a domain $[0, 2] \times [0, 1]$ with 5×10 uniform mesh. We enumerate $x = 0, 0 \leq y \leq 1$ first vertically as $1, 2, , \cdots, 6$. Then along $x = 0.2, 0 \leq y \leq 1$ we enumerate as $7, 8, \cdots, 12$ etc. Then $d = 6$(assuming one degree of freedom on each node), while if we enumerate horizontally, we would have $d = 11$. In a typical finite element method with linear basis functions $n = O(h^{-2})$ and $d = O(h^{-1})$. Hence total cost is $O(nd^2) = O(h^{-4})$.

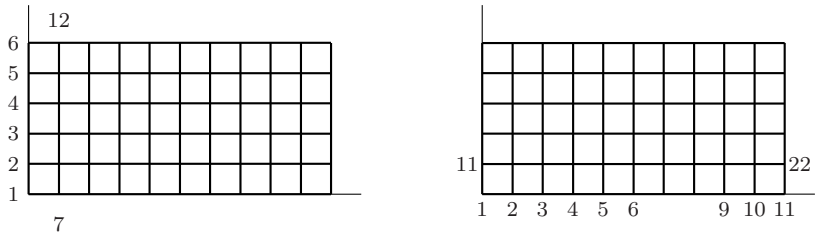


Figure 5.3: Numbering nodes horizontally or vertically

Storage of banded matrix

We store a banded matrix as a vector columnwise or alternatively as $2d-1$-vectors (d-vectors for symmetric matrix). One can also store a matrix with variable band width, in this case, one need store the index of diagonal entries also, referred to as a skyline method of storage.

Fill in

During the elimination, matrix entries with zero elements are filled with nonzero values. To avoid this, one can use Frontal methods or Nested dissection method.

Exercise 5.2.28. (1) Let A, B are two $n \times n$ matrices. Show that

(2) Let

$$\kappa_2 := \|A\|_2 \cdot \|A^{-1}\|_2 = \sqrt{\rho(A^*A)\rho(A^{-1}A^{*-1})}.$$

Show that if μ is an eigenvalue of A^*A, $\mu \neq 0$ then μ^{-1} is an eigenvalue value of $(A^*A)^{-1}$, thus, $\kappa_2 = \sqrt{\frac{\mu_1}{\mu_n}}$, where $\mu_1 \geq \cdots > \mu_n$ are eigenvalues of A^*A. Furthermore, if A is Hermitian, $A^* = A$. Hence $\|A\|_2 = \rho(A)$ and

$$\kappa_2 = \left|\frac{\lambda_1}{\lambda_n}\right|, \qquad |\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n| > 0.$$

If A is symmetric, positive definite, then $\kappa_2 = \frac{\lambda_1}{\lambda_n}$.

(3) Let $D = \mathrm{diag}(2, 11)$ and

$$A = \begin{pmatrix} 1 & 1 \\ 1 & 10 \end{pmatrix}.$$

Calculating the condition numbers of A and $D^{-1}A$, conclude that diagonal scale can improve the condition number.

(4) However, for a matrix A satisfying

$$\sum_{k=1}^{n} |a_{jk}| = 1,$$

we have

$$\kappa_\infty(A) \le \kappa_\infty(D^{-1}A)$$

for all diagonal matrix D.(It suffices to consider the case $d_i \ge 1$ for all i.)

$$\|(D^{-1}A)^{-1}\|_\infty = \|A^{-1}D\|_\infty = \max_j \sum_{k=1}^{n} |a_{jk}^{-1}||d_k| \ge \max_k |d_k| \|A^{-1}\|_\infty.$$

Thus

$$\kappa_\infty(D^{-1}A) = \|D^{-1}A\|_\infty \|A^{-1}D\|_\infty \ge \|A\|_\infty \|A^{-1}\|_\infty.$$

(5) Show the tridiagonal matrix in Theorem 5.2.27 is nonsingular under the following assumptions: $|a_1| > |c_1| > 0$, $\quad |a_n| > |b_n| > 0$ and

$$|a_i| \ge |b_i| + |c_i|, \quad b_i c_i \ne 0,\ i = 2, 3, \cdots, n-1.$$

(It suffices to consider the case $a_i > 0$. Use induction.)

(6) Derive a LU-decomposition for the tridiagonal matrix in Theorem 5.2.27, where L is the unit lower triangular matrix.

(7) Explain a similar algorithm for Cholesky decomposition as in Theorem 5.2.27.

(8) If $f(x) = \sum_{i=0}^{n} x^i$, then we have $\mathbf{f} = (\sum_{i=0}^{n} \frac{1}{i+j+1})_{j=0}^{n}$ in (5.21). Compute $\mathbf{c}$ by solving the system by LU-decomposition (a) with no pivoting and (b) with pivoting. Do this for $n = 10$ and $n = 15$ or larger. Measure the maximum error of $\mathbf{c}$ by printing out the computed solution. Does the partial pivoting work ? QR-decomposition may be better.

(9) Consider solving the two point boundary value problem

$$-u'' = f \text{ on } (0, 1)$$

with B.C.

$$u(0) = a, \quad u(1) = b$$

by the finite difference method. First subdivide the interval by n-equal intervals $h = 1/n$.

$$x_0 = 0, x_1 = h, \cdots, x_i = ih, \cdots, x_n = 1.$$

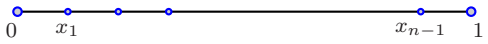


Figure 5.4: meshes for boundary value problem

We get

$$Au_h = f_h,$$

where A_h is $(n-1) \times (n-1)$ system given by

$$A_h = \frac{1}{h^2} \begin{bmatrix} 2 & -1 & & \\ -1 & 2 & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 2 \end{bmatrix}$$

and

$$f_h = \left(f(x_1) + \frac{u(0)}{h^2}, f(x_2), \cdots, f(x_{n-1}) + \frac{u(1)}{h^2} \right)^T.$$

If $f = x(x-1)$, we have $u = -\frac{x^4}{12} + \frac{x^3}{6} - \frac{x}{12}$. Eigenvalues and eigenvectors of $(n-1) \times (n-1)$ matrix $h^2 A$ are (with $h = \frac{1}{n}$)

$$\lambda_j = 2(1 - \cos j\pi h), \quad \mathbf{x}_j = (\sin j\pi h, \cdots, \sin(n-1)j\pi h),\ j = 1, \cdots, n-1.$$

For $n = 10$, the eigenvalues are

$$3.90211, 3.61803, 3.1755, 2.618, 1.9999, 1.3819, 0.8244, 0.38196, 0.9788.$$

(10) Prove Theorem 5.2.17.

(11) Solve (5.22) with $u = (5x + y)e^{x+2y}$ and $f = -(14 + 25x + 5y)e^{x+2y}$.

5.3 Iterative method for large system of equation

Motive

This is exactly the idea behind iterative refinement:

(1) Get an approximate solution $A\hat{\mathbf{x}}_1 = \mathbf{b}$.

(2) Compute the residual $r = \mathbf{b} - A\hat{\mathbf{x}}_1$ (to good accuracy).

(3) Approximately solve $A\delta\mathbf{x}_1 = \mathbf{r}$.

(4) Get a new approximate solution $\hat{\mathbf{x}}_2 = \hat{\mathbf{x}}_1 + \delta\mathbf{x}_1$; repeat as needed.

If we know A and $\mathbf{b}$, a reasonable way to evaluate an approximate solution $\hat{\mathbf{x}}$ is through the residual $\mathbf{r} = \mathbf{b} - A\hat{\mathbf{x}}$. The approximate solution satisfies

$$A\hat{\mathbf{x}} = \mathbf{b} - \mathbf{r};$$

so if we subtract from $A\mathbf{x} = \mathbf{b}$, we have

$$\mathbf{x} - \hat{\mathbf{x}} = A^{-1}\mathbf{r}.$$

We can use this to get the error estimate.

$$\|\mathbf{x} - \hat{\mathbf{x}}\| \leq \|A^{-1}\mathbf{r}\|.$$

This is exactly the idea behind iterative refinement:

(1) Get an approximate solution $A\hat{\mathbf{x}}_1 \approx \mathbf{b}$.

(2) Compute the residual $\mathbf{r} = \mathbf{b} - A\hat{\mathbf{x}}_1$ (to good accuracy).

(3) Approximately solve $A\delta\mathbf{x}_1 \approx \mathbf{r}$.

(4) Get a new approximate solution $\hat{\mathbf{x}}_2 = \hat{\mathbf{x}}_1 + \delta\mathbf{x}_1$; repeat as needed.

Recall from last lecture that if we have a solver for $\hat{A} = A+E$ with E small, then we can use iterative refinement to clean up the solution. The matrix $\hat{A}$ could come from finite precision Gaussian elimination of A, for example, or from some factorization of a nearby easier matrix. To get the refinement iteration, we take the equation

$$A\mathbf{x} = \hat{A}\mathbf{x} - E\mathbf{x} = \mathbf{b}; \tag{5.27}$$

and think of $\mathbf{x}$ as the fixed point for an iteration

$$\hat{A}\mathbf{x}_{k+1} - E\mathbf{x}_k = \mathbf{b}. \tag{5.28}$$

Note that this is the same as

$$\hat{A}\mathbf{x}_{k+1} - (\hat{A} - A)\mathbf{x}_k = \mathbf{b}$$

or

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \hat{A}^{-1}(\mathbf{b} - A\mathbf{x}_k). \tag{5.29}$$

Note that this latter form is the same as inexact Newton iteration on the equation $A\mathbf{x}_k - \mathbf{b} = 0$ where Jacobian matrix $\hat{A}$ was replaced by $\hat{A}$. If we subtract (5.27) from (5.28), we see

$$\hat{A}(\mathbf{x}_{k+1} - \mathbf{x}) - E(\mathbf{x}_k - \mathbf{x}) = 0;$$

or

$$\mathbf{x}_{k+1} - \mathbf{x} = \hat{A}^{-1}E(\mathbf{x}_k - \mathbf{x}).$$

Taking norms, we have

$$\|\mathbf{x}_{k+1} - \mathbf{x}\| \leq \|\hat{A}^{-1}E\| \, \|\mathbf{x}_k - \mathbf{x}\|.$$

Thus, if $\|\hat{A}^{-1}E\| < 1$, we are guaranteed that $\mathbf{x}_k \to \mathbf{x}$ as $k \to \infty$. At least, this is what happens in exact arithmetic. In practice, the residual is usually computed with only finite precision, and so we might expect to stop making progress at some point.

This is a starting point of iterative refinement.

Jacobi - method

In this section we consider an iterative method to find the solution of $A\mathbf{x} = \mathbf{b}$. We start from a splitting of A as $A = D - T$. First observe

$$\begin{aligned} (D-T)\mathbf{x} &= \mathbf{b} \\ D\mathbf{x} &= T\mathbf{x} + \mathbf{b} \\ \mathbf{x} &= D^{-1}T\mathbf{x} + D^{-1}\mathbf{b}. \end{aligned}$$

We now define a sequence $\mathbf{x}^{(k)}$ of approximation to $\mathbf{x}$ via

$$\begin{aligned} \mathbf{x}^{(k)} &= D^{-1}T\mathbf{x}^{(k-1)} + D^{-1}\mathbf{b} \\ &= D^{-1}(D-A)\mathbf{x}^{(k-1)} + D^{-1}\mathbf{b} \\ &= (I - D^{-1}A)\mathbf{x}^{(k-1)} + D^{-1}\mathbf{b}. \end{aligned}$$

Note

$$\begin{aligned} \mathbf{x}^{(k)} - \mathbf{x}^{(k-1)} &= (-D^{-1}A)\mathbf{x}^{(k-1)} + D^{-1}A\mathbf{x} \\ &= D^{-1}(\mathbf{b} - A\mathbf{x}^{(k-1)}) \end{aligned}$$

is the form described in (5.29).

Question. Does $\mathbf{x}^{(k)} \to \mathbf{x}$? and how fast? How to choose $D - T$? The idea is to choose D so that $D\mathbf{x} = \mathbf{b}$ can be solved easily. Diagonal of A is a often a good choice for D(called a **Jacobi method**). If D is diagonal, the Jacobi method is defined as

$$x_i^{(k)} = \frac{1}{a_{ii}}\left[\sum_{j \neq i} -a_{ij}x_j^{(k-1)} + b_i\right], \qquad i = 1, 2, \ldots, n.$$

Gauss-Seidel iterative method

Suggestion. In each computation of Jacobi method, $x_i^{(k-1)}$ was already updated for $j < i$. Why don't we utilize the most recent information? The result is

$$x_i^{(k)} = \frac{1}{a_{ii}}\left[\sum_{j=1}^{i-1} -a_{ij}x_j^{(k)} + \sum_{j=i+1}^{n} -a_{ij}x_i^{(k-1)} + f_i\right].$$

If we use the splitting $A = D - L - U$, then the scheme is, in matrix form

$$\mathbf{x}^{(k)} = (D-L)^{-1}U\mathbf{x}^{(k-1)} + (D-L)^{-1}\mathbf{f}. \tag{5.30}$$

Block Jacobi or block Gauss-Seidel method

$$\bar{x}_i^{(k)} = A_{ii}^{-1}\left(\sum_{j\neq i} -A_{ij}\bar{x}_j^{(k-1)} + \bar{k}_i\right)$$

$$A = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}$$

Successive over relaxation method (SOR)

(1) Given $\mathbf{x}^{(k)}$

(2) Compute $\tilde{x}_i^{(k+1)}$, the i-th entry of Gauss-Seidel

(3) Define $x_i^{(k+1)} = (1-\omega)x_i^{(k)} + \omega\tilde{x}_i^{(k+1)}$, ω: relaxation factor

Note: if $\omega = 1$, SOR $\equiv$ Gauss-Seidel
Algorithm:

$$\begin{aligned}
a_{ii}\tilde{x}_i^{(k+1)} &= -\sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)} + f_i, \qquad 1 \le i \le n, \\
a_{ii}x_i^{(k+1)} &= (1-\omega)a_{ii}x_i^{(k)} + \omega a_{ii}\tilde{x}_i^{(k+1)} \\
&= (1-\omega)a_{ii}x_i^{(k)} + \omega\left\{-\sum_{j\le i-1} a_{ij}x_j^{(k+1)} - \sum_{j\ge i+1} a_{ij}x_j^{(k)} + f_i\right\} \\
&= a_{ii}x_i^{(k)} + \omega\left\{-\sum_{j\le i-1} a_{ij}x_j^{(k+1)} - \sum_{j\ge i+1} a_{ij}x_j^{(k)} + f_i - a_{ii}x_i^{(k)}\right\}
\end{aligned}$$

Let

$$\begin{aligned} D &= diag\{a_{11}, \ldots, a_{nn}\} \\ L &= -\begin{pmatrix} 0 & \ldots & 0 \\ a_{21} & \ddots & \vdots \\ & \ldots & 0 \end{pmatrix}, \qquad U = -\begin{pmatrix} 0 & a_{12} & \ldots & a_{1n} \\ \vdots & \ddots & & \\ 0 & \ldots\ldots & & 0 \end{pmatrix} \end{aligned}$$

In matrix form with the splitting $A = D - L - U$, we have

$$\begin{aligned} (D - \omega L)\mathbf{x}^{(k+1)} &= \{(1-\omega)D + \omega U\}\mathbf{x}^{(k)} + \omega\mathbf{f} \\ \mathbf{x}^{(k+1)} &= (D - \omega L)^{-1}\{(1-\omega)D + \omega U\}\mathbf{x}^{(k)} + (D - \omega E)^{-1}\omega\mathbf{f}. \end{aligned}$$

This again is of the form,

$$\mathbf{x}^{(k+1)} = L_\omega \mathbf{x}^{(k)} + \mathbf{g}.$$

How "fast" does the iterative scheme converge? The convergence will depend on $\rho(L_\omega)$.

5.3.1 Convergence of iterative scheme

Theorem 5.3.1 (Gershgorin disk theorem)**.** *For $A \in \mathbb{C}^{n,n}$, all the eigenvalues of A satisfies*

$$\lambda \in \bigcup_{i=1}^{n} \left\{ \xi : |\xi - a_{ii}| \leq \sum_{j \neq i} |a_{ij}| \right\} := \bigcup_{i=1}^{n} D_i(a_{ii}; \rho_i).$$

Each disk on the right hand side is called a **Gershgorin disk***. Furthermore, if an eigenvalue lies on the boundary of a disk, then every such disk must pass the eigenvalue.*

Proof. We first normalize $\|\mathbf{x}\|_\infty = 1$, so that $|x_r| = 1$ for some r. Then since $\sum_j a_{ij}x_j = \lambda x_i$, we have

$$(\lambda - a_{rr})x_r = \sum_{j \neq r} a_{rj}x_j \tag{5.31}$$

$$|\lambda - a_{rr}| \leq \sum_{j \neq r} |a_{rj}||x_j| \leq \sum_{j \neq r} |a_{rj}| := \rho_r. \tag{5.32}$$

Thus all eigenvalue λ must belong to the union of disks: $\bigcup_{i=1}^{n} D_i(a_{ii}; \rho_i)$. The rest of the proof is left as an exercise. □

Definition 5.3.2. An $n \times n$ matrix A is called **reducible** if there exists a permutation matrix P such that

$$PA = \begin{pmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{pmatrix},$$

where A_{11} and A_{22} are square matrices. If a matrix is not reducible, it is called **irreducible**.

Lemma 5.3.3. *A is reducible iff there is an index set $J \subset \{1, 2, \cdots, n\}$ such that*

$$a_{kj} = 0, \text{ for } k \in J, j \notin J.$$

Definition 5.3.4 (diagonal dominance)**.** If $|a_{ii}| \geq \sum_{j \neq i} |a_{ij}|$, $i = 1, \ldots, n$, and strict inequality holds at least for one i, then it is called **weakly diagonally dominant**. If the strict inequality holds for all i, it is called **strictly diagonally dominant**.

Theorem 5.3.5. *If A is strictly diagonally dominant, then A is nonsingular.*

Proof. Use G-disk theorem to conclude $\lambda_i \neq 0$. □

Theorem 5.3.6. *If A is irreducible and weakly diagonally dominant, then A is nonsingular.*

Proof. Suppose $A\mathbf{x} = 0$ for some $\mathbf{x} \neq 0$. Let m be the index such that

$$|a_{mm}| > \sum_{j \neq m} |a_{mj}|. \tag{5.33}$$

Let

$$J = \{k : |x_k| \geq |x_i|, \forall i, \quad |x_k| > |x_j| \text{ for some } j\}.$$

Clearly J is nonempty, for this would imply that $|x_1| = |x_2| = \cdots |x_n| \neq 0$. Hence

$$|a_{mm} x_m| \leq \sum_{j \neq m} |a_{mj} x_j|$$

which in turn implies

$$|a_{mm}| \leq \sum_{j \neq m} |a_{mj}|$$

which is a contradiction to (5.33).

Now for any $k \in J$,

$$|a_{kk}| \leq \sum_{j \neq k} |a_{kj}||x_j|/|x_k| \leq \sum_{j \neq k} |a_{kj}| \leq |a_{kk}|,$$

where the second inequality is strict whenever $|x_k| > |x_j|$, (i.e., $j \notin J$). So $a_{kj} = 0$ for all $k \in J$, $j \notin J$. Then A is reducible, a contradiction. □

Corollary 5.3.7. *If A is symmetric, irreducible and weakly diag. dominant with nonnegative diagonal elements, then A is positive definite.*

Proof. Since A is symmetric, eigenvalues are real and by G-disk theorem the G-disks are lying in the right upper half plane and eigenvalues are nonzero, hence all eigenvalues are positive. □

Convergence of Jacobi method

Let $G = I - D^{-1}A$. Then we have

$$\mathbf{x}^{(k)} = G\mathbf{x}^{(k-1)} + D^{-1}\mathbf{b}.$$

The true solution satisfies $\mathbf{x} = MG\mathbf{x} + D^{-1}\mathbf{b}$. Subtracting, we get

$$\mathbf{x}^{(k)} - \mathbf{x} = G(\mathbf{x}^{(k-1)} - \mathbf{x}).$$

Thus, the error $\varepsilon^{(k)} = \mathbf{x}^{(k)} - \mathbf{x}$ satisfies

$$\varepsilon^{(k)} = G\varepsilon^{(k-1)} = \cdots = G^k\varepsilon^{(0)}.$$

The error approaches to 0 iff $G^k \to 0$ iff $\rho(G) < 1$.

Theorem 5.3.8. *If A is irreducible and weakly diagonally dominant(called **irreducibly diagonally dominant**), then the Jacobi method converges.*

Proof. If λ is an eigenvalue of $G = I - D^{-1}A$, then

$$\begin{aligned} 0 = \det(G - \lambda I) &= \det(I - D^{-1}A - \lambda I) \\ &= \det(D^{-1}[L + U - \lambda D]) \\ &= \lambda^n \det(D^{-1}) \det\left(\frac{L+U}{\lambda} - D\right). \end{aligned}$$

Suppose $|\lambda| \geq 1$. then $\lambda \neq 0$, hence

$$\det\left(\frac{L+U}{\lambda} - D\right) = 0.$$

Since $|\lambda| \geq 1$, the matrix $\lambda^{-1}(L+U) - D$ is weakly diagonal dominant since the sum of absolute value of off-diagonal do note exceed the diagonal. Also it is irreducible since A is(they have exactly the same structure!). Now by Theorem 5.3.6 it is nonsingular and hence λ cannot be an eigenvalue of G. This is a contradiction. □

Example 5.3.9. Consider a partial differential equation on a square domain.

$$\begin{aligned} -\Delta u &= f \text{ in } \Omega \\ u &= g \text{ on } \partial\Omega. \end{aligned}$$

We discretize it by the 5-point stencil, then we get

$$A = \left[0, 0, -1, 0, 0, -1, 4, -1, 0, 0, -1, 0, 0\right].$$

$$A = D - T, \quad D = 4I, \quad G = D^{-1}T.$$

Note that the iteration matrix for Jacobi method is $G = I - D^{-1}A = D^{-1}T$. Even though Theorem 5.3.8 guarantees the convergence, in this special case we can argue directly that the Jacobi method is convergent. In fact, by Gershgorin-Disk theorem, all eigenvalues of G are ≤ 1 because all disks have radius ≤ 1 with center at the origin. Assume $\lambda = 1$ is an eigenvalue of G. Then since G is irreducible, all Gershgorin disk must pass through 1. This is a contradiction to the Gershgorin disk theorem(part 2).

Convergence of Gauss-Seidel method

Theorem 5.3.10. *If A is irreducible and weakly diagonally dominant, then the Gauss-Seidel method converges, i.e., $\rho(G) < 1$.*

Proof. Note that with $\bar{L} = D^{-1}L, \bar{U} = D^{-1}U$ Gauss-Seidel iteration matrix can be written as

$$G = (D - L)^{-1}U = (D(I - D^{-1}L))^{-1}U = (I - \bar{L})^{-1}\bar{U}.$$

If λ is an eigenvalue of G, then

$$\begin{aligned} 0 = \det(G - \lambda I) &= \det((I - \bar{L})^{-1}\bar{U} - \lambda I) \\ &= \det((I - \bar{L})^{-1}[\bar{U} - \lambda(I - \bar{L})]) \\ &= \det(I - \bar{L})^{-1} \det(\bar{U} - \lambda(I - \bar{L})) \\ &= \det(\bar{U} + \lambda\bar{L} - \lambda I). \end{aligned}$$

Let $\bar{G} = \frac{1}{\lambda}\bar{U} + \bar{L} - I$. If $|\lambda| \geq 1$, then $\lambda \neq 0$ and

$$\det\left(\frac{1}{\lambda}\bar{U} + \bar{L} - I\right) = 0.$$

The matrix $\frac{1}{\lambda}\bar{U} + \bar{L} - I$ is weakly diagonally dominant since the sum of absolute value of off-diagonal do note exceed the diagonal. Also $\bar{G}$ is irreducible since A is. Now by Theorem 5.3.6 $\bar{G}$ is nonsingular and hence λ cannot be an eigenvalue of G. This means $|\lambda| < 1$ for all λ, thus completes the proof of the theorem. □

Some general convergence theory

Definition 5.3.11 (Partial ordering for vectors)**.** For any two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, We write

$$\mathbf{x} \leq \mathbf{y} \text{ if and only if } x_i \leq y_i,\ i = 1, \cdots, n.$$

Analogously, we compare two matrices whenever possible, i.e,

$$A \leq B, \text{ if and only if } a_{ij} \leq b_{ij}.$$

Definition 5.3.12 (nonnegative matrix)**.** A matrix A is nonnegative if $A \geq 0$.

