matrix algebra

Numerical Methods for IT

solution of linear algebraic equations

Gaussian elimination with Back-substitution

The basic idea of Gauss-Jordin elimination is to add or subtract linear combination of the given equations $\boldsymbol{Ax} = \boldsymbol{b}$ until each equation contains only one of the unknown, thus giving an immediate solution.
Gauss-Jordin elimination produces both the solution of the equations for one or more right-hand side vector $\boldsymbol{b}$, $\boldsymbol{b} = [\boldsymbol{b_1} \dots \boldsymbol{b_m}]$, and the matrix inverse $\boldsymbol{A^{-1}}$.

Gaussian elimination reduces a matrix not all the way to the identity matrix, but only halfway, to a matrix whose components on the diagonal and above.

$$\begin{bmatrix} a_{11} a_{12} a_{13} a_{14} \\ a_{21} a_{22} a_{23} a_{24} \\ a_{31} a_{23} a_{33} a_{34} \\ a_{41} a_{24} a_{43} a_{44} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} \Longrightarrow \begin{bmatrix} a'_{11} a'_{12} a'_{13} a'_{14} \\ 0 \;\; a'_{22} a'_{23} a'_{24} \\ 0 \;\; 0 \;\; a'_{33} a'_{34} \\ 0 \;\; 0 \;\; 0 \;\; a'_{44} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} b'_1 \\ b'_2 \\ b'_3 \\ b'_4 \end{bmatrix} \Longleftrightarrow \begin{matrix} x_4 = \dfrac{b'_{44}}{a'_{44}} \\ x_i = \dfrac{\left(b'_i - \sum_{j=i+1}^{n} a'_{i} * x_j\right)}{a'_{ii}} \end{matrix}$$

Iterative improvement of a solution

Suppose that a vector $\boldsymbol{x}$ is the exact solution of

$\boldsymbol{A} \cdot \boldsymbol{x} = \boldsymbol{b}$

We only know some lightly wrong solution

$\boldsymbol{x} + \boldsymbol{\delta x}$,

where $\boldsymbol{\delta x}$ is the unknow error.

So

$\boldsymbol{A} \cdot (\boldsymbol{x} + \boldsymbol{\delta x}) = \boldsymbol{b} + \boldsymbol{\delta b}.$

$\boldsymbol{A} \cdot \boldsymbol{\delta x} = \boldsymbol{\delta b}.$

$\boldsymbol{A} \cdot \boldsymbol{\delta x} = \boldsymbol{A} \cdot (\boldsymbol{x} + \boldsymbol{\delta x}) - \boldsymbol{b}.$

Let $\boldsymbol{X} = \boldsymbol{\delta x}$, $\boldsymbol{B} = \boldsymbol{A} \cdot (\boldsymbol{x} + \boldsymbol{\delta x}) - \boldsymbol{b}$

Solve $\boldsymbol{A} \cdot \boldsymbol{X} = \boldsymbol{B}$ using Gaussian elimination with Back-substitution.

inverse matrix

definition

$A \in \mathbb{R}^{n\times n}$ is invertible $\Leftrightarrow \exists X \in \mathbb{R}^{n\times n}: X \cdot A = I = A \cdot X$. Denote $X = A^{-1}$

Given an invertible $A \in \mathbb{R}^{n\times n}, A^{-1} = X = [X_1 \dots X_n]$ will be the solution of $A \cdot X = I = [e_1 \dots e_i]$

Sherman-Morrison formula

Suppose that we have A^{-1} which is invertible matrix of $A \in \mathbb{R}^{n\times n}$.
Now we want to make a small change in A, for example change one element a_{ij}, or a few elements, or one row, or one column:

$A \to (A + u \otimes v); u, v \in \mathbb{R}^n$, where $u \otimes v = \left[w_{ij} = u_i \times v_j\right] \in \mathbb{R}^{n\times n}$.

The Sherman-Morrison formula give the inverse $(A + u \otimes v)^{-1}$:

$(A + u \otimes v)^{-1} = \frac{(A^{-1} \cdot u) \otimes (v \cdot A^{-1})}{\lambda}$, where $\lambda = v \cdot A^{-1} \cdot u$.

Given A^{-1} and the vectors u and v, we need only perform two matrix multiplications and a vector dot product, $z \equiv A^{-1} \cdot u, w \equiv (A^{-1})^T \cdot v, \lambda = v \cdot z$, to get the desired change in the inverse:

$A^{-1} \to A^{-1} - \frac{z \otimes w}{1+\lambda}$.

