MA 8019: Numerical Analysis I
Solving Systems of Linear Equations

Suh-Yuh Yang (楊肅煜)

Department of Mathematics, National Central University
Jhongli District, Taoyuan City 320317, Taiwan

First version: May 4, 2018 Last updated: October 22, 2024

A system of linear equations

We are interested in solving systems of linear equations having the form:

$$\left\{ \begin{array}{rcl} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n & = & b_1, \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2n}x_n & = & b_2, \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \cdots + a_{3n}x_n & = & b_3, \\ \vdots & \vdots & \vdots \\ a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \cdots + a_{nn}x_n & = & b_n. \end{array} \right.$$

This is a system of n equations in the n unknowns, $x_1, x_2, \cdots, x_n$. The elements a_{ij} and b_i are assumed to be prescribed real numbers.

$Ax = b$

We can rewrite this system of linear equations in a matrix form:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{bmatrix}.$$

We can denote these matrices by A, x, and b, giving the simpler equation:

$$Ax = b.$$

Matrix

A matrix is a rectangular array of numbers such as

$$\begin{bmatrix} 3.0 & 1.1 & -0.12 \\ 6.2 & 0.0 & 0.15 \\ 0.6 & -4.0 & 1.3 \\ 9.3 & 2.1 & 8.2 \end{bmatrix}, \quad \begin{bmatrix} 3 & 6 & \frac{11}{7} & -17 \end{bmatrix}, \quad \begin{bmatrix} 3.2 \\ -4.7 \\ 0.11 \end{bmatrix}.$$

4×3 matrix

1×4 matrix
a row vector

3×1 matrix
a column vector

Matrix properties

- If A is a matrix, the notation a_{ij}, $(A)_{ij}$, or $A(i,j)$ is used to denote the element at the intersection of the ith row and the jth column. For example, let A be the first matrix on the previous slide. Then $a_{32} = (A)_{32} = A(3,2) = -4.0$.
- The transpose of a matrix is denoted by $A^\top$ and is the matrix defined by $(A^\top)_{ij} = a_{ji}$. The transpose of the matrix A is:

$$A^\top = \begin{bmatrix} 3.0 & 6.2 & 0.6 & 9.3 \\ 1.1 & 0.0 & -4.0 & 2.1 \\ -0.12 & 0.15 & 1.3 & 8.2 \end{bmatrix}.$$

- If $A = A^\top$, we say that matrix A is symmetric.
- The $n \times n$ matrix

$$I := I_n := I_{n\times n} := \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$$

is called an identity matrix. Notice that $IA = A = AI$ for any $n \times n$ matrix A.

Algebraic operations

- *Scalar * Matrix:* If A is a matrix and λ is a scalar, then λA is defined by $(\lambda A)_{ij} = \lambda a_{ij}$.
- *Matrix + Matrix:* If $A = (a_{ij})$ and $B = (b_{ij})$ are $m \times n$ matrices, then $A + B$ is defined by $(A + B)_{ij} = a_{ij} + b_{ij}$.
- *Matrix * Matrix:* If A is an $m \times p$ matrix and B is a $p \times n$ matrix, then AB is an $m \times n$ matrix defined by:

$$(AB)_{ij} = \sum_{k=1}^{p} a_{ik} b_{kj}, \qquad 1 \leq i \leq m,\ 1 \leq j \leq n.$$

What is the cost of AB?

Answer: mnp multiplications and $mn(p-1)$ additions.

Right inverse and left inverse

If A and B are two matrices such that $AB = I$, then we say that B is a right inverse of A and that A is a left inverse of B. For example,

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ \alpha & \beta \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = I_{2\times 2}, \qquad \forall \alpha, \beta \in \mathbb{R}.$$

$$\begin{bmatrix} 1 & 0 & \alpha \\ 0 & 1 & \beta \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = I_{2\times 2}, \qquad \forall \alpha, \beta \in \mathbb{R}.$$

Notice that right inverse and left inverse *may not unique.*

1. **Theorem:** *A square matrix can possess at most one right inverse.*

   Proof: Let $AB = I$. Then $\sum_{j=1}^{n} b_{jk} A^{(j)} = I^{(k)}$, $1 \le k \le n$. So, the columns of A form a basis for $\mathbb{R}^n$. Therefore, the coefficients b_{jk} above are uniquely determined. $\square$

2. **Theorem:** *If A and B are square matrices such that $AB = I$, then $BA = I$.*

   Proof: Let $C = BA - I + B$. Then $AC = ABA - AI + AB = A - A + I = I$.
   Since right inverse for square matrix is at most one, $B = C$.
   Hence, $C = BA - I + B = BA - I + C$, i.e., $BA = I$. $\square$

Inverse

1. If a square matrix A has a right inverse B, then B is unique and $BA = AB = I$. We then call B the inverse of A and say that A is invertible or nonsingular. We denote $B = A^{-1}$.
2. **Example:**

$$\begin{aligned} \begin{bmatrix} -2 & 1 \\ \frac{3}{2} & -\frac{1}{2} \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} &= \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} -2 & 1 \\ \frac{3}{2} & -\frac{1}{2} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = I_{2\times 2}. \end{aligned}$$

3. If A is invertible, then the system of equations $Ax = b$ has the solution $x = A^{-1}b$. If A^{-1} is not available, then in general, A^{-1} should not be computed solely for the purpose of obtaining x.
4. How do we get this A^{-1}?

Equivalent systems

1. Let two linear systems be given, each consisting of n equations with n unknowns:

$$Ax = b \quad \textit{and} \quad Bx = d.$$

If the two systems have precisely the same solutions, we call them equivalent systems.

2. Note that A and B can be very different.

3. Thus, to solve a linear system of equations, we can instead solve any equivalent system. *This simple idea is at the heart of our numerical procedures.*

Elementary operations

1. Let $\mathcal{E}_i$ denote the i-th equation in the system $Ax = b$. The following are the elementary operations which can be performed:
   - Interchanging two equations in the system: $\mathcal{E}_i \leftrightarrow \mathcal{E}_j$;
   - Multiplying an equation by a nonzero number: $\lambda \mathcal{E}_i \to \mathcal{E}_i$;
   - Adding to an equation a multiple of some other equation: $\mathcal{E}_i + \lambda \mathcal{E}_j \to \mathcal{E}_i$.
2. **Theorem on equivalent systems:** *If one system of equations is obtained from another by a finite sequence of elementary operations, then the two systems are equivalent.*

Elementary operations (cont'd)

1. An elementary matrix is defined to be an $n \times n$ matrix that arises when an elementary operation is applied to the $n \times n$ identity matrix.
2. Let A_i be the i-th row of matrix A. The elementary operations expressed in terms of the rows of matrix A are:
   - The interchange of two rows in A: $A_i \leftrightarrow A_j$;
   - Multiplying one row by a nonzero constant: $\lambda A_i \to A_i$;
   - Adding to one row a multiple of another: $A_i + \lambda A_j \to A_i$.
3. *Each elementary row operation on A can be accomplished by multiplying A on the left by an elementary matrix.*

Examples

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{31} & a_{32} & a_{33} \\ a_{21} & a_{22} & a_{23} \end{bmatrix}.$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ \lambda a_{21} & \lambda a_{22} & \lambda a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}.$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \lambda & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ \lambda a_{21} + a_{31} & \lambda a_{22} + a_{32} & \lambda a_{23} + a_{33} \end{bmatrix}.$$

Invertible matrix

1. If matrix A is invertible, then there exists a sequence of elementary row operations can be applied to A, reducing it to I,

$$E_m E_{m-1} \cdots E_2 E_1 A = I.$$

2. This gives us an equation for computing the inverse of a matrix:

$$A^{-1} = E_m E_{m-1} \cdots E_2 E_1 = E_m E_{m-1} \cdots E_2 E_1 I.$$

Remark: This is not a practical method to compute A^{-1}.

Eigenvalue and eigenvector

Definition: *Let $A \in \mathbb{C}^{n \times n}$ be a square matrix. If there exists a nonzero vector $x \in \mathbb{C}^n$ and a scalar $\lambda \in \mathbb{C}$ such that*

$$Ax = \lambda x,$$

then λ is called an eigenvalue of A and x is called the corresponding eigenvector of A.

Remark: Computing λ and x is a major task in numerical linear algebra, see Chapter 5.

Theorem on nonsingular matrix properties

For an $n \times n$ real matrix A, the following properties are equivalent:

1. The inverse of A exists; that is, A is nonsingular
2. The determinant of A is nonzero
3. The rows of A form a basis for $\mathbb{R}^n$
4. The columns of A form a basis for $\mathbb{R}^n$
5. As a map from $\mathbb{R}^n$ to $\mathbb{R}^n$, A is injective (one to one)
6. As a map from $\mathbb{R}^n$ to $\mathbb{R}^n$, A is surjective (onto)
7. The equation $Ax = 0$ implies $x = 0$
8. For each $b \in \mathbb{R}^n$, there is exactly one $x \in \mathbb{R}^n$ such that $Ax = b$
9. A is a product of elementary matrices
10. 0 is not an eigenvalue of A

Note: *We can view an $n \times n$ real matrix A as a linear transformation $A : \mathbb{R}^n \to \mathbb{R}^n$. Then by the rank-nullity theorem, we have*

$$\dim(\text{kernel}(A)) + \dim(\text{image}(A)) = \dim(\mathbb{R}^n) = n.$$

Positive definiteness (review)

- Let $A \in \mathbb{C}^{n\times n}$ be a square matrix and $x, y \in \mathbb{C}^n$. Define $x^* := \overline{x}^\top$, $(x,y) := y^*x \in \mathbb{C}$. *Then* $(Ax, x) = x^*Ax$ *is called a quadratic form.*
- **Definition:** Let $A \in \mathbb{C}^{n\times n}$.

$$A \text{ is positive definite } \iff (Ax, x) > 0, \quad \forall\, 0 \neq x \in \mathbb{C}^n.$$

- **Note 1:** $A = A^* (:= \overline{A}^\top) \iff (Ax, x) \in \mathbb{R}, \forall\, x \in \mathbb{C}^n$.
- **Note 2:** If $A \in \mathbb{C}^{n\times n}$ is positive definite, then $A = A^*$. (by Note 1)
- **Note 3:** Let $A \in \mathbb{R}^{n\times n}$. A is positive definite $\iff A = A^\top$ and $(Ax, x) > 0, \forall\, 0 \neq x \in \mathbb{R}^n$.

  Proof: $(\Rightarrow)$ Trivial!

  $(\Leftarrow)$ Let $0 \neq x := x_1 + ix_2 \in \mathbb{C}^n$. Then $x_1 \neq 0$ or $x_2 \neq 0$.

  $$\therefore (A(x_1 + ix_2), (x_1 + ix_2)) = (Ax_1, x_1) - i(Ax_1, x_2) + i(Ax_2, x_1) + (Ax_2, x_2)$$
  $$\because -i(Ax_1, x_2) = -i(x_1, A^*x_2) = -i(x_1, A^\top x_2) = -i(x_1, Ax_2) = -i(Ax_2, x_1)$$
  $$\therefore (A(x_1 + ix_2), (x_1 + ix_2)) = (Ax_1, x_1) + (Ax_2, x_2) > 0$$

- **Note 4:** Let $A \in \mathbb{C}^{n\times n}$ and $A = A^*$. Then A is positive definite $\iff$ all of its eigenvalues are real and positive.

Proof of Note 1

$(\Rightarrow) \because\ (Ax,x) = x^*Ax = (Ax)^*x = (x,Ax) = \overline{(Ax,x)}, \forall\, x \in \mathbb{C}^n$
$\therefore\ (Ax,x) \in \mathbb{R}, \forall\, x \in \mathbb{C}^n$

$(\Leftarrow)$ $\forall\, x, y \in \mathbb{C}^n$, we have
$\mathbb{R} \ni (x+y)^*A(x+y) = x^*Ax + y^*Ay + x^*Ay + y^*Ax.$
$\therefore\ x^*Ay + y^*Ax \in \mathbb{R}$

- Let $x = e_j \in \mathbb{R}^n, y = e_k \in \mathbb{R}^n$. Then $\mathbb{R} \ni x^*Ay + y^*Ax = a_{jk} + a_{kj}$

  $\therefore\ Im(a_{jk}) = -Im(a_{kj})$

  $\therefore\ a_{jk} := a + bi$ and $a_{kj} := c - bi$ for some $a, b, c \in \mathbb{R}$
- Let $x = ie_j \in \mathbb{C}^n, y = e_k \in \mathbb{R}^n$. Then

  $$\mathbb{R} \ni x^*Ay + y^*Ax = -ia_{jk} + ia_{kj} = (-ia+b) + (ci+b) = (c-a)i + 2b.$$

  $\therefore\ c = a$. Then $a_{jk} := a + bi = \overline{a - bi} = \overline{a_{kj}}$

  $\therefore\ A = \overline{A}^\top = A^*$

Example

The following 2×2 real matrix

$$\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

is positive definite since $A = A^\top$ and

$$x^\top A x = [x_1, x_2] \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = (x_1 + x_2)^2 + x_1^2 + x_2^2 > 0,$$

$$\forall\, 0 \neq (x_1, x_2)^\top \in \mathbb{R}^2.$$

Partitioned matrices

Let A, B, C be matrices that have been partitioned into submatrices:

$$A = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{bmatrix}, \quad B = \begin{bmatrix} B_{11} & B_{12} & \cdots & B_{1k} \\ B_{21} & B_{22} & \cdots & B_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ B_{n1} & B_{n2} & \cdots & B_{nk} \end{bmatrix},$$

$$C = \begin{bmatrix} C_{11} & C_{12} & \cdots & C_{1k} \\ C_{21} & C_{22} & \cdots & C_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ C_{m1} & C_{m2} & \cdots & C_{mk} \end{bmatrix}.$$

If each product $A_{is}B_{sj}$ can be formed and $C_{ij} = \sum_{s=1}^{n} A_{is}B_{sj}$, then $C = AB$.

(see pp.146-147 for the proof)

Partitioned matrices - an example

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

$$= \begin{bmatrix} \begin{bmatrix} 1 & 2 & 7 \end{bmatrix} & \begin{bmatrix} 2 & 5 \end{bmatrix} \\ \begin{bmatrix} -3 & 1 & 3 \\ -3 & 3 & 2 \\ 4 & 0 & -1 \\ 2 & 3 & 2 \end{bmatrix} & \begin{bmatrix} 0 & 2 \\ -2 & 1 \\ 0 & 1 \\ 1 & 6 \end{bmatrix} \end{bmatrix}.$$

Some easy-to-solve systems

Diagonal Structure:

We consider
$$\begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ 0 & a_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}.$$

The solution is: (provided $a_{ii} \neq 0$ for all $i = 1, 2, \cdots, n$)

$$x = \left(\frac{b_1}{a_{11}}, \frac{b_2}{a_{22}}, \cdots, \frac{b_n}{a_{nn}} \right)^\top.$$

- If $a_{ii} = 0$ for some index i, and if $b_i = 0$ also, then x_i can be any real number. The number of solutions is infinity.
- If $a_{ii} = 0$ and $b_i \neq 0$, no solution of the system exists.
- What is the complexity of the method? n divisions.