Definition 5.3.13 (M-matrix)**.** A matrix A is an M(onotone)-matrix if A is invertible, $A^{-1} \geq 0$ and $a_{ij} \leq 0$ for all $i, j = 1, \cdots, n, i \neq j$. A symmetric M-matrix is a Stieltjes matrix.

The following result can be found in R.A. Horn and C.R. Johnson, Topics in Matrix Analysis, Cambridge University Press, 1991.

Theorem 5.3.14. *Let A be real $n \times n$ matrix with nonpositive off-diagonal entries. The followings are equivalent:*

(1) A is an M-matrix;

(2) A is nonsingular and $A^{-1} \geq 0$;

(3) All eigenvalues of A have positive real part;

(4) Every real eigenvalue of A is positive.

Lemma 5.3.15. *A nonnegative matrix preserves partial ordering.*

For a vector $\mathbf{x}$, let us denote $|\mathbf{x}| = (|x_1|, \cdots, |x_n|)$.

Lemma 5.3.16. *If G is irreducible and satisfies*

$$\sum_{j=1}^{n} |g_{ij}| \leq 1, \quad \textit{for all } i \tag{5.34}$$

and strict inequality holds at least for one i, then $\rho(G) < 1$.

Proof. Clearly $\rho(G) \leq \|G\| \leq 1$. Let $\lambda(|\lambda| = 1)$ be an eigenvalue of G. Then $\lambda I - B$ is singular, but

$$|\lambda - g_{ii}| \geq 1 - |g_{ii}| \geq \sum_{j \neq i}^{n} |g_{ij}|$$

by the hypothesis, and a strict inequality holds for some i. Hence $\lambda I - G$ is weakly diag. dominant. Of course is irreducible. Hence $\lambda I - G$ is nonsingular by theorem 5.3.6, which is a contradiction. □

Remark 5.3.17. This lemma prove the convergence of Jacobi and G-S method at one stroke.

Theorem 5.3.18. *Let $A \in R^{n,n}$ with $a_{ij} \leq 0, i \neq j$. Then A is an M-matrix if and only if*

(1) the diagonal entries are positive and

(2) the matrix $G = I - D^{-1}A, D = \text{diag}(a_{11}, \cdots, a_{nn})$ satisfies $\rho(G) < 1$.

Proof. Suppose (1) and (2) are satisfied. Then $G = D^{-1}(D - A) \geq 0$ by (1). Since $\rho(G) < 1$, $I - G$ is invertible, A^{-1} exists and $(I - G)^{-1} = (D^{-1}A)^{-1}$. By Neumann lemma, $(I - G)^{-1} \geq 0$. and since $D \geq 0$, we have $A^{-1} \geq 0$.

Conversely, if A is an M-matrix, then the diagonal entries are positive, for if some $a_{ii} \leq 0$, then the i-th column $\mathbf{a}^i$ is nonpositive and hence $\mathbf{e}_i = A^{-1}\mathbf{a}^i \leq 0$. This is a contradiction. Hence $D > 0$ and D is invertible. It follows that $(I - G) = D^{-1}A$ is invertible and $(I - G)^{-1} = A^{-1}D \geq 0$ and again by Neumann lemma,we have $\rho(G) < 1$. □

Corollary 5.3.19. *If A is an M-matrix, then Jacobi method converges.*

A criterion for M-matrix is the following.

Theorem 5.3.20. *Let A be strictly diagonally dominant, or weakly diagonally dominant and irreducible and assume $a_{ij} \leq 0$ $i \neq j$ and that $a_{ii} > 0$, for $i = 1, \cdots, n$ then A is an M-matrix.*

Proof. Define $G = I - D^{-1}A$. Then it suffices to show $\rho(G) < 1$. By diagonal dominance of A, the condition (5.34) holds for G. If A is strictly diagonally dominant, a strict inequality holds in (5.34) in which case, $\|G\|_\infty < 1$. If A is weakly diagonally dominant and irreducible, the result follows from Lemma 5.3.16. □

Theorem 5.3.21 (Perron-Frobenius)**.** *If A, $a_{ij} \geq 0$, irreducible, then*

(1) A has a simple positive eigenvalue $\rho(A)$ (called the Perron root of A).

(2) $|\lambda_i| \leq \rho$ for every eigenvalues of A.

*(3) $A\mathbf{x} = \rho\mathbf{x}$ for some **positive** vector $\mathbf{x}$.*

See R. Varga "matrix iterative analysis."

Theorem 5.3.22 (Neumann Lemma)**.** *Let $B \in \mathbb{C}^{n,n}$ and assume $\rho(B) < 1$. Then $(I - B)^{-1}$ exists and*

$$(I - B)^{-1} = \lim_{k\to\infty} \sum_{i=0}^{k} B^i.$$

Proof. Since $\rho(B) < 1$, $(I - B)$ has no zero eigenvalues and hence invertible. To show the series note

$$(I - B)(I + B + \cdots + B^{k-1}) = I - B^k.$$

Thus

$$I + B + \cdots + B^{k-1} = (I - B)^{-1} - (I - B)^{-1}B^k.$$

Since $\rho(B) < 1$, there exists a norm $\|\cdot\|_\beta$ such that $\|B\|_\beta < 1$. For this norm, we see $\lim_{k\to\infty} B^k = 0$ and hence the above series converges to $(I - B)^{-1}$. Hence

$$\lim_{k\to\infty} \|\sum_{i=0}^{k} B^i - (I - B)^{-1}\|_\beta = 0.$$

Since all norms are equivalent in finite dimensional space, we see the convergence is guaranteed for any norm. □

As a consequence, we see that if $\|B\| < 1$, $I - B$ is invertible and

$$\|(I - B)^{-1}\| \leq \sum_{i=0}^{\infty} \|B\|^i = \frac{1}{(1 - \|B\|)}.$$

Theorem 5.3.23 (Neumann Lemma, positive entry)**.** *Let $B \in \mathbb{R}^{n,n}$ and assume that $B \geq 0$. Then $(I - B)^{-1}$ exists and nonnegative if and only if $\rho(B) < 1$.*

Proof. If $\rho(B) < 1$ then Neumann Lemma assures the existence of $(I-B)^{-1} = \sum_{i=0}^{\infty} B^i$ and each term of the sum is nonnegative. Hence $(I - B)^{-1} \geq 0$. Conversely, assume $(I - B)^{-1} \geq 0$ and let λ be any eigenvalue of B with eigenvector $\mathbf{x}$. Then $|\lambda||\mathbf{x}| \leq B|\mathbf{x}|$ so that

$$(I - B)|\mathbf{x}| \leq (1 - |\lambda|)|\mathbf{x}|.$$

Consequently,

$$|\mathbf{x}| \leq (1 - |\lambda|)(I - B)^{-1}|\mathbf{x}|,$$

from which it follows that $|\lambda| < 1$ since $(I - B)^{-1} \geq 0$ and $\mathbf{x} \neq 0$. □

Theorem 5.3.24. *If D, L are nonnegative, D is diagonal and invertible, and L is strictly lower triangular, then $(D - L)^{-1} \geq 0$.*

Proof. Use above Theorem. □

Theorem 5.3.25 (Perturbation lemma)**.** *Let $A, C \in \mathbb{C}^{n,n}$ and assume A is invertible with $\|A^{-1}\| \leq \alpha$. If $\|A - C\| \leq \beta$ and $\beta\alpha < 1$, then C is invertible and*

$$\|C^{-1}\| \leq \frac{\alpha}{1 - \alpha\beta}.$$

Proof. Since $\|I - A^{-1}C\| = \|A^{-1}(A - C)\| \leq \alpha\beta < 1$ and $A^{-1}C = I - (I - A^{-1}C)$, it follows from Neumann lemma that $A^{-1}C$ is invertible. Since

$$C = AA^{-1}C = A[I - (I - A^{-1}C)],$$

we have $C^{-1} = (A^{-1}C)^{-1}A^{-1} = [I - (I - A^{-1}C)]^{-1}A^{-1}$ and

$$\|C^{-1}\| = \|[I - (I - A^{-1}C)]^{-1}A^{-1}\| \leq \alpha \sum_{i=0}^{\infty} (\alpha\beta)^i = \frac{\alpha}{1 - \alpha\beta}.$$

□

Average reduction of a iterative method

The error of the linear iteration $\mathbf{x}^{(k)} = G\mathbf{x}^{(k-1)} + \mathbf{b}$ satisfies $\mathbf{e}^{(k)} = G^k\mathbf{e}^{(0)}$ hence $\|\mathbf{e}^{(k)}\| \leq \|G^k\|\|\mathbf{e}^{(0)}\|$ or

$$\frac{\|\mathbf{e}^{(k)}\|}{\|\mathbf{e}^{(0)}\|} \leq \|G^k\| \leq \|G\|^k$$

which suggests us to define: $\sigma = \left(\frac{\|\mathbf{e}^{(k)}\|}{\|\mathbf{e}^{(0)}\|}\right)^{1/k}$: The **average reduction** per iteration for k iterations

$$\sigma \leq \|G^k\|^{\frac{1}{k}} = e^{\frac{1}{k}\ln\|G^k\|}.$$

Thus, the decay rate is $-\ln\|G^k\|^{\frac{1}{k}} \equiv R(G^k)$. "The bigger the better."

Theorem 5.3.26 (Stein-Rosenberg Theorem)**.** *Let J be the Jacobi iteration matrix. G be Gauss-Seidel iteration matrix. If $J \geq 0$ (which is often the case), then one of the following holds:*

(1) $\rho(J) = \rho(G) = 0$

(2) $0 < \rho(G) < \rho(J) < 1$ *(i.e., when they converges G-S is faster)*

(3) $\rho(G) = \rho(J) = 1$

(4) $1 < \rho(J) < \rho(G)$ *(i.e., where they diverge, G-S diverges faster).*

The hyposthesis is satisfied if A is an L-matrix, i.e, $a_{ii} > 0$ and $a_{i,j} \leq 0$ for $i \neq j$.

Theorem 5.3.27 (Kahan's Theorem)**.** *In SOR iteration, we have $\rho(\mathcal{L}_\omega) \geq |\omega - 1|$, for all ω real.*

Proof. Recall the SOR iteration is given by

$$\mathbf{x}^{(k+1)} = (D - \omega L)^{-1}\{(1-\omega)D + \omega U\}\mathbf{x}^{(k)} + (D - \omega E)^{-1}\omega\mathbf{f}.$$

The eigenvalues μ_i of L_ω are the zeros of characteristic polynomials, thus satisfy

$$\prod_{i=1}^{n} \mu_i = det(L_\omega).$$

Since the matrix $(D - \omega L)^{-1}\{(1-\omega)D + \omega U\}$ is triangular, we see

$$\prod_{i=1}^{n} \mu_i = det(L_\omega) = (1-\omega)^n.$$

Thus $\rho(L_\omega) \geq |1 - \omega|$. □

Corollary 5.3.28. *SOR Converges only when $0 < \omega < 2$.*

For a class of matrices the best parameter is attained at $\omega_b = \frac{2}{1+\sqrt{1-\rho^2(J)}}$,where J is the iteration matrix for the Jacobi method. There are many case that $\rho^2(J)$ is close to 1 so that ω_b is close to 2.

R. S. Varga: Matrix Iterative Analysis, 1965, Prentice Hall. D. M. Young: Iterative Solution of Large Linear Systems, AP, 1971.

Direct	Iterative
Gaussian Elim.	Jacobi(Parellel), Gauss-Seidel, SOR(sequential)
Deterministic	Stopping criterior
Costly	Capitalize on sparseness
Generally Applicable	Convergence restricted to certain classes
only small # of unknowns	large # of unknowns allowed

Table of Comparison

Exercise 5.3.29. (1) Extend the Neumann Lemma to the Banach space as follows: Let $B : X \to X$ be a bounded linear operator on a Banach space X with $\|B\| < 1$ and let I be the identity on X. Then for each $f \in X$ the successive approximation

$$x_{n+1} = Bx_n + f,\ n = 0, 1, 2, \cdots,$$

with arbitrary x_0 converges to an element $x \in X$ and this x satisfies the equation

$$x - Bx = f.$$

In other words, the mapping $(I - B)$ is onto. Furthermore, we have

$$\|x_n - x\| \leq \frac{1}{1 - \|B\|}\|x_1 - x_0\|.$$

(Hint. Use Banach theorem and proof used there.)

(2) Prove part 2 of Theorem 5.3.1(Gershgorin disk theorem): If an eigenvalue λ lies on the boundary of a disk, then every disk must pass through λ.

(3) Let $A = (a_{ij})$ be a $n \times n$ nonnegative matrix and let $\rho(A) < 1$. Show that $(I - A)^{-1}$ is nonnegative.

Chapter 6

Algebraic eigenvalue problem

Eigenvalue problem has many important applications other than the eigenvalue itself. For example, we convert problem of finding zeros of polynomial equation $p(x) = x^n + a_{n-1}x^{n-1} \cdots + a_0 = 0$ to an eigenvalue problem: Let

$$C = \begin{bmatrix} 0 & 1 & 0 & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ -a_0 & & \cdots & -a_{n-1} \end{bmatrix}$$

be the companion matrix. Then $\det(xI - C) = p(x)$.(Expand with respect to first column) We solve this eigenvalue problem to find zeros of a polynomial but we never solve polynomial equation to find eigenvalues of a matrix.

6.1 Inclusion or exclusion theorem

Theorem 6.1.1 (Gershgorin disk theorem)**.** $A \in \mathbb{C}^{n,n}$, $A\mathbf{x} = \lambda\mathbf{x}$, $\mathbf{x} \neq 0$.

$$\lambda \in \bigcup \left\{ \xi \mid |\xi - a_{ii}| \leq \sum_{j \neq i} |a_{ij}| \right\}.$$

Example 6.1.2.

$$A = \begin{bmatrix} 3+4i & 1 & 0 & 2 \\ 1 & -3+4i & 2 & 0 \\ 1 & 3 & -4-3i & 2 \\ 3 & 1 & 2 & 4-3i \end{bmatrix}$$

$$\rho_1 = 1 + 2 = 3,\ \rho_2 = 3,\ \rho_3 = 6,\ \rho_4 = 6.$$

Theorem 6.1.3. *If the union of m disks are disjoint from the other disks, then the union contains precisely m eigenvalues.*

Proof. Let $\Gamma = \bigcup_i^n D_i$, $D_i = \{\xi \mid |\xi - a_{ii}| \le \rho_i\}$ and $\Gamma_1 = \bigcup_{j=1}^m D_j$ be a union of disks disjoint from $\Gamma_2 = \bigcup_k D_k$, i.e,

$$\Gamma_1 \cap \Gamma_2 = \phi, \qquad \Gamma_1 \cup \Gamma_2 = \Gamma.$$

Set for $0 \le \varepsilon \le 1$

$$A(\varepsilon) \;=\; \varepsilon[A - \mathrm{diag}\{a_{11}, \ldots, a_{nn}\}] + \mathrm{diag}\{a_{11}, \ldots, a_{nn}\}.$$

Then

$$\begin{aligned} A(1) &= A, \\ A(0) &= \mathrm{diag}\{a_{11}, \ldots, a_{nn}\}. \end{aligned}$$

i.e, $A(\varepsilon)$ is a homotopy and the eigenvalues of A are exactly $\bigcup_1^m D_i(0)$. The G-disks of $A(\varepsilon)$ are

$$\begin{aligned} D_i(\varepsilon) &= \{\xi \mid |\xi - a_{ii}| \le \varepsilon\rho_i\} \subset D_i(1), \qquad i = 1, 2, \ldots, n. \\ \text{Thus, } \Gamma_1(\varepsilon) &= \bigcup_1^m D_i(\varepsilon) \subset \Gamma_1 \equiv \Gamma_1(1). \end{aligned}$$

From the first Gershgorin theorem, the eigenvalues of $A(\varepsilon)$ varies within $\Gamma_1(\varepsilon) \cup \Gamma_2(\varepsilon)$ as ε varies from 0 to 1. But any eigenvalue of $A(\varepsilon)$ for $\varepsilon < 1$ contained in $\Gamma_1(\varepsilon) \subset \Gamma_1(1)$ must stay there $\Gamma_1(1)$ as ε increases. (The eigenvalues are continuous function of ε and $A(0)$ has exactly m-eigenvalues in $\Gamma_1(0)$) □

Isolation of eigenvalues

Eigenvalues are contained in the union of G-disks, i.e,

$$\lambda \in \bigcup_i G_i, \quad G_i = \{\xi : |\xi - a_{ii}| \le \rho_i = \sum_{j \ne i} |a_{ij}|\}.$$

Note that $D^{-1}AD$ has the same eigenvalues as A. Set $D = \mathrm{diag}\{d_1, \ldots, d_n\}$ $d_i \ne 0$, $i = 1, \ldots, n$ and apply G-disk theorem to $D^{-1}AD$ so that for any eigenvalue λ of A,

$$\lambda \in \bigcup_i G_i^* = \bigcup_i \{z \mid |z - a_{ii}| \le \rho_i^* = \frac{1}{|d_i|} \sum_{j \ne i} |a_{ij} d_j|\}.$$

This is called 'isolation' or 'contraction' of Gershgorin disks. See Varga's book.

exer. Find a better estimate of eigenvalues for Example 6.1.2.

Theorem 6.1.4. *Assume $A \in \mathbb{C}^{n,n}$. Then all the eigenvalues of A belong to the union of sets c_{ij} (called* **Cassini ovals***) given by*

$$c_{ij} = \{\xi \mid |\xi - a_{ii}| \cdot |\xi - a_{jj}| \le \rho_i \rho_j\}, \quad \rho_i = \sum_{j \ne i} |a_{ij}|.$$

Proof. Assume $|x_s| = \max_{1\le i\le n} |x_i|$, $|x_r| = \max_{i\neq s} |x_i|$. Then we see

$$(\lambda - a_{rr})x_r = \sum_{j\neq r} a_{rj}x_j.$$

Hence

$$|\lambda - a_{rr}| \cdot |x_r| \quad \le \quad \sum_{j\neq r} |a_{rj}|\,|x_j| \le \rho_r |x_s|.$$

By the same way

$$|\lambda - a_{ss}| \cdot |x_s| \quad \le \quad \sum_{j\neq s} |a_{sj}| \cdot |x_j| \le \rho_s |x_r|.$$

Multiplying two equations, we have

$$|\lambda - a_{rr}| \cdot |\lambda - a_{ss}| \quad \le \quad \rho_r \rho_s.$$

□

Theorem 6.1.5. *The union of the ovals is always contained in the union of G-disks.*

Eigenvalue problem

Object: Construct a sequence of easily invertible matrices $\{P_k\}$ in the sequence $\{A_k\}$ given by $A_{k+1} = P_k^{-1} A_k P_k$, where the eigenvalue problem for A_{k+1} is more easily solved than that for A_k.

Ideal Algorithm: For some finite value of k, this algorithm yields a triangular matrix.

Remark 6.1.6. Such $\{P_k\}$ does not exists in general.

6.2 Jacobi algorithm (Hermitian case)

In 1846, Jacobi found a method of annihilating off diagonal elements by **similarity transform** using elementary rotation. As a motive, we can imagine removing the cross term in the equation of quadratic curves by rotating the axes. For simplicity, we assume A is real symmetric. It is easily generalized to Hermitian case.

Let $R(p,q)$ be the (plane) rotation matrix with $p < q$(in Y-G book, the order is reversed), called 'elementary rotation matrix'(corresponding to counterclockwise direction)

$$R(p,q) = \begin{bmatrix} 1 & \vdots & & \vdots & \\ \dots & c & \dots & -s & \dots \\ & \vdots & 1 & \vdots & \\ \dots & s & \dots & c & \dots \\ & \vdots & & \vdots & 1 \end{bmatrix}$$

Recall 2×2 case: Any quadratic form $ax^2 + bxy + cy^2$ can be transformed into $a'\xi^2 + b'\eta^2 = 1$ by a rotation matrix $R(R^T \cdot R = I)$. Let $A = \{a_{ij}\}$ be symmetric, then $A_1 = R^T AR$ is also symmetric. Let $R(p,q)$ be the (plane) rotation matrix (same as Givens) with $p < q$, called 'elementary rotation matrix'(corresponding to counterclockwise direction)

$$R_{pq} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

$$R_{pq}^T A R_{pq} = \begin{bmatrix} c & s \\ -s & c \end{bmatrix} \begin{bmatrix} a_{pp} & a_{pq} \\ a_{pq} & a_{qq} \end{bmatrix} \begin{bmatrix} c & -s \\ s & c \end{bmatrix} = \begin{bmatrix} d_1 & 0 \\ 0 & d_2 \end{bmatrix}$$

The entries of $R^T AR$ are computed as follows:

$$a'_{pi} = (R^T AR)_{p,i} = \sum_{k,\ell} r_{kp} a_{k\ell} r_{\ell i}.$$

Since $r_{pp} = \cos\theta$, $r_{pq} = -\sin\theta$, $r_{qp} = \sin\theta$, $r_{qq} = \cos\theta$, $r_{ii} = 1$ and $r_{ij} = 0$ for all other i, j, we see $a'_{pi} = \sum_k r_{kp} a_{ki}$. Hence we have

$$\begin{aligned} a'_{pi} &= a_{pi}\cos\theta + a_{qi}\sin\theta, \\ a'_{qi} &= -a_{pi}\sin\theta + a_{qi}\cos\theta, \end{aligned} \qquad i \neq p, q. \tag{6.1}$$

Similarly, from

$$a'_{ip} = (R^T AR)_{i,p} = \sum_{k,\ell} r_{ki} a_{k\ell} r_{\ell p},$$

we have

$$\begin{aligned} a'_{ip} &= a_{ip}\cos\theta + a_{iq}\sin\theta, \\ a'_{iq} &= -a_{ip}\sin\theta + a_{iq}\cos\theta, \end{aligned} \qquad i \neq p, q \tag{6.2}$$

and

$$\begin{aligned} a'_{pq} &= (R^T AR)_{p,q} = \sum_{k,\ell} r_{kp} a_{k\ell}\, r_{\ell q} \\ &= \sum_{k} r_{kp}[a_{kp}(-\sin\theta) + a_{kq}\cos\theta] \\ &= \cos\theta[a_{pp}(-\sin\theta) + a_{pq}\cos\theta] + \sin\theta[a_{qp}(-\sin\theta) + a_{qq}\cos\theta] \\ &= (a_{qq} - a_{pp})\sin\theta\cos\theta + a_{pq}(\cos^2\theta - \sin^2\theta) \\ &= \frac{(a_{qq} - a_{pp})}{2}\sin 2\theta + a_{pq}\cos 2\theta. \end{aligned} \tag{6.3}$$

If we choose θ so that $\cot 2\theta = (a_{pp} - a_{qq})/2a_{pq}$, the transformation(plane rotation) annihilates the (p, q) element of A. (If $a_{pp} - a_{qq} = 0$, we choose $\theta = \pi/4$). Let $\phi \equiv \cot 2\theta$, $t = s/c$ then

$$\phi = \frac{c^2 - s^2}{2sc} \Rightarrow t^2 + 2\phi t - 1 = 0 \Rightarrow t = -\phi \pm \sqrt{\phi^2 + 1}. \tag{6.4}$$

To avoid the cancelation we choose the sign so that following holds.

$$t = -\phi \pm \sqrt{\phi^2 + 1} = \frac{1}{\phi \pm \sqrt{\phi^2 + 1}} = \frac{\text{sgn }(\phi)}{|\phi| + \sqrt{\phi^2 + 1}}. \tag{6.5}$$

If ϕ is so large that ϕ^2 would overflow, we use the relation

$$\frac{1}{\phi} = \tan 2\theta \approx 2\tan\theta$$

so set $t = 1/(2\phi)$. Hence

$$c = \frac{1}{\sqrt{t^2 + 1}}, \quad s = tc. \tag{6.6}$$

The square sum of all other elements do not change as the following theorem shows.