For some other sparse problem, the Sherman-Morris formula cannot be directly applied for the simple reason that the storage of whole inverse matrix A^{-1} is not feasible.
To add only a single correction ò the form $u \otimes v$ and solve the linear system $(A + u \otimes v) \cdot x = b$:
. Solve the two auxiliary problem $A \cdot y = b$ and $A \cdot x = u$ for the vectors y and z.
. And $x = y - \left[\frac{v \cdot y}{1 + (v \cdot z)}\right]$.

eigen value and eigen vector

Definition

Given $A \in \mathbb{R}^{n\times n}$

$$\det(A) = \begin{cases} a_{11}, n = 1 \\ \Sigma_{i=1}^{n}(-1)^{i+j}\det\left(M_{ij}\right) = \Sigma_{j=1}^{n}(-1)^{i+j}\det\left(M_{ij}\right), n > 1 \end{cases}$$

where M_{ij} is a matrix of order $(n-1)$ extracted from A by removing row i and column j.

Given $A \in \mathbb{R}^{n\times n}$. If there are $0 \neq x \in \mathbb{R}^n$ and $\lambda \in \mathbb{R}$ such that $A \cdot x = \lambda x$, then λ is called an eigenvalue of A, and x – eigenvector corresponding to λ.

To find λ and x, because
$\mathrm{A} \cdot x = \lambda x \iff A \cdot x = \lambda x \cdot I \iff (\lambda I - A) \cdot x = 0, \lambda \neq 0\ and\ x \neq 0,$
So $\det(\lambda I - A) = 0 = p(\lambda) = \lambda^n + c_1\lambda^{n-1} + \cdots + c_n\lambda^0$. $p(\lambda)$ is called the characteristic polynomial of A.
. Eigenvalue λ is a solution of the equation $p(\lambda) = 0$.
. Given eigen λ, equation system $(\lambda I - A) \cdot x = 0$ has a non-trivial solution x which is eigenvector.

Matrix Diagonalization

Given $A, B \in \mathbb{R}^{n\times n}$ are called similar if there exists an invertible matrix P such that $B = P^{-1} \cdot A \cdot P$

Let $P = [p_1 \dots p_n]$ and suppose that $D = \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_n \end{bmatrix}$,

$A \cdot P = A \cdot [p_1 \dots p_n] = [A \cdot p_1 \dots A \cdot p_n]$ and $P \cdot D = [\lambda_1 p_1 \dots \lambda_n p_n]$.
If $p_1, \dots, p_n$ are linearly independent, and $\lambda_1, \dots, \lambda_n$ are eigenvalue of A, then
$A \cdot P = A \cdot [p_1 \dots p_n] = [A \cdot p_1 \dots A \cdot p_n] = [\lambda_1 p_1 \dots \lambda_n p_n] = P \cdot D$

Algorithm (Diagonalization)
. If $p_1, \dots, p_n$ are distinct eigenvectors then A is diagonalizable.
. Let $P = [p_1 \dots p_n]$
. Then $D = P^{-1} \cdot A \cdot P$

matrix decomposition

QR decomposition

Problem: Given $u_1, \dots, u_r \in \mathbb{R}^n$. Find $v_1, \dots, v_r$: $< v_i, v_{j\neq i} > = 0$, if $u_1, \dots, u_r$ is a family of linearly independent vectors.

Algorithm (Gram-Schmidt)

. $v_1 = u_1$ (if $v_1 = 0$ then $\{u_1, \dots, u_r\}$ is linearly dependent family).

. $v_i = u_i - \Sigma_{j=1}^{i-1} \frac{(<u_i, v_j>)}{\|v_j\|^2} v_j, i = 2, \dots, r$ (if $v_j = 0$ then $\{u_1, \dots, u_r\}$ is linearly dependent family).

. $q_i = \frac{v_i}{\|v_i\|}, i = 1, \dots, r$ (if we need a family of orthonormal vectors).

Proposition: If $A = [u_1 \dots u_n] \in \mathbb{R}^{m\times n}$ has n linearly independent column vectors, then A can decompose to $A = Q \cdot R$, where $Q = [q_1 \dots q_n] \in \mathbb{R}^{m\times n}$ which has n orthonormal vectors, and $R \in \mathbb{R}^{n\times n}$ is an invertible upper triangular matrix.