Lower triangular systems

We consider
$$\begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ a_{21} & a_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}.$$

- If $a_{11} \neq 0$, then we have $x_1 = b_1/a_{11}$. Once we have x_1, we can simplify the second equation, $x_2 = (b_2 - a_{21}x_1)/a_{22}$, provided that $a_{22} \neq 0$. Similarly, we can continue this process.
- In general, to find the solution to this system, we use forward substitution (assume that $a_{ii} \neq 0$ for all i):

input n, (a_{ij}), $b = (b_1, b_2, \cdots, b_n)^\top$
for $i = 1$ **to** n **do**

$$x_i \leftarrow \left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j\right)/a_{ii}$$

end do
output $x = (x_1, x_2, \cdots, x_n)^\top$

Lower triangular systems (continued)

- Complexity of forward substitution:
  - n divisions; n subtractions;
  - the number of multiplications: 0 for x_1, 1 for x_2, 2 for x_3, $\cdots$ $0 + 1 + 2 + \cdots + (n-1) \approx 1 + 2 + \cdots + n = (n+1)n/2$, $\therefore$ total = $O(n^2)$.
  - the number of additions: same as multiplications $= O(n^2)$.
- The complexity of an algorithm is often measured using the unit called flop:

$$\textit{one flop} = \textit{one addition} + \textit{one multiplication}.$$

- Forward substitution is an $O(n^2)$ algorithm.
- **Remark:** forward substitution is a sequential algorithm (not parallel at all).

Upper triangular systems

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ 0 & a_{22} & a_{23} & \cdots & a_{2n} \\ 0 & 0 & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{bmatrix}.$$

The formal algorithm to solve for x is called backward substitution. It is also an $O(n^2)$ algorithm. Assume that $a_{ii} \neq 0$ for all i:

input n, (a_{ij}), $b = (b_1, b_2, \cdots, b_n)^\top$
for $i = n : -1 : 1$ **do**

$$x_i \leftarrow \left(b_i - \sum_{j=i+1}^{n} a_{ij} x_j \right) / a_{ii}$$

end do
output $x = (x_1, x_2, \cdots, x_n)^\top$

Another simple systems

For example, consider the following linear system:

$$\begin{bmatrix} a_{11} & a_{12} & 0 \\ a_{21} & a_{22} & a_{23} \\ a_{31} & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}.$$

If we reorder these equations, we can get a lower triangular system:

$$\begin{bmatrix} a_{31} & 0 & 0 \\ a_{11} & a_{12} & 0 \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_3 \\ b_1 \\ b_2 \end{bmatrix}.$$

Another Simple Systems (continued)

How do we solve $Ax = b$ if A is a permuted lower or upper triangular matrix?

Assuming that the permutation vector $(p_1, p_2, \cdots, p_n)$ is known, we modify the forward substitution algorithm for a permuted lower triangular system:

input n, (a_{ij}), $b = (b_1, b_2, \cdots, b_n)^\top$, $(p_1, p_2, \cdots, p_n)$
for $i = 1$ **to** n **do**

$$x_i \leftarrow \left(b_{p_i} - \sum_{j=1}^{i-1} a_{p_i j} x_j \right) / a_{p_i i}$$

end do
output $x = (x_1, x_2, \cdots, x_n)^\top$

LU decomposition (factorization)

- Suppose that A can be factored into the product of a lower triangular matrix L and an upper triangular matrix U:

$$A = LU.$$

- Then,

$$Ax = LUx = L(Ux).$$

Thus, to solve the system of equations $Ax = b$, it is enough to solve this problem in two stages:

$$\begin{aligned} Lz &= b \quad \textit{solve for } z, \\ Ux &= z \quad \textit{solve for } x. \end{aligned}$$

LU decomposition (continued)

- We begin with an $n \times n$ matrix A and search for matrices:

$$L = \begin{bmatrix} \ell_{11} & 0 & \cdots & 0 \\ \ell_{21} & \ell_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \ell_{n1} & \ell_{n2} & \cdots & \ell_{nn} \end{bmatrix}, \; U = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_{nn} \end{bmatrix}$$

  such that $A = LU$. When this is possible, we say that A has an LU decomposition.

- It turns out if we compare $A = LU$, we have more unknowns $n^2 + n$ than equations n^2. Hence, L and U are not uniquely determined by $A = LU$.

- One simple choice is to make L unit lower triangular ($\ell_{ii} = 1$ for each i). Another obvious choice is to make U unit upper triangular ($u_{ii} = 1$ for each i).

LU decomposition (continued)

Using the formula for matrix multiplication, we have

$$a_{ij} = \sum_{s=1}^{n} \ell_{is} u_{sj} = \sum_{s=1}^{\min(i,j)} \ell_{is} u_{sj}. \qquad (*)$$

Notice that $\ell_{is} = 0$ for $s > i$ and $u_{sj} = 0$ for $s > j$. At each new step k, we know rows $1, 2, \cdots, (k-1)$ for U and columns $1, 2, \cdots, (k-1)$ for L. We wish to know formulas at k by setting $i = j = k$, $i = k$, and $j = k$ in (*), respectively. We obtain

$$a_{kk} = \sum_{s=1}^{k-1} \ell_{ks} u_{sk} + \ell_{kk} u_{kk}, \text{ specify } \ell_{kk} = 1 \text{ or } u_{kk} = 1 \Rightarrow \text{obtain } \ell_{kk} \text{ and } u_{kk}$$

$$a_{kj} = \sum_{s=1}^{k-1} \ell_{ks} u_{sj} + \ell_{kk} u_{kj}, \ k+1 \le j \le n \Rightarrow \text{ obtain } u_{kj}$$

$$a_{ik} = \sum_{s=1}^{k-1} \ell_{is} u_{sk} + \ell_{ik} u_{kk}, \ k+1 \le i \le n \Rightarrow \text{ obtain } \ell_{ik}$$

Note: ℓ_{kk} and $u_{kk} \Longrightarrow u_{kj}$ for $j = k+1, k+2, \cdots, n$ (kth row of U)
$\Longrightarrow \ell_{ik}$ for $i = k+1, k+2, \cdots, n$ (kth column of L)

LU decomposition (continued)

- This algorithm is known as Doolittle's decomposition when L is a unit lower triangular and as Crout's decomposition when U is a unit upper triangular.
- When $U = L^\top$, so that $\ell_{ii} = u_{ii}$ for $1 \le i \le n$, the algorithm is called Cholesky's decomposition (will be discussed later).
- **Homework:** find the Doolittle, Crout, and Cholesky decompositions of the matrix

$$A = \begin{bmatrix} 1 & 2 \\ 2 & 7 \end{bmatrix}.$$

LU decomposition (continued)

The algorithm for the general LU decomposition is as follows:

input $n, (a_{ij})$
for $k = 1$ **to** n **do**
  specify a nonzero value for either
    ℓ_{kk} or u_{kk} and compute the other from
    $\ell_{kk}u_{kk} = a_{kk} - \sum_{s=1}^{k-1} \ell_{ks}u_{sk}$
  for $j = k+1$ **to** n **do**
    $u_{kj} \leftarrow \left(a_{kj} - \sum_{s=1}^{k-1} \ell_{ks}u_{sj}\right) / \ell_{kk}$
  end do
  for $i = k+1$ **to** n **do**
    $\ell_{ik} \leftarrow \left(a_{ik} - \sum_{s=1}^{k-1} \ell_{is}u_{sk}\right) / u_{kk}$
  end do
end do
output $(\ell_{ij}), (u_{ij})$

Operation counts (cf. the algorithm)

- Consider the number of multiplications ($\approx$ additions),

$$k=1: \quad 0+((n-1)*0)*2,$$

$$k=2: \quad 1+((n-2)*1)*2,$$

$$k=i: \quad (i-1)+((n-i)*(i-1))*2, \quad \cdots$$

$$k=n: \quad (n-1)+((n-n)*(n-1))*2.$$

$$\begin{aligned} Total &= \sum_{i=1}^{n}(i-1)+2\sum_{i=1}^{n}(n-i)*(i-1) \approx \sum_{i=1}^{n} i + 2\sum_{i=1}^{n}(n-i)*i \\ &= \sum_{i=1}^{n} i + 2n\sum_{i=1}^{n} i - 2\sum_{i=1}^{n} i^2 = (2n+1)\sum_{i=1}^{n} i - 2\sum_{i=1}^{n} i^2 \\ &= (2n+1)n(n+1)/2 - 2n(n+1)(2n+1)/6 \\ &= \frac{1}{6}n(n+1)(2n+1) = O(\frac{1}{3}n^3). \end{aligned}$$

- The number of subtractions = the number of divisions = $n+2(1+2+\cdots+(n-1)) \approx 2(1+2+\cdots+n) = O(n^2)$.

Basic steps for solving a linear system

- Want to solve
$$Ax = b.$$
- Obtain a LU decomposition,
$$A = LU.$$
- Solve a lower triangular system
$$Lz = b.$$
- Solve an upper triangular system
$$Ux = z.$$

Total cost

- In the LU decomposition phase, the cost is $O(n^3)$.
- In solving triangular systems phases, the cost is $O(n^2)$.
- Total cost is $O(n^3)$ or more precisely

$$O(\frac{1}{3}n^3) + O(n^2).$$

- **Remark:** Once L and U are obtained, A is no longer needed. One can over-write A with L and U.

Theorem on LU decomposition

If all n leading principal submatrices of the $n \times n$ matrix A are nonsingular, then A has an LU-decomposition, where L is unit lower triangular.

Proof is omitted. See the textbook, pp. 156-157 (by induction).

Recall that the kth leading principal submatrix of the matrix A is the matrix:

$$A_k := \begin{bmatrix} a_{11} & a_{22} & \cdots & a_{1k} \\ a_{21} & a_{22} & \cdots & a_{2k} \\ \vdots & \vdots & \cdots & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{kk} \end{bmatrix}.$$

Cholesky Theorem on $LL^\top$ decomposition

If A is a real, symmetric and positive definite matrix, then it has a unique factorization, $A = LL^\top$, in which L is a lower triangular matrix with positive diagonal.

Proof: Some key steps:

- Prove that A has an LU-decomposition (L unit lower triangular) by showing that all leading principal submatrices of A are SPD. ($\because x^\top Ax > 0$ for all $x = (x_1, \cdots x_k, 0, \cdots, 0)^\top \neq 0 \quad \therefore A_k$ is SPD)
- Show that $A = LDL^\top$ by considering $LU = A = A^\top = U^\top L^\top$
$$\Longrightarrow \underbrace{U(L^\top)^{-1}}_{upper\Delta} = \underbrace{L^{-1}U^\top}_{lower\Delta} \text{ (p. 158, \#1)} \Longrightarrow \exists\, D \text{ s.t. } D = U(L^\top)^{-1}$$
$$\Longrightarrow DL^\top = U \Longrightarrow A = LU = LDL^\top.$$
- $\because\ A = LU = LDL^\top$ and L is nonsingular
$\therefore D$ is SPD (cf. p. 160, #26) $\quad \therefore\ d_{ii} > 0$ for all i
$\therefore\ A = LDL^\top = LD^{\frac{1}{2}}D^{\frac{1}{2}}L^\top := \widetilde{L}\widetilde{L}^\top, \widetilde{\ell}_{ii} = \ell_{ii}\sqrt{d_{ii}} = \sqrt{d_{ii}} > 0\ \forall\ i$
- uniqueness (p. 158, #2, L and U are unique $\Rightarrow \widetilde{L}$ unique).

Cholesky decomposition for SPD matrices

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} \ell_{11} & & & \\ \ell_{21} & \ell_{22} & & \\ \vdots & \vdots & \ddots & \\ \ell_{n1} & \ell_{n2} & \cdots & \ell_{nn} \end{bmatrix} \begin{bmatrix} \ell_{11} & \ell_{21} & \cdots & \ell_{n1} \\ & \ell_{22} & \cdots & \ell_{n2} \\ & & \ddots & \vdots \\ & & & \ell_{nn} \end{bmatrix}$$

- $\ell_{kk} \neq 1$ in general.
- Need a square root to compute the diagonal entry:

$$\ell_{kk} = \left(a_{kk} - \sum_{s=1}^{k-1} \ell_{ks}^2 \right)^{1/2}.$$

- Cost = $O(n^3) + O(n^2)$ + “n square roots.”

Some remarks

- If A is SPD, then all the leading principal submatrices of A are also SPD.
- Since $\ell_{kk} = \left(a_{kk} - \sum_{s=1}^{k-1} \ell_{ks}^2\right)^{1/2}$, we have for $j \leq k$

$$a_{kk} = \sum_{s=1}^{k} \ell_{ks}^2 \geq \ell_{kj}^2$$

and

$$|\ell_{kj}| \leq \sqrt{a_{kk}} \qquad (1 \leq j \leq k).$$

Hence, the elements of L do not become large relative to A even without any pivoting (pivoting will be explained later).

$LDL^\top$ decomposition for SPD matrices

$$A = \begin{bmatrix} 1 & & & \\ \ell_{21} & 1 & & \\ \vdots & \vdots & \vdots & \\ \ell_{n1} & \ell_{n2} & \cdots & 1 \end{bmatrix} \begin{bmatrix} d_{11} & & & \\ & d_{22} & & \\ & & & \\ & & & d_{nn} \end{bmatrix} \begin{bmatrix} 1 & \ell_{21} & \cdots & \ell_{n1} \\ & 1 & \cdots & \ell_{n2} \\ & & \vdots & \vdots \\ & & & 1 \end{bmatrix}.$$

No need to compute square roots.

If $A = LDL^\top$, then solve $Ax = b$ in three stages: $Lz = b$, $Dw = z$, and $L^\top x = w$.

How to get $A = LDL^\top$? e.g.,

A is tridiagonal & SPD. (why SPD? cf. proof of Cholesky Theorem)

Banded matrices

- $A = (a_{ij})$ with upper bandwidth q and lower bandwidth p:
  $a_{ij} = 0$ if $j > i + q$,
  $a_{ij} = 0$ if $i < j + p$.
- total bandwidth = $p + q + 1$.
- **Theorem:** *If A has an LU decomposition then U has an upper bandwidth q and L has a lower bandwidth p (L is unit lower triangular).*
- **Remark:** Both L and U can be stored in A.