Theorem 6.2.1. *The plane rotation reduces the sum of squares of off-diagonal elements of A by $2a_{pq}^2$.*

Proof. See (6.1)-(6.3) and note that for $i \neq p, j \neq q$,

$$\begin{aligned} (a'_{pi})^2 + (a'_{qi})^2 &= a_{pi}^2 + a_{qi}^2 \\ (a'_{ip})^2 + (a'_{iq})^2 &= a_{ip}^2 + a_{iq}^2 \\ a'_{pq} &= a'_{qp} = 0. \end{aligned}$$

All other elements a_{ij}, for $i \neq p, j \neq q$ do not change. Hence the square sum is reduced by $2a_{pq}^2$. □

As a reference, the values a'_{pp}, a'_{qq} are computed as

$$\begin{aligned} a'_{pp} &= (a_{pp}\cos\theta + a_{pq}\sin\theta)\cos\theta \\ &\quad +(a_{pq}\cos\theta + a_{qq}\sin\theta)\sin\theta \\ &= a_{pp}\cos^2\theta + 2a_{pq}\sin\theta\cos\theta + a_{qq}\sin^2\theta, \\ a'_{qq} &= (-a_{qp}\sin\theta + a_{qq}\cos\theta)\cos\theta \\ &\quad +(-a_{pp}\sin\theta + a_{qp}\cos\theta)(-\sin\theta) \\ &= a_{pp}\sin^2\theta - 2a_{qp}\cos\theta\sin\theta + a_{qq}\cos^2\theta. \end{aligned}$$

Theorem 6.2.2 (Jacobi)**.** *Let A be symmetric. If a sequence is given by*

$$A_{k+1} = R^T(p_k, q_k)A_k R(p_k, q_k)$$

which annihilates off diagonal element of A_k of largest modulus, then A_k approaches a diagonal matrix.

Theorem 6.2.3 (Goldsteine, Murray, Neumann,1959)**.** *The sequence above converges to a diagonal matrix when the order is chosen so that A_{k+1} annihilates off diagonal element of A_k whose entry is greater than the average of off diagonal elements.*

Remark 6.2.4. In practice, the search for largest off-diagonal element is costly ($O(n^2)$ more than updating). So use cyclic Jacobi. The Jacobi method is essential eclipsed by (symmetric) QR(below), it is efficient in parallel computation. It is also useful to find eigenvalues of nearly diagonal matrices. Advantage of Jacobi method is that it produces eigenvector at the same time, since

$$A_{m+1} = R^T A_m R \sim \Lambda$$

so that $AR \sim R\Lambda$.

Special cyclic Jacobi- Henrici 1958

An alternative of selecting the order of elimination is "Typewriter fashion": $(2,1),(3,1),(3,2),(4,1),(4,2),(4,3),\cdots,(n,1),(n,2),\cdots,(n,n-1)$. It is known to converges, as long as we can choose $|\theta| \le \pi/4$.

$$\begin{bmatrix} * & * & \cdots & * & * \\ \cancel{0} & * & \cdots & * & * \\ \cancel{0} & \cancel{0} & \cdots & * & * \\ \cancel{0} & \cancel{0} & \to & * & * \\ \cancel{0} & \cancel{0} & \cdots & \cancel{0} & * \end{bmatrix}$$

Remark 6.2.5. In Jacobi algorithm the 0's created by a plane rotation do not in general remain zeros in the subsequence rotations. This motivated Givens(1954) to introduce an algorithm (see section 6.3) which keeps the zeros once they are created.

Givens rotation

Consider the plane rotation $R(p,q)$ introduced in Jacobi. Let $\mathbf{y} = R(p,q)\mathbf{x}$. Then

$$y_j = \begin{cases} cx_p - sx_q, & j = p \\ sx_p + cx_q, & j = q \\ x_j, & j \neq p, q. \end{cases}$$

If we choose $c := \cos\theta, s := \sin\theta$, where

$$c = \frac{x_p}{(x_p^2 + x_q^2)^{1/2}}, \ s = \frac{-x_q}{(x_p^2 + x_q^2)^{1/2}}, \tag{6.7}$$

then we have $y_q = 0$. Thus we arrived at a (onesided) method of annihilating the q-th entry of a vector using p and q-th entries. The angle is chosen to satisfy $\tan\theta = -x_q/x_p$.[1]

$$\begin{bmatrix} c & -s \\ s & c \end{bmatrix} \begin{bmatrix} x_p \\ x_q \end{bmatrix} = \begin{bmatrix} \alpha \\ 0 \end{bmatrix}$$

Here

$$c = \frac{x_p}{(x_p^2 + x_q^2)^{1/2}}, \ s = \frac{-x_q}{(x_p^2 + x_q^2)^{1/2}}.$$

Note that this Givens rotations are usually used for QR decomposition as a one sided rotation to transform $[x_p, x_q]$ to $[\alpha, 0]$ ($\alpha = \sqrt{x_p^2 + x_q^2}$). The matrix has the same form as jacobi rotation but the way to choose the pivot and the angle is different.

6.3 Givens algorithm for tridiagonalization

As mentioned in the previous section, Givens(1954) proposed an algorithm for *tridiagonalizing* a symmetric matrix, which preserves the zeros in the off diagonal positions once they are created by using the plane rotation. Instead, *we have to be content to stop when the matrix is tridiagonal. This allows the procedure to be carried out in a finite number of steps, unlike the Jacobi method, which requires iteration to convergence.*

Example 6.3.1. We choose the rotation angle in equation (6.3) so as to zero an element that is not at one of the four corners, i.e., not a_{pp}, a_{pq} or a_{qq} in equation (11.1.3). Specifically, we first choose R_{23} to annihilate a_{31} (and, by symmetry, a_{13}). Choose angle so that

$$a'_{31} = ca_{31} - sa_{21} = 0.$$

[1] Note that in Jacobi, we choose $\tan 2\theta = 2a_{pq}/(a_{pp} - a_{qq})$ to diagonalize by similarity transform.

Hence $\tan\theta = \frac{a_{31}}{a_{21}}$. Next we choose R_{24} to annihilate a_{41}. Once a_{31} has been changed to 0, the rotation R_{24}(annihilating a_{41}) do not change 3rd row. Hence a_{31} entry remain 0. Similarly, the subsequence rotations $R_{25}, \cdots, R_{2n}$ annihilate a_{51}, a_{61}, ... ,a_{n1} while they do not change 4-th row element, 5-th row, etc. Hence the first column below the subdiagonal element is annihilated.

$$R_{23}A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c & -s & 0 \\ 0 & s & c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a'_{21} & a'_{22} & a'_{23} & a'_{24} \\ 0 & a'_{32} & a'_{33} & a'_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} = A_1$$

$$R_{24}A_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c & 0 & -s \\ 0 & 0 & 1 & 0 \\ 0 & s & 0 & c \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a'_{21} & a'_{22} & a'_{23} & a'_{24} \\ 0 & a'_{32} & a'_{33} & a'_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a'_{21} & a'_{22} & a'_{23} & a'_{24} \\ 0 & a'_{32} & a'_{33} & a'_{34} \\ 0 & a'_{42} & a'_{43} & a'_{44} \end{bmatrix}$$

Also, the transpose of the same matrix has to be multiplied on the right.

In general, we choose the sequence

$$R_{23}, R_{24}, \cdots, R_{2n}, R_{34}, \cdots, R_{3n}, \cdots, R_{n-1,n}$$

where R_{jk} annihilates $a_{k,j-1}$.

$$\begin{bmatrix} \times & \times & \times & \dots & \times & \times \\ \times & \times & \times & \dots & \times & \times \\ \times & \times & \times & \dots & \times & \times \\ \times & \times & \times & \dots & \times & \times \\ \times & \times & \times & \dots & \times & \times \\ \times & \times & \times & \dots & \times & \times \end{bmatrix} \underset{\substack{(3,1)\\(4,1)\\ \dots \\ (n,1)}}{\Longrightarrow} \begin{bmatrix} \times & \times & 0 & \dots & 0 & 0 \\ \times & \times & \times & \dots & \times & \times \\ 0 & \times & \times & \dots & \times & \times \\ 0 & \times & \times & \dots & \times & \times \\ 0 & \times & \times & \dots & \times & \times \\ 0 & \times & \times & \dots & \times & \times \end{bmatrix} \underset{\substack{(4,2)\\(5,2)\\ \dots \\ (n,2)}}{\Longrightarrow} \begin{bmatrix} \times & \times & 0 & 0 & 0 & 0 \\ \times & \times & \times & 0 & 0 & 0 \\ 0 & \times & \times & \dots & \times & \times \\ 0 & 0 & \times & \dots & \times & \times \\ 0 & 0 & \times & \dots & \times & \times \\ 0 & 0 & \times & \dots & \times & \times \end{bmatrix}$$

This is called the Givens algorithm (for tridiagonalization). This procedure has the advantage that it requires only a finite number of similarity transformations. (Next step is to compute eigenvalues of a tridiagonal matrix by a root finding algorithm-'bisection'.) Evidently, of order $n^2/2$ rotations are required, and the number of multiplications in a straightforward implementation is of order $4n^3/3$, not counting those for keeping track of the product of the transformation matrices, required for the eigenvectors.

Later Householder(1958) proposed an algorithm which is just as stable as the Givens and twice as fast as Givens algorithm, so the Givens method is not generally used.

Remark 6.3.2. Recent work has shown that the Givens reduction can be reformulated to reduce the number of operations by a factor of 2, and also avoid the necessity of taking square roots. This appears to make the algorithm competitive with the Householder reduction. However, this fast Givens reduction has to be monitored to avoid overflows, and the variables have to be periodically rescaled. There does not seem to be any compelling reason to prefer the Givens reduction over the Householder method.

Summary of Givens algorithm

(1) Tridiagonalization by a finite number of similarity transformation (Originally by Givens later improved by Householder- See next section). It takes $(n-1)(n-2)/2$ rotations to produce a tridiagonal matrix.

(2) A simple root finding algorithm (bisection see below) for the eigenvalues of tridiagonal matrix.

Definition 6.3.3. A matrix $A = \{a_{ij}\}$ is called an **Upper-Hessenberg** matrix if $a_{ij} = 0$ for $i > j + 1$. i.e., the following type is an upper-Hessenberg matrix. A Symmetric upper Hessenberg matrix is a tridiagonal matrix.

$$\begin{bmatrix} \times & \times & \dots & & \times \\ \times & \times & \dots & & \times \\ 0 & \times & \dots & & \times \\ 0 & 0 & \dots & & \\ 0 & 0 & \dots & \times & \times \end{bmatrix}$$

6.4 Householder transformations

Now we provide a tridiagonalization by similarity transformation modified by Householder. Instead of using $n-k-1$ plane rotations to create $n-k-1$ zeros in the k-th column of A as Givens does, Householder uses a *single orthogonal similarity transform* to do the job(as in Schur' lemma). In this way, tridiagonalization can be completed in exactly $n-2$ Householder transformations. This is twice as fast as the original Givens algorithm.

Householder transformations introduces many zeros at once. Although efficient, this approach is too heavy for selectivity is needed. So it is sometimes better to use Givens rotations which introduces zeros one at a time.

6.4.1 Reduction to U-H form by elementary reflector

Let $H = I - 2\mathbf{w}\mathbf{w}^*$ be an elementary reflector($\mathbf{w}^*\mathbf{w} = I$). It satisfies $H^* = H^{-1} = H$. Let us write A in a block matrix form:

$$A = A^{(0)} = \begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \end{bmatrix} = \begin{bmatrix} A_{11}^{(0)} & A_{12}^{(0)} \\ A_{21}^{(0)} & A_{22}^{(0)} \end{bmatrix},$$

where $A_{11}^{(0)} = a_{11}$ and $A_{21}^{(0)}$ denotes the first column of A less a_{11}, and $A_{22}^{(0)}$ denotes the $(n-1) \times (n-1)$ submatrix of A obtained by deleting first row and column etc.

Let $P_1 = \begin{bmatrix} 1 & 0 \\ 0 & I - 2\mathbf{w}\mathbf{w}^* \end{bmatrix}$, where $\mathbf{w} \in \mathbb{R}^{n-1}$ satisfies $\|\mathbf{w}\|_2 = 1$. Then we see $P_1^* P_1 = I$.(an elementary reflector) We seek to determine $\mathbf{w}$ such that

$$P_1 A_0 = \left[\begin{array}{c|c} a_{11} & A_{12}^{(0)} \\ \hline \mathbf{y} & A_{22}^{(1)} \end{array}\right] = \left[\begin{array}{c|ccc} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \hline 0 & \times & \times & \times \\ \vdots & \multicolumn{3}{c}{\cdots\cdots\ \times} \\ 0 & \multicolumn{3}{c}{\cdots\cdots\ \times} \end{array}\right], \tag{6.8}$$

where $\mathbf{y} = (I - 2\mathbf{w}\mathbf{w}^*)A_{21}^{(0)} = [*, 0, \cdots, 0]^T \in \mathbb{R}^{n-1}$ and $A_{22}^{(1)} = (I - 2\mathbf{w}\mathbf{w}^*) \cdot A_{22}^{(0)}$.

Thus if we can determine $\mathbf{w}$ such that

$$(I - 2\mathbf{w}\mathbf{w}^*)A_{21}^{(0)} = -\operatorname{sgn}(a_{21})\|A_{21}^{(0)}\|_2 \cdot \mathbf{e}_1, \tag{6.9}$$

then the 1st column of $P_1 A$ is U-H form. (The sign of $\|A_{21}^{(0)}\|_2$ is determined so that subtractive cancelation would not occur. i.e, we choose the opposite sign of the first entry of $A_{21}^{(0)}$.) Let $\mathbf{a}_2 = [a_{21}, a_{31}, \cdots, a_{n1}]$ be the first column of A without the first entry and $\alpha = -\operatorname{sgn} a_{21}\|\mathbf{a}_2\|$. As discussed earlier, we set

$$\mathbf{w} = \frac{\mathbf{a}_2 - \alpha\mathbf{e}_1}{\|\mathbf{a}_2 - \alpha\mathbf{e}_1\|} \in \mathbb{R}^{n-1}.$$

Then we have

$$(I - 2\mathbf{w}\mathbf{w}^*)\mathbf{a}_2 = \alpha\mathbf{e}_1. \tag{6.10}$$

This is exactly the inverse case of Schur lemma, (except that we are working in $\mathbb{R}^{n-1}$, see (5.1)). Thus

$$(I - 2\mathbf{w}\mathbf{w}^*)\mathbf{e}_1 = \mathbf{a}_2. \tag{6.11}$$

Also, note that the first row of $P_1 A^{(0)}$ is the same as that of $A^{(0)}$. Hence if we apply P_1^T on the right, we get

$$P_1 A^{(0)} P_1^T = \left[\begin{array}{c|c} a_{11} & \mathbf{a}_2' \\ \hline \alpha & \\ 0 & \\ \cdot & A_{22}^{(1)} \\ 0 & \end{array}\right] \left[\begin{array}{c|c} 1 & 0 \\ \hline 0 & \\ \vdots & I - 2\mathbf{w}\mathbf{w}^* \\ 0 & \end{array}\right] = \left[\begin{array}{c|c} a_{11} & \mathbf{a}_2'(I - 2\mathbf{w}\mathbf{w}^*) \\ \hline \alpha & \\ 0 & \\ \cdot & A_{22}^{(1)}(I - 2\mathbf{w}\mathbf{w}^*) \\ 0 & \end{array}\right]. \tag{6.12}$$

Thus for $A_{22}^{(1)} \in \mathbb{R}^{(n-1)\times(n-1)}$

$$P_1 A_0 = \left[\begin{array}{c|cc} \times & \cdots & \times \\ \times & \cdots & \times \\ \hline 0 & & \\ \vdots & A_{22}^{(1)} & \\ 0 & & \end{array}\right]$$

Write

$$A_{22}^{(1)} = \left[\begin{array}{c|c} \times & \cdots \quad \times \\ \hline A_{21}^{(1)} & \tilde{A}_{22}^{(1)} \end{array}\right],$$

where $\tilde{A}_{22}^{(1)} \in \mathbb{R}^{(n-2)\times(n-2)}$. Next we put

$$P_2 = \left[\begin{array}{cc|c} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \hline 0 & 0 & \\ \cdot & \cdot & I - 2\mathbf{w}\mathbf{w}^* \\ 0 & 0 & \end{array}\right]$$

and find $\mathbf{w} \in \mathbb{R}^{n-2}$ exactly the same way as above. i.e,

$$P_2P_1A_0 = \left[\begin{array}{c|c|c} \times & \cdots & \times \\ \times & \cdots & \times \\ \hline 0 & & \\ \vdots & \mathbf{y} & A_{22}^{(2)} \\ 0 & & \end{array}\right] = \left[\begin{array}{c|c|cc} \times & \times & \cdots & \times \\ \times & \times & \cdots & \times \\ \hline 0 & \times & & \\ \vdots & 0 & A_{22}^{(2)} & \\ 0 & 0 & & \end{array}\right]$$

where $\mathbf{y} = (I - 2\mathbf{w}\mathbf{w}^*)A_{21}^{(1)} = [*, 0 \cdots, 0]^T$ and $A_{22}^{(2)} \in \mathbb{R}^{(n-2)\times(n-2)}$. Apply $P_1^TP_2^T$ to the right, we still obtain an (semi) upper Hessenberg form $P_2P_1A^{(0)}P_1^TP_2^T$. Repeating the process, we arrive at an upper-Hessenberg form

$$P_{n-1}\cdots P_1AP_1^T\cdots P_{n-1}^T, \quad (A^{(k)} := P_k\cdots P_1AP_1^T\cdots P_{(k)}^T,).$$

If A is symmetric we arrive at a tridiagonal matrix: Thus we only need to compute the eigenvalues of an upper Hessenberg form(or tridiagonal if A is symmetric).

Remark 6.4.1. In actual computation, one never form the product $P_m\cdots P_1AP_1^T\cdots P_m^T$. Instead, in each step one computes $\mathbf{w}$ and compute $A_{22}^{(1)}$ by

$$A_{22}^{(1)} = (I - 2\mathbf{w}\mathbf{w}^*)\cdot A_{22}^{(0)} = A_{22}^{(0)} - 2\mathbf{w}\mathbf{w}^*A_{22}^{(0)}.$$

Here one first computes $\mathbf{u}^T := \mathbf{w}^*A_{22}^{(0)}$ by vector-matrix multiplication and then computes $2\mathbf{w}\mathbf{w}^*A_{22}^{(0)}$ by the vector-vector multiplication $2\mathbf{w}\mathbf{u}^T$. So the cost (multiplication only) is $2(n-k)^2$, $k = 1, 2\cdots, n-1$, total of $4n^3/3$.

Remark 6.4.2. Reduction to upper Hessenberg form by elementary matrices(permutation plus M_i like matrices in Gaussian elimination) are also possible and costs half of Householder's transformation. (See Y-G. p. 924)

Remark 6.4.3. As a global orthogonal transform, Householder is faster then Jacobi(Givens), but when we need to zero in a particular element, Jacobi(Givens) is better.

Remark 6.4.4. $\mathbf{a}_2' = \mathbf{a}_2^T$ if A is symmetric and hence

$$\mathbf{a}_2^T(I - 2\mathbf{w}\mathbf{w}^*) = \alpha\mathbf{e}_1^T.$$

Hence we have $a = b = 0$. If we were to transform the first column of A to $\alpha\mathbf{e}_1$ instead of $\alpha\mathbf{e}_2$, it would change the first row, hence the post multiplication by P_1^T would not, in general produce $\alpha\mathbf{e}_1^T$. This is why we can only suffice with tri-diagonal(or U-H) form.

6.4.2 Reduction to tridiagonal matrix when A is symmetric

Now assume A is symmetric. Change the notation: With $\mathbf{w}$ above being replaced by $\mathbf{v}$, we have $P_1 = \begin{bmatrix} 1 & 0 \\ 0 & I - 2\mathbf{v}\mathbf{v}^T \end{bmatrix}$. Let $\bar{P}_1 = (I - 2\mathbf{v}\mathbf{v}^T) \in \mathbb{R}^{n-1}$. Then we see

$$\begin{aligned}(I - 2\mathbf{v}\mathbf{v}^T)A^{(1)}(I - 2\mathbf{v}\mathbf{v}^T) &= A^{(0)} - 2\mathbf{v}(\mathbf{v}^T A^{(0)}) - 2(A^{(0)}\mathbf{v})\mathbf{v}^T + 4\mathbf{v}[(\mathbf{v}^T A^{(0)})\mathbf{v}]\mathbf{v}^T \\ &= A^{(0)} - \mathbf{v}\mathbf{p}^T - \mathbf{p}\mathbf{v}^T + 2(\mathbf{p}^T\mathbf{v})\mathbf{v}\mathbf{v}^T \\ &= A^{(0)} - \mathbf{v}(\mathbf{p} - (\mathbf{p}^T\mathbf{v})\mathbf{v})^T - (\mathbf{p} - (\mathbf{p}^T\mathbf{v})\mathbf{v})\mathbf{v}^T \\ &= A^{(0)} - \mathbf{v}\mathbf{w}^T - \mathbf{w}\mathbf{v}^T \ (\mathbf{w} = \mathbf{p} - (\mathbf{p}^T\mathbf{v})\mathbf{v}).\end{aligned}$$

For $i = 1 : n - 2$

(1) $\mathbf{v} = A^{(i-1)}\mathbf{e}_1 \in \mathbb{R}^{n-i}$ (first column of $A^{(i-1)}$ from $i+1$-th entry to n)

(2) $\mu = \mathbf{v}^T\mathbf{v}$

(3) $\mathbf{v}[1] = \mathbf{v}[1] + sign(\mathbf{v}(1))\sqrt{\mu}$ $(\mathbf{v} - \alpha\mathbf{e}_1,\ \alpha = -\text{ sgn } v_1\|\mathbf{v}\|)$

(4) $\mathbf{p} = \frac{2}{\mathbf{v}^T\mathbf{v}}A^{(i-1)}\mathbf{v}$ and $\mathbf{w} = \mathbf{p} - \frac{\mathbf{p}^T\mathbf{v}}{\mathbf{v}^T\mathbf{v}}\mathbf{v}$

(5) $\bar{P}_k^T A^{(i-1)}\bar{P}_k = A^{(i-1)} - \mathbf{v}\mathbf{w}^T - \mathbf{w}\mathbf{v}^T$

Note

$$\begin{aligned}\phi^2 &:= \|\mathbf{v} - \alpha\mathbf{e}_1\|^2 = \|\mathbf{v}\|^2 - 2\alpha\mathbf{v}[1] + \alpha^2 \\ &= 2\mu + 2|\mathbf{v}[1]|\sqrt{\mu} \\ &= 2(\mu + |\mathbf{v}[1]|\sqrt{\mu}).\end{aligned}$$

Using this we normalize $\mathbf{v}$ in 3rd step. Then next step is simpler.