Banded matrices (continued)

- **Cost:** If $p \le q$,

$$npq - 1/2pq^2 - 1/6p^3 + pn.$$

- **Remark:** If p and q are much smaller than n, then the algorithm is linear in n.

- **Remark:** If A is banded and SPD, then the cost of Cholesky decomposition is

$$1/2np^2 + p^3 + 3/2(np - p^2) + n \textit{ square roots}$$

In the case when p is small, the square root calculation can be a significant part of the decomposition. $LDL^\top$ is preferred!

Tridiagonal & SPD matrices

Find the $LDL^\top$ decomposition of a tridiagonal SPD matrix A:

$$A = \begin{bmatrix} a_{11} & a_{21} & & \\ a_{21} & a_{22} & a_{23} & \\ & \ddots & \ddots & \ddots \\ & & a_{n,n-1} & a_{nn} \end{bmatrix}.$$

Suppose that

$$A = \begin{bmatrix} 1 & & & \\ e_1 & 1 & & \\ & \ddots & \ddots & \\ & & e_{n-1} & 1 \end{bmatrix} \begin{bmatrix} d_1 & & & \\ & d_2 & & \\ & & \ddots & \\ & & & d_n \end{bmatrix} \begin{bmatrix} 1 & e_1 & & \\ & 1 & e_2 & \\ & & \ddots & \ddots \\ & & & 1 \end{bmatrix}.$$

Tridiagonal & SPD matrices (continued)

Then we have

$$
\begin{aligned}
A &= \begin{bmatrix} 1 & & & \\ e_1 & 1 & & \\ & \ddots & \ddots & \\ & & e_{n-1} & 1 \end{bmatrix} \begin{bmatrix} d_1 & d_1 e_1 & & \\ & d_2 & d_2 e_2 & \\ & & \ddots & \ddots \\ & & & d_n \end{bmatrix} \\
&= \begin{bmatrix} d_1 & d_1 e_1 & & \\ e_1 d_1 & d_2 + d_1 e_1^2 & d_2 e_2 & \\ & & \ddots & \ddots \\ & & \ddots & d_n + d_{n-1} e_{n-1}^2 \end{bmatrix}.
\end{aligned}
$$

Tridiagonal & SPD matrices (continued)

- Comparing with the elements in A, we obtain:

  $a_{11} = d_1.$

  $a_{kk-1} = e_{k-1} d_{k-1}.$

  $a_{kk} = d_k + d_{k-1} e_{k-1}^2.$

- A simple observation:

  $a_{kk} = d_k + d_{k-1} e_{k-1}^2 = d_k + (d_{k-1} e_{k-1}) e_{k-1} = d_k + a_{kk-1} e_{k-1}.$

- **Algorithm:**

  $d_1 = a_{11}.$

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

  $\qquad e_{k-1} = a_{kk-1} / d_{k-1}.$

  $\qquad d_k = a_{kk} - e_{k-1} a_{kk-1}.$

  end do

- Total cost $\approx n$ multiplications + n divisions + n subtractions.

Tridiagonal & SPD matrices (continued)

- Solving a tridiagonal & SPD system:
  - step 1: obtain the $LDL^\top$ decomposition ($\approx 2n$ flops).
  - step 2: solve the lower triangular system (n flops).
  - step 3: solve the diagonal system (n divisions $\approx n$ flops).
  - step 4: solve the upper triangular system (n flops).
- Total cost $\approx 5n$ flops.

Basic Gaussian elimination

Let $A^{(1)} = (a_{ij}^{(1)}) = A = (a_{ij})$ and $b^{(1)} = b$. Consider the following linear system $Ax = b$:

$$\begin{bmatrix} 6 & -2 & 2 & 4 \\ 12 & -8 & 6 & 10 \\ 3 & -13 & 9 & 3 \\ -6 & 4 & 1 & -18 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 12 \\ 34 \\ 27 \\ -38 \end{bmatrix}.$$

pivot row = row1.

pivot element: $a_{11}^{(1)} = 6$.

row2 - (12/6)*row1 $\to$ row2.

row3 - (3/6)*row1 $\to$ row3.

row4 - (-6/6)*row1 $\to$ row4.

$$\Longrightarrow \begin{bmatrix} 6 & -2 & 2 & 4 \\ 0 & -4 & 2 & 2 \\ 0 & -12 & 8 & 1 \\ 0 & 2 & 3 & -14 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 12 \\ 10 \\ 21 \\ -26 \end{bmatrix}.$$

multipliers: $12/6, 3/6, -6/6$.

Basic Gaussian elimination (continued)

We have the following equivalent system $A^{(2)}x = b^{(2)}$:

$$\begin{bmatrix} 6 & -2 & 2 & 4 \\ 0 & -4 & 2 & 2 \\ 0 & -12 & 8 & 1 \\ 0 & 2 & 3 & -14 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 12 \\ 10 \\ 21 \\ -26 \end{bmatrix}.$$

pivot row = row2.
pivot element $a_{22}^{(2)} = -4$.
row3 - (-12/-4)*row2 $\rightarrow$ row3.
row4 - (2/-4)*row2 $\rightarrow$ row4.

$$\Longrightarrow \begin{bmatrix} 6 & -2 & 2 & 4 \\ 0 & -4 & 2 & 2 \\ 0 & 0 & 2 & -5 \\ 0 & 0 & 4 & -13 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 12 \\ 10 \\ -9 \\ -21 \end{bmatrix}.$$

multiplier: $-12/-4, 2/-4$.

Basic Gaussian elimination (continued)

We have the following equivalent system $A^{(3)}x = b^{(3)}$:

$$\begin{bmatrix} 6 & -2 & 2 & 4 \\ 0 & -4 & 2 & 2 \\ 0 & 0 & 2 & -5 \\ 0 & 0 & 4 & -13 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 12 \\ 10 \\ -9 \\ -21 \end{bmatrix}.$$

pivot row = row3.
pivot element $a_{33}^{(3)} = 2$.
row4 - (4/2)*row3 $\rightarrow$ row4.

$$\Longrightarrow \begin{bmatrix} 6 & -2 & 2 & 4 \\ 0 & -4 & 2 & 2 \\ 0 & 0 & 2 & -5 \\ 0 & 0 & 0 & -3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 12 \\ 10 \\ -9 \\ -3 \end{bmatrix}.$$

multiplier: 4/2.

Basic Gaussian elimination (continued)

Finally, we have the following equivalent upper triangular system $A^{(4)}x = b^{(4)}$:

$$\begin{bmatrix} 6 & -2 & 2 & 4 \\ 0 & -4 & 2 & 2 \\ 0 & 0 & 2 & -5 \\ 0 & 0 & 0 & -3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 12 \\ 10 \\ -9 \\ -3 \end{bmatrix}.$$

Using the backward substitution, we have

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 1 \\ -3 \\ -2 \\ 1 \end{bmatrix}.$$

The LU decomposition

Display the multipliers in an unit lower triangular matrix $L = (\ell_{ij})$:

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ \frac{1}{2} & 3 & 1 & 0 \\ -1 & -\frac{1}{2} & 2 & 1 \end{bmatrix}.$$

Let $U = (u_{ij})$ be the final upper triangular matrix $A^{(4)}$. Then we have

$$U = \begin{bmatrix} 6 & -2 & 2 & 4 \\ 0 & -4 & 2 & 2 \\ 0 & 0 & 2 & -5 \\ 0 & 0 & 0 & -3 \end{bmatrix}$$

and one can check that $A = LU$ (the Doolittle Decomposition).

Some remarks

- The entire elimination process will break down if any of the pivot elements are 0.
- The total number of arithmetic operations:

$$M/D = \frac{n^3}{3} + n^2 - \frac{n}{3};$$

$$A/S = \frac{n^3}{3} + \frac{n^2}{2} - \frac{5n}{6}.$$

$\therefore$ The GE is an $O(n^3)$ algorithm.

Pivoting

For example, the above technique doesn't work if we have

$$\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

and works incorrectly if we have ($\varepsilon > 0$ is sufficiently small)

$$\begin{bmatrix} \varepsilon & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}.$$

Using the above case as an example: row2 - $(1/\varepsilon)$*row1 $\rightarrow$ row2, we have

$$\begin{bmatrix} \varepsilon & 1 \\ 0 & 1 - 1/\varepsilon \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 - 1/\varepsilon \end{bmatrix}.$$

Example

$$\begin{bmatrix} \varepsilon & 1 \\ 0 & 1-1/\varepsilon \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 2-1/\varepsilon \end{bmatrix}.$$

- Using the backward substitution, we have

$$x_2 = \frac{2-1/\varepsilon}{1-1/\varepsilon}, \qquad x_1 = \frac{1-x_2}{\varepsilon}.$$

  If we let $0 < \varepsilon \ll 1$, then $(1/\varepsilon) \gg 1$, and then x_2 goes to 1 and x_1 goes to 0.

- However, the exact solution should be close to $x_1 = 1$ and $x_2 = 1$.

  What's wrong?

Example (continued)

- Maybe that is because the pivot element $a_{11} = \varepsilon$ is too small. So we multiply row1 by $1/\varepsilon$ before perform GE.

$$\begin{bmatrix} 1 & 1/\varepsilon \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1/\varepsilon \\ 2 \end{bmatrix}.$$

- However, it does not help too much since

$$x_2 = \frac{2 - 1/\varepsilon}{1 - 1/\varepsilon} \approx 1, \qquad x_1 = \frac{1}{\varepsilon} - \frac{x_2}{\varepsilon} \approx 0.$$

- In fact, it is not actually the smallness of the coefficient a_{11} that is causing trouble. Rather, it is the smallness of a_{11} relative to the other elements in its row.

Example (continued)

- An equivalent linear system: exchanging equations 1 and 2, we have

$$\begin{bmatrix} 1 & 1 \\ \varepsilon & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}.$$

- Using the same algorithm, we obtain $x_2 = (1 - 2\varepsilon)/(1 - \varepsilon)$, which is close to 1 and $x_1 = 2 - x_2$ is also close to 1.

Partial pivoting and complete pivoting

- **GE with partial pivoting:** select the largest element (in $|\cdot|$) in the column as the pivot element ($\Longrightarrow$ exchange rows).
- **GE with complete pivoting:** select the largest element (in $|\cdot|$) in the whole matrix as the pivot element ($\Longrightarrow$ exchange rows and columns).
- After the first round of elimination, we obtain an $(n-1)\times(n-1)$ linear system to solve. The same idea is used for this subsystem, and so on.

Gaussian elimination with scaled row pivoting

- The algorithm consists of two parts:
  - a factorization phase (also called forward elimination);
  - a solution phase (involving updating and backward substitution).
- In a factorization phase, first compute the scale of each row

$$s_i = \max_{1 \le j \le n} |a_{ij}| = \max\{|a_{i1}|, |a_{i2}|, \cdots, |a_{in}|\}.$$

  Do it for $1 \le i \le n$.

- To get started, we choose the pivot row for which $|a_{i1}|/s_i$ is largest. The index p_1 is associated to the index i, where $|a_{p_1 1}|/s_{p_1} \ge |a_{i1}|/s_i$ for $1 \le i \le n$.
- Zeros are created by subtracting multiples of row p_1 and so on (see next example).
- The permutation vector $(1, 2, \cdots, n) \Longrightarrow (p_1, p_2, \cdots, p_n)$ and we obtain a permutation matrix P according to the permutation vector $(p_1, p_2, \cdots, p_n)$.

Example

$$A = \begin{bmatrix} 2 & 3 & -6 \\ 1 & -6 & 8 \\ 3 & -2 & 1 \end{bmatrix}.$$

- First compute the scales $s = (6, 8, 3)$ and initialize $p = (p_1, p_2, p_3) = (1, 2, 3)$.
- Select the first pivot row from ratios, $\{2/6, 1/8, 3/3\}$. Since 3th row has the largest ratio, the row3 is selected to be the first pivot. Change the permutation vector by $p_1 \leftrightarrow p_3$ and then $p = (p_1, p_2, p_3) = (3, 2, 1)$.
- Perform row1$-(2/3)$row3 and row2$-(1/3)$row3, we have

$$\begin{bmatrix} 0 & 13/3 & -20/3 \\ 0 & -16/3 & 23/3 \\ 3 & -2 & 1 \end{bmatrix}.$$

Example (continued)

- From the previous page, $s = (6, 8, 3)$, $p = (p_1, p_2, p_3) = (3, 2, 1)$,

$$\begin{bmatrix} 0 & 13/3 & -20/3 \\ 0 & -16/3 & 23/3 \\ 3 & -2 & 1 \end{bmatrix}.$$

- Select the next pivot row from ratios, $\{\frac{16/3}{8}, \frac{13/3}{6}\} = \{2/3, 13/18\}$. Since $p_3(= 1)$th row has the largest ratio, the rowp_3 (row1) is selected to be the pivot row and $p_2 \leftrightarrow p_3$. Then $p = (p_1, p_2, p_3) = (3, 1, 2)$.
- Perform row2$-(-16/13)$row1 to obtain

$$\begin{bmatrix} 0 & 13/3 & -20/3 \\ 0 & 0 & -7/13 \\ 3 & -2 & 1 \end{bmatrix}.$$

Example (continued)

At the end, we have a decomposition for $PA = LU$, where

$$PA = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 2 & 3 & -6 \\ 1 & -6 & 8 \\ 3 & -2 & 1 \end{bmatrix},$$

$$LU = \begin{bmatrix} 1 & 0 & 0 \\ 2/3 & 1 & 0 \\ 1/3 & -16/13 & 1 \end{bmatrix} \begin{bmatrix} 3 & -2 & 1 \\ 0 & 13/3 & -20/3 \\ 0 & 0 & -7/13 \end{bmatrix}.$$

$\because Ax = b. \qquad \therefore PAx = Pb.$

In the solution phase, we consider two equations: $Lz = Pb$ and $Ux = z$.

$Pb \to b \Longrightarrow$ solve $Lz = b \Longrightarrow z \to b \Longrightarrow$ solve $Ux = b$.
This procedure is called updating b.

Vector norm

Let V be a vector space over $\mathbb{R}$, e.g., $V = \mathbb{R}^n$. A norm is a real-valued function $\|\cdot\| : V \to \mathbb{R}$ that satisfies

- $\|x\| \geq 0, \forall\, x \in V$, and $\|x\| = 0$ if and only if $x = 0$;
- $\|\lambda x\| = |\lambda| \|x\|, \forall\, x \in V$ and $\lambda \in \mathbb{R}$;
- $\|x + y\| \leq \|x\| + \|y\|, \forall\, x, y \in V$ (triangle inequality).

Note: $\|x\|$ is called the norm of x, the length or magnitude of x.

Some vector norms on $\mathbb{R}^n$

Let $x = (x_1, x_2, \cdots, x_n)^\top \in \mathbb{R}^n$:

- The 2-norm (Euclidean norm, or ℓ^2 norm):

$$\|x\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2}.$$

- The infinity norm (ℓ^∞-norm):

$$\|x\|_\infty = \max_{1 \le i \le n} |x_i|.$$

- The 1-norm (ℓ^1-norm):

$$\|x\|_1 = \sum_{i=1}^{n} |x_i|.$$

The difference between the above norms

- Take three vectors $x = (4, 4, -4, 4)^\top$, $v = (0, 5, 5, 5)^\top$, $w = (6, 0, 0, 0)^\top$:

	$\|\cdot\|_1$	$\|\cdot\|_2$	$\|\cdot\|_\infty$
x	16	8	4
v	15	8.66	5
w	6	6	6

- What is the unit ball $\{x \in \mathbb{R}^2 : \|x\| \leq 1\}$ for the three norms above?
    - 2-norm: a circle
    - ∞-norm: a square
    - 1-norm: a diamond

Matrix norm

Let A be an $n \times n$ real matrix. If $\|\cdot\|$ is any norm on $\mathbb{R}^n$, then

$$\|A\| := \sup\{\|Ax\| : x \in \mathbb{R}^n, \|x\| = 1\} \Leftrightarrow \|A\| := \sup\{\frac{\|Ax\|}{\|x\|} : x \in \mathbb{R}^n, x \neq 0\}$$

defines a norm on the vector space of all $n \times n$ real matrices. (This is called the matrix norm associated with the given vector norm)

Proof:

- $\because \|Ax\| \geq 0 \, \forall \, x \in \mathbb{R}^n, \|x\| = 1. \therefore \|A\| \geq 0.$
  Exercise: $\|A\| = 0$ if and only if $A = 0$.
- $\|\lambda A\| = \sup\{\|\lambda Ax\| : \|x\| = 1\} = \sup\{|\lambda|\|Ax\| : \|x\| = 1\}$
  $= |\lambda| \sup\{\|Ax\| : \|x\| = 1\} = |\lambda|\|A\|.$
- $\|A + B\| = \sup\{\|(A + B)x\| : \|x\| = 1\} \leq \sup\{\|Ax\| + \|Bx\| : \|x\| = 1\}$
  $\leq \sup\{\|Ax\| : \|x\| = 1\} + \sup\{\|Bx\| : \|x\| = 1\} = \|A\| + \|B\|.$

Some additional properties

- $\|Ax\| \leq \|A\|\|x\|, \forall\, x \in \mathbb{R}^n$.

  Proof:

  Let $x \neq 0$. Then $v = \dfrac{x}{\|x\|}$ is of norm 1.

  $\therefore \|A\| \geq \|Av\| = \dfrac{\|Ax\|}{\|x\|}$.

- $\|I\| = 1$.
- $\|AB\| \leq \|A\|\|B\|$.

  Proof:

  $\|AB\| := \sup\{\|(AB)x\| : x \in \mathbb{R}^n, \|x\| = 1\}$

  $\leq \sup\{\|A\|\|Bx\| : x \in \mathbb{R}^n, \|x\| = 1\}$

  $\leq \sup\{\|A\|\|B\|\|x\| : x \in \mathbb{R}^n, \|x\| = 1\} = \|A\|\|B\|$.

Some matrix norms

Let $A_{n\times n} = (a_{ij})$ be an $n \times n$ real matrix. Then

- The ∞-matrix norm:

$$\|A\|_\infty = \max_{1\le i\le n} \sum_{j=1}^{n} |a_{ij}|.$$

- The 1-matrix norm:

$$\|A\|_1 = \max_{1\le j\le n} \sum_{i=1}^{n} |a_{ij}|.$$

- The 2-matrix norm (ℓ^2-matrix norm):

$$\|A\|_2 = \sup_{\|x\|_2=1} \|Ax\|_2.$$

The 2-matrix norm

- $\|A\|_2$ is not easy to compute.
- Since $A^\top A$ is symmetric, $A^\top A$ has n real eigenvalues, $\lambda_1, \lambda_2, \cdots, \lambda_n \in \mathbb{R}$. Moreover, one can prove that they are all nonnegative. Then

$$\rho(A^\top A) := \max_{1 \le i \le n} \{\lambda_i\} \ge 0.$$

  is called the spectral radius of $A^\top A$.
- Then the ℓ^2-matrix norm of A is given by

$$\|A\|_2 = \sqrt{\rho(A^\top A)}.$$

- The ℓ^2-matrix norm is also called the spectral norm.

ℓ^2-matrix norm of A

Singular value decomposition (SVD): *Let $A \in \mathbb{R}^{m\times n}$. Then we have*

$$A = U\Sigma V^\top := \begin{bmatrix} u_1 & u_2 & \dots & u_m \end{bmatrix}_{m\times m} \Sigma \begin{bmatrix} v_1 & v_2 & \dots & v_n \end{bmatrix}_{n\times n}^\top,$$

where U and V are orthogonal matrices,

$$UU^\top = U^\top U = I_{m\times m}, \quad VV^\top = V^\top V = I_{n\times n},$$

$\Sigma = diag(\sigma_1, \dots, \sigma_r, 0 \dots, 0) \in \mathbb{R}^{m\times n}$ *with*

$$\sigma_1 \geq \cdots \geq \sigma_r > 0 = \sigma_{r+1} = \cdots = \sigma_{\min\{m,n\}}$$

is a diagonal matrix of singular values, and $r = \mathrm{rank}(A)$.

Given $x = \sum_{i=1}^{n} \alpha_i v_i \in \mathbb{R}^n$ with $\|x\|_2 = 1$, then $1 = \|x\|_2^2 = \sum_{i=1}^{n} \alpha_i^2$ and

$$Ax = \sum_{i=1}^{n} \alpha_i A v_i = \sum_{i=1}^{r} \alpha_i \sigma_i u_i \;\Rightarrow\; \|Ax\|_2^2 = \sum_{i=1}^{r} \alpha_i^2 \sigma_i^2 \leq \sigma_1^2 \sum_{i=1}^{r} \alpha_i^2 \leq \sigma_1^2.$$

Moreover, we have $\|Av_1\|_2 = \|\sigma_1 u_1\|_2 = \sigma_1$. Therefore,

$$\|A\|_2 := \max_{\|x\|_2=1} \|Ax\|_2 = \sigma_1 = \sqrt{\rho(A^\top A)}.$$

Some error analysis

- Suppose that we want to solve the linear system $Ax = b$, but b is somehow perturbed to $\widetilde{b}$ (this may happen when we convert a real b to a floating-point b).
- Then actual solution would satisfy a slightly different linear system
$$A\widetilde{x} = \widetilde{b}.$$
- **Question**: Is $\widetilde{x}$ very different from the desired solution x of the original system?

  The answer should depend on how good the matrix A is.
- Let $\|\cdot\|$ be a vector norm, we consider two types of errors:
  - absolute error: $\|x - \widetilde{x}\|$?
  - relative error: $\|x - \widetilde{x}\| / \|x\|$?

The absolute error

For the absolute error, we have

$$\|x - \widetilde{x}\| = \|A^{-1}b - A^{-1}\widetilde{b}\| = \|A^{-1}(b - \widetilde{b})\| \leq \|A^{-1}\| \|b - \widetilde{b}\|.$$

Therefore, the absolute error of x depends on two factors: the absolute error of b and the matrix norm of A^{-1}.

The relative error

For the relative error, we have

$$\begin{aligned} \|x - \tilde{x}\| &= \|A^{-1}b - A^{-1}\tilde{b}\| = \|A^{-1}(b - \tilde{b})\| \\ &\le \|A^{-1}\| \|b - \tilde{b}\| = \|A^{-1}\| \|Ax\| \frac{\|b - \tilde{b}\|}{\|b\|} \\ &\le \|A^{-1}\| \|A\| \|x\| \frac{\|b - \tilde{b}\|}{\|b\|}. \end{aligned}$$

That is

$$\frac{\|x - \tilde{x}\|}{\|x\|} \le \|A^{-1}\| \|A\| \, \frac{\|b - \tilde{b}\|}{\|b\|}.$$

Therefore, the relative error of x depends on two factors: the relative error of b and $\|A\| \|A^{-1}\|$.

Condition number

- Therefore, we define a condition number of the matrix A as

$$\kappa(A) := \|A\| \|A^{-1}\|.$$

$\kappa(A)$ measures how good the matrix A is.

- Example: Let $\varepsilon > 0$ and

$$A = \begin{bmatrix} 1 & 1+\varepsilon \\ 1-\varepsilon & 1 \end{bmatrix} \Longrightarrow A^{-1} = \varepsilon^{-2} \begin{bmatrix} 1 & -1-\varepsilon \\ -1+\varepsilon & 1 \end{bmatrix}.$$

Then $\|A\|_\infty = 2+\varepsilon$, $\|A^{-1}\|_\infty = \varepsilon^{-2}(2+\varepsilon)$, and $\kappa(A) = \left(\frac{2+\varepsilon}{\varepsilon}\right)^2 \geq \frac{4}{\varepsilon^2}$.

Condition number (continued)

- For example, if $\varepsilon = 0.01$, then $\kappa(A) \geq 40000$.
- What does this mean?

  It means that the relative error in x can be 40000 times greater than the relative error in b.
- If $\kappa(A)$ is large, we say that A is ill-conditioned, otherwise A is well-conditioned.
- In the ill-conditioned case, the solution is probably very sensitive to the small changes in the right-hand vector b (higher precision in b may be needed).

Another way to measure the error

Consider the linear system $Ax = b$. Let $\tilde{x}$ be a computed solution (an approximation to x).

- Residual vector:
$$r = b - A\tilde{x}.$$
- Error vector:
$$e = x - \tilde{x}.$$
- They satisfy
$$Ae = r.$$
(Proof: $Ae = Ax - A\tilde{x} = b - A\tilde{x} = r$)
- Moreover, we have
$$\frac{1}{\kappa(A)}\frac{\|r\|}{\|b\|} \leq \frac{\|e\|}{\|x\|} \leq \kappa(A)\frac{\|r\|}{\|b\|}.$$
(Theorem on bounds involving condition number)

Proof of the Theorem

$\because Ae = r.$

$\therefore e = A^{-1}r.$

$\therefore \|e\|\,\|b\| = \|A^{-1}r\|\,\|Ax\| \le \|A^{-1}\|\,\|r\|\,\|A\|\,\|x\|.$

$\therefore \dfrac{\|e\|}{\|x\|} \le \kappa(A)\,\dfrac{\|r\|}{\|b\|}.$

On the other hand, we have

$\|r\|\,\|x\| = \|Ae\|\,\|A^{-1}b\| \le \|A\|\,\|e\|\,\|A^{-1}\|\,\|b\|.$

$\therefore \dfrac{1}{\kappa(A)}\,\dfrac{\|r\|}{\|b\|} \le \dfrac{\|e\|}{\|x\|}.$

Concept of convergence in a vector space

- If a vector space V is assigned a norm $\|\cdot\|$, then the pair $(V, \|\cdot\|)$ is a normed linear space.
- Consider a sequence of vectors $v^{(1)}, v^{(2)}, \cdots$ in a normed space $(V, \|\cdot\|)$. Then we say that the given sequence converges to a vector $v \in V$ and write $\lim_{k\to\infty} v^{(k)} = v$ if

$$\lim_{k\to\infty} \|v^{(k)} - v\| = 0.$$

- **Theorem:** *Any two norms $\|\cdot\|_a$ and $\|\cdot\|_b$ on a finite-dimensional vector space V are equivalent, i.e., $\exists\, C_1, C_2 > 0$ such that*

$$C_1\|v\|_b \le \|v\|_a \le C_2\|v\|_b, \quad \forall\, v \in V,$$

which leads to the same concept of convergence.

- **Caution:** This theorem does not apply in infinite-dimensional normed linear spaces. (See Problem 4.5, #20, p. 206)

An example in $\mathbb{R}^4$

- Let

$$v^{(k)} = \begin{bmatrix} 3 - k^{-1} \\ -2 + k^{-1/2} \\ (k+1)k^{-1} \\ e^{-k} \end{bmatrix} \quad \text{and} \quad v = \begin{bmatrix} 3 \\ -2 \\ 1 \\ 0 \end{bmatrix}.$$

Then

$$v^{(k)} - v = \begin{bmatrix} -k^{-1} \\ k^{-1/2} \\ k^{-1} \\ e^{-k} \end{bmatrix}.$$

- Then $\lim_{k\to\infty} \|v^{(k)} - v\|_\infty = 0.$

Neumann series

Theorem on Neumann series: *If A is an $n \times n$ matrix such that $\|A\| < 1$ then $I - A$ is invertible and*

$$(I-A)^{-1} = \sum_{k=0}^{\infty} A^k.$$

Proof: Suppose that $I - A$ is not invertible.
Then $\exists\, 0 \neq x$ with $\|x\| = 1$ such that $(I-A)x = 0$.
$\therefore 1 = \|x\| = \|Ax\| \leq \|A\|\|x\| = \|A\| < 1$, a contradiction!

Claim: $\sum_{k=0}^{\infty} A^k = (I-A)^{-1}$, i.e., $\lim_{m\to\infty} (I-A) \sum_{k=0}^{m} A^k = I$.

$\because (I-A) \sum_{k=0}^{m} A^k = \sum_{k=0}^{m} (A^k - A^{k+1}) = A^0 - A^{m+1} = I - A^{m+1}$

$\therefore 0 \leq \|(I-A) \sum_{k=0}^{m} A^k - I\| = \|-A^{m+1}\| \leq \|A\|^{m+1} \to 0$ as $m \to \infty$

Iterative refinement

- Let $x^{(0)}$ be an approximate solution of
$$Ax = b.$$
Then the residual vector is
$$r^{(0)} = b - Ax^{(0)}.$$
and the error vector is
$$e^{(0)} = x - x^{(0)}.$$
- Since $Ae^{(0)} = Ax - Ax^{(0)} = b - Ax^{(0)} = r^{(0)}$, we have
$$Ae^{(0)} = r^{(0)},$$
which is not too expensive to solve at this point. Why?
We also know that the exact solution
$$x = x^{(0)} + e^{(0)}.$$

Iterative refinement (continued)

Consider the linear system: $Ax = b$. Let $x^{(0)}$ be an approximation to the exact solution x. Then

$$\begin{aligned} r^{(0)} &= b - Ax^{(0)}, \\ Ae^{(0)} &= r^{(0)}. \end{aligned}$$

Let $\widetilde{e}^{(0)}$ be an approximate solution of $e^{(0)}$. Then define $x^{(1)} := x^{(0)} + \widetilde{e}^{(0)}$. Repeat this process, we have $x^{(2)}, x^{(3)}, \cdots$

Example

Consider the linear system:

$$\begin{bmatrix} 420 & 210 & 140 & 105 \\ 210 & 140 & 105 & 84 \\ 140 & 105 & 84 & 70 \\ 105 & 84 & 70 & 60 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 875 \\ 539 \\ 399 \\ 319 \end{bmatrix}.$$

- Exact solution $x = (1, 1, 1, 1)^\top$.
- GE with partial pivoting:

$$\begin{aligned} x^{(0)} &= (0.999988, 1.000137, 0.999670, 1.000215)^\top, \\ x^{(1)} &= (0.999994, 1.000069, 0.999831, 1.000110)^\top, \\ x^{(2)} &= (0.999996, 1.000046, 0.999891, 1.000070)^\top, \\ x^{(3)} &= (0.999993, 1.000080, 0.999812, 1.000121)^\top, \\ x^{(4)} &= (1.000000, 1.000006, 0.999984, 1.000010)^\top. \end{aligned}$$

A comparison

- We have been studying direct methods for solving the matrix problem $Ax = b$, e.g., LU-decomposition and GE.
  - large operation count.
  - needs lot of memory.
  - hard to do on parallel machines.
  - a solution will be found, and we know how long and how much memory it takes.
- Iterative methods produce a sequence of vectors that ideally converges to the solution.
  - much smaller operation counts.
  - needs much less memory.
  - a lot easier to implement on parallel computers.
  - not as reliable or predicable (the number of iterations is not known in advance).