For $i = 1 : n - 2$

(1) $\mathbf{v} = A^{(i-1)}\mathbf{e}_1 \in \mathbb{R}^{n-i}$ (first column of $A^{(i-1)}$ from $i+1$-th entry to n)

(2) $\mu = \mathbf{v}^T\mathbf{v}$

(3) $\mathbf{v}[1] = \mathbf{v}[1] + sign(\mathbf{v}(1))\sqrt{\mu}$ $(\mathbf{v} - \alpha\mathbf{e}_1,\ \alpha = -\text{ sgn } v_1\|\mathbf{v}\|)$

(4) $\phi = \sqrt{2(\mu + |\mathbf{v}[1]|\sqrt{\mu})}$

(5) Set $\mathbf{v} = \mathbf{v}/\phi$

(6) $\mathbf{p} = 2A^{(i-1)}\mathbf{v}/\phi$ and $\mathbf{w} = \mathbf{p} - (\mathbf{p}^T\mathbf{v})\mathbf{v}$

(7) Then $\bar{P}_i^T A^{(i-1)}\bar{P}_i = A^{(i-1)} - \mathbf{v}\mathbf{w}^T - \mathbf{w}\mathbf{v}^T$ is the $(n-i)\times(n-i)$ submatrix of $A^{(i)}$.

6.5 Eigenvalues of symmetric tridiagonal matrix(Givens)

As we have seen in the previous section, we can use the similarity transform by Householder to reduce A to a U-H form (tridaigonal when A is symmetric). Now we turn to Givens procedure for finding eigenvalues of a symmetric tridiagonal matrix.

$$T = \begin{bmatrix} a_1 & b_1 & 0 & \dots & 0 \\ b_1 & a_2 & \ddots & \ddots & \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & b_{n-1} \\ 0 & & 0 & b_{n-1} & a_n \end{bmatrix} \qquad P_n(\lambda) = \det(T - \lambda I)$$

If any one of b_i is 0, we decompose $T = \begin{bmatrix} \times & 0 \\ 0 & \times \end{bmatrix}$. Thus we may assume $b_i \neq 0$ for all i.

$$\text{Let } P_i(\lambda) = \det \begin{bmatrix} a_1 - \lambda & b_1 & 0 & \dots & 0 \\ b_1 & a_2 - \lambda & \ddots & & \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & b_{i-1} \\ 0 & \dots & 0 & b_{i-1} & a_i - \lambda \end{bmatrix} = \det \text{ of principal}$$

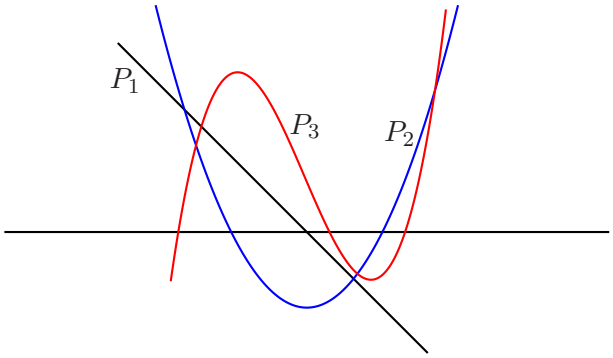
minor.

Theorem 6.5.1. *We have*

$$P_i(\lambda) = (a_i - \lambda)P_{i-1}(\lambda) - b_{i-1}^2 P_{i-2}(\lambda),$$

where we define $P_{-1}(\lambda) = 0, P_0(\lambda) = 1, b_0 = 0 (or P_1(\lambda) = a_1 - \lambda)$. *For derivatives we have(Wilkinson 423 or T.Y. Li paper)*

$$P_i'(\lambda) = (a_i - \lambda)P_{i-1}'(\lambda) - P_{i-1}(\lambda) - b_{i-1}^2 P_{i-2}'(\lambda), i = 2, 3, \cdots$$



Separation of roots of P_i

where we define $P_0'(\lambda) = 0, P_1'(\lambda) = -1$.

$$P_i''(\lambda) = (a_i - \lambda)P_{i-1}''(\lambda) - 2P_{i-1}'(\lambda) - b_{i-1}^2 P_{i-2}''(\lambda), i = 2, 3, \cdots$$

where we define $P_0''(\lambda) = 0, P_1''(\lambda) = 0$. *But direct use of these relation suffer from severe under-overflow.*

The Laguerre Iteration in Solving the Symmetric Tridiagonal Eigenproblem, Revisited, T. Y. Li and Zhonggang Zeng, SIAM J. Sci. Comput., 15(5), 1145-1173.

Proof. Expand and use induction. □

We note that the eigenvalues of each principal submatrix of T are real.

Definition 6.5.2. Let $A(\lambda)$ be the number of sign changes between consecutive members of the sequence.

Theorem 6.5.3 (Givens-The Sturm sequence property)**.** *The roots of* $P_i(\lambda)$ *are distinct (because* $b_i \neq 0$*) for each* i *and are separated by the roots of* $P_{i-1}(\lambda)$. *Furthermore,* $A(\lambda)$*, the number of sign changes between consecutive members of the following sequence is equal to the number of eigenvalues strictly less than* λ.

$$\{P_0(\lambda), P_1(\lambda), \cdots, P_n(\lambda)\}.$$

Proof. First we show $P_{i-1}(\lambda)$ and $P_{i-2}(\lambda)$ cannot have common roots. If $P_{i-2}(\lambda^*) = P_{i-1}(\lambda^*) = 0$, then $P_{i-3}(\lambda^*) = 0 \cdots \Rightarrow P_1(\lambda^*) = a_1 - \lambda^* = 0$. But then $P_2(\lambda^*) = 0 = (a_1 - \lambda^*)(a_2 - \lambda^*) - b_1^2 = -b_1^2 < 0$, contradiction.

Now separation property: Let us use an induction on i, P_i.

$i = 1$:

$$\begin{aligned} P_1(\lambda) &= a_1 - \lambda \\ P_2(\lambda) &= (a_2 - \lambda)(a_1 - \lambda) - b_1^2. \end{aligned}$$

Therefore, the roots of P_2 are clearly separated by the root of P_1. So the roots of P_2 are distinct.

Assume the roots of $P_{i-2}(\lambda)$ separates the roots of $P_{i-1}(\lambda)$. Let $r_1 < r_2 < \cdots < r_{i-1}$ be the roots of $P_{i-1}(\lambda)$. Then from recurrence formula,

$$P_i(r_k) = (a_i - r_k)P_{i-1}(r_k) - b_{i-1}^2 P_{i-2}(r_k) = -b_{i-1}^2 P_{i-2}(r_k), \qquad 1 \le k \le i-1. \tag{6.13}$$

Thus $P_i(r_k)$ and $P_{i-2}(r_k)$ have opposite sign for all k. But $P_{i-2}(x)$ changes sign between r_k and r_{k+1} for $k = 1, 2, \cdots, i-2$. Hence $P_i(\lambda)$ also changes sign because it has opposite signs. In other words, $P_i(\lambda) = 0$ has a root between each pair of adjacent roots of $P_{i-1}(\lambda) = 0$.

We have to find two more roots outside the interval (r_1, r_{i-1}). We see from (6.13), it holds that $P_i(r_1) \cdot P_{i-2}(r_1) < 0$ but $P_i(\lambda)P_{i-2}(\lambda) \to +\infty$ as $\lambda \to -\infty$ as a polynomial of even degree. Hence $P_i(\lambda)P_{i-2}(\lambda)$(hence $P_i(\lambda)$) must change sign on $(-\infty, r_1)$. By the same reason $P_i(\lambda)$ must change sign on (r_{i-1}, ∞). □

Computing eigenvalues by solving $P_n(\lambda) = 0$

In general, finding all the the zeros of a polynomial is not easy. But if we know the approximate interval where the roots belong, we can find them relatively easily. One way of obtaining such interval is as follows. Since $\rho(T) \le \|T\|_\beta$ for any norm, all eigenvalues lie in the interval $[-\|T\|_\infty, \|T\|_\infty]$. One can use bisection method,(since all roots are distinct) to narrow down the interval in which the eigenvalue we seek belong to. In this algorithm, we may find *one eigenvalue at a time*. Another way is to use G-disk theorem, which tells us that eigenvalues lies in the interval

$$[a_1 - |b_1|, a_1 + |b_1|]$$

$$[a_i - |b_{i-1}| - |b_i|, a_i + |b_{i-1}| + |b_i|], i = 2, 3, \cdots, n-1.$$

$$[a_n - |b_{n-1}|, a_n + |b_{n-1}|].$$

For eigenvectors it seems best to use inverse power method. But Jacobi algorithm is good in the sense that it produces all eigenvalues and eigenvectors together.

Remark 6.5.4. A remark from 'An Accelerated Bisection Method for the Calculation of Eigenvalues of a Symmetric Tridiagonal Matrix Herbert J. Bernstein, Numer. Math. 43, 153-160 (1984) Numerische.' .... Indeed, many obvious techniques for converging to eigenvalues faster than with bisection

turn out to take fewer steps, but considerably more time, because so much more has to be done in the inner loop. Further, when calculations to near the limiting accuracy of the machine used are required, bisection has the distinct advantage of avoiding convergence questions. We can retain the guaranteed convergence of bisection.

.....

6.6 Eigenval of symm. tridiag by bisection-Wilkinson

6.6.1 The Sturm sequence property

$$T = \begin{bmatrix} a_1 & b_1 & 0 & \dots & 0 \\ b_1 & a_2 & \ddots & \ddots & \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & b_{n-1} \\ 0 & & 0 & b_{n-1} & a_n \end{bmatrix} \qquad f_n(\lambda) = \det(T - \lambda I)$$

Then the Sturm sequence is

$$\begin{aligned} f_0(\lambda) &= 1, \\ f_1(\lambda) &= a_1 - \lambda \\ f_i(\lambda) &= (a_i - \lambda) f_{i-1}(\lambda) - b_{i-1}^2 f_{i-2}(\lambda) \quad (3) \end{aligned}$$

Associate a sign with each of the $f_i(\lambda)$ as follows. If $f_i(\lambda) \neq 0$, then this sign is the true sign of $f_i(\lambda)$. If $f_i(\lambda) = 0$ we take its sign to be the opposite of that of $f_{i-1}(\lambda)$(a convention).

With this interpretation the number, recall the result above:

Theorem 6.6.1 (Givens)**.** *The roots of $f_i(\lambda)$ are separated by the roots of $f_{i-1}(\lambda)$. Furthermore, $A(\lambda)$, the number of sign changes between consecutive members of the sequence (3) is equal to the number of eigenvalues strictly less than λ.*

The condition $b_i = 0$ considerably simplifies the statement of the theorem. It ensures that there are no true multiple eigenvalues and that no two consecutive members of the sequence are both zero.

The theorem may be applied to unsymmetric tridiagonal matrices with upper off-diagonal elements a_i and lower off-diagonal elements b_i provided $a_i b_i > 0$ for all i. We merely replace b_{i-1}^2 in (3) by $a_{i-1} b_{i-1}$.

6.6.2 The bisection process-Wilkinson original method

J. H. Wilkinson, Numer Math 4, 362–367 (1962) (In practice, we use the stabilized version developed by Barth wilkin 1967 see below.) Suppose the eigenvalues are ordered so that $\lambda_1 > \lambda_2 > \cdots > \lambda_n$. (Equality is excluded since we are assuming $b_i \neq 0$) and it is known that for some g_0 and h_0

$$g_0 \geq \lambda_r > h_0, \ \ (4) \tag{6.14}$$

so that

$$A(h_0) \geq r; \quad A(g_0) < r \ \ (5). \tag{6.15}$$

Then in t bisection steps we can locate λ_i in an interval (h_i, g_i) of width $(g_0 - h_0)/2^t$ as follows. Suppose in i steps we have established that

$$A(h_i) \geq r; \qquad A(g_i) < r, \tag{6.16}$$

$$g_i - h_i \ = \ (g_0 - h_0)/2^i. \tag{6.17}$$

In the $i+1$-th step we compute $A[\frac{1}{2}(g_i + h_i)]$. Then:

$$\text{if } A[\frac{1}{2}(g_i + h_i)] \geq r; \qquad \text{we take } h_{i+1} = \frac{1}{2}(g_i + h_i); \ g_{i+1} = g_i, (8) \tag{6.18}$$

$$\text{if } A[\frac{1}{2}(g_i + h_i)] < r; \qquad \text{we take } g_{i+1} = \frac{1}{2}(g_i + h_i); \ h_{i+1} = h_i. (9) \tag{6.19}$$

In either ease we have

$$(g_{i+1} - h_{i+1}) \ = \ \frac{1}{2}(g_i - h_i) \ \ (10) \tag{6.20}$$

$$A(h_{i+1}) \geq r; \qquad A(g_{i+1}) < r \ \ (11) \tag{6.21}$$

so that λ_r is in the interval (h_{i+1}, g_{i+1}).

Values g_0 and h_0 satisfying (4) and (5) are given by

$$h_0 = -norm; \quad g_0 = norm \ \ (12), \tag{6.22}$$

where norm is the infinity norm of the tridiagonal matrix.

Note that if a general matrix is reduced to tridiagonal form by orthogonal similarity transformations, then for exact computation multiple eigenvalues imply that some off-diagonal elements must be zero.

6.6.3 A modified method to avoid under/overflow

"Calculation of the Eigenvalues of a Symmetric Tridiagonal Matrix by the Method of Bisection" W. Barth, R. S. Martin and J. H. Wilkinson, Numer Math 9, 386– 393 (1967) This algorithm seems best for single process and

the algol code included there. For a slight improved version see "An Accelerated Bisection Method for the Calculation of Eigenvalues of a Symmetric Tridiagonal Matrix", Herbert J. Bernstein, Numer. Math. 43, 153-160 (1984).

Let (see the index change from above $b_1 => b_2$)

$$T = \begin{bmatrix} c_1 & b_2 & 0 & \dots & 0 \\ b_2 & c_2 & \ddots & \ddots & \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & b_n \\ 0 & & 0 & b_n & c_n \end{bmatrix} \qquad f_n(\lambda) = \det(T - \lambda I)$$

Givens' method is numerically very stable. The error in the eigenvalues is no worse than 15 times the truncation error 2^t, for t-bit mantissa binary machines, independent of the order of T. However, the direct calculation of the sequence $f(v)$ can easily lead to underflow or overflow. Barth, Martin and Wilkinson [1] avoid the need to rescale by using the sequence

$$g_i(v) = f_i(v)/f_{i-1}(v), i = 1, ..., n \tag{6.23}$$

alternatively given as

$$g_1(v) = c_1 - v, \tag{6.24}$$

$$g_i(v) = (c_i - v) - b_i^2/g_{i-1}(v), i = 2, ..., n, \tag{6.25}$$

with zeros replaced by small non-zero quantities. Counting positive g_i is the same as counting sign agreements of f_i, which yields the count of the number of eigenvalues of $A \geq v$. Equivalently, counting negative g_i is the same as counting sign disagreements of f_i, which yields the count of eigenvalues of $A < v$.

find eigenvalues $\lambda_{m1}, \cdots, \lambda_{m2} (\lambda_{i+1} \geq \lambda_i)$ a symm. tridiag matrix of order n.

Input to procedure **bisect**

c	an $n \times 1$ array giving the diagonal elements of a tridiagonal matrix.
b	an $n \times 1$ array giving the subdiagonal elements of a tridiagonal matrix.
β	an $n \times 1$ array giving $\beta[i] = b^2[i]$
$m_1 \leq m_2$	the eigenvalues $\lambda_{m1}, \cdots, \lambda_{m2}$ are calculated
ϵ_1	a quantity affecting the precision to which the eigenvalues are computed.
$relfeh$	the smallest number for which $1 + relfeh > 1$ on the computer.

Output of procedure bisect

ϵ_2	gives information concerning the accuracy of results
z	total number of bisections to find all required eigenvalues.
x	array $x[m_1 : m_2]$ contains the computed eigenvalues.

procedure bisect (c, b, etc)

```
begin  real h, xmin, xmax;              int i;
       β[1] := b[1] = 0;
          xmin := c[n] − abs(b[n]);     xmax := c[n] + abs(b[n]);
       for i := n − 1 to 1
       begin h := abs(b[i])             +abs(b[i + 1]);
          if c[i] + h > xmax then       xmax := c[i] + h;
          if c[i] − h < xmin then       xmin := c[i] − h;
       end i;
       ε2 := rel × temp          (if xmin + xmax > 0 then temp = xmax else − xmin);
       if ε1 ≤ 0 then ε1 = ε2;
       ε2 := 0.5ε1 + 7ε2
       innerblock                       compute λm1, ··· , λm2
       begin int a, k; real q, x1, xu,  xo; array wu[m1 : m2];
             xo := xmax;
             for i := m1 to m2
             begin x[i] := xmax;        wu[i] := xmin; end i;
             z := 0;
             for                        k := m2 to m1 (k-th eigen)
             begin                      xu := xmin;
                                        for i := k to m1
                                        begin if xu < wu[i] then
                                              begin xu := wu[i]; go to contin
                                              end
                                        end i;
               contin: if xo > x[k] then xo := x[k]; (interval (xu, xo) chosen)
                       for x1 := (xu + xo)/2(initializa) while xo − xu >
                                    2 × rel × (abs(xu) + abs(xo)) + ε1 do
                       begin z := z + 1; ( #of bisections needed )
                                        a := 0; q := 1;
                                        for i := 1 → n do (Sturm sequence)
                                        begin
                                          q := c[i] − x1 − (if q ≠ 0 then β[i]/q
                                                           else abs(b[i])/rel);
                                          if q < 0 then a := a + 1; (#of eig less than x1)
                                        end i;
                                        if a < k then
                                        begin if a < m1 then
                                              xu := wu[m1] := x1;
                                              else
                                              begin xu := wu[a + 1] := x1;
                                                    if x[a] > x1 then x[a] := x1;
                                              end
                                        end
                                        else xo := x1;
                       end x1;
                       x[k] := (xo + xu)/2; (k-th eigvalue found)
             end k;
       end inner block
end bisect;
```

Remark 6.6.2. Other way of finding eigenvalues of tri diagonal matrix — QR algorithm with origin shift to find eigenvalues of tridiagonal matrix. Stewart[1970] applied this method for symmetric matrices after Householder's algorithm to tridiagonalize the matrix.

Remark 6.6.3. Jacobi Method works only for symmetric matrix. $R_m^T A R_m \to \text{diag}(\lambda_1, \ldots, \lambda_n)$, $R_m = \prod_1^m R_i$, R^i are orthogonal matrix. This produces eigenvalues altogether and eigenvectors are displayed as columns of R. Givens-Householder works for non symmetric case, but used only for symmetric case. In general, QR method (below) is preferred after reducing the matrix to U-H form to save computational times.

Eigenvectors

For symmetric matrices, Givens-Householder algorithm produces eigenvectors. We have

$$A = PTP^T,$$

where T is tridiagonal. If λ is an eigenvalue of T corresponding to eigenvector $\mathbf{y}$(relatively easy), then $T\mathbf{y} = \lambda\mathbf{y}$. Hence $\lambda(P\mathbf{y}) = PT\mathbf{y} = PTP^T(P\mathbf{y}) = A(P\mathbf{y})$. Thus λ is an eigenvalue of A with corresponding eigenvector $\mathbf{x} = P\mathbf{y}$. Then use power or inverse power method to find eigenvectors of T.

Non Hermitian case can be handled similarly, except that we have for some unitary matrix U and an upper Hessenberg matrix H,

$$A = UHU^H.$$

If λ is an eigenvalue of H corresponding to eigenvector $\mathbf{y}$, then $H\mathbf{y} = \lambda\mathbf{y}$, hence $\lambda(U\mathbf{y}) = UH\mathbf{y} = UHU^H(U\mathbf{y}) = A(U\mathbf{y})$. Thus λ is an eigenvalue of A with corresponding eigenvector $\mathbf{x} = U\mathbf{y}$.

Recall that in the QR -iteration, when the sequence $\{A_k\}$ converges to a diagonal matrix, the product $Q_0Q_1Q_2\cdots Q_k$ converges to a matrix whose columns are eigenvectors. (costly!)

However, if eigenvalues are known, finding the corresponding eigenvector $\mathbf{x}$ is relatively easy, for example by (inverse) power method to be described below.

6.7 Gram-Schmidt Orthogonalization and the QR Decomposition

6.7.1 Classical Gram-Schmidt

Given k linearly independent vectors $\{\mathbf{v}_1, \cdots, \mathbf{v}_k\}$, we consider the span $V = span\{\mathbf{v}_1, \cdots, \mathbf{v}_k\}$ and we would like to find an orthonormal basis for V. The

usual Gram-Schmidt process is to find $\mathbf{u}_1, \cdots, \mathbf{u}_k$ as follows:

$$\begin{array}{rcll} \mathbf{u}_1 &=& \frac{\mathbf{v}_1}{\|\mathbf{v}_1\|} & \\ \mathbf{u}_2' &=& \mathbf{v}_2 - (\mathbf{v}_2, \mathbf{u}_1)\mathbf{u}_1, & \mathbf{u}_2 = \frac{\mathbf{u}_2'}{\|\mathbf{u}_2'\|} \\ \mathbf{u}_3' &=& \mathbf{v}_3 - (\mathbf{v}_3, \mathbf{u}_2)\mathbf{u}_2 - (\mathbf{v}_3, \mathbf{u}_1)\mathbf{u}_1, & \mathbf{u}_3 = \frac{\mathbf{u}_3'}{\|\mathbf{u}_3'\|} \\ &=& \cdots & \\ \mathbf{u}_k' &=& \mathbf{v}_k - \sum_{j=1}^{k-1}(\mathbf{v}_k, \mathbf{u}_j)\mathbf{u}_j, & \mathbf{u}_k = \frac{\mathbf{u}_k'}{\|\mathbf{u}_k'\|}. \end{array} \tag{6.26}$$

Suppose A is an $m \times k$ matrix with full column rank(this requires $m \geq k$) whose successive column vectors are $\{\mathbf{w}_1, \cdots, \mathbf{w}_k\}$. If Gram-Schmidt process is applied to these vectors to produce an orthonormal basis $\{\mathbf{q}_1, \cdots, \mathbf{q}_k\}$ for the column space of A, and Q is the matrix whose column vectors are $\{\mathbf{q}_1, \cdots, \mathbf{q}_k\}$ in order, what is the relationship between A and Q ?