Example

$$\begin{bmatrix} 7 & -6 \\ -8 & 9 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3 \\ -4 \end{bmatrix}.$$

How can this be solved by an iterative process?

Rewrite the system of equations as

$$\begin{aligned} x_1 &= \frac{6}{7}x_2 + \frac{3}{7}, \\ x_2 &= \frac{8}{9}x_1 - \frac{4}{9}. \end{aligned}$$

Jacobi method

$$
\begin{aligned}
x_1^{(k)} &= \frac{6}{7} x_2^{(k-1)} + \frac{3}{7}, \\
x_2^{(k)} &= \frac{8}{9} x_1^{(k-1)} - \frac{4}{9}.
\end{aligned}
$$

Here are some values of the iterates of the Jacobi method for this example:

k	$x_1^{(k)}$	$x_2^{(k)}$
0	0.00000	0.00000
10	0.14865	-0.19820
20	0.18682	-0.24909
30	0.19662	-0.26215
40	0.19913	-0.26637
50	0.19978	-0.26637

Gauss-Seidel method

$$
\begin{aligned}
x_1^{(k)} &= \frac{6}{7} x_2^{(k-1)} + \frac{3}{7}, \\
x_2^{(k)} &= \frac{8}{9} x_1^{(k)} - \frac{4}{9}.
\end{aligned}
$$

Some output from this method:

k	$x_1^{(k)}$	$x_2^{(k)}$
0	0.00000	0.00000
10	0.21978	-0.24909
20	0.20130	-0.26531
30	0.20009	-0.26659
40	0.20001	-0.26666
50	0.20000	-0.26667

Basic concepts

In general, to solve the system

$$Ax = b$$

using an iterative process, we prescribe a matrix Q, called the splitting matrix. We can rewrite the original system of equations as:

$$Qx = (Q - A)x + b.$$

The iterations are defined as follows:

$$Qx^{(k)} = (Q - A)x^{(k-1)} + b \qquad (k \geq 1),$$

where $x^{(0)}$ is an initial vector. The goal is to choose Q so that the following conditions hold:

- The sequence $\{x^{(k)}\}$ is easily computed.
- The sequence $\{x^{(k)}\}$ converges rapidly to a solution.

Theoretical analysis

$$x^{(k)} = (I - Q^{-1}A)x^{(k-1)} + Q^{-1}b. \qquad (*)$$

The actual solution x satisfies

$$x = (I - Q^{-1}A)x + Q^{-1}b. \qquad (**)$$

Thus, x is a fixed point of the mapping

$$x \longmapsto (I - Q^{-1}A)x + Q^{-1}b.$$

Subtracting (**) from (*) yields

$$x^{(k)} - x = (I - Q^{-1}A)(x^{(k-1)} - x).$$

Theoretical analysis (continued)

Using a convenient vector norm and its associated matrix norm,

$$\|x^{(k)} - x\| \leq \|I - Q^{-1}A\| \|x^{(k-1)} - x\|.$$

Repeating this step, we obtain

$$\|x^{(k)} - x\| \leq \|I - Q^{-1}A\|^k \|x^{(0)} - x\|.$$

Thus, if $\|I - Q^{-1}A\| < 1$ then

$$\lim_{k\to\infty} \|x^{(k)} - x\| = 0$$

for any initial vector $x^{(0)}$.

Note: According to Theorem on Neumann series, $\|I - Q^{-1}A\| < 1$ implies the invertibility of $Q^{-1}A$ and of A.

Theorem on iterative method convergence

If $\|I - Q^{-1}A\| < 1$ for some vector induced matrix norm (also called subordinate matrix norm), then the sequence produced by

$$Qx^{(k)} = (Q - A)x^{(k-1)} + b$$

converges to the solution of $Ax = b$ for any initial vector $x^{(0)}$.

Note: If $\{x^{(k)}\}$ converges, it converges in any norm.

Richardson method

- Q is chosen to be the identity matrix. In this case, the iterates are given by:

$$x^{(k)} = (I - A)x^{(k-1)} + b = x^{(k-1)} + r^{(k-1)},$$

where $r^{(k-1)}$ is the residual vector, $r^{(k-1)} := b - Ax^{(k-1)}$.

- According to the above theorem, Richardson method will converges to solution of $Ax = b$ if $\|I - A\| < 1$ for some vector induced matrix norm.
- There are two classes of matrices having the required property (cf. page 229, problems 2 & 3):
  - unit row strictly diagonally dominant matrices:

$$a_{ii} = 1 > \sum_{j=1, j\neq i}^{n} |a_{ij}| \qquad (1 \leq i \leq n) \Longrightarrow \|I - A\|_\infty < 1$$

  - unit column strictly diagonally dominant matrices:

$$a_{jj} = 1 > \sum_{i=1, i\neq j}^{n} |a_{ij}| \qquad (1 \leq j \leq n) \Longrightarrow \|I - A\|_1 < 1$$

An example

Compute 100 iterates using the Richardson method, starting with $x = (0, 0, 0)^\top$.

$$\begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{3} & 1 & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{3} & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} \frac{11}{18} \\ \frac{11}{18} \\ \frac{11}{18} \end{bmatrix}.$$

A few of the iterates:

$$\begin{aligned} x^{(0)} &= (0.00000, 0.00000, 0.00000)^\top, \\ x^{(1)} &= (0.61111, 0.61111, 0.61111)^\top, \\ x^{(10)} &= (0.27950, 0.27950, 0.27950)^\top, \\ &\vdots \\ x^{(40)} &= (0.33311, 0.33311, 0.33311)^\top, \\ &\vdots \\ x^{(80)} &= (0.33333, 0.33333, 0.33333)^\top. \end{aligned}$$

Diagonally dominant matrices

- **Definition:** The $n \times n$ matrix $A = (a_{ij})$ is called strictly diagonally dominant if

$$|a_{ii}| > \sum_{j=1, j \neq i}^{n} |a_{ij}| \qquad (1 \leq i \leq n).$$

- **Example:**

$$\begin{bmatrix} 4 & -1 & 0 & -1 \\ -1 & 4 & 0 & -1 \\ -1 & 0 & 4 & -1 \\ 0 & -1 & -1 & 4 \end{bmatrix}$$

is strictly diagonally dominant.

Jacobi method

- In the Jacobi iteration, Q is a diagonal matrix whose diagonal entries are the same as those in the matrix A.
- One can verify that

$$\|I - Q^{-1}A\|_\infty = \max_{1 \le i \le n} \sum_{j=1, j\neq i}^{n} \left|\frac{a_{ij}}{a_{ii}}\right|.$$

- **Theorem on Convergence of Jacobi Method:**
  If A is strictly diagonally dominant, then the sequence produced by the Jacobi iteration converges to the solution of $Ax = b$ for any starting vector.

Algorithm for the Jacobi method

input n, (a_{ij}), (b_i), (x_i), M
for $k = 1$ **to** M **do**
  for $i = 1$ **to** n **do**

$$u_i \leftarrow \left(b_i - \sum_{j=1, j\neq i}^{n} a_{ij} x_j \right) \Big/ a_{ii}$$

  end do
  for $i = 1$ **to** n **do**
    $x_i \leftarrow u_i$
  end do
  output k, (x_i)
end do

Some remarks

- Some divisions can be avoided by preprocessing the system.

  for $i = 1$ **to** n **do**
      $d = 1/a_{ii}$
      $b_i \leftarrow db_i$
      for $j = 1$ **to** n **do**
          $a_{ij} = da_{ij}$
      end do
  end do

  Then the replacement statement for u_i becomes simply

  $$u_i \leftarrow b_i - \sum_{j=1, j\neq i}^{n} a_{ij}x_j.$$

- Another way to interpret this is that the original system $Ax = b$ has been replaced by:

  $$D^{-1}Ax = D^{-1}b,$$

  where $D = \mathrm{diag}(a_{ii})$.

How to stop the iterations?

- Residual norm: $\|r\| = \|b - Ax\|$.
- Where is r_i in the computer program? (if without preprocessing)

$$r_i = b_i - \sum_{j=1, j\neq i}^{n} a_{ij}x_j - a_{ii}x_i = a_{ii}u_i - a_{ii}x_i.$$

- Or, one can implement the Jacobi algorithm differently:

$$x^{(k+1)} = (I - Q^{-1}A)x^{(k)} + Q^{-1}b.$$

is the same as

$$x^{(k+1)} = x^{(k)} - Q^{-1}(b - Ax^{(k)}) = x^{(k)} - Q^{-1}r^{(k)}.$$

Spectral radius

- The spectral radius of A is defined by

$$\rho(A) = \max\{|\lambda| : \det(A - \lambda I) = 0\}.$$

- Thus, $\rho(A)$ is the smallest number such that a circle with that radius centered at 0 in the complex plane will contain all the eigenvalues of A.
- **Theorem on Spectral Radius:** *The spectral radius function satisfies the equation:*

$$\rho(A) = \inf_{\|\cdot\|} \|A\|,$$

in which the infimum is taken over all subordinate matrix norms.

Proof: see pp. 214-215.

- **Corollary on Spectral Radius:**
  - $\rho(A) \leq \|A\|$, for any subordinate matrix norm.
  - If $\rho(A) < 1$ then $\|A\| < 1$ for some subordinate matrix norm.

Analysis

In general, an iterative method defined by

$$Qx^{(k)} = (Q - A)x^{(k-1)} + b.$$

Let $G = I - Q^{-1}A$ and $c = Q^{-1}b$. Then we consider the iterative process in the following form:

$$x^{(k)} = Gx^{(k-1)} + c.$$

Suppose that it converges, then the solution must satisfy

$$x = Gx + c,$$

or

$$(I - G)x = c,$$

or

$$x = (I - G)^{-1}c.$$

Necessary and sufficient conditions for convergence

For the iteration formula

$$x^{(k)} = Gx^{(k-1)} + c$$

to produce a sequence converging to $(I - G)^{-1}c$*, for any* c *and starting vector* $x^{(0)}$*, it is necessary and sufficient that the spectral radius of* G *be less than* 1*, i.e.,* $\rho(G) < 1$.

Proof of the Theorem

Suppose that $\rho(G) < 1$. Then there is a subordinate matrix norm such that $\|G\| < 1$. From the iteration formula, we have

$$\begin{aligned} x^{(1)} &= Gx^{(0)} + c, \\ x^{(2)} &= G^2 x^{(0)} + Gc + c, \\ &\cdots \\ x^{(k)} &= G^k x^{(0)} + \sum_{j=0}^{k-1} G^j c. \qquad (\star) \end{aligned}$$

Using the matrix norm (and corresponding vector norm) that satisfies the spectral radius theorem:

$$\|G^k x^{(0)}\| \le \|G^k\| \|x^{(0)}\| \le \|G\|^k \|x^{(0)}\| \to 0 \quad as \quad k \to \infty.$$

The second term on RHS of $(\star)$ as $k \to \infty$ is given by

$$\sum_{j=0}^{\infty} G^j c = (I - G)^{-1} c,$$

when $\|G\| < 1$ by Neumann series. Thus, by letting $k \to \infty$, we obtain

$$\lim_{k \to \infty} x^{(k)} = (I - G)^{-1} c.$$

Proof of the Theorem (continued)

For the converse, suppose that $\rho(G) \geq 1$. Select u and λ so that

$$Gu = \lambda u,$$

where $|\lambda| \geq 1$ and $u \neq 0$. Recall that $x^{(k)} = G^k x^{(0)} + \sum_{j=0}^{k-1} G^j c$. Let $c = u$ and $x^{(0)} = 0$. Then we have

$$x^{(k)} = \sum_{j=0}^{k-1} G^j u = \sum_{j=0}^{k-1} \lambda^j u.$$

- If $\lambda = 1$, $x^{(k)} = ku$, this diverges as $k \to \infty$.
- If $\lambda \neq 1$, $x^{(k)} = (\lambda^k - 1)(\lambda - 1)^{-1} u$, this diverges as $k \to \infty$ and this diverges also because $\lim_{k\to\infty} \lambda^k$ does not exist.

For both cases, $\{x^{(k)}\}$ diverges, a contradiction! Therefore, $\rho(G) < 1$.

Gauss-Seidel method

- In the Gauss-Seidel iteration, Q is the lower triangular part of A, including the diagonal.
- **Theorem on Gauss-Seidel Method Convergence:**

  If A is strictly diagonally dominant, then the Gauss-Seidel method converges for any starting vector.

  Proof: It suffices to prove that $\rho(I - Q^{-1}A) < 1$. Let λ be any eigenvalue of $I - Q^{-1}A$ and let x be a corresponding eigenvector. Without loss of generality, we assume that $\|x\|_\infty = 1$. Then $(I - Q^{-1}A)x = \lambda x$ or $Qx - Ax = \lambda Qx$.

$$-\sum_{j=i+1}^{n} a_{ij}x_j = \lambda \sum_{j=1}^{i} a_{ij}x_j, \qquad (1 \le i \le n).$$

  By transposing terms in this equation, we obtain

$$\lambda a_{ii}x_i = -\lambda \sum_{j=1}^{i-1} a_{ij}x_j - \sum_{j=i+1}^{n} a_{ij}x_j, \qquad (1 \le i \le n).$$

Theorem on Gauss-Seidel method convergence (continued)

Since $\|x\|_\infty = 1$, we can select an index i such that $|x_i| = 1 \geq |x_j|$ for all j. Then

$$|\lambda||a_{ii}| \leq |\lambda| \sum_{j=1}^{i-1} |a_{ij}| + \sum_{j=i+1}^{n} |a_{ij}|.$$

Solving for $|\lambda|$ and using the strictly diagonal dominance of A, we have

$$|\lambda| \leq \frac{\sum_{j=i+1}^{n} |a_{ij}|}{|a_{ii}| - \sum_{j=1}^{i-1} |a_{ij}|} < 1.$$

Therefore, $\rho(I - Q^{-1}A) < 1$.

Algorithm for the Gauss-Seidel iteration

input n, (a_{ij}), (b_i), (x_i), M
for $k = 1$ **to** M **do**
    for $i = 1$ **to** n **do**

$$x_i \leftarrow \left(b_i - \sum_{j=1, j \neq i}^{n} a_{ij} x_j \right) \bigg/ a_{ii}$$

    end do
    output k, (x_i)
end do

Example

Consider the linear system:

$$\begin{bmatrix} 2 & -1 & 0 \\ 1 & 6 & -2 \\ 4 & -3 & 8 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 2 \\ -4 \\ 5 \end{bmatrix}.$$

Start with $x^{(0)} = (0,0,0)^\top$. Scaling using the equation $D^{-1}Ax = D^{-1}b$ where $D = \mathrm{diag}(A)$, we obtain:

$$\begin{bmatrix} 1 & -\frac{1}{2} & 0 \\ \frac{1}{6} & 1 & -\frac{1}{3} \\ \frac{1}{2} & -\frac{3}{8} & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ -\frac{2}{3} \\ \frac{5}{8} \end{bmatrix}.$$

Example (continued)

Referring to this system as $Ax = b$, we take Q to be the lower triangular part of A. The Gauss-Seidel iteration is given by:

$$Qx^{(k)} = (Q - A)x^{(k-1)} + b$$

or

$$\begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{6} & 1 & 0 \\ \frac{1}{2} & -\frac{3}{8} & 1 \end{bmatrix} \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ x_3^{(k)} \end{bmatrix} = \begin{bmatrix} 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{3} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1^{(k-1)} \\ x_2^{(k-1)} \\ x_3^{(k-1)} \end{bmatrix} + \begin{bmatrix} 1 \\ -\frac{2}{3} \\ \frac{5}{8} \end{bmatrix}.$$

Example (continued)

We obtain $x^{(k)}$ by solving a lower triangular system:

$$\begin{aligned} x_1^{(k)} &= \tfrac{1}{2}x_2^{(k-1)} + 1, \\ x_2^{(k)} &= -\tfrac{1}{6}x_1^{(k)} + \tfrac{1}{3}x_3^{(k-1)} - \tfrac{2}{3}, \\ x_3^{(k)} &= -\tfrac{1}{2}x_1^{(k)} + \tfrac{3}{8}x_2^{(k)} + \tfrac{5}{8}. \end{aligned}$$

The following iterates are obtained $(x^{(13)}$ is the correct solution):

$$\begin{aligned} x^{(1)} &= (1.000000, -0.833333, -0.187500)^\top, \\ &\vdots \\ x^{(5)} &= (0.622836, -0.760042, 0.028566)^\top, \\ &\vdots \\ x^{(10)} &= (0.620001, -0.760003, 0.029998)^\top, \\ &\vdots \\ x^{(13)} &= (0.620000, -0.760000, 0.030000)^\top. \end{aligned}$$

Basic iterative methods

For any nonsingular matrix Q, the system

$$Ax = b$$

can be rewritten as:

$$Qx = (Q - A)x + b.$$

An iterative method can be defined as follows:

$$Qx^{(k)} = (Q - A)x^{(k-1)} + b$$

or

$$x^{(k)} = (I - Q^{-1}A)x^{(k-1)} + Q^{-1}b.$$

Here $G = I - Q^{-1}A$ is called the iteration matrix.

More about iteration matrices

Suppose A is partitioned into

$$A = D - C_L - C_U,$$

where $D = \mathrm{diag}(A)$, C_L is the negative of the strictly lower part of A, and C_U is the negative of the strictly upper part of A.

- **Richardson:**

$$\begin{cases} Q = I, & (\textit{splitting matrix}) \\ G = I - A. & (\textit{iteration matrix}) \end{cases}$$

$$x^{(k)} = (I - A)x^{(k-1)} + b.$$

More about iteration matrices (continued)

- **Jacobi:**

$$\begin{cases} Q = D, & \textit{(splitting matrix)} \\ G = D^{-1}(C_L + C_U). & \textit{(iteration matrix)} \end{cases}$$

$$Dx^{(k)} = (C_L + C_U)x^{(k-1)} + b.$$

- **Gauss-Seidel:**

$$\begin{cases} Q = D - C_L, & \textit{(splitting matrix)} \\ G = (D - C_L)^{-1}C_U. & \textit{(iteration matrix)} \end{cases}$$

$$(D - C_L)x^{(k)} = C_U x^{(k-1)} + b.$$

- **Successive over-relaxation (SOR):**

$$\begin{cases} Q = \omega^{-1}(D - \omega C_L), & \textit{(splitting matrix)} \\ G = (D - \omega C_L)^{-1}\Big((1-\omega)D + \omega C_U\Big). & \textit{(iteration matrix)} \end{cases}$$

$$(D - \omega C_L)x^{(k)} = \Big((1-\omega)D + \omega C_U\Big)x^{(k-1)} + \omega b.$$

Another viewpoint of SOR

$x_i^{(k)}$ is obtained by a weighted sum of $x_i^{(k-1)}$ and the GS iteration:

$$
\begin{aligned}
& x_i^{(k)} = (1-\omega)x_i^{(k-1)} + \frac{\omega}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k-1)}\right) \\
\Longleftrightarrow \quad & a_{ii}x_i^{(k)} + \omega\sum_{j=1}^{i-1} a_{ij}x_j^{(k)} = (1-\omega)a_{ii}x_i^{(k-1)} - \omega\sum_{j=i+1}^{n} a_{ij}x_j^{(k-1)} + \omega b_i \\
\Longleftrightarrow \quad & (D-\omega C_L)x^{(k)} = \Big((1-\omega)D + \omega C_U\Big)x^{(k-1)} + \omega b \\
\Longleftrightarrow \quad & x^{(k)} = (D-\omega C_L)^{-1}\Big((1-\omega)D + \omega C_U\Big)x^{(k-1)} + \omega(D-\omega C_L)^{-1}b
\end{aligned}
$$

Remarks:

- $0 < \omega < 1$: under-relaxation methods and can be used to obtain convergence of some systems that are not convergent by the GS.
- $1 < \omega$: over-relaxation methods, which are used to accelerate the convergence for systems that are convergent by the GS.
- Methods are abbreviated SOR (successive over-relaxation).

Recall - linear algebra

- Let $\gamma \in \mathbb{C}$ and be written as $\gamma = \alpha + i\beta$, where α and β are real and $i^2 = -1$. The conjugate of γ is defined to be $\bar{\gamma} = \alpha - i\beta$.
- In $\mathbb{C}^n$, the inner product is defined as $< x, y >= y^* x = \sum_{i=1}^n x_i \bar{y}_i$. Here y^* is the conjugate transpose of y, i.e., $y^* = \bar{y}^\top$.
- Some properties: $x, y, z \in \mathbb{C}^n$, $\alpha, \beta, \lambda \in \mathbb{C}$, $A \in \mathbb{C}^{n \times n}$.
  - $< x, x >> 0, \quad$ (if $x \neq 0$).
  - $< x, \lambda y >= \bar{\lambda} < x, y >$.
  - $< x, y >= \overline{< y, x >}$.
  - $< \alpha x + \beta y, z >= \alpha < x, z > + \beta < y, z >$.
  - $< Ax, y >=< x, A^* y >$ and $< x, Ay >=< A^* x, y >$.
  - $\|x\|_2 = \sqrt{< x, x >} = \sqrt{x^* x}$.
- A is Hermitian if $A^* = A$, where A^* is conjugate transpose of A.
- A is positive definite if $< Ax, x >> 0$ for all $0 \neq x \in \mathbb{C}^n$.
- If A is Hermitian, then $< Ax, y >=< x, A^* y >=< x, Ay >$.

A general theory for SOR

Theorem on SOR convergence: A is Hermitian and positive definite
In the SOR method, suppose that the splitting matrix Q is chosen to be $\alpha D - C$, where α is a real parameter, D is any positive definite Hermitian matrix, and C is any matrix satisfying $C + C^* = D - A$. If A is positive definite Hermitian, if Q is nonsingular, and if $\alpha > \frac{1}{2}$, then the SOR iteration converges for any starting vector.

Proof: Let $G := I - Q^{-1}A$ be the iteration matrix. We wish to show that $\rho(G) < 1$. Let λ be an eigenvalue of G and x be a corresponding eigenvector. Let $y = (I - G)x$. Then we have

$$y = x - Gx = x - \lambda x = Q^{-1}Ax, \qquad (1)$$

$$Q - A = (\alpha D - C) - (D - C - C^*) = \alpha D - D + C^*. \qquad (2)$$

From (1), we have

$$(\alpha D - C)y = Qy = Ax. \qquad (3)$$

By (1), (2), (3), we obtain

$$(\alpha D - D + C^*)y = (Q - A)y = A(x - y) = A(x - Q^{-1}Ax) = AGx. \qquad (4)$$

A general theory for SOR (continued)

From (3) and (4), we have

$$\alpha < Dy, y > - < Cy, y > = < Ax, y >, \qquad (5)$$
$$\alpha < y, Dy > - < y, Dy > + < y, C^* y > = < y, AGx > . \qquad (6)$$

On adding (5) and (6), we have

$$2\alpha < Dy, y > - < y, Dy > = < Ax, y > + < y, AGx >,$$

which implies

$$(2\alpha - 1) < Dy, y > = < Ax, y > + < y, AGx > . \qquad (7)$$

Since $y = (1 - \lambda)x$ and $Gx = \lambda x$, equation (7) yields

$$(2\alpha - 1)|1 - \lambda|^2 < Dx, x > = (1 - \overline{\lambda}) < Ax, x > + \overline{\lambda}(1 - \lambda) < x, Ax >$$
$$= (1 - |\lambda|^2) < Ax, x > .$$

If $\lambda \neq 1$ then LHS is positive, RHS must be positive and $|\lambda| < 1$.
If $\lambda = 1$ then $y = x - \lambda x = 0 = Q^{-1}Ax$. So, $Ax = 0$. This is a contradiction, since $< Ax, x >> 0$. Therefore, we have $\rho(G) < 1$.

A general theory for SOR (continued)

- In practice, we let D be the diagonal of A, and $-C$ be the strictly lower triangular part of A, i.e., $C = C_L$.
- In the most popular SOR method,

$$Q = \omega^{-1}(D - \omega C_L) = \alpha D - C_L.$$

  This implies that $\omega^{-1} = \alpha$. Therefore, $\alpha > 1/2 \Longleftrightarrow 0 < \omega < 2$.
- $\omega = 1$, we have the Gauss-Seidel method.

Homework

Consider the linear system $Ax = b$, where

$$A = \begin{bmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{bmatrix}_{10\times 10}, \quad b = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}_{10\times 1}$$

Using $x^{(0)} = (1, 0, 0, \cdots, 0)^\top$ as an initial vector, write Matlab files for the Jacobi, Gauss-Seidel, SOR with $\omega = 1.25$ to solve the system.

Extrapolation

- The extrapolation technique can be used to improve the convergence properties of a linear iterative process.
- Consider the iteration formula:

$$x^{(k)} = Gx^{(k-1)} + c. \qquad (*)$$

- We introduce a parameter, $\gamma \neq 0$ and consider

$$\begin{aligned} x^{(k)} &= \gamma(Gx^{(k-1)} + c) + (1-\gamma)x^{(k-1)} \\ &= G_\gamma x^{(k-1)} + \gamma c, \end{aligned}$$

where

$$G_\gamma = \gamma G + (1-\gamma)I.$$

- Notice that when $\gamma = 1$, we recover the original iteration (*).

Extrapolation (continued)

- If the iteration converges,

$$x = \gamma(Gx + c) + (1 - \gamma)x.$$

or

$$x = Gx + c,$$

since $\gamma \neq 0$.

- If $G = I - QA^{-1}$ and $c = Q^{-1}b$, then this iteration corresponds to solving $Ax = b$.

Extrapolation (continued)

- **Theorem on Eigenvalues of $p(A)$:** *If λ is an eigenvalue of a matrix A and if p is a polynomial, then $p(\lambda)$ is an eigenvalue of $p(A)$.*
- The convergence of the extrapolated method is guaranteed if $\rho(G_\gamma) < 1$.

$$\begin{aligned} \rho(G_\gamma) &= \max_{\lambda \in \Lambda(G_\gamma)} |\lambda| = \max_{\lambda \in \Lambda(G)} |\gamma\lambda + 1 - \gamma| \\ &\leq \max_{a \leq \lambda \leq b} |\gamma\lambda + 1 - \gamma|, \end{aligned}$$

if we know only an interval $[a, b] \subseteq \mathbb{R}$ that contain all eigenvalues of G.
- We can prove that if $1 \notin [a, b]$ then γ can be chosen so that $\rho(G_\gamma) < 1$. The best choice for γ is $2/(2 - a - b)$, and in such case $\rho(G_\gamma) \leq 1 - |\gamma| d$, d is the distance from 1 to $[a, b]$ (see pp. 222-223).

An example

If A is a matrix whose eigenvalues $\lambda_1, \lambda_2, \cdots, \lambda_n$ are all real, define

$$m(A) = \min_i \lambda_i \qquad M(A) = \max_i \lambda_i.$$

Example: Determine the spectral radius of the optimal extrapolated Richardson method.

In Richardson iteration, $Q = I$ and $G = I - A$.

$$M(G) = 1 - m(A) \qquad m(G) = 1 - M(A).$$

The optimal γ is:

$$\gamma = 2/(m(A) + M(A)).$$

The resulting spectral radius is:

$$\rho(G_\gamma) = (M(A) - m(A))/(M(A) + m(A)).$$

SPD linear systems

- Let $A \in \mathbb{C}^{n\times n}$ be a square matrix and $x, y \in \mathbb{C}^n$. Define $x^* := \overline{x}^\top$, $(x, y) := y^*x \in \mathbb{C}$. *Then* $(Ax, x) = x^*Ax$ *is called a quadratic form.*
- **Definition:** Let $A \in \mathbb{C}^{n\times n}$.

$$A \text{ is positive definite } \iff (Ax, x) > 0, \quad \forall\, 0 \neq x \in \mathbb{C}^n.$$

- **Note 1:** $A = A^* (:= \overline{A}^\top) \iff (Ax, x) \in \mathbb{R}, \forall\, x \in \mathbb{C}^n$.
- **Note 2:** If $A \in \mathbb{C}^{n\times n}$ is positive definite, then $A = A^*$. (by Note 1)
- **Note 3:** Let $A \in \mathbb{R}^{n\times n}$. A is positive definite $\iff A = A^\top$ and $(Ax, x) > 0, \forall\, 0 \neq x \in \mathbb{R}^n$.
- **Note 4:** Let $A \in \mathbb{C}^{n\times n}$ and $A = A^*$. Then A is positive definite $\iff$ all of its eigenvalues are real and positive.