Let A and Q be the matrices having $\mathbf{w}_i$ and $\mathbf{q}_i$ as columns, i.e.,

$$A = [\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_k], \quad Q = [\mathbf{q}_1, \mathbf{q}_2, \cdots, \mathbf{q}_k].$$

We can express the vector $\mathbf{w}_i$ in terms of orthonormal column vectors of Q as

$$\mathbf{w}_i = \sum_{j=1}^{k} c_{ij}\mathbf{q}_j.$$

By orthonormal property of $\mathbf{q}_j$'s, we see $c_{ij} = \mathbf{w}_i \cdot \mathbf{q}_j$ and hence

$$\begin{array}{rcl} \mathbf{w}_1 &=& (\mathbf{w}_1 \cdot \mathbf{q}_1)\mathbf{q}_1 + (\mathbf{w}_1 \cdot \mathbf{q}_2)\mathbf{q}_2 + \cdots + (\mathbf{w}_1 \cdot \mathbf{q}_k)\mathbf{q}_k \\ \mathbf{w}_2 &=& (\mathbf{w}_2 \cdot \mathbf{q}_1)\mathbf{q}_1 + (\mathbf{w}_2 \cdot \mathbf{q}_2)\mathbf{q}_2 + \cdots + (\mathbf{w}_2 \cdot \mathbf{q}_k)\mathbf{q}_k \\ &=& \cdots \\ \mathbf{w}_k &=& (\mathbf{w}_k \cdot \mathbf{q}_1)\mathbf{q}_1 + (\mathbf{w}_k \cdot \mathbf{q}_2)\mathbf{q}_2 + \cdots + (\mathbf{w}_k \cdot \mathbf{q}_k)\mathbf{q}_k. \end{array}$$

Note $\mathbf{q}_j$ is orthogonal to $\mathbf{w}_i$ when $i < j$. Hence we have

$$\begin{array}{rcl} \mathbf{w}_1 &=& (\mathbf{w}_1 \cdot \mathbf{q}_1)\mathbf{q}_1 \\ \mathbf{w}_2 &=& (\mathbf{w}_2 \cdot \mathbf{q}_1)\mathbf{q}_1 + (\mathbf{w}_2 \cdot \mathbf{q}_2)\mathbf{q}_2 \\ &=& \cdots \\ \mathbf{w}_k &=& (\mathbf{w}_k \cdot \mathbf{q}_1)\mathbf{q}_1 + (\mathbf{w}_k \cdot \mathbf{q}_2)\mathbf{q}_2 + \cdots + (\mathbf{w}_k \cdot \mathbf{q}_k)\mathbf{q}_k. \end{array}$$

Let us form the upper triangular matrix

$$R = \begin{bmatrix} (\mathbf{w}_1 \cdot \mathbf{q}_1) & (\mathbf{w}_2 \cdot \mathbf{q}_1) & \cdots & (\mathbf{w}_k \cdot \mathbf{q}_1) \\ 0 & (\mathbf{w}_2 \cdot \mathbf{q}_2) & \cdots & (\mathbf{w}_k \cdot \mathbf{q}_2) \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & (\mathbf{w}_k \cdot \mathbf{q}_k) \end{bmatrix} \tag{6.27}$$

Then we can see that $AQ = R$, i.e.,

$$\begin{array}{ccccc} [\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_k] & = & [\mathbf{q}_1, \mathbf{q}_2, \cdots, \mathbf{q}_k] & \begin{bmatrix} (\mathbf{w}_1 \cdot \mathbf{q}_1) & (\mathbf{w}_2 \cdot \mathbf{q}_1) & \cdots & (\mathbf{w}_k \cdot \mathbf{q}_1) \\ 0 & (\mathbf{w}_2 \cdot \mathbf{q}_2) & \cdots & (\mathbf{w}_k \cdot \mathbf{q}_2) \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & (\mathbf{w}_k \cdot \mathbf{q}_k) \end{bmatrix} \\ A & = & Q & R. \end{array} \tag{6.28}$$

Theorem 6.7.1. *If A is an $m \times n (m \geq n)$ matrix with full column rank, then A can be factored as*

$$A = QR, \tag{6.29}$$

where Q is $m \times n$ matrix whose column vectors form an orthonormal basis for the column space of A, and R is a $n \times n$ invertible upper triangular matrix.

In general a matrix factorization of the form $A = QR$, where column vectors of Q are orthonormal and R is invertible, upper triangular is called a **QR-decomposition**.

CGS-**Algorithm**: Given $\mathbf{v}_1, \cdots, \mathbf{v}_n \in \mathbb{R}^m$
for $k = 1, 2, \ldots, n$
  for $i = 1, 2, \ldots, k-1$
    $r_{ik} = \mathbf{q}_i^T \mathbf{v}_k$,
    $\mathbf{v}_k \leftarrow \mathbf{v}_k - r_{ik}\mathbf{q}_i$
  end
  $r_{kk} = \|\mathbf{v}_k\|$,
  $\mathbf{q}_k = \frac{\mathbf{v}_k}{r_{kk}}$
end k-loop

Here $Q = (q_{ij})$ and $R = (r_{ij})$.

Example 6.7.2. Find a QR-decomposition of the following matrix using the Gram-Schmidt process.

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

sol.

$$\mathbf{w}_1 = [1,0,1]^T,\ \mathbf{w}_2 = [-1,1,1]^T,\ \mathbf{w}_3 = [0,1,1]^T$$

$$\begin{aligned}
\mathbf{v}_1 &= \mathbf{w}_1 = [1,0,1]^T \\
\mathbf{v}_2 &= \mathbf{w}_2 - \frac{\mathbf{w}_2 \cdot \mathbf{v}_1}{\|\mathbf{v}_1\|^2}\mathbf{v}_1 = [-1,1,1]^T - 0 \\
\mathbf{v}_3 &= \mathbf{w}_3 - \frac{\mathbf{w}_3 \cdot \mathbf{v}_1}{\|\mathbf{v}_1\|^2}\mathbf{v}_1 - \frac{\mathbf{w}_3 \cdot \mathbf{v}_2}{\|\mathbf{v}_2\|^2}\mathbf{v}_2 = [0,1,1]^T - \frac{1}{2}\cdot[1,0,1]^T - \frac{2}{3}[-1,1,1]^T \\
&= [\frac{1}{6}, \frac{1}{3}, -\frac{1}{6}]^T
\end{aligned}$$

$$\mathbf{q}_1 = \frac{1}{\sqrt{2}}[1,0,1]^T,\ \mathbf{q}_2 = \frac{1}{\sqrt{3}}[-1,1,1]^T,\ \mathbf{q}_3 = \frac{1}{\sqrt{6}}[1,2,-1]^T$$

$$R = \begin{bmatrix} (\mathbf{w}_1 \cdot \mathbf{q}_1) & (\mathbf{w}_2 \cdot \mathbf{q}_1) & (\mathbf{w}_3 \cdot \mathbf{q}_1) \\ 0 & (\mathbf{w}_2 \cdot \mathbf{q}_2) & (\mathbf{w}_3 \cdot \mathbf{q}_2) \\ 0 & 0 & (\mathbf{w}_3 \cdot \mathbf{q}_3) \end{bmatrix} = \begin{bmatrix} \sqrt{2} & 0 & \frac{1}{\sqrt{2}} \\ 0 & \sqrt{3} & \frac{2}{\sqrt{3}} \\ 0 & 0 & \frac{1}{\sqrt{6}} \end{bmatrix}$$

□

6.7.2 Other Ways to Obtain QR-decomposition

We study (Modified) Gram-Schmidt, Givens Rotation and Householder Reflection here.

$$R = \begin{bmatrix} \downarrow & \downarrow & \cdots & \downarrow \\ 0 & \downarrow & \cdots & \downarrow \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \downarrow \end{bmatrix}, \qquad R = \begin{bmatrix} \rightarrow & \rightarrow & \cdots & \rightarrow \\ 0 & \rightarrow & \cdots & \rightarrow \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \rightarrow \end{bmatrix}$$

CGS vs. MGS

It is known that the classical Gram-Schmidt QR-decomposition (6.27) produces large round off error numerically. Hence it is not recommended to use in numerical purpose. One remedy is to rearrange the order of orthogonalization. Note that the order of generating R is columnwise in (6.27). If we change the order row wise, we get **Modified Gram-Schmidt**. Another method is to use Householder transformation. Still another method is to use the Givens rotation.

Modified Gram-Schmidt

Unfortunately, this is a numerically bad procedure: Large round off errors and hence loss of orthogonality of Q. For better numerical behavior, use the

following modified Gram-Schmidt. The idea is to orthogonalize all vectors w.r.t orthogonal vector $\mathbf{q}_i$ as soon as they are available.

Given linearly independent vectors $\{\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_n\}$,

To save space A is overwritten by Q-orthogonal vectors.

Modified-GS Algorithm -vector version: Given $\mathbf{v}_1, \cdots, \mathbf{v}_n \in \mathbb{R}^m$
for $k = 1, 2, \ldots, n$

$r_{kk} = \|\mathbf{v}_k\|$

$\mathbf{q}_k = \frac{\mathbf{v}_k}{r_{kk}}$ (normalize k-th vector $\mathbf{v}_k$)

for $j = k+1, \ldots, n$ (row wise)

$r_{kj} = \mathbf{q}_k^T \cdot \mathbf{v}_j$

$\mathbf{v}_j \leftarrow \mathbf{v}_j - < \mathbf{q}_k, \mathbf{v}_j > \mathbf{q}_k$

end j-loop

end k-loop

Here $Q = \{q_{ij}\}$ and $R = \{r_{ij}\}$.

QR by Householder reflection

Recall that one way of expressing the Gaussian elimination algorithm is in terms of Gauss transformations that serve to introduce zeros into the lower triangle of a matrix. Householder transformations are *orthogonal* transformations (refections) that can have similar effect. Reflection across the plane orthogonal to a unit normal vector $\mathbf{w}$(**Householder reflection, elementary reflector**) can be expressed in a matrix form as

$$H_{\mathbf{w}} = I - 2\mathbf{w}\mathbf{w}^* = I - 2proj_{\mathbf{w}}.$$

This is symmetric and orthogonal($H_{\mathbf{w}}^* = H_{\mathbf{w}}$ and $H_{\mathbf{w}}^* H_{\mathbf{w}} = I$). The idea is :

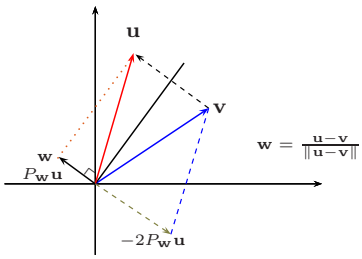


Figure 6.1: Action of an elementary reflector $H_{\mathbf{w}}$

Given $\mathbf{u}$, $\mathbf{v}$ with $\|\mathbf{u}\| = \|\mathbf{v}\|$, find an orthogonal matrix $H_{\mathbf{w}}$ so that $H_{\mathbf{w}}\mathbf{u} = \mathbf{v}$. Thus from the figure 6.7.2 we can see

$$\mathbf{w} = \frac{\mathbf{u} - \mathbf{v}}{\|\mathbf{u} - \mathbf{v}\|}. \tag{6.30}$$

By multiplying a sequence of Householder transformations we turn all the subdiagonal elements into zeros. This leads us to the following algorithm to compute the QR decomposition: For simplicity assume $m = n$ here. But the QR transform is valid for the case $n \leq m$ also. Now the first step of QR is to reduce the first column to a multiple of $\mathbf{e}_1$: Let $A = [\mathbf{a}_1, A_2]$. Then we can find an orthogonal matrix $Q_1 = H_{\mathbf{w}}$, which maps the first column of A onto $\alpha\mathbf{e}_1$. Since $\|H_{\mathbf{w}}\mathbf{a}_1\| = \|\mathbf{a}_1\| = \|\alpha\mathbf{e}_1\|$, we have $\alpha = \pm\|\mathbf{a}_1\|$. Thus $\mathbf{w}$ is given by

$$\mathbf{w} = \frac{\mathbf{a}_1 - \alpha\mathbf{e}_1}{\|\mathbf{a}_1 - \alpha\mathbf{e}_1\|}. \tag{6.31}$$

and

$$Q_1 A = \left[\begin{array}{c|ccc} \alpha & b_2 & \dots & b_n \\ \hline 0 & & & \\ \vdots & & A_{12} & \\ 0 & & & \end{array}\right].$$

We repeat the same process. If $(n-1)\times(n-1)$ matrix Q_2' is the elementary reflector used to reduce the first column of A_{12}, then the matrix

$$Q_2 = \begin{pmatrix} 1 & 0 \\ 0 & Q_2' \end{pmatrix}$$

makes

$$Q_2 Q_1 A = \left[\begin{array}{c|ccc} \alpha & b_2 & \dots & b_n \\ 0 & \alpha_2 & * & * \\ \hline \vdots & 0 & A_{23}' & \\ 0 & 0 & & \end{array}\right].$$

How to choose the sign of α? The idea is to let we have less round off error (i.e, to induce less subtractive cancelation). Thus we choose the sign as $-$ sgn a_{11}.

Continuing this process, we arrive at

$$Q_n \cdots Q_2 Q_1 A = R(\text{ upper triangular}).$$

Hence the QR is obtained as:

$$A = R Q_n \cdots Q_2 Q_1(\text{ upper triangular}).$$

In fact, we can use QR transform to a $m \times n (m \geq n)$ matrix A. The following algorithm is from Heath book.

QR transform by Householder : Given $\mathbf{a}_1, \cdots, \mathbf{a}_n \in \mathbb{R}^m$
for $k = 1, 2, \dots, \min(n, m-1)$
$\alpha_k = -sgn(a_{kk})\sqrt{a_{kk}^2 + \cdots + a_{mk}^2}$
$\mathbf{w}_k = [0, \cdots, 0, a_{kk}, \cdots, a_{mk}]^T - \alpha_k \mathbf{e}_k$ (compute $\alpha\mathbf{e}_1 - \mathbf{a}_1$)

$\beta_k = \mathbf{w}_k^T \mathbf{w}_k$ (compute $\|\mathbf{w}\|$)
If $\beta_k = 0$, continue with next k (current column is zero)
else for $j = k$ to n
$\gamma_j = \mathbf{w}_k^T \cdot \mathbf{a}_j$
$\mathbf{a}_j = \mathbf{a}_j - (2\gamma_j/\beta_k)\mathbf{w}_k$ (perform $(I - 2\mathbf{w}\mathbf{w}^T)\mathbf{a}_j$)
end j-loop
end k-loop

We write above decomposition by $Q_1 R_1$. Then we can extend the QR decomposition. Let $Q = [Q_1\ Q_2]$, where Q_1 is the $m \times n$ matrix (first n columns just computed above) and the columns of Q_2 form an orthonormal basis for the orthogonal complement of span(A). Then we have

$$A = QR = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} R_1 & \mathbf{0} \\ \mathbf{0} & R_2 \end{bmatrix} = \begin{bmatrix} Q_1 R_1 & Q_2 R_2 \end{bmatrix}$$

QR by Givens rotation for upper-Hessenberg form

When H is an upper-Hessenberg matrix, the Givens rotation is cheaper than Householder transform to form a QR-factorization.

6.8 Subspace Iterations

6.8.1 Power method - a few extreme eigenvalues

Hypothesis :

(1) A is diagonalizable, i.e, $A\mathbf{v}^{(i)} = \lambda_i \mathbf{v}^{(i)}$ for linearly independent eigenvectors $\mathbf{v}^{(i)}$.

(2) $|\lambda_1| > |\lambda_j|$, $j = 2, \ldots, n$(i.e, λ_1 is a dominant eigenvalue.)

Power method consists of generating sequences of vector converging to an eigenvector and a sequence of scalar converging to dominant eigenvalue λ_1. We assume $\{\mathbf{v}^{(i)}\}_{i=1}^n$ is a normalized sets with respect to some norm $\|\cdot\|$.

Let $\mathbf{y} \in \mathbb{C}^n$ be any vector whose projection against $\{\mathbf{v}^{(1)}\}$ is nonzero. Then one write $\mathbf{y} = \sum_1^n c_i \mathbf{v}^{(i)}$ with $c_1 \neq 0$. With $\mathbf{y}^{(0)} = \mathbf{y}$, set

$$\mathbf{y}^{(k)} = A\mathbf{y}^{(k-1)}, \qquad k = 1, 2, \ldots$$

Then

$$\begin{aligned} \mathbf{y}^{(k)} &= A^k \mathbf{y}^{(0)} = A^k (\sum c_i \mathbf{v}^{(i)}) = \sum c_i \lambda_i^k \mathbf{v}^{(i)} \\ &= \lambda_1^k \left[c_1 \mathbf{v}^{(1)} + \sum_2^n c_i (\frac{\lambda_i}{\lambda_1})^k \mathbf{v}^{(i)} \right] \approx \lambda_1^k c_1 \mathbf{v}^{(1)}, \end{aligned}$$

since $|\lambda_1| > |\lambda_i|$, $i = 2, \ldots, n$. Similarly, $\mathbf{y}^{(k-1)} \approx \lambda_1^{k-1} c_1 \mathbf{v}^{(1)}$. Hence we see

$$\mathbf{y}^{(k)} \approx \lambda_1 \mathbf{y}^{(k-1)}$$

from which we may compute λ_1. Let $\mathbf{w}$ be any vector not orthogonal to $\mathbf{v}^{(1)}$. Then

$$\lambda_1 \approx \frac{(\mathbf{w}, \mathbf{y}^{(k)})}{(\mathbf{w}, \mathbf{y}^{(k-1)})} + O(\frac{\lambda_2}{\lambda_1})^k.$$

But, this process is unstable numerically because it may happen that $\mathbf{y}^{(k)} \to 0$ or ∞. To avoid it, we normalize the vector at each step.

Power method is especially useful when the matrix is sparse.

$\mathbf{x}^{(0)}$ arbitrary
for $k = 1, 2, \cdots, n$ do
  $\mathbf{y}^{(k)} = A\mathbf{x}^{(k-1)}$
  $\mathbf{x}^{(k)} = \frac{\mathbf{y}^k}{\|\mathbf{y}^k\|_2}$
  $\lambda^{(k)} = \frac{(\mathbf{w}, \mathbf{y}^{(k)})}{(\mathbf{w}, \mathbf{y}^{(k-1)})}$.

In practice we use the following variation(Rayleigh quotient): Setting $\mathbf{w} = \mathbf{y}^{(k-1)}$ we get

for $k = 1, 2, \cdots, n$ do
  $\mathbf{y}^{(k)} = A\mathbf{x}^{(k-1)}$
  $\mathbf{x}^{(k)} \leftarrow \frac{\mathbf{y}^{(k)}}{\|\mathbf{y}^{(k)}\|_2}$
  $\lambda^{(k)} = (\mathbf{x}^{(k-1)}, \mathbf{y}^{(k)}) = (\mathbf{x}^{(k-1)}, A\mathbf{x}^{(k-1)})$

One can easily show the error is $O(\frac{\lambda_2}{\lambda_1})^{2k}$ for general matrix and cubic for normal (and symmetric) matrix.

Some problems related with power method

(1) What if $c_1 = 0$?

(2) What if $\lambda_1/\lambda_2 \approx 1$?

(3) What about λ_j for $j \neq 1$?

Roots of same modulus

We assume full linearly independent eigenvectors exist corresponding to the multiple eigenvalues.

The case $|\lambda_1| = |\lambda_2|$

(Faddeev, Faddeeva.) Suppose $|\lambda_1| = |\lambda_2| > |\lambda_3| \cdots$ and $\lambda_2 = \pm\lambda_1$. Then

$$\frac{y_j^{(k+2)}}{y_j^{(k)}} = \frac{\lambda_1^{k+2}[c_1\mathbf{v}^{(1)} + c_2(-1)^{k+2}\mathbf{v}^{(2)} + \varepsilon^{(k+2)}]_j}{\lambda_1^k[c_1\mathbf{v}^{(1)} + c_2(-1)^k\mathbf{v}^{(2)} + \varepsilon^{(k)}]_j} \to \lambda_1^2.$$

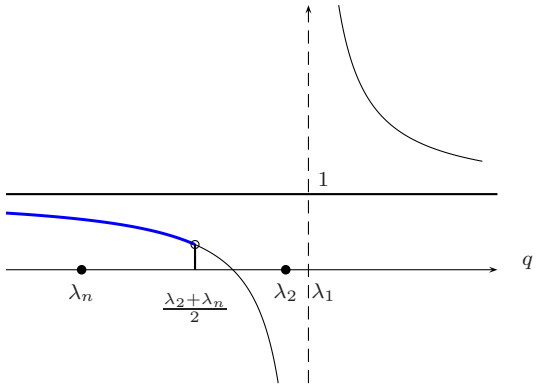


Figure 6.2: Graph of $\frac{\lambda_2-q}{\lambda_1-q} = 1 + \frac{\lambda_2-\lambda_1}{\lambda_1-q}$

If three eigenvalues have same modulus, and $(\frac{\lambda_i}{\lambda_1})^3 = 1$, $i = 1, 2$ then

$$\frac{y_j^{(k+3)}}{y_j^{(k)}} = \frac{\lambda_1^{k+3}[c_1\mathbf{v}^{(1)} + c_2(\frac{\lambda_2}{\lambda_1})^{k+3}\mathbf{v}^{(2)} + c_3(\frac{\lambda_3}{\lambda_1})^{k+3}\mathbf{v}^{(3)} + \varepsilon(k+3)]_j}{\lambda_1^k[c_1\mathbf{v}^{(1)} + c_2(\frac{\lambda_2}{\lambda_1})^k\mathbf{v}^{(2)} + c_3(\frac{\lambda_3}{\lambda_1})^k\mathbf{v}^{(3)} + \varepsilon(k)]_j} \to \lambda_1^3.$$

If $\lambda_1 = 1$, $\lambda_2 = 1$, $\lambda_3 = -1$, try $y_j^{(k+6)}/y_j^{(k)}$.

Speed up of power method-origin shift

Lemma 6.8.1. *If $(\lambda, \mathbf{v})$ is an eigenpair of A, then $(\lambda - q, \mathbf{v})$ is an eigenpair of $A - qI$. If the eigenvalues of A satisfy $0 < \lambda_n \le \lambda_{n-1} \le \cdots \le \lambda_1$, then for any real number q, the dominant eigenvalue of $A - qI$ is $\lambda_1 - q$ or $\lambda_n - q$.*

The convergence ratio of power method depends on the ratio of second largest to largest eigenvalue. Thus, we have the following result.

Theorem 6.8.2. *The value $q = \frac{1}{2}(\lambda_2 + \lambda_n)$ gives the best rate of convergence for $\lambda_1 - q$, while $q = \frac{1}{2}(\lambda_1 + \lambda_{n-1})$ gives the best rate of convergence to $\lambda_n - q$. In general, the idea is to find an optimum q such that*

$$\max_{2\le i\le n} \frac{|\lambda_i - q|}{|\lambda_1 - q|} \text{ or } \max_{1\le i\le n-1} \frac{|\lambda_i - q|}{|\lambda_n - q|}$$

is minimum.

Proof. For the sake of simplicity, we may assume $0 < \lambda_n \le \lambda_{n-1} \le \cdots \le \lambda_1$. It suffices to seek $\lambda_n < q < \lambda_1$. Case II: $|\lambda_n - q| \ge |\lambda_2 - q|$, i.e, $q \ge \frac{\lambda_2+\lambda_n}{2}$. □

6.8.2 Inverse power method

Assume $0 < |\lambda_1| < |\lambda_2| \leq |\lambda_3| \cdots$. Then the power method applied to A^{-1} lead to the computation of the smallest eigenvalue. Start from any $\mathbf{y}^{(0)}$ and define

$$A\mathbf{y}^{(k)} = \mathbf{y}^{(k-1)}, \qquad k = 1, 2, \cdots .$$

Then

$$\begin{aligned} \mathbf{y}^{(k)} &= A^{-k}\mathbf{y}^{(0)} = A^{-k}(\sum c_i \mathbf{v}^{(i)}) = \sum c_i \lambda_i^{-k} \mathbf{v}^{(i)} \\ &= \lambda_1^{-k}\left[c_1\mathbf{v}^{(1)} + \sum_2^n c_i (\frac{\lambda_i}{\lambda_1})^{-k}\mathbf{v}^{(i)}\right] \approx \lambda_1^{-1}\mathbf{y}^{(k-1)}. \end{aligned}$$

In practice, we always normalize $\mathbf{y}^{(k-1)}$ each step, so that if $\|\mathbf{y}^{(k-1)}\|_2 = |y_j^{(k-1)}| = 1$ for some j, then $y_j^{(k)} \approx \frac{1}{\lambda_i} y_j^{(k-1)}$.