SPD linear systems (continued)

Let $A \in \mathbb{R}^{M\times M}$ be a SPD sparse matrix. Define $f : \mathbb{R}^M :\to \mathbb{R}$ by

$$f(\eta) = \frac{1}{2}\eta \cdot A\eta - b \cdot \eta.$$

- **Problem** (1)**:** Find $\xi \in \mathbb{R}^M$ such that $f(\xi) = \min_{\eta \in \mathbb{R}^M} f(\eta)$.
- **Problem** (2)**:** Find $\xi \in \mathbb{R}^M$ such that $A\xi = b$.

  Note: $\exists\,!$ solution ξ such that $A\xi = b$, since A is SPD.

Theorem: Problem (1) $\Longleftrightarrow$ Problem (2).

See next two pages for the proof.

Proof of the Theorem

- Problem (1) ($\Longrightarrow$) Problem (2):

  Let $\xi \in \mathbb{R}^M$ be such that $f(\xi) = \min_{\eta \in \mathbb{R}^M} f(\eta)$. Given $0 \neq \eta \in \mathbb{R}^M$, we have

$$
\begin{aligned}
g(\varepsilon) \quad &:= \quad f(\xi + \varepsilon\eta) = \frac{1}{2}(\xi + \varepsilon\eta) \cdot A(\xi + \varepsilon\eta) - b \cdot (\xi + \varepsilon\eta) \\
&= \quad \frac{1}{2}\xi \cdot A\xi + \frac{1}{2}\varepsilon\xi \cdot A\eta + \frac{1}{2}\varepsilon\eta \cdot A\xi + \frac{1}{2}\varepsilon^2\eta \cdot A\eta - b \cdot \xi - \varepsilon b \cdot \eta \\
&= \quad \frac{1}{2}\varepsilon^2\eta \cdot A\eta + \varepsilon\eta \cdot A\xi - \varepsilon b \cdot \eta + \frac{1}{2}\xi \cdot A\xi - b \cdot \xi,
\end{aligned}
$$

  where we use
  $\xi \cdot A\eta = (\xi, A\eta) = (A^\top \xi, \eta) = (A\xi, \eta) = (\eta, A\xi) = \eta \cdot A\xi.$

  $\therefore g$ is a quadratic poly. in ε with leading coefficient $\frac{1}{2}\eta \cdot A\eta > 0$

  $\because g(0) = f(\xi) = \min_{\eta \in \mathbb{R}^M} f(\eta) \quad \therefore g'(0) = 0$ (by Fermat's Thm)

  $\therefore 0 = g'(0) = \left(\varepsilon\eta \cdot A\eta + \eta \cdot A\xi - b \cdot \eta\right)\Big|_{\varepsilon=0} = \eta \cdot (A\xi - b)$

  $\therefore A\xi = b$

Proof of the Theorem (continued)

- Problem (2) ($\Longrightarrow$) Problem (1):

  Assume that $A\xi = b$. Let $\eta \in \mathbb{R}^M$. Define $w := \eta - \xi$. Then $\eta = w + \xi$. We have

$$
\begin{aligned}
f(\eta) &= \frac{1}{2}\eta \cdot A\eta - b \cdot \eta = \frac{1}{2}(w+\xi) \cdot A(w+\xi) - b \cdot (w+\xi) \\
&= \frac{1}{2}w \cdot Aw + w \cdot A\xi + \frac{1}{2}\xi \cdot A\xi - b \cdot w - b \cdot \xi \\
&= \frac{1}{2}w \cdot Aw + w \cdot A\xi - b \cdot w + f(\xi) \\
&\geq w \cdot A\xi - b \cdot w + f(\xi) \qquad (\because A \text{ is SPD } \therefore \frac{1}{2}w \cdot Aw \geq 0) \\
&= w \cdot b - b \cdot w + f(\xi) = f(\xi).
\end{aligned}
$$

  $\therefore f(\xi) = \min_{\eta \in \mathbb{R}^M} f(\eta)$.

Minimization algorithms

Given an initial approximation $\xi^0 \in \mathbb{R}^M$ of the exact solution ξ, find $\xi^k \in \mathbb{R}^M$, $k = 1, 2, \cdots$ of the form

$$\xi^{k+1} = \xi^k + \alpha_k d^k, \quad k = 0, 1, \cdots,$$

where $d^k \in \mathbb{R}^M$ is the search direction, $\alpha_k > 0$ is the step size (length).

We will focus on two methods:

- The gradient method
- The conjugate gradient method

Some notation

Let $g : \mathbb{R}^M \to \mathbb{R}$ be a smooth function and $\eta \in \mathbb{R}^M$.

- gradient of g at η

$$= g'(\eta) := \nabla g(\eta) := \left(\frac{\partial g}{\partial \eta_1}(\eta), \frac{\partial g}{\partial \eta_2}(\eta), \cdots, \frac{\partial g}{\partial \eta_M}(\eta)\right)^\top.$$

- Hessian of g at η,

$$\begin{aligned}
g''(\eta) &= \begin{bmatrix} \frac{\partial^2 g}{\partial \eta_1^2}(\eta) & \frac{\partial^2 g}{\partial \eta_1 \partial \eta_2}(\eta) & \cdots & \frac{\partial^2 g}{\partial \eta_1 \partial \eta_M}(\eta) \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial^2 g}{\partial \eta_M \partial \eta_1}(\eta) & \frac{\partial^2 g}{\partial \eta_M \partial \eta_2}(\eta) & \cdots & \frac{\partial^2 g}{\partial \eta_M^2}(\eta) \end{bmatrix}_{M \times M} \\
&= \left(\nabla \frac{\partial g}{\partial \eta_1}(\eta), \cdots, \nabla \frac{\partial g}{\partial \eta_M}(\eta)\right) \\
&:= \nabla \left(\frac{\partial g}{\partial \eta_1}(\eta), \cdots, \frac{\partial g}{\partial \eta_M}(\eta)\right) \\
&= \nabla \left(g'(\eta)^\top\right) = \nabla \left(\nabla g(\eta)^\top\right).
\end{aligned}$$

Homework

Assume that $A \in \mathbb{R}^{M\times M}$ is a symmetric matrix, $b \in \mathbb{R}^M$ is a given vector, and $f : \mathbb{R}^M \to \mathbb{R}$ is defined by $f(\eta) := \frac{1}{2}\eta \cdot A\eta - b \cdot \eta$.

Prove that $\forall\, \eta \in \mathbb{R}^M$,

- $f'(\eta) = A\eta - b$;
- $f''(\eta) = A$.

Hint:

- $\eta \cdot A\eta = \eta_1(A_{1\cdot} \cdot \eta) + \eta_2(A_{2\cdot} \cdot \eta) + \cdots + \eta_M(A_{M\cdot} \cdot \eta)$.
- $f''(\eta) = \nabla(\nabla f(\eta)^\top) = \nabla((A\eta - b)^\top) = \nabla(A_{1\cdot} \cdot \eta - b_1, \cdots, A_{M\cdot} \cdot \eta - b_M)$.

Taylor's expansion of a smooth function g at ξ^k

Let $g : \mathbb{R}^M \to \mathbb{R}$ be a smooth function. By Taylor's expansion,

$$\begin{aligned} g(\xi^{k+1}) &= g(\xi^k) + \nabla g(\xi^k) \cdot (\xi^{k+1} - \xi^k) + (\xi^{k+1} - \xi^k) \cdot \frac{g''(\eta)}{2!}(\xi^{k+1} - \xi^k), \\ &\quad \textit{for some } \eta \in \overline{\xi^k \xi^{k+1}}. \\ &= g(\xi^k) + \alpha_k g'(\xi^k) \cdot d^k + \frac{\alpha_k^2}{2!} d^k \cdot g''(\eta) d^k, \quad \textit{if } \xi^{k+1} = \xi^k + \alpha_k d^k. \end{aligned}$$

$\therefore g(\xi^{k+1}) = g(\xi^k) + \alpha_k g'(\xi^k) \cdot d^k + O(\alpha_k^2)$, if the entries in $g''(\eta)$ are bounded in a neighborhood containing $\overline{\xi^k \xi^{k+1}}$.

$\therefore$ If $g'(\xi^k) \cdot d^k < 0$ and $\alpha_k > 0$ is sufficiently small, $g(\xi^{k+1}) < g(\xi^k)$.
In this case, we call d^k a descent direction.

The gradient method

Let us go back to the case of $g = f$, where $f(\eta) := \frac{1}{2}\eta \cdot A\eta - b \cdot \eta$ and A is SPD.

If we choose $d^k = -f'(\xi^k) = -(A\xi^k - b)$ and if $f'(\xi^k) \neq 0$,

then we have $f'(\xi^k) \cdot d^k = -\|f'(\xi^k)\|_2^2 < 0$.

We obtain the so-called gradient method or the steepest descent method.

Note: If $f'(\xi^k) = 0$ then $A\xi^k - b = 0 \Longrightarrow A\xi^k = b \Longrightarrow \xi^k$ is the exact solution.

How to choose $\alpha_k > 0$ in the gradient method?

Determine optimal α_k such that $f(\xi^k + \alpha_k d^k) = \min_{\alpha\in\mathbb{R}} f(\xi^k + \alpha d^k)$.

Notice that $f(\xi^k + \alpha d^k)$ can be viewed as a quadratic function in α with positive leading coefficient.

If α_k is optimal, then $\dfrac{d}{d\alpha} f(\xi^k + \alpha d^k)\Big|_{\alpha=\alpha_k} = 0.$

$\therefore f'(\xi^k + \alpha d^k)\cdot d^k\Big|_{\alpha=\alpha_k} = 0. \quad \therefore f'(\xi^k + \alpha_k d^k)\cdot d^k = 0.$

$$\begin{aligned} \Longrightarrow 0 &= f'(\xi^k + \alpha_k d^k)\cdot d^k = \Big(A(\xi^k + \alpha_k d^k) - b\Big)\cdot d^k \\ &= (A\xi^k - b)\cdot d^k + \alpha_k d^k \cdot A d^k. \end{aligned}$$

$\therefore \alpha_k = -\dfrac{(A\xi^k - b)\cdot d^k}{d^k\cdot A d^k} = \dfrac{d^k\cdot d^k}{d^k\cdot A d^k}$, provided
$d^k = -f'(\xi^k) = -(A\xi^k - b) \neq 0$

$\because A$ is SPD $\therefore d^k\cdot A d^k > 0$, provided $d^k = -f'(\xi^k) = -(A\xi^k - b)\neq 0$

$\therefore \alpha_k > 0$, provided $d^k = -f'(\xi^k) = -(A\xi^k - b) \neq 0$

The gradient method with optimal step length α_k

Given $\xi^0 \in \mathbb{R}^M$, define

$$\xi^{k+1} = \xi^k + \alpha_k d^k, k = 0, 1, \cdots$$

$$d^k = -(A\xi^k - b).$$

$$\alpha_k = \frac{d^k \cdot d^k}{d^k \cdot Ad^k}.$$

Recall of the condition number

Let $A \in \mathbb{R}^{M\times M}$ be a SPD matrix.

Let $0 < \lambda_1 \le \lambda_2 \le \cdots \le \lambda_M$ be the eigenvalues of A.

Then $0 < \dfrac{1}{\lambda_M} \le \dfrac{1}{\lambda_{M-1}} \le \cdots \le \dfrac{1}{\lambda_1}$ are the eigenvalues of A^{-1}.

Let $\rho(A)$ denote the spectral radius of A, i.e., the maximum size of the eigenvalues of A. That is, $\rho(A) = \max\limits_{\lambda \text{ is an e.v. of } A} |\lambda|$

$$\begin{aligned}
&\textit{condition number } \kappa(A) \\
&:= \|A\|_2 \|A^{-1}\|_2 = \sqrt{\rho(A^*A)}\sqrt{\rho((A^{-1})^*A^{-1})} \\
&= \sqrt{\rho(A^\top A)}\sqrt{\rho((A^{-1})^\top A^{-1})} = \sqrt{\rho(A^2)}\sqrt{\rho((A^{-1})^2)} \\
&= \sqrt{\lambda_M^2}\sqrt{\frac{1}{\lambda_1^2}} = \frac{\lambda_M}{\lambda_1}.
\end{aligned}$$

$\therefore \kappa(A) = \dfrac{\lambda_{\max}}{\lambda_{\min}}.$

The gradient method with constant step length

Given ξ_0, $\alpha > 0$ sufficiently small.

$\xi^{k+1} = \xi^k + \alpha d^k, k = 0, 1, \cdots$

$d^k = -f'(\xi^k) = -(A\xi^k - b).$

Let ξ be the exact solution, $A\xi = b. \Longrightarrow \xi = \xi - \alpha(A\xi - b).$

Let $e^k := \xi - \xi^k. \Longrightarrow e^{k+1} = e^k - \alpha(Ae^k) = (I - \alpha A)e^k, k = 0, 1, 2 \cdots$

$\therefore e^{k+1} = (I - \alpha A)^{k+1} e^0.$

$\lim_{k\to\infty} e^{k+1} = 0$ for every $e^0 \Longleftrightarrow \lim_{k\to\infty} (I - \alpha A)^{k+1} e^0 = 0$ for every e^0

$\Longleftrightarrow \rho(I - \alpha A) < 1 \Longleftrightarrow \max_j |1 - \alpha\lambda_j| < 1$
$\Longleftrightarrow -1 < 1 - \alpha\lambda_j < 1, j = 1, 2, \cdots, M$

$\Longleftrightarrow 1 - \alpha\lambda_{\max} > -1 \Longleftrightarrow \alpha\lambda_{\max} < 2.$

The gradient method with constant step length (continued)

If we choose $\alpha = \frac{1}{\lambda_{\max}} > 0$, then we have

$$\begin{aligned} \|e^{k+1}\|_2 &= \|(I - \alpha A)e^k\|_2 \le \|I - \alpha A\|_2 \|e^k\|_2 \le \left(1 - \frac{1}{\lambda_{\max}} \lambda_{\min}\right) \|e^k\|_2 \\ &= \left(1 - \frac{1}{\kappa(A)}\right) \|e^k\|_2. \end{aligned}$$

$\therefore \|e^k\|_2 \le \left(1 - \frac{1}{\kappa(A)}\right)^k \|e^0\|_2$ (small $\kappa(A)$ is better).