The following algorithm is based on Rayleigh Quotient:

Algorithm: Basic Inverse power method

For $k = 1, 2, \cdots$, do

Solve $A\mathbf{y}^{(k)} = \mathbf{x}^{(k-1)}$

$\mathbf{x}^{(k)} \leftarrow \frac{\mathbf{y}^{(k)}}{\|\mathbf{y}^{(k)}\|_2}$

$\lambda^{(k)} = \frac{1}{(\mathbf{x}^{(k-1)}, \mathbf{y}^{(k)})}$

6.8.3 Inverse power method with origin shift

Now we present a method to find any eigenvalue by shifting the origin and then using the inverse power method. Assume we know $q \approx \lambda_i$ and $\mathbf{y}^{(0)}$ has a nonzero component along $\mathbf{v}^{(i)}$. Then

$$\begin{aligned} \mathbf{y}^{(k)} &= (A - qI)^{-1}\mathbf{y}^{(k-1)} \\ &\cdots \\ &= (A - qI)^{(-k)}\mathbf{y}^{(0)} = (A - qI)^{-k}(\sum c_i \mathbf{v}^{(i)}) = \sum_1^n \frac{c_i}{(\lambda_i - q)^k}\mathbf{v}^{(i)} \\ &= \frac{1}{(\lambda_i - q)^k}\left[c_1\left(\frac{\lambda_i - q}{\lambda_1 - q}\right)^k \mathbf{v}^{(1)} + \cdots + c_i\mathbf{v}^{(i)} + \cdots + c_n\left(\frac{\lambda_i - q}{\lambda_n - q}\right)^k \mathbf{v}^{(n)}\right] \\ &= \frac{1}{(\lambda_i - q)^k}[c_i\mathbf{v}^{(i)} + \varepsilon^{(k)}] \approx \frac{1}{(\lambda_i - q)}\mathbf{y}^{(k-1)} \end{aligned}$$

since we assumed λ_i is closest to q.

This method finds an eigenvalue closest to q. Thus inverse power method with origin shift useful when one knows an approximate value.

In practice, we use the following variation: The following algorithm is the shifted inverse power method based on Rayleigh Quotient iteration:

Shifted Inverse power method- Rayleigh quotient iteration

For $k = 1, 2, \cdots$, do

$\sigma_{(k)} = \frac{\mathbf{x}_{(k-1)}^T A \mathbf{x}_{(k-1)}}{\mathbf{x}_{(k-1)}^T \mathbf{x}_{(k-1)}}$

Solve $(A - \sigma_{(k)} I)\mathbf{y}_{(k)} = \mathbf{x}_{(k-1)}$

$\mathbf{x}_{(k)} \leftarrow \frac{\mathbf{y}_{(k)}}{\|\mathbf{y}_{(k)}\|_2}$

Here $\sigma_{(k)} \to \lambda$.

Remark 6.8.3. One can use a variable shift by taking $q \approx \lambda^{(k-1)}$. This yields a rapid convergence.

Advantage of inverse power method.

(1) We can find any eigenvalue.

(2) Each iteration takes more time than the direct power method. However, if we have an approximate value of λ_i, the inverse power method will yields λ_i in a few steps.

6.8.4 Deflation method

Suppose A is symmetric and some eigenpair $\{(\lambda_1, \mathbf{x}_1)\}$ is known. Choose a matrix H s.t. $H\mathbf{x}_1 = \alpha \mathbf{e}_1$ (Householder is good). Then (HW. Show this)

$$A\mathbf{x}_1 = \lambda_1 \mathbf{x}_1 \Rightarrow HA\mathbf{x}_1 = \lambda_1 \alpha \mathbf{e}_1 \Rightarrow \alpha HAH^{-1}\mathbf{e}_1 = \lambda_1 \alpha \mathbf{e}_1$$

Hence

$$HAH^{-1} = \begin{bmatrix} \lambda_1 & b^T \\ \mathbf{0} & B \end{bmatrix},$$

where B has eigenvalues $\lambda_2, \cdots, \lambda_n$.

Let $B\mathbf{y}_2 = \lambda_2 \mathbf{y}_2$ and set $\mathbf{x}_2 = H^{-1} \begin{bmatrix} \gamma \\ \mathbf{y}_2 \end{bmatrix}$ and determine γ so that $\mathbf{x}_2$ is the eigenvector of A corresponding to λ_2. We have

$$\begin{aligned} A\mathbf{x}_2 = \lambda_2 \mathbf{x}_2 \Rightarrow AH^{-1} \begin{bmatrix} \gamma \\ \mathbf{y}_2 \end{bmatrix} &= \lambda_2 H^{-1} \begin{bmatrix} \gamma \\ \mathbf{y}_2 \end{bmatrix} \\ HAH^{-1} \begin{bmatrix} \gamma \\ \mathbf{y}_2 \end{bmatrix} &= \lambda_2 \begin{bmatrix} \gamma \\ \mathbf{y}_2 \end{bmatrix} \\ \begin{bmatrix} \lambda_1 & \mathbf{b}^T \\ 0 & B \end{bmatrix} \begin{bmatrix} \gamma \\ \mathbf{y}_2 \end{bmatrix} &= \begin{bmatrix} \lambda_1 \gamma + \mathbf{b}^T \mathbf{y}_2 \\ \lambda_2 \mathbf{y}_2 \end{bmatrix} \end{aligned}$$

Thus $\lambda_2 \gamma = \lambda_1 \gamma + \mathbf{b}^T \mathbf{y}_2$. We can continue deflation. But this process loses stability and is not recommended. Instead use the following simultaneous Iteration.

6.8.5 Simultaneous Iteration

Read Understanding the QR Algorithm, David S. Watkins SIAM Review, Vol. 24, No. 4 (Oct., 1982).

Now consider a method to compute several eigenvalues simultaneously. Simplest way is to use power method with several starting vectors called simultaneous iteration. This is a generalizes the power method.

Algorithm 4.5: Simultaneous iteration
1. X_0 arbitrary $n \times p$ matrix of rank p
2. for $k = 1, 2, \cdots$ do
3. $\quad X_k := AX_{k-1}$ Normalize once in a while
4. end

Let $S_0 = span(X_0)$ and let S be the invariant subspace spanned by $\mathbf{v}_1, \cdots, \mathbf{v}_p$ corresponding to the p largest eigenvalues $\lambda_1, \cdots, \lambda_p$. The columns of $X_k = A^k X_0$ will form a basis for p-dim subspace $S_k = A^k S_0$ provided $|\lambda_p| > |\lambda_{p+1}|$. Hope that this subspaces converge to S. But there are a number of problems associated with this approach:

(1) Normalization of columns are necessary to avoid overflow

(2) Each column of X_k tends to converge a multiple of dominant eigenvector, so the condition number will grow quickly.

• Both of these problems can be partly resolved by *orthogonalization* such as QR.

So we suggest the following

Algorithm 4.6-1: Orthogonal iteration w/ shift 2
1. X_0: arbitrary $n \times p$ matrix of rank p
2. for $k = 1, 2, \cdots$ do
3. $\quad X_{k-1} = \hat{Q}_k R_k$ (once in a while is enough, see the ppt of Hammarling)
4. $\quad X_k = (A - \sigma_k I)\hat{Q}_k$
5. End

:

(1) choose p larger than the number of required eigenvalues.

(2) Extraction by Rayleigh Ritz

(3) When a eigenvector is converged, lock it.(probably small ones near p. Then the rest of iteration converges much faster, since the ratio λ_p/λ_{p+k} is much better than λ_p/λ_{p+1})

(4) If $(A - \sigma I)$ is easily factorized, use inverse simul. iteration $(A - \sigma I)^{-1}$

(5) reliable for symmetric/Hermitian matrices

It can be shown that the sequence converges to $AQ = QB$. Then

$$(A - \sigma I_n)Q = QB - \sigma_p Q = Q(B - \sigma I_p).$$

(From Kubong Note on invariant space) Apply the result to $(A - qI)$ to see $(A-qI)^k$ converges to the invariant subspace $V_q := span\{\mathbf{v}_{i(1)}, \cdots, \mathbf{v}_{i(p)}\}$. Here the order was chosen so that

$$|\lambda_{i(1)} - q| \geq |\lambda_{i(2)} - q| \geq \cdots \geq |\lambda_{i(p)} - q| \cdots \geq |\lambda_{i(n)} - q|.$$

So if $n = 100$ and $p = 50$ we get $\lambda_1 - q, \cdots, \lambda_{25} - q$ and $\lambda_{100} - q, \cdots, \lambda_{75} - q$. But after a few step we may deflate the smallest ones, say, $\lambda_{25} - q$ and $\lambda_{75} - q$. As in power method, the subspace iteration converges to the space generated by the 50 largest eigenvalues, but among them the vectors associated with the smallest one converges first.(Like QR-explain) Same idea may apply to Arnoldi method.

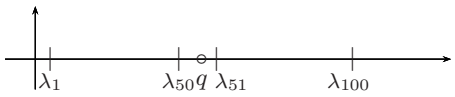


Figure 6.3: distribution of $\lambda_i - q$

Here $\hat{Q}_k$ is $n \times p$ matrix having orthonormal columns and R_k is $p \times p$ upper triangular matrix. $\hat{Q}_k R_k$ is the reduced QR. For the time being assume the shift $\sigma_k = 0$. Under certain conditions, X_k converge to a $n \times p$ matrix X whose columns form a basis for the invariant subspace spanned by the p *dominant* eigenvectors. Hence the matrix $\hat{Q}_k$ will converge to $\hat{Q}$ whose columns form a basis for the invariant subspace S. Hence there exists a $p \times p$ matrix B (Section 4.4.1) s.t.

$$A\hat{Q} = \hat{Q}B.$$

Assume $|\lambda_j| > |\lambda_{j+1}|$ for all $j = 1, 2, \cdots, p$. Then $A\hat{Q}_j \subset Span(\hat{Q}_j) := S^{(j)}$, and

$$A\hat{Q}_j = \hat{Q}_j B_j,$$

Then for any j, $1 \leq j \leq p$ the first j columns of $\hat{Q}$ are the same as if the iteration had been executed on only the first j columns of A. Hence by considering the following multiplication diagram for $j = 1, 2, \cdots, p$ in the order,

$$A\hat{Q}_j = A\left[\mathbf{q}_1, \mathbf{q}_2, \cdots, \mathbf{q}_j\right] = \left[\sum_{i=1}^{j} \mathbf{q}_i b_{i1}, \sum_{i=1}^{j} \mathbf{q}_i b_{i2}, \cdots, \sum_{i=1}^{j} \mathbf{q}_i b_{ij}\right] = \hat{Q}B_j.$$

Note that for each j, $AS^{(j)} \subset S^{(j)}$, hence reading the above equation columnwise, we see

$$A\mathbf{q}_1 \in S^{(1)},\ A\mathbf{q}_2 \in S^{(2)}, \cdots, A\mathbf{q}_j \in S^{(j)}.$$

The first relation implies $A\mathbf{q}_1$ is a multiple of $\mathbf{v}_1$ (hence a multiple of $\mathbf{q}_1$), and the second relation implies $A\mathbf{q}_2 = c_{21}\mathbf{v}_1 + c_{22}\mathbf{v}_2$. Since $span\{\mathbf{q}_1, \mathbf{q}_2\} = span\{\mathbf{v}_1, \mathbf{v}_2\}$ we have $A\mathbf{q}_2 = b_{21}\mathbf{q}_1 + b_{22}\mathbf{q}_2$ for some coefficients b_{21}, b_{22}. Continue this process, we see

$$\begin{aligned} A\mathbf{q}_1 &= b_{11}\mathbf{q}_1, && (6.32)\\ A\mathbf{q}_2 &= b_{21}\mathbf{q}_1 + b_{22}\mathbf{q}_2, && (6.33)\\ A\mathbf{q}_3 &= b_{31}\mathbf{q}_1 + b_{32}\mathbf{q}_2 + b_{33}\mathbf{q}_3, && (6.34)\\ &\cdots, && (6.35)\\ A\mathbf{q}_j &= \sum_{i=1}^{j} \mathbf{q}_i b_{ij}. && (6.36) \end{aligned}$$

Hence we conclude that B is a triangular matrix for all $j = 1, 2, \cdots, p$. By adding any complementary vectors we can assume

$$A\begin{bmatrix}\hat{Q}_1 & \hat{Q}_2\end{bmatrix} = \begin{bmatrix}\hat{Q}_1 & \hat{Q}_2\end{bmatrix}\begin{bmatrix}B & * \\ \mathbf{0} & C\end{bmatrix}$$

Hence the diagonals of B are eigenvalues of A. If some eigenvalues are multiple, then B will be a block triangular matrix.

Assume A is symmetric. Then by multiplying $\mathbf{q}_2$ to the first equation and $\mathbf{q}_1$ to the second equation of (6.32), etc., we see

$$\begin{aligned} \mathbf{q}_2^T A\mathbf{q}_1 &= b_{11}\mathbf{q}_2^T\mathbf{q}_1 = 0, && (6.37)\\ 0 = \mathbf{q}_1^T A\mathbf{q}_2 &= b_{21}\mathbf{q}_1^T\mathbf{q}_1 + b_{22}\mathbf{q}_1^T\mathbf{q}_2 = b_{21}, && (6.38)\\ \mathbf{q}_3^T A\mathbf{q}_1 &= b_{11}\mathbf{q}_3^T\mathbf{q}_1 = 0, && (6.39)\\ \mathbf{q}_1^T A\mathbf{q}_3 &= b_{31}\mathbf{q}_1^T\mathbf{q}_1, && (6.40)\\ &\cdots, && (6.41)\\ & && (6.42) \end{aligned}$$

Thus B is diagonal.

Example 6.8.4. $a_{ij} = \frac{1}{1+i+j}$, $10 \le i, j \le 50$

Now we consider general nonsymmetric case. First we recall the QR-decomposition studied before.

Theorem 6.8.5 (1961, Francis, Kublanovskaya)**.** *Let A be a $n \times n$ nonsingular matrix. Then $\exists!$ Q, R such that*

(1) $A = QR$

(2) Q is unitary

(3) R is upper triangular

(4) $r_{ii} > 0$.

Uniqueness : Assume r_{ii} are chosen to be positive. If $Q_1R_1 = Q_2R_2$, Q_i unitary, R_i upper triangular with positive diagonal. Then $Q_2^*Q_1 = R_2R_1^{-1}$ is unitary upper triangular with positive diagonal. Since $Q = H_n^* \dots H_1^*$ is Hermitian, so is $R_2R_1^{-1}$. Therefore, $R_2R_1^{-1}$ is diagonal, unitary, with positive diagonal part, $R_2R_1^{-1} = \text{diag}(d_1, \cdots, d_n)$. This means $d_i^2 = 1$, $\forall\ i$ and hence $d_i = 1$.

6.8.6 QR - Iteration

Now we introduce a QR - Iteration. In principle, it is the simultaneous iteration in the above case, $n = p$ with $X_0 = I$. The QR factorization $X_0 = \hat{Q}_0R_0$ gives $\hat{Q}_0 = R_0 = I$. So $X_1 = A\hat{Q}_0 = A$. For next, we need to form a QR factorization of $X_1 := \hat{Q}_1R_1 = A$. Define for $k = 0, 1, 2, \cdots$,

$$A_k = \hat{Q}_k^H A \hat{Q}_k. \tag{6.43}$$

Thus we get

$$A_1 = \hat{Q}_1^H A \hat{Q}_1 = \hat{Q}_1^H \hat{Q}_1 R_1 \hat{Q}_1 = R_1 \hat{Q}_1,$$

hence A_1 is the product of the QR factorization of A in the reverse order.

Induction

Let $A_0 = A$ and for $k = 1, 2, \cdots$, assume $Q_kR_k = A_{k-1}$ is given with $\hat{Q}_0 = I$, $\hat{Q}_1 = Q_1$. Then simultaneous iteration becomes

$$X_k = A\hat{Q}_{k-1} = \hat{Q}_{k-1}\hat{Q}_{k-1}^H A \hat{Q}_{k-1} = \hat{Q}_{k-1}A_{k-1} = \hat{Q}_{k-1}Q_kR_k := \hat{Q}_kR_k$$

Thus the QR factor of A_{k-1} gives the QR factor of X_k. Meanwhile

$$\begin{aligned} A_k &= \hat{Q}_k^H A \hat{Q}_k = (\hat{Q}_{k-1}Q_k)^H A (\hat{Q}_{k-1}Q_k) \\ &= Q_k^H \hat{Q}_{k-1}^H A \hat{Q}_{k-1} Q_k \\ &= Q_k^H A_{k-1} Q_k \\ &= Q_k^H Q_k R_k Q_k \\ &= R_k Q_k. \end{aligned} \tag{6.44}$$

Thus the reverse product R_kQ_k from $Q_kR_k = A_{k-1}$ gives A_k, from which we can continue QR factorization of X_{k+1}.

$$X_{k+1} = \hat{Q}_{k+1}R_{k+1} = \hat{Q}_kQ_{k+1}R_{k+1} = \cdots = Q_1Q_2 \cdots Q_kQ_{k+1}R_{k+1}.$$

Summarizing, we have

$$A_{k-1} = Q_kR_k, \quad X_k = Q_1Q_2 \cdots Q_{k-1}Q_kR_k,\ A_k = R_kQ_k \tag{6.45}$$

This shows the relation between two QR- decompositions as well as the relation between A_{k-1} and A_k.

We can repeat this process to obtain the following QR-iteration.

QR-iteration:
Let $A_0 = A$
For $k = 1, 2, \ldots$.
factor $A_{k-1} = Q_k R_k$
form reverse product $A_k = R_k Q_k$.

In general, we have

$$\begin{aligned} A_k &= R_k Q_k = Q_k^{-1} A_{k-1} Q_k \\ &= Q_k^H A_{k-1} Q_k \\ &= Q_k^H Q_{k-1}^H A_{k-2} Q_{k-1} Q_k \\ &= \cdots \\ &= (Q_1 Q_2 \cdots Q_k)^H A (Q_1 Q_2 \cdots Q_k). \end{aligned}$$

The matrices A_k converge at least to block triangular form (Schur form).

Computation of eigenvectors -symmetric case

Eigenvectors are bi-product in the Jacobi iteration. But with QR -iterations, the eigenvectors are not directly obtained. We need some extra work. Note that in case of symmetric matrix without multiple eigenvalues, the columns of $Q := Q_1 Q_2 \cdots Q_k$ converge to the eigenvectors, but very impractical to compute such products!!!

Assume A is symmetric. If $A = PTP^T$(T is tridiagonal) where $PP^T = I$, then

$$T\mathbf{y} = \lambda \mathbf{y} \Rightarrow AP\mathbf{y} = \lambda P\mathbf{y}$$

To find the eigenvectors of T, use inverse power with shift σ close (not not equal) to λ. Since T is tridiagonal, Gauss elimination is quick. Also if $P = P_2 P_3 \cdots P_{n-1}$, the the evaluation

$$\mathbf{x} = P_2 P_3 \cdots P_{n-1}\mathbf{y} = [I - \alpha_2 \mathbf{v}_2 \mathbf{v}_2^T] \cdots [I - \alpha_{n-1} \mathbf{v}_{n-1} \mathbf{v}_{n-1}^T]\mathbf{y}$$

can be carried out quickly since

$$[I - \alpha \mathbf{v}\mathbf{v}^T]\mathbf{y} = \mathbf{y} - \alpha(\mathbf{v}^T\mathbf{y})\mathbf{v}.$$

QR-algorithm with origin shift

From here and thereafter we assume A is reduced to upper Hessenberg form of matrix H and discuss matters on H. It should be pointed out that when

the algorithm converges(see Naiser[1967], p.40) it is observed that the convergence of subdiagonal entries $h_{i,i-1}$ (of a Hessenberg form) depends on the ratio $|\lambda_i|/|\lambda_{i-1}|$. In this discussion, we are assuming

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

By shifting the origin, i.e, for $A - qI$, the convergence will depends on $|\lambda_i - q|/|\lambda_{i-1} - q|$. By suitable shifting we can accelerate the convergence. Now consider a QR-factorization to $A_k - q_k I$:

Shifted QR-Algorithm:
Put $A_0 = A$
For $k = 1, 2, \ldots.$
Choose shift σ_k.
Factor $A_{k-1} - \sigma_k I = Q_k R_k$
Form $A_k = R_k Q_k + \sigma_k I$.

Then

$$\begin{aligned} A_k &= R_k Q_k + q_k I \\ &= Q_k^H (A_{k-1} - q_k I) Q_k + q_k I \\ &= Q_k^H A_{k-1} Q_k. \end{aligned}$$

From the factorization $A_{k-1} = Q_k R_k$, we observe

$$A_{k-1}^{-H} = Q_k R_k^{-H} \text{ hence } A_k^{-H} = R_k^{-H} Q_k$$

Also from $A^k = \hat{Q}_k \hat{R}_k$, we have

$$(A^{-H})^k = \hat{Q}_k \hat{R}_k^{-H}.$$

Note that R_k^{-H} is a lower-triangular matrix. So the QR- decomposition is the same as the inverse QL- iteration (i.e., from column n to column 1) to A^H. The columns of $\hat{Q}_k$ are the same as those produced by inverse iteration to A_k^H.

Now we know the QR iteration is an implicit form of inverse iteration (to A^H). Since inverse iteration converges very fast with suitable choice of shift, this fact can be exploited to choose the shift σ_k.

Lower right corner entry of $A_{k-1}, a_{n,n}^{(k-1)}$ is an approximate eigenvalue, since it is the Rayleigh quotient corresponding to the last columns of $\hat{Q}_k$. From these, we can see the QR produce smaller eigenvalues more quickly. Once it is obtained within the machine accuracy, we can deflate it and work with leading submatrix of size $n - 1$ (See M. Heath book p. 185 for details.) See the Example 4.16. It produces 1, then 2 etc.

A more effective shift is Wilkinson shift defined as the eigenvalue of

$$\begin{bmatrix} a_{n-1,n-1} & b_n \\ b_n & a_{nn} \end{bmatrix}$$

which is closer to a_{nn}. Thus it is

$$\mu = a_{nn} + d - sgn(d)\sqrt{d^2 + b_n^2} \text{ where } d = (a_{n-1,n-1} - a_{nn})/2. \qquad (6.46)$$

Implementation of QR-U-Hessenberg form

An upper-Hessenberg form: $h_{ij} = 0$ if $i > j + 1$.

$$H = \begin{bmatrix} h_{11} & h_{12} & \dots & \dots & h_{1n} \\ h_{21} & h_{21} & \dots & \dots & h_{2n} \\ 0 & h_{32} & \dots & \dots & \\ 0 & 0 & \dots & \dots & \\ 0 & 0 & h_{i,i-1} & \dots & \\ 0 & 0 & \dots & \dots & \\ 0 & 0 & 0 & h_{n,n-1} & h_{nn} \end{bmatrix}$$

Theorem 6.8.6. *The upper Hessenberg form of a matrix is preserved under a QR transformation.*

Thus according to the remark and the Theorem it is practical to perform the QR algorithm in two steps: First reduce the matrix to an U-H form (by Householder), then apply the QR algorithm.

Remark 6.8.7. Two step QR algorithm : First reduce the matrix to an U-H form (by Givens-Householder), then apply the QR algorithm. It is cheaper to use Givens rotation than Householder to apply QR to U-H form.

Theorem 6.8.8. *If $A_k = \hat{Q}_k^H A \hat{Q}_k$ converges, it converges to a block triangular matrix U whose diagonal block is either 1×1 or 2×2 matrices.*

Theorem 6.8.9. *If A is nonsingular and its eigenvalue have distinct moduli, then $A_k = \hat{Q}_k^H A \hat{Q}_k \to U$ (upper triangular) essentially displaying eigenvalues on the diagonal. The columns of the product $Q_1 Q_2 \cdots Q_k$ are the eigenvectors. So $A \sim \hat{Q}_k U \hat{Q}_k^H$ is the Schur decomposition.*

Remark 6.8.10. (1) A single factorization of QR costs $2n^3/3$ and $n \cdot n^2/2$ for RQ product, total $7n^3/6$ FLOPS. It is still impractical to use QR directly.

(2) For upper-Hessenberg matrix, one QR step (Givens rotation is cheaper than Householder's elementary reflector, in this case) costs $2n^2$ together with RQ factorization ($2n^2$). Hence total cost is $4n^2$.

(3) Noting that an upper-Hessenberg matrix form is preserved under QR transformation(Exercise), we usually transform A by Householder's method to an U-Hessenberg form and then use QR (with n^2 multip.) method.

(4) The computational complexity in step (3) is $2\{n^2 + (n-1)^2 + \cdots + 1^2\} = \frac{n(n+1)(2n+1)}{3}$ (to reduce to upper Hessenberg form) and $4(n + n - 1 + \cdots + 1) = 2n(n + 1)$ for QR factorization plus $2n^2$ for RQ.

(5) If the upper-Hessenberg form H is reducible(i.e, having zeros on the subdiagonal) then we can apply QR to each sub matrices.

(6) One advantage of QR is that if H is unreduced(i.e., every subdiagonal element is nonzero) singular Hessenberg matrix, the zero eigenvalue is revealed after one step.(Parlett 1966).

(7) Note a similarity between QR factorization and Gram-Schimdt process. (They both orthogonalize the given set of vectors, columns of A). That is, Span $Col(A)$ = Span $Col(Q_k)$.

6.8.7 Krylov Subspace Methods

QR is based on the simultaneous iteration starting with an orthonormal set (columns of identity matrix). An alternative is to build a subspace incrementally.

$$K_k = [\mathbf{x}_0, A\mathbf{x}_0, \cdots, A^{k-1}\mathbf{x}_0]$$

For $k = n$, we have

$$\begin{aligned} AK_n &= [A\mathbf{x}_0, A\mathbf{x}_1, \cdots, A\mathbf{x}_{n-1}] \\ &= [\mathbf{x}_1, \cdots, \mathbf{x}_{n-1}, \mathbf{x}_n] \\ &= K_n[\mathbf{e}_2, \cdots, \mathbf{e}_n, \mathbf{a}] := K_n C_n \end{aligned}$$

where $\mathbf{a} = K_n^{-1}\mathbf{x}_n$. Therefore

$$K_n^{-1} A K_n = C_n$$

where C_n is upper Hessenberg form (in fact it is a companion matrix). Thus we derived a method for similarity transform to an Hessenberg form. But due to the illcondition problem we orthogonalize it each step by QR

$$Q_n R_n = K_n.$$

$$Q_n^H A Q_n = (K_n R_n^{-1})^{-1} A K_n R_n^{-1} = R_n K_n^{-1} A K_n R_n^{-1} = R_n C_n R_n^{-1} := H$$

since the Hessenberg form of C_n is preserved under pre-post multiplication by an upper triangular matrix, $R_n C_n R_n^{-1}$ is an Hessenberg form.

Equating k th column of $AQ_n = Q_n H$, we obtain

$$A\mathbf{q}_k = (Q_n H)_k = Q_n \mathbf{h}_k = h_{1k}\mathbf{q}_1 + \cdots + h_{kk}\mathbf{q}_k + h_{k+1,k}\mathbf{q}_{k+1},$$

where $\mathbf{h}_k$ is the k-th column of H. Premultiplying by $\mathbf{q}_k^H$ we see $h_{jk} = \mathbf{q}_j^H A \mathbf{q}_k$, $j = 1, \cdots, k$. With normalization, this yields Arnoldi process.

We can obtain eigenvalue and eigenvector at a given step k using Rayleigh -Ritz procedure, which projects the matrix A onto the Krylov subspace K_k. Let

$$Q_k = [\mathbf{q}_1, \cdots, \mathbf{q}_k]$$

contain first k Arnoldi vectors, let $n \times (n-k)$ matrix

$$U_k = [\mathbf{q}_{k+1}, \cdots, \mathbf{q}_n]$$

the remaining Arnoldi vectors to be computed. Let $Q_n = [Q_k, U_k]$. Then

$$H = Q_n^H A Q_n = \begin{bmatrix} Q_k^H \\ U_k^H \end{bmatrix} A[Q_k, U_k] = \begin{bmatrix} Q_k^H A Q_k & Q_k^H A U_k \\ U_k^H A Q_k & U_k^H A U_k \end{bmatrix} = \begin{bmatrix} H_k & M \\ \tilde{H}_k & N \end{bmatrix}$$

where M, N are yet to be computed. Since H is upper Hessenberg, so is $H_k := Q_k^H A Q_k$ and $\tilde{H}_k$ has only one nonzero entry $h_{k+1,k}$. The eigenvalues of H_k (Ritz value) converges to the eigenvalues of A and $Q_k\mathbf{y}$, $\mathbf{y}$ eigenvectors of H_k (Ritz vector), converges to the eigenvectors of A.

One must still compute eigenvalues and eigenvectors of H_k by some other methods (QR). This process produces extreme eigenvalues quickly. One needs to store Q_k of Arnoldi vectors and H_k. Restart once in a while.

Arnoldi

$\mathbf{q}_1$ arb. normalized vector
for $k = 1, 2, \ldots$
  $\mathbf{u}_k = A\mathbf{q}_k$ (generate next vector)
  for $j = 1, 2, \cdots k$ (Gram Schmidt)
    $h_{j,k} = \mathbf{q}_j^H \mathbf{u}_k$ (orthogonalize new vector)
    $\mathbf{u}_k = \mathbf{u}_k - h_{j,k}\mathbf{q}_j$ (against previous vector)
  end
  $h_{k+1,k} = \|\mathbf{u}_k\|_2$
  If $h_{k+1,k} = 0$ stop (Reducible)
  $\mathbf{q}_{k+1} = \mathbf{u}_k / h_{k+1,k}$
end

Lanczos algorithm (Symmetric version of Arnoldi)

The Arnoldi algorithm becomes much simpler if A is symmetric. $H)k$ is now becomes tridiagonal, denoted by T_k. Hence the algorithm is

$\mathbf{q}_0 = 0$, $\beta = 0$ arb. normalized vector
$\mathbf{q}_1$ arb. normalized vector
for $k = 1, 2, \ldots$
  $\mathbf{u}_k = A\mathbf{q}_k$ (generate next vector)
  $\alpha_k = \mathbf{q}_j^H \mathbf{u}_k$ (orthogonalize new vector)
  $\mathbf{u}_k = \mathbf{u}_k - \beta_{k-1}\mathbf{q}_{k-1} - \alpha_k\mathbf{q}_k$ (against previous two vectors)
  $\beta_{k-1} = \|\mathbf{u}_k\|_2$ (against previous vector)
  If $\beta_k = 0$ stop (Reducible)
  $\mathbf{q}_{k+1} = \mathbf{u}_k / \beta_k$
end

Not that the orthogonalization step is three term relationship. The α_k and β_k are diagonal and subdiagonal entries, resp. of $T_k (= H_k)$, whose eigenvalues

and eigenvectors must be computed by some other methods (QR using Givens, say).

This process generates extreme eigenvalues rapidly (either end), Fig 4.4

(1) In QR, one needs reduce A to U-H form, then use shifted QR(Rayleigh)

(2) Arnoldi(Lanczos), one computes H_k (T_k) by $\mathbf{u}_k = A\mathbf{q}_k$ and then use shifted QR(Rayleigh)

Bisection or Spectrum-Slicing

(1) Let A be symmetric. Then *inertia* of A is a triple (p, n, z) consisting of the number of positive, negative and zero eigenvalues of A.

(2) SAS^T is a congruence transformation

(3) Sylvester law of inertia says, a congruence transformation preserve the inertia.

(4) Find a good congruence transformation that makes the inertia easy to determine.

(5) Apply it to $A - \sigma I$ to determine the number of eigenvalues to the left or right of σ

(6) LDL^T is a good candidate

(7) Bisection method to find good σ

Divide and Conquer

(1) Assume a rank one modification of a diagonal matrix $D + \beta\mathbf{v}\mathbf{v}^T$.

$$D + \beta\mathbf{v}\mathbf{v}^T - \lambda I = (D - \lambda I)(I + \beta(D - \lambda I)^{-1}\mathbf{v}\mathbf{v}^T)$$

Hence $p(\lambda) = \prod_i (\lambda - d_i)\, det(I + \beta(D - \lambda I)^{-1}\mathbf{v}\mathbf{v}^T)$ and by exer. 4.25

$$det(I + \beta(D - \lambda I)^{-1}\mathbf{v}\mathbf{v}^T) = (1 + \beta\mathbf{v}^T(D - \lambda I)^{-1}\mathbf{v}) = 1 + \beta\sum_{j=1}^{n}\frac{v_j^2}{d_j - \lambda}$$

Then apply bisection method to find zeros of this function.

(2) Any real symmetric tri-diagonal matrix can be written as a rank one modification of a block tridiagonal matrix of the form $T + \beta\mathbf{u}\mathbf{u}^T$, where $T = \begin{bmatrix} T_1 & 0 \\ 0 & T_2 \end{bmatrix}$, where $\beta = t_{k,k+1} = t_{k+1,k}$, T_1 is $k \times k$, T_2 is $(n-k)\times(n-k)$ tridiagonal, $\mathbf{u} = \mathbf{e}_k + \mathbf{e}_{k+1}$.

(3) Compute eigenvalues/eigenvectors of smaller matrices T_1 and T_2 to get

$$Q_1^T T_1 Q_1 = D_1 \text{ and } Q_2^T T_2 Q_2 = D_2,$$

then

(4) with $D = diag(D_1, D_2)$, $Q = diag(Q_1, Q_2)$, we have

$$Q^T T Q = D + \beta \mathbf{v}\mathbf{v}^T$$

where $\mathbf{v} = Q^T \mathbf{u}$. Now we can apply the previous method.

This Divide and Conquer method is often significantly faster than QR iteration if all the eigenvalues and eigenvectors of a symmetric matrix are needed.

Relatively Robust Representation

Eigenvalues of a tri-diagonal matrix is sensitive to perturbations of its entries, but much less sensitive to LDL^T factors. Use twisted factorization.

6.9 Generalized Eigenvalue Problems

$$A\mathbf{x} = \lambda B\mathbf{x}$$

A naive approach is to convert it to the standard problem as

$$B^{-1}A\mathbf{x} = \lambda\mathbf{x} \text{ or } A^{-1}B\mathbf{x} = 1/\lambda\mathbf{x}$$

But such methods are not recommended since it may cause

(1) Loss of accuracy due to the rounding error

(2) Loss of symmetry when A and B are symmetric.

If A and B are symmetric and one of them is positive definite, then symmetry still can be retained by Cholesky decomposition. Example, if $B = LL^T$, then the eigenvalue problem can be transformed into ($L^T\mathbf{x} = \mathbf{y}$)

$$L^{-1}AL^{-T}\mathbf{y} = \lambda\mathbf{y}$$

Still loss of accuracy etc.

A better method is QZ algorithms. Idea: transform A, B simultaneously to upper triangular form by orthogonal transform.

(1) Apply an orthogonal transform Q_0 to reduce B to a upper triangular form.

(2) Apply a sequence of orthogonal transforms Q_k to reduce A to an upper Hessenberg form.

(3) ....

6.10 Stability and Condition number for the eigenvalue problem

Eigenvalues are continuous functions of the entries. However, they can be sensitive with small change. We will see some example.

Example 6.10.1. Let A and its small perturbation $A+E$ are given as follow.

$$A = \begin{bmatrix} a & & & \\ 1 & \ddots & & 0 \\ & \ddots & \ddots & \\ & 0 & \ddots & \ddots \\ & & & 1 \quad a \end{bmatrix} \qquad A+E = \begin{bmatrix} a & & & \varepsilon \\ 1 & \ddots & & 0 \\ & \ddots & \ddots & \\ & 0 & \ddots & \ddots \\ & & & 1 \quad a \end{bmatrix}$$

Let C_{A+E} denote the characteristic polynomial of $A+E$. Then

$$C_{A+E}(\lambda_\varepsilon) = (a-\lambda_\varepsilon)^n + \varepsilon(-1)^{n-1},$$

Hence

$$\lambda_\varepsilon = a \pm \omega^r \cdot \varepsilon^{1/n}, r = 1, \cdots,$$

where $\omega = \frac{2\pi}{n} + i \sin\frac{2\pi}{n}$ is the root of unity. Even if ε is very small, $\varepsilon^{1/n}$ can be non negligible. For example, $\varepsilon = 10^{-10}$ and $n = 100$, then $\varepsilon^{1/100} = 10^{-1/10} \approx 1$. Error is close to 1.

Example 6.10.2. Eigenvectors are not continuous functions of ε(even for Hermitian matrix).

$$A^{(\varepsilon)} = \begin{bmatrix} 1+\varepsilon\cos\frac{2}{\varepsilon} & -\varepsilon\sin\frac{2}{\varepsilon} \\ -\varepsilon\sin\frac{2}{\varepsilon} & 1-\varepsilon\cos\frac{2}{\varepsilon} \end{bmatrix}, \qquad A^{(0)} = I$$

eigenvalues of $A^{(\varepsilon)}$ satisfies $(1+\varepsilon\cos\frac{2}{\varepsilon} - \lambda)(1-\varepsilon\cos\frac{2}{\varepsilon} - \lambda) - \varepsilon^2\sin^2\frac{2}{\varepsilon} = 0$.

$$(1-\lambda_\varepsilon)^2 - \varepsilon^2 = 0 \qquad \therefore \quad \lambda_\varepsilon = 1 \pm \varepsilon$$

Eigenvectors are

$$X^{(1)} = \begin{bmatrix} \sin(\frac{1}{\varepsilon}) \\ \cos(\frac{1}{\varepsilon}) \end{bmatrix}, \qquad X^{(2)} = \begin{bmatrix} -\cos(\frac{1}{\varepsilon}) \\ \sin(\frac{1}{\varepsilon}) \end{bmatrix}$$

$\lim_{\varepsilon\to 0} X^{(1)}$ does not exist!! Extremely ill-conditioned.

We want to investigate what kind of things influence the change in the eigenvalues when entries are perturbed a bit. Consider the case of a simple eigenvalue

$$A\mathbf{x}_1 = \lambda_1\mathbf{x}_1, \qquad \lambda_1 \neq \lambda_j, \quad j = 2, \ldots, n.$$

Its perturbed problem is

$$(A + \varepsilon B)\mathbf{x}_1(\varepsilon) = \lambda_1(\varepsilon)\mathbf{x}_1(\varepsilon), \tag{6.47}$$

where $|B_{ij}| < 1$. Assume

$$\begin{aligned} \lambda_1(\varepsilon) &= \lambda_1 + k_1\varepsilon + k_2\varepsilon^2 + \cdots \\ &\quad k_1 : \text{This number is important. Try to make it small.} \\ \mathbf{x}_1(\varepsilon) &= \mathbf{x}_1 + E_1\varepsilon + E_2\varepsilon^2 + \cdots \end{aligned}$$

and A has linear elementary divisors, i.e., diagonalizable.

Theorem 6.10.3. *Let A be diagonalizable. Then there exist n linearly independent right and left eigenvectors $\mathbf{x}_j$ and $\mathbf{y}_i$ such that*

$$P^{-1}AP = \Lambda$$

where

$$\mathbf{x}_j = \mathrm{col}_j(P) \qquad \mathbf{y}_i^T = \mathrm{Row}_i(P^{-1})$$

and Λ is the diagonal matrix displaying eigenvalues.

Assume $\|\mathbf{x}_i\|_2 = \|\mathbf{y}_j\|_2 = 1$ for the time being. Note that $\mathbf{x}_i^T\mathbf{y}_i \neq 1$.

Lemma 6.10.4. $\mathbf{y}_i^T\mathbf{x}_j = 0$ *if* $\lambda_i \neq \lambda_j$.

Proof. $A\mathbf{x}_j = \lambda_j\mathbf{x}_j \Rightarrow \mathbf{y}_i^TA\mathbf{x}_j = \lambda_j\mathbf{y}_i^T\mathbf{x}_j$. But $\mathbf{y}_i^TA = \lambda_i\mathbf{y}_i^T \Rightarrow \mathbf{y}_i^TA\mathbf{x}_j = \lambda_i\mathbf{y}_i^T\mathbf{x}_j$. Therefore, $(\lambda_i - \lambda_j)\mathbf{y}_i^T\mathbf{x}_j = 0$. Therefore we conclude $\mathbf{y}_i^T\mathbf{x}_j = 0$. □

Expand E_i and $\mathbf{x}_1(\varepsilon)$ in terms of eigenvectors of A.

$$\begin{aligned} E_i &= \sum_{j=1}^{n} s_{ji}\mathbf{x}_j, \qquad i = 1, 2, \ldots, \\ \mathbf{x}_1(\varepsilon) &= \mathbf{x}_1 + \varepsilon\left(\sum_j s_{j1}\mathbf{x}_j\right) + \varepsilon^2\left(\sum_j s_{j2}\mathbf{x}_j\right) + \ldots \\ &= (1 + \varepsilon s_{11} + \varepsilon^2 s_{12} + \cdots)\mathbf{x}_1 + (\varepsilon s_{21} + \varepsilon^2 s_{22} + \cdots)\mathbf{x}_2 + (\varepsilon s_{31} + \varepsilon^2 s_{32} + \cdots)\mathbf{x}_3 \cdots \end{aligned}$$

Normalize $\mathbf{x}_1(\varepsilon)$ by dividing by $(1 + \varepsilon s_{11} + \varepsilon^2 s_{12} + \cdots)$,

$$\mathbf{x}_1^{(\varepsilon)} = \mathbf{x}_1 + (\varepsilon t_{21} + \varepsilon^2 t_{22} + \cdots)\mathbf{x}_2 + \cdots + (\varepsilon t_{n1} + \varepsilon^2 t_{n2} + \cdots)\mathbf{x}_n$$

and substitute into the perturbed equation (6.47) to get

$$(A + \varepsilon B)[\mathbf{x}_1 + (\quad)\mathbf{x}_2 + \cdots + (\quad)\mathbf{x}_n] = \lambda_1(\varepsilon)[\mathbf{x}_1 + (\quad)\mathbf{x}_2 + \cdots + (\quad)\mathbf{x}_n]$$

where $\lambda_1(\varepsilon) = \lambda_1 + k_1\varepsilon + k_2\varepsilon^2 + \cdots$

Equating the 0-th order term (i.e, terms with no ε factor)

$$A\mathbf{x}_1 = \lambda_1 \mathbf{x}_1.$$

Equating 1-st order term

$$A\left(\sum_{j=2}^{n} t_{j1}\mathbf{x}_j\right) + B\mathbf{x}_1 = \lambda_1\left(\sum_{j=2}^{n} t_{j1}\mathbf{x}_j\right) + k_1\mathbf{x}_1.$$

Therefore, $\sum_{j=2}^{n}(\lambda_j - \lambda_1)t_{j1}\mathbf{x}_j + B\mathbf{x}_1 = k_1\mathbf{x}_1$. Here k_1 measures error size. Multiply by $\mathbf{y}_1^T$, to obtain

$$\sum_{j=2}^{n}(\lambda_j - \lambda_1)t_{j1}\mathbf{y}_1^T\mathbf{x}_j + \mathbf{y}_1^T B\mathbf{x}_1 = k_1\mathbf{y}_1^T\mathbf{x}_1.$$

Since $\mathbf{y}_1^T\mathbf{x}_j = 0$ for $J = 2, \cdots, n$, we have

$$k_1 = \frac{\mathbf{y}_1^T B\mathbf{x}_1}{\mathbf{y}_1^T\mathbf{x}_1}, \qquad |k_1| \le \frac{\|\mathbf{y}_1\|_2\|B\|_2\|\mathbf{x}_1\|_2}{|\mathbf{y}_1^T \cdot \mathbf{x}_1|} = \frac{\sqrt{\rho(B^TB)}}{|\mathbf{y}_1^T \cdot \mathbf{x}_1|}$$

Since we do not have any information on the spectral radius of B, we can only see the effect of $|\mathbf{y}_1^T \cdot \mathbf{x}_1|^{-1}$. Thus we define it as the condition number of A (with respect to) λ_1 when $\mathbf{x}_i$, $\mathbf{y}_i$ are normalized.

In general,

$$\frac{1}{|s_i|} = \frac{1}{|\mathbf{y}_i^T\mathbf{x}_i|}$$

are defined as condition numbers for the eigenvalue problem.

Theorem 6.10.5. *We have $|s_i| \le 1$.*

Proof. $|s_i| = |\mathbf{y}_i^T\mathbf{x}_i| \le \|\mathbf{y}_i\| \cdot \|\mathbf{x}_i\| = 1$. Thus, $|s_i|^{-1} \ge 1$. □

Theorem 6.10.6. *If $A^H = A$, then $P^{-1}AP = P^HAP = \Lambda$. Therefore, $\mathbf{y}_i^T = \mathbf{x}_i^H$ and hence $s_i = \mathbf{y}_i^T \cdot \mathbf{x}_i = \mathbf{x}_i^H \cdot \mathbf{x}_i = 1$. $\therefore$ Hermitian matrix is perfectly conditioned for eigenvalue problem.*

6.10.1 Error bound of eigenvalues

Assume A is diagonalizable i.e., $P^{-1}AP = \Lambda$ (diagonal). Suppose $A + E$ is a perturbation of A and let $(\mu, \mathbf{y})$ be an eigenpair of $A + E$. Then

$$(A + E)\mathbf{y} = \mu\mathbf{y} \Rightarrow (\mu I - A)\mathbf{y} = E\mathbf{y}$$

For now assume μ is not an eigenvalue of A. Then

$$P^{-1}(\mu I - A)PP^{-1}\mathbf{y} = P^{-1}EPP^{-1}\mathbf{y}$$

$$(\mu I - \Lambda)P^{-1}\underset{\mathbf{w}}{\mathbf{y}} = P^{-1}EP \cdot P^{-1}\underset{\mathbf{w}}{\mathbf{y}}\,, \qquad (\mu I - \Lambda)\mathbf{w} = P^{-1}EP \cdot \mathbf{w}.$$

Since $\mu I - \Lambda$ is invertible, $\mathbf{w} = (\mu I - \Lambda)^{-1}P^{-1}EP \cdot \mathbf{w}$ and thus

$$\begin{aligned} \|\mathbf{w}\| &\leq \|(\mu I - \Lambda)^{-1}\| \cdot \|P^{-1}EP\| \cdot \|\mathbf{w}\| \\ 1 &\leq \|(\mu I - \Lambda)^{-1}\| \cdot \|P^{-1}EP\|. \end{aligned}$$

Choose L_2 norm

$$\|(\mu I - \Lambda)^{-1}\|_2 = \max_i |\mu - \lambda_i|^{-1} = 1/\min_i |\mu - \lambda_i|$$

$$\therefore \qquad \min_i |\mu - \lambda_i| = \|(\mu I - \Lambda)^{-1}\|_2^{-1} \leq \|P^{-1}\|_2 \cdot \|P\|_2 \|E\|_2 = \kappa_2(P)\|E\|_2.$$

Here λ_i is the eigenvalue of A closest to μ.

Theorem 6.10.7. *If A is diagonalizable, any eigenvalue μ of $A + E$ satisfies*

$$|\mu - \lambda| \leq \kappa_2(P)\|E\|_2$$

for some eigenvalue λ of A. i.e., if μ is an eigenvalue of a perturbed matrix, it lies in a disk center at some eigenvalue of A of radius $\kappa_2(P)\|E\|_2$.

* Above theorem holds even when μ is an eigenvalue of A.