Given $0 < \varepsilon < 1$, find the smallest n such that $\|e^n\|_2 \le \varepsilon \|e^0\|_2$.

$\therefore$ We require $\left(1 - \frac{1}{\kappa(A)}\right)^n \le \varepsilon$.

The gradient method with constant step length (continued)

$$\left(1-\frac{1}{\kappa(A)}\right)^n \le \varepsilon \iff n\ln\left(1-\frac{1}{\kappa(A)}\right) \le \ln(\varepsilon)$$

$$\iff n\left(-\ln\left(1-\frac{1}{\kappa(A)}\right)\right) \ge \ln\left(\frac{1}{\varepsilon}\right) \iff n \ge \frac{\ln\left(\frac{1}{\varepsilon}\right)}{-\ln\left(1-\frac{1}{\kappa(A)}\right)}.$$

$$\because -\ln(1-x) = \sum_{i=1}^{\infty}\frac{x^i}{i} > x \text{ for } 0<x<1.$$

$$\therefore -\ln\left(1-\frac{1}{\kappa(A)}\right) > \frac{1}{\kappa(A)}.$$

$\therefore$ We take $n \ge \kappa(A)\ln\left(\frac{1}{\varepsilon}\right)$.

$\therefore$ The required number of iterations in the gradient method is proportional to the condition number $\kappa(A)$. If $\kappa(A)$ is large, then the gradient method is not efficient.

The conjugate gradient method

- Roughly speaking, the conjugate gradient method $\approx$ the gradient method + optimal step length, but with different search direction.
- Let A be a SPD real $M \times M$ matrix. Define $< \zeta, \eta > := \zeta \cdot A\eta$, $\forall\, \zeta, \eta \in \mathbb{R}^M$. Then $< \cdot, \cdot >$ is a scalar product on $\mathbb{R}^M$.

  Proof: check
  - it is a symmetric bilinear form;
  - $< v, v > \geq 0 \;\forall\, v \in \mathbb{R}^M$, and $< v, v > = 0 \iff v = 0$.
- Define the energy norm: $\|\eta\|_A := < \eta, \eta >^{1/2}, \forall\, \eta \in \mathbb{R}^M$.

The conjugate gradient method (continued)

Given $\xi^0 \in \mathbb{R}^M$, $d^0 := -r^0 := -f'(\xi^0) = -(A\xi^0 - b)$,

find ξ^1 & d^1, ξ^2 & $d^2, \cdots$, such that for $k = 0, 1, \cdots$,

$$\begin{aligned} \xi^{k+1} &= \xi^k + \alpha_k d^k, \\ \alpha_k &= -\frac{r^k \cdot d^k}{< d^k, d^k >} \qquad \textit{(optimal step length)}, \\ d^{k+1} &= -r^{k+1} + \beta_k d^k \qquad \textit{(for next step)}, \end{aligned}$$

where

$$\begin{aligned} r^k &:= f'(\xi^k) = A\xi^k - b, \\ \beta_k &:= \frac{< r^{k+1}, d^k >}{< d^k, d^k >}. \end{aligned}$$

Some remarks

- The new search direction d^{k+1} is a linear combination of r^{k+1} and the old search direction d^k.
- Notice that

$$\beta_k = \frac{< r^{k+1}, d^k >}{< d^k, d^k >} \quad \Longleftrightarrow \quad \beta_k < d^k, d^k > - < r^{k+1}, d^k >= 0$$
$$\Longleftrightarrow \quad < -r^{k+1} + \beta_k d^k, d^k >=< d^{k+1}, d^k >= 0.$$

- Suppose that $d^0, d^1, \cdots, d^{k-1} \neq 0$. If $d^k = 0$ then

$$-r^k + \beta_{k-1} d^{k-1} = 0 \Longrightarrow r^k = \beta_{k-1} d^{k-1} = \frac{< r^k, d^{k-1} >}{< d^{k-1}, d^{k-1} >} d^{k-1}$$
$$\Longrightarrow \cdots \Longrightarrow r^k = 0 \,?$$

- α_k is the optimal step length.

Lemma 1

Notation: Let $\eta^0, \eta^1, \cdots, \eta^m \in \mathbb{R}^M$. Define $[\eta^0, \eta^1, \cdots, \eta^m] := \text{span}\{\eta^0, \eta^1, \cdots, \eta^m\}$.

Lemma 1: For $m = 0, 1, \cdots$, we have

$$[d^0, d^1, \cdots, d^m] = [r^0, r^1, \cdots, r^m] = [r^0, Ar^0, \cdots, A^m r^0].$$

Proof: We will use the induction to prove the assertion.
$m = 0$: It is trivial, since $[d^0] = [-r^0] = [r^0] = [A^0 r^0]$.
Suppose that the assertion holds for $m \leq k$. Consider the case $m = k$, we have $[d^0, d^1, \cdots, d^k] = [r^0, r^1, \cdots, r^k] = [r^0, Ar^0, \cdots, A^k r^0]$.

$\because \xi^{k+1} = \xi^k + \alpha_k d^k$.

$\therefore A\xi^{k+1} = A\xi^k + \alpha_k A d^k$.

$\therefore A\xi^{k+1} - b = A\xi^k - b + \alpha_k A d^k$.

$\therefore r^{k+1} = r^k + \alpha_k A d^k$.

Proof of Lemma 1 (continued)

$\because d^k \in [r^0, Ar^0, \cdots, A^k r^0]$ and $r^k \in [r^0, Ar^0, \cdots, A^k r^0]$.

$\therefore Ad^k \in [r^0, Ar^0, \cdots, A^{k+1} r^0]$ and
$r^{k+1} = r^k + \alpha_k Ad^k \in [r^0, Ar^0, \cdots, A^{k+1} r^0]$.

$\therefore [r^0, r^1, \cdots, r^{k+1}] \subseteq [r^0, Ar^0, \cdots, A^{k+1} r^0]$.

$\because A^k r^0 \in [d^0, d^1, \cdots, d^k] = [r^0, r^1, \cdots, r^k] = [r^0, A^1 r^0, \cdots, A^k r^0]$.

$\therefore A^{k+1} r^0 \in [Ad^0, Ad^1, \cdots, Ad^k]$.

Notice that $d^0 \in [r^0] \;\Rightarrow\; Ad^0 \in [r^0, Ar^0] = [r^0, r^1]$.

Similarly, $Ad^1 \in [r^0, r^1, r^2], \cdots, Ad^{k-1} \in [r^0, r^1, \cdots, r^k]$,

and $r^{k+1} = r^k + \alpha_k Ad^k$ implies $Ad^k \in [r^k, r^{k+1}]$.

$\therefore A^{k+1} r^0 \in [r^0, r^1, \cdots, r^{k+1}]$.
$\therefore [r^0, Ar^0, \cdots, A^{k+1} r^0] \subseteq (\Rightarrow =) [r^0, r^1, \cdots, r^{k+1}]$.

On the other hand,

$\because [r^0, r^1, \cdots, r^k] = [d^0, d^1, \cdots, d^k]$ and $d^{k+1} = -r^{k+1} + \beta_k d^k$.

$\therefore [r^0, r^1, \cdots, r^{k+1}] = [d^0, d^1, \cdots, d^{k+1}]$.

Lemma 2

- $r^i \cdot r^j = 0$ if $i \neq j$ (orthogonal).
- $< d^i, d^j >= 0$ if $i \neq j$ (conjugate).

Proof: We use induction on n $(i, j \leq n)$.

$n = 1$:

- $\because r^1 = r^0 + \alpha_0 A d^0$ with $\alpha_0 = \dfrac{-r^0 \cdot d^0}{< d^0, d^0 >}$, $r^0 = -d^0$.

  $\therefore r^1 \cdot r^0 = (-d^0) \cdot (-d^0) - \dfrac{-d^0 \cdot d^0}{< d^0, d^0 >} A d^0 \cdot (-d^0) =$
  $d^0 \cdot d^0 - (d^0 \cdot d^0) = 0.$
- $< d^1, d^0 >=< -r^1 + \beta_0 d^0, d^0 >=< -r^1, d^0 > +\beta_0 < d^0, d^0 >=$
  $- < r^1, d^0 > + \dfrac{< r^1, d^0 >}{< d^0, d^0 >} < d^0, d^0 >= 0.$

  Note: If $< d^0, d^0 >= 0 \iff d^0 \cdot A d^0 = 0 \iff d^0 = 0 \iff r^0 =$
  $0 \iff A\xi^0 - b = 0 \iff A\xi^0 = b$.

Proof of Lemma 2 (continued)

Suppose that these two properties hold for $n \leq k$.

$\because [d^0, d^1, \cdots, d^{k-1}] = [r^0, r^1, \cdots, r^{k-1}]$

$\therefore r^k \cdot d^j = 0$ for $j = 0, 1, \cdots, k-1$

$\because r^{k+1} = r^k + \alpha_k + Ad^k$

$\therefore$ For $j = 0, 1, \cdots, k-1$, $r^{k+1} \cdot d^j = r^k \cdot d^j + \alpha_k < d^k, d^j >= 0$

Notice that

$$\begin{aligned} r^{k+1} \cdot d^k &= f'(\xi^{k+1}) \cdot d^k = f'(\xi^k + \alpha_k d^k) \cdot d^k \\ &= \frac{d}{d\alpha} f(\xi^k + \alpha d^k)|_{\alpha=\alpha_k} = 0 \qquad (\because\ \alpha_k \textit{ is optimal}). \end{aligned}$$

$\therefore r^{k+1} \cdot d^j = 0$ for $j = 0, 1, \cdots, k$

$\because [r^0, r^1, \cdots, r^k] = [d^0, d^1, \cdots, d^k]$

$\therefore r^{k+1} \cdot r^j = 0$ for $j = 0, 1, \cdots, k$. That is, the first property holds.

Proof of Lemma 2 (continued)

$\because r^{k+1} = r^k + \alpha_k + Ad^k.$

$\therefore Ad^j \in [r^0, r^1, \cdots, r^{j+1}]$ for any $j = 0, 1, \cdots$

$\therefore r^{k+1} \cdot Ad^j =< r^{k+1}, d^j >= 0$ for $j = 0, 1, \cdots, k-1.$

$\therefore < d^{k+1}, d^j >=< -r^{k+1}, d^j > +\beta_k < d^k, d^j >= 0 + 0 = 0$ for $j = 0, 1, \cdots, k-1.$

$\because$

$$< d^{k+1}, d^k >=< -r^{k+1} + \beta_k d^k, d^k >= - < r^{k+1}, d^k > +\beta_k < d^k, d^k >$$

$$= - < r^{k+1}, d^k > + \frac{< r^{k+1}, d^k >}{< d^k, d^k >} < d^k, d^k >= 0.$$

$\therefore < d^{k+1}, d^j >= 0$ for $j = 0, 1, \cdots, k.$

$\therefore$ The second property holds.

Theorem on the conjugate gradient method

$\exists\, m \le M$ such that $A\xi^m = b$.

Proof:

$\because r^j, j = 0, 1, 2, \cdots$ are pairwise orthogonal
($\Rightarrow$ linearly independent if nonzero) and $\dim \mathbb{R}^M = M$

$\therefore \exists\, r^m \in \{r^0, r^1, \cdots, r^M\}, 0 \le m \le M$, such that $r^m = 0$

$\therefore A\xi^m - b = 0 \Rightarrow A\xi^m = b$

Theorem on the conjugate gradient method (continued)

- **Theorem:** *Let x be the exact solution, then*

$$\|x - x^k\|_A \le 2 \left(\frac{\sqrt{\kappa(A)} - 1}{\sqrt{\kappa(A)} + 1} \right)^k \|x - x_0\|_A.$$

- In order to have

$$\|x - x^k\|_A \le \varepsilon \|x - x^0\|_A,$$

  for some given ϵ, we must have

$$n \ge \frac{1}{2} \sqrt{\kappa(A)} \ln \frac{2}{\varepsilon}.$$

- Compare with the gradient method with constant step length

$$n \ge \kappa(A) \ln \frac{1}{\varepsilon}.$$

  The number of iterations is large for ill-conditioned matrices.

- Can we change the condition number without changing the solution of a given system?

Preconditioning

(1) $$\min_{\eta \in \mathbb{R}^M} f(\eta) = \min_{\eta \in \mathbb{R}^M} \left(\frac{1}{2} \eta \cdot A\eta - b \cdot \eta \right).$$

The gradient method with constant step length α is

$$\eta^{k+1} = \eta^k - \alpha(A\eta^k - b).$$

Let E be a nonsingular $M \times M$ matrix. Let $\zeta = E\eta \Longrightarrow \eta = E^{-1}\zeta$. Then

$$\begin{aligned} \widetilde{f}(\zeta) &:= f(\eta) = f(E^{-1}\zeta) = \frac{1}{2}(E^{-1}\zeta) \cdot A(E^{-1}\zeta) - b \cdot E^{-1}\zeta \\ &= \frac{1}{2}\zeta \cdot E^{-\top}AE^{-1}\zeta - E^{-\top}b \cdot \zeta = \frac{1}{2}\zeta \cdot \widetilde{A}\zeta - \widetilde{b} \cdot \zeta, \end{aligned}$$

where $\widetilde{A} := E^{-\top}AE^{-1}$ and $\widetilde{b} := E^{-\top}b$.

Preconditioning (continued)

$$\text{(2)} \qquad \min_{\zeta \in \mathbb{R}^M} \left(\frac{1}{2} \zeta \cdot \widetilde{A} \zeta - \widetilde{b} \cdot \zeta \right).$$

The gradient method with constant step length α is

$$\zeta^{k+1} = \zeta^k - \alpha(\widetilde{A}\zeta^k - \widetilde{b}).$$

If $\kappa(\widetilde{A}) \ll \kappa(A)$ then the gradient method for problem (2) will converge much faster than the same method applied to problem (1).

Preconditioning (continued)

$\because \zeta = E\eta.$

$\therefore E\eta^{k+1} = E\eta^k - \alpha(\widetilde{A}E\eta^k - \widetilde{b}).$

$\therefore$
$\eta^{k+1} = \eta^k - \alpha E^{-1}(E^{-\top}AE^{-1}E\eta^k - E^{-\top}b) = \eta^k - \alpha E^{-1}E^{-\top}(A\eta^k - b).$
Let $C := E^\top E$. Then $C^{-1} = E^{-1}E^{-\top}$ and

$$\eta^{k+1} = \eta^k - \alpha C^{-1}(A\eta^k - b).$$

This is the preconditioned version of the gradient method for problem (1) with preconditioner C.

To compute η^{k+1} from η^k, we have to solve

$$C\theta^k = (A\eta^k - b).$$

Note that do not need the explicit form of C^{-1}.