* Thus the condition # of P has some effect on the condition # of A's eigenvalue problem.

Corollary 6.10.8. *If A is Hermitian, then P is unitary and hence $\|P\|_2 = \|P^{-1}\|_2 = 1$ and so, $\min_i |\mu - \lambda_i| = |\mu - \lambda_p| \leq \|E\|_2$.*

Each disk of radius $\|E\|_2$ contains an eigenvalue of $A+E$ (may be complex). In other word, the eigenvalue of A is stable under perturbation and furthermore, its error is less than the L_2 norm of perturbation matrix.

Recall Perron-Frobenius theorem: If $A \geq 0$, irreducible, then $\rho(A)$ is a simple eigenvalue and $|\lambda_j| < \rho(A)$, $j = 2, \ldots, n$. Thus, if A is not irreducible, we apply (P-F) to $P^T AP = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix}$, block matrix with

$$Spec(A) = \mathrm{Spec}(A_{11}) \cup \mathrm{Spec}(A_{22}).$$

Thus we have the same result for any nonnegative matrix A except that $\rho(A)$ may be an eigenvalue of multiplicity greater than 1.

Let $\ell(\mathbf{v})$ be a linear ftnal such as $e_1(\mathbf{v})$ or $e_\infty(\mathbf{v})$.

Algorithm: Inverse power method with fixed shift

For $k = 1, 2, \cdots$, do

$\mathbf{v}^{(i+1)} = (A - \sigma I)^{-1}u_i$ and $k_{i+1} = \ell(\mathbf{v}_{i+1})$

Set $\mathbf{u}_{i+1} = \mathbf{v}_{i+1}/k_{i+1}$

Convergence rate is $\left|\frac{\lambda_1 - \sigma}{\lambda_2 - \sigma}\right|$.

Exercise 6.10.9. (1) The eigenvalue problem, $A\mathbf{x} = \lambda\mathbf{x}$, $\mathbf{x}^T\mathbf{x} = 1$ can be regarded as nonlinear system of equations $\mathbf{f}(\mathbf{x}, \lambda) = \mathbf{0}$ where

$$\mathbf{f}(\mathbf{x}, \lambda) = \begin{pmatrix} A\mathbf{x} - \lambda\mathbf{x} \\ \mathbf{x}^T\mathbf{x} - 1 \end{pmatrix}$$

Write down a Newton's method to solve this system. Or $A\mathbf{x} = \lambda\mathbf{x}$, $\ell^T(\mathbf{x}) = 1$ (ℓ^T is a fixed linear ftnal) can be regarded as nonlinear system of equations $\mathbf{f}(\mathbf{x}, \lambda) = \mathbf{0}$ where

$$\mathbf{f}(\mathbf{x}, \lambda) = \begin{pmatrix} A\mathbf{x} - \lambda\mathbf{x} \\ \ell^T(\mathbf{x}) - 1 \end{pmatrix}$$

Ans:

$$D\mathbf{f} \quad = \quad \begin{pmatrix} A - \lambda I & -\mathbf{x} \\ \ell^T & 0 \end{pmatrix}$$

The Newton can be written by component

$$(A - \lambda_i I)\mathbf{x}_{i+1} = (\lambda_{i+1} - \lambda_i)\mathbf{x}_i \tag{6.48}$$

$$\ell^T(\mathbf{x}_{i+1}) = 1 \tag{6.49}$$

Let

$$\mathbf{v}_{i+1} = \frac{\mathbf{x}_{i+1}}{\lambda_{i+1} - \lambda_i}.$$

Substituting into (6.48) we get,

$$(A - \lambda_i I)\mathbf{v}_{i+1} \quad = \quad \mathbf{x}_i$$

by (6.49)

$$k_{i+1} = \ell^T(\mathbf{v}_{i+1}) = \frac{\ell^T(\mathbf{x}_{i+1})}{\lambda_{i+1} - \lambda_i} = \frac{1}{\lambda_{i+1} - \lambda_i}$$

It follows that

$$\lambda_{i+1} = \lambda_i + \frac{1}{k_{i+1}}$$

A good initial guess is $\mathbf{x}_0, \|\mathbf{x}_0\| = 1, \lambda_0 = \mathbf{x}_0^T A\mathbf{x}_0$. Thus (6.48) and (6.49) are identified with inverse power (variant shift) of the following:

Algorithm: Inverse power method with variant shift
For $k = 1, 2, \cdots$, do
$\mathbf{v}^{(i+1)} = (A - \sigma_i I)^{-1}\mathbf{u}_i$ and $k_{i+1} = \ell(\mathbf{v}_{i+1})$
Set $\mathbf{u}_{i+1} = \mathbf{v}_{i+1}/k_{i+1}$ and $\sigma_{i+1} = \sigma_i + 1/k_{i+1}$.

6.11 Least Square Methods - by QR

When A is $m \times n$ matrix ($m \geq n$), the system $A\mathbf{x} = \mathbf{b}$ usually does not have solution. In this case we can consider a least square solution of $A\mathbf{x} = \mathbf{b}$. The following QR decomposition is possible:

$$A = QR,$$

where Q is $m \times m$ orthogonal matrix($Q^T = Q^{-1}$) and R is an $m \times n$ upper triangular matrix. Suppose A is a square and have a QR decomposition, then we can solve $A\mathbf{x} = \mathbf{b}$:

$$\begin{aligned} A\mathbf{x} &= \mathbf{b} \\ QR\mathbf{x} &= \mathbf{b} \\ R\mathbf{x} &= Q^T\mathbf{b} \end{aligned}$$

Since R is upper triangular, this is easy to solve provided that diagonal entries of R are nonzero.

Now consider the case of **nonsquare** matrix. The case $m > n$(more equations than the unknowns) is more interesting. Since the solution of $A\mathbf{x} = \mathbf{b}$ does not exist in general, we are forced to find a best alternative: i.e., the minimizer $\mathbf{x}$ of

$$\|A\mathbf{x} - \mathbf{b}\|^2$$

which is equivalent to minimizing

$$(A\mathbf{x} - \mathbf{b})^T(A\mathbf{x} - \mathbf{b}) = \mathbf{x}^T A^T A\mathbf{x} - \mathbf{b}^T A\mathbf{x} - \mathbf{x}^T A^T\mathbf{b} + \mathbf{b}^T\mathbf{b}.$$

Taking the derivative w.r.t $\mathbf{x}$, we have

$$\mathbf{x}^T A^T A - \mathbf{b}^T A = 0 \text{ or } A^T A\mathbf{x} = A^T\mathbf{b}.$$

Interpretation:

Since $A\mathbf{x}$ lies in $\mathbb{R}^n$(represented by the x-axis in the figure 6.11), the minimum is attained when $A\mathbf{x}$ is the projection of $\mathbf{b}$ on the $\mathbb{R}^n$ space. Thus $\mathbf{r}$ is orthogonal to $\mathbb{R}^n$(the column space of A). In algebraic notation, we must have $\mathbf{r}^T A = 0$ which is equivalent to

$$\begin{aligned} A^T(\mathbf{b} - A\mathbf{x}) &= 0 \\ A^T A\mathbf{x} &= A^T\mathbf{b}. \end{aligned}$$

This is called the **normal equation**. From now on we assume $m \geq n$ and the rank of A is n. Solving such problems, one could consider using LU decomposition, in this case Cholesky decomposition. However, experience says it is expensive and very sensitive. So we solve it by QR as follows:

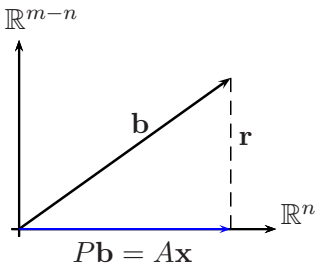


Figure 6.4: least square fit

Solving with QR

Consider solving $A^TA\mathbf{x} = A^T\mathbf{b}$. If we have a QR decomposition $A = QR$,

Two ways:

(a) (Q is $m \times m$, $Q^TQ = I_m$ and R is $m \times n$)(Leader book) or

(b) (Q is $m \times n$, $Q^TQ = I_n$ and R is $n \times n$-Givens rotation, See Golub p 226) This one takes much less space and time.

$$\begin{aligned} (QR)^T(QR)\mathbf{x} &= (QR)^T\mathbf{b} \\ R^TQ^T(QR)\mathbf{x} &= R^TQ^T\mathbf{b} \\ R^TR\mathbf{x} &= R^TQ^T\mathbf{b}. \end{aligned}$$

Note that R is $m \times n$ upper triangular matrix. Thus

$$R = \begin{pmatrix} R_1 \\ \mathbf{0} \end{pmatrix},$$

where R_1 is $n \times n$ upper triangular matrix and $\mathbf{0}$ is $(m-n) \times n$ zero matrix.

$$\begin{aligned} \begin{pmatrix} R_1 \\ \mathbf{0} \end{pmatrix}^T \begin{pmatrix} R_1 \\ \mathbf{0} \end{pmatrix} \mathbf{x} &= \begin{pmatrix} R_1 \\ \mathbf{0} \end{pmatrix}^T Q^T\mathbf{b} \\ \begin{pmatrix} R_1^T & \mathbf{0}^T \end{pmatrix} \begin{pmatrix} R_1 \\ \mathbf{0} \end{pmatrix} \mathbf{x} &= \begin{pmatrix} R_1^T & \mathbf{0}^T \end{pmatrix} Q^T\mathbf{b} \\ R_1^TR_1\mathbf{x} &= \begin{pmatrix} R_1^T & \mathbf{0}^T \end{pmatrix} Q^T\mathbf{b}. \end{aligned}$$

Since R_1 is nonsingular, we have

$$R_1\mathbf{x} = \begin{pmatrix} I_n & \mathbf{0}^T \end{pmatrix} Q^T\mathbf{b}.$$

That is,

$$R_1\mathbf{x} = (Q^T\mathbf{b})_n (\text{first } n \text{ components of } Q^T\mathbf{b}) \tag{6.50}$$

This is a square system with nonsingular upper $n \times n$ triangular matrix R_1. Thus it can be solved by backward substitution.

6.12 Singular Value decomposition

SVD of Rectangular Matrix

As we go from Gauss Elim to Gauss-Jordan, we can further transform QR to have R as diagonal for, by right multiplying an orthogonal matrix, we can obtain the following.

Theorem 6.12.1 (Singular value decomposition)**.** *For $A \in \mathbb{R}^{m\times n}$, there exists orthogonal matrices $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ such that*

$$A = U\Sigma V^T \text{ or } AV = U\Sigma, \tag{6.51}$$

where Σ is $m \times n$ matrix and k is the rank of A and $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k \geq 0$.

Proof. Let $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{y} \in \mathbb{R}^m$ such that $\|\mathbf{x}\|_2 = \|\mathbf{y}\|_2 = 1$ and $A\mathbf{x} = \sigma\mathbf{y}$ with $\sigma = \|A\|_2$. Let

$$V = (\mathbf{x}, V_1) \in \mathbb{R}^{n\times n}$$

and

$$U = (\mathbf{y}, U_1) \in \mathbb{R}^{m\times m}$$

be orthogonal. Thus U^TAV has the following form:

$$A_1 = U^TAV = \begin{bmatrix} \sigma & \mathbf{w}^T \\ 0 & B \end{bmatrix} \begin{matrix} 1 \\ m-1 \end{matrix}$$
$$\begin{matrix} \\ 1 \end{matrix} \quad \begin{matrix} \\ n-1 \end{matrix}$$

Since

$$\left\| A_1 \begin{pmatrix} \sigma \\ \mathbf{w} \end{pmatrix} \right\|_2^2 \geq (\sigma^2 + \mathbf{w}^T\mathbf{w})$$

it follows that $\|A_1\|_2^2 \geq \sigma^2 + \mathbf{w}^T\mathbf{w}$. But since $\sigma^2 = \|A\|_2^2 = \|A_1\|_2^2$, we must have $\mathbf{w} = 0$. Now induction proves the argument. □

The values σ_i are the singular values of A and vectors $\mathbf{u}_i$ and $\mathbf{v}_i$(columns of U and V) are left singular and right singular vectors. We see

$$\begin{matrix} A\mathbf{v}_i = \sigma_i\mathbf{u}_i \\ A^T\mathbf{u}_i = \sigma_i\mathbf{v}_i \end{matrix}, \quad i = 1, \cdots, p$$

SVD to Least Square problem

SVD provides another way of solving least square problem. Consider solving an over-determined system $A\mathbf{x} = \mathbf{b}$ by least square method (Although it is possible to form SVD for any m, n, we assume $m > n$(over-determined) and the rank of A is n). The the reduced SVD is

$$A = U\Sigma V^T = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 \\ \mathbf{0} \end{bmatrix} V^T = U_1\Sigma_1 V^T, \tag{6.52}$$

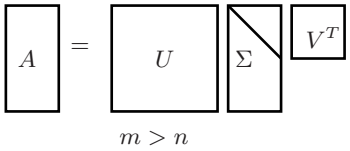


Figure 6.5: Σ of SVD

where U_1 is an $m \times n$, Σ_1 is $n \times n$ nonsingular. Using this, we can solve the normal equation $A^T A\mathbf{x} = A^T\mathbf{b}$. We have

$$\begin{aligned} V\Sigma_1^T U_1^T U_1 \Sigma_1 V^T \mathbf{x} &= V\Sigma_1^T U_1^T \mathbf{b} \\ \Sigma_1 V^T \mathbf{x} &= U_1^T \mathbf{b} \\ V^T \mathbf{x} &= \Sigma_1^{-1} U_1^T \mathbf{b} \\ \mathbf{x} &= V\Sigma_1^{-1} U_1^T \mathbf{b}. \end{aligned}$$

The solution with minimal norm is given by

$$\mathbf{x} = \Sigma_{\sigma_i \neq 0} \frac{\mathbf{u}_i^T \mathbf{b}}{\sigma_i} \mathbf{v}_i$$

If SVD is useful for ill-conditioned or nearly singular problem, since we can drop the term of small singular values.

Other Applications of SVD

We see from (6.52), that $A = U\Sigma V^T$ equals

$$\left[\mathbf{u}_1, \cdots, \mathbf{u}_n\right] \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \\ & \mathbf{0} & \end{bmatrix} \begin{bmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \\ \vdots \\ \mathbf{v}_n^T \end{bmatrix} = \left[\mathbf{u}_1, \cdots, \mathbf{u}_n\right] \begin{bmatrix} \sigma_1 \mathbf{v}_1^T \\ \sigma_2 \mathbf{v}_2^T \\ \vdots \\ \sigma_n \mathbf{v}_n^T \end{bmatrix} = \sum_{i=1}^{n} \sigma_i \mathbf{u}_i \mathbf{v}_i^T \tag{6.53}$$

If the entries σ_j $(j = k+1, \cdots, n)$ are small compared with σ_j for $j = 1, \cdots, k$, we can drom them and obtain

$$A \doteq \sum_{i=k+1}^{n} \sigma_i \mathbf{u}_i \mathbf{v}_i^T \equiv A_1 \tag{6.54}$$

and see that

$$\|A - A_1\|_E \leq \sum_{i=k+1}^{n} \sigma_i \|\mathbf{u}_i\|_2 \|\mathbf{v}_i^T\|_2 \leq \sum_{i=k+1}^{n} \sigma_i.$$

Thus we have restored A from the vectors

$$\sigma_i, \mathbf{u}_i, \mathbf{v}_i,\ i = 1, \cdots, k$$

The total storage is $(2m+1)k$ while the whole storage of A requires mn. The number k indicates the (essentially) linearly independent vectors among the column vectors of A. This technique is used in the image compression.

You may try to us MGS with column pivoting to get similar result.

6.13 Solving linear system by minimizing the residual

We will consider an iterative method to solve a nonsymmetric linear problem $A\mathbf{x} = \mathbf{b}$. Given an initial guess $\mathbf{x}_0$, using the same notation for space and the corresponding matrix, define

$$K_k = [\mathbf{x}_0, A\mathbf{x}_0, \cdots, A^{k-1}\mathbf{x}_0] = span\{\mathbf{x}_0, A\mathbf{x}_0, \cdots, A^{k-1}\mathbf{x}_0\}$$

$$V_k = \mathbf{x}_0 + K_k.$$

This is an **affine space**.

A natural way to define an approximation from V_k is to let $\mathbf{x}_k$ be the minimizer of the residual

$$\|\mathbf{b} - A\mathbf{x}\|.$$

Equivalently

$$\mathbf{x}_k = argmin_{\mathbf{x}\in V_k}\|\mathbf{b} - A\mathbf{x}\|^2. \tag{6.55}$$

Let $\mathbf{z}_k = \mathbf{x}_k - \mathbf{x}_0$. Then $\mathbf{z}_k \in K_k$ and

$$\begin{aligned} \mathbf{z}_k &= argmin_{\mathbf{z}\in K_k}\|\mathbf{b} - A(\mathbf{z} + \mathbf{x}_0)\|^2 \\ &= argmin_{\mathbf{z}\in K_k}\|\mathbf{b} - A\mathbf{x}_0 - A\mathbf{z}\|^2 \\ &= argmin_{\mathbf{z}\in K_k}\|\mathbf{r}_0 - A\mathbf{z}\|^2 \end{aligned} \tag{6.56}$$

where $\mathbf{r}_0$ is the initial error.

GMRES

We will solve (6.56) for $k = 1, 2, 3, \cdots$. **GMRES**(General minimized residual method). stopping criterion is the relative residual $\frac{\|\mathbf{r}_k\|}{\|\mathbf{b}\|}$. We solve least square problem by QR decomposition. Define the Krylov matrix

$$\Gamma_k := K_k \equiv [\mathbf{b}\,|A\mathbf{b}\,|A^2\mathbf{b}\,|\cdots|A^{k-1}\mathbf{b}].$$

Since every vector in K_k is a linear combination of columns of Γ_k, there exists some vector $\mathbf{y}$ such that

$$\mathbf{x}_k - \mathbf{x}_0 = \Gamma_k\mathbf{y}.$$

In terms of (6.56) this problem becomes finding the solution $\mathbf{y}$ of

$$\mathbf{y} = argmin_{\mathbf{y}\in\mathbb{R}^k}\|\mathbf{r}_0 - A\Gamma_k\mathbf{y}\|^2 = argmin_{\mathbf{y}\in\mathbb{R}^k}\|\mathbf{r}_0 - AQ_k\mathbf{y}\|^2. \tag{6.57}$$

This is a standard minimization problem over $\mathbb{R}^k$ which could be solved by QR decomposition.

Implementation of GMRES

From the Arnoldi algorithm, we set $\mathbf{q}_1 = \mathbf{r}_0/\|\mathbf{r}_0\|$ then $Q_k^T\mathbf{r}_0 = \rho\mathbf{e}_1$. We observe

$$\mathbf{u}_{k+1} = A\mathbf{q}_k - \sum_{j=1}^{k} < \mathbf{q}_j, A\mathbf{q}_k > \mathbf{q}_j.$$

Or

$$A\mathbf{q}_k = \sum_{j=1}^{k} \mathbf{q}_j^T A\mathbf{q}_k\mathbf{q}_j + \mathbf{u}_{k+1}. \tag{6.58}$$

The summation on the right hand side is nothing but a linear combination of columns of $\mathbf{q}_j, (j = 1, 2, \cdots, k)$ with coefficients $h_{jk} = \mathbf{q}_j^T A\mathbf{q}_k$. Hence

$$\sum_{j=1}^{k} \mathbf{q}_j^T A\mathbf{q}_k\mathbf{q}_j = [\mathbf{q}_1\,|\mathbf{q}_2\,|\mathbf{q}_3\,|\cdots|\mathbf{q}_k]Q_k^T A\mathbf{q}_k = Q_kQ_k^T A\mathbf{q}_k.$$

Thus, with $\mathbf{h}_k :=$ k-th column of H_k,

$$A\mathbf{q}_k = Q_kQ_k^T A\mathbf{q}_k + \|\mathbf{u}_{k+1}\|\mathbf{q}_{k+1} = Q_k\mathbf{h}_k + \|\mathbf{u}_{k+1}\|\mathbf{q}_{k+1}. \tag{6.59}$$

Hence

$$\begin{aligned} AQ_k &= Q_kH_k + \|\mathbf{u}_{k+1}\|\mathbf{q}_{k+1}\mathbf{e}_k^T = Q_kH_k + [\mathbf{0}, \cdots, \mathbf{0}, \mathbf{u}_{k+1}] \\ &= [Q_k, \mathbf{q}_{k+1}]\begin{bmatrix} H_k \\ \|\mathbf{u}_{k+1}\|\mathbf{e}_k^T \end{bmatrix} \\ &= Q_{k+1}\tilde{H}_k, \end{aligned}$$

where $Q_{k+1} = [Q_k, \mathbf{q}_{k+1}]$ and $\tilde{H}_k$ is obtained by augmenting $[0, \cdots, 0, h_{k+1,k}], h_{k+1,k} = \|\mathbf{u}_{k+1}\|$ after the last row of H_k. Substituting in (6.57) we get

$$\mathbf{y}_k = argmin_{\mathbf{y}\in\mathbb{R}^k}\|\mathbf{r}_0 - Q_{k+1}\tilde{H}_k\mathbf{y}\|^2.$$

Since the LS solution is solved on the orthogonal projection onto column space, this can be simplified as

$$\begin{aligned} \mathbf{y}_k &= argmin_{\mathbf{y}\in\mathbb{R}^k}\|Q_{k+1}^T(\mathbf{r}_0 - Q_{k+1}\tilde{H}_k\mathbf{y})\|^2 \\ &= argmin_{\mathbf{y}\in\mathbb{R}^k}\|Q_{k+1}^T\mathbf{r}_0 - \tilde{H}_k\mathbf{y}\|^2 \\ &= argmin_{\mathbf{y}\in\mathbb{R}^k}\|\rho\mathbf{e}_1 - \tilde{H}_k\mathbf{y}\|^2. \end{aligned}$$

Here $\mathbf{e} = (1, 0, \cdots, 0) \in \mathbb{R}^{k+1}$, and we used the form

$$Q_{k+1}^T \mathbf{r}_0 = \begin{bmatrix} Q_k^T \\ \mathbf{q}_{k+1}^T \end{bmatrix} \mathbf{r}_0 = \|\mathbf{r}_0\| \mathbf{e}_1 = \rho \mathbf{e}_1.$$

We can use QR decomposition (say, Givens rotation) to solve the least square problem

$$\mathbf{y} = a r g m i n_{\mathbf{y} \in \mathbb{R}^k} \|\rho \mathbf{e}_1 - \tilde{H}_k \mathbf{y}\|^2 \tag{6.60}$$

and set $\mathbf{x}_k = \mathbf{x}_0 + Q_k \mathbf{y}_k$.

GMRES -Algorithm:
Set $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0, \rho = \|\mathbf{r}_0\|, \mathbf{q}_1 = \frac{\mathbf{r}_0}{\rho}$.
For $k = 1, 2, \ldots,$
  For $i = 1, 2, \ldots, k$
    Set $h_{ik} = \mathbf{q}_i^T A \mathbf{q}_k$
  End i-loop
  Set $\mathbf{u}_{k+1} = A\mathbf{q}_k - \sum_{i=1}^k h_{ik} \mathbf{q}_i$.
  Set $h_{k+1,k} = \|\mathbf{u}_{k+1}\|$.
  Set $\mathbf{q}_{k+1} = \frac{\mathbf{u}_{k+1}}{h_{k+1,k}}$.
  Find the minimizer of $\|\rho \mathbf{e}_1 - \tilde{H}_k \mathbf{y}_k\|$ by QR -factorization (Givens)
  If stopping criterion is met, then set $\mathbf{x}_k = \mathbf{x}_0 + Q_k \mathbf{y}_k$.
End k-loop