Algorithm (QR-decomposition)

. $\mathrm{u}_1, \dots, u_n \leftarrow A$

. $Q \leftarrow$ Gram-Schmidt($u_1, \dots, u_n$)

. $q_1, \dots, q_n \leftarrow Q$

. $R = \begin{cases} < u_i, q_j > if\ i \leq j \\ 0, \qquad\qquad else \end{cases}$

LU decomposition

Problem: Given $A \in \mathbb{R}^{n\times n}$. Find a lower triangular L and a upper triangular U such that $A = L \cdot U$:

$$\begin{bmatrix} \alpha_{11} & 0 & 0 & 0 \\ \alpha_{21} & \alpha_{22} & 0 & 0 \\ \alpha_{31} & \alpha_{32} & \alpha_{33} & 0 \\ \alpha_{41} & \alpha_{42} & \alpha_{43} & \alpha_{44} \end{bmatrix} \cdot \begin{bmatrix} \beta_{11} & \beta_{12} & \beta_{13} & \beta_{14} \\ 0 & \beta_{22} & \beta_{23} & \beta_{24} \\ 0 & 0 & \beta_{33} & \beta_{34} \\ 0 & 0 & 0 & \beta_{44} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{23} & a_{33} & a_{34} \\ a_{41} & a_{24} & a_{43} & a_{44} \end{bmatrix} \Leftrightarrow \alpha_{i1}\beta_{1j} + \alpha_{i2}\beta_{2j} + \cdots = a_{ij}; i, j = 1, \ldots, n$$

We have:
$$\begin{aligned} i < j: &\quad \alpha_{i1}\beta_{1j} + \cdots + \alpha_{ii}\beta_{ij} = a_{ij} \\ i = j: &\quad \alpha_{i1}\beta_{1j} + \cdots + \alpha_{ii}\beta_{jj} = a_{ij} \\ i > j: &\quad \alpha_{i1}\beta_{1j} + \cdots + \alpha_{ij}\beta_{jj} = a_{ij} \end{aligned}$$
. There are total n^2 equations for $n^2 + n$ unknows α, β.

Algorithm (LU decomposition)

. Set $\alpha_{ii} = 1, \forall i = 1, \ldots, n$

. For each $j = 1,2, \ldots, n$:

(a) $\forall i = 1, \ldots, j$: $\beta_{ij} = a_{ij} - \Sigma_{k=1}^{i-1} \alpha_{ik}\beta_{kj}$,

(b) $\forall i = j + 1, \ldots, N$: $\alpha_{ij} = \frac{1}{\beta_{jj}}(a_{ij} - \Sigma_{k=0}^{j-1} \alpha_{ik}\beta_{kj})$

Application of LU decomposition: $\boldsymbol{A = L \cdot U}$

(1.) **Determinant**

$\det(A) = \det(L \cdot U) = \det(L) \times \det(U) = \Pi_{i=1}^{n} \beta_{ii}$.

(2.) **Solve equations**

$A \cdot x = b \Leftrightarrow (L \cdot U)x = b \Leftrightarrow L(U \cdot x) = b$

$$\Leftrightarrow \begin{cases} L \cdot y = b \\ U \cdot x = y \end{cases}$$

SVD – Singular Value Decomposition

In many cases where Gaussian elimination and LU decomposition fail to satisfactory results. SVD will diagnose precisely what the problem is.
SVD is also the method of choice for solving most linear least-square problems.

Theorem. Any $A \in \mathbb{R}^{M \times N}$ can be written as the product of a column-orthogonal matrix $U \in \mathbb{R}^{M \times N}$, a diagonal matrix W $\in \mathbb{R}^{N \times N}$ with positive or zero elements (the singular), and the transpose of orthogonal matrix $V \in \mathbb{R}^{N \times N}$.

Algorithm (SVD)
(a) $V = [v_1 \dots v_n] \leftarrow Gram - Schmid$ of $A^T \cdot A$
(b) Let $\sigma_1 = \sqrt{\lambda_1}, \dots, \sigma_k = \sqrt{\lambda_k}$, with $\lambda_1, \dots, \lambda_k$ are eigenvalues of $A^T \cdot A$, then $D = \begin{bmatrix} \sigma_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_k \end{bmatrix}$
(c) The column vectors of V are arranged in order corresponding to $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k > 0$.
(d) $u_i = \frac{A \cdot v_i}{\|A \cdot v_i\|} = \frac{1}{\sigma_i} A \cdot v_i, i = 1, \dots, k$.
(e) $\{u_1, \dots, u_k\}$ is an orthogonal basis for A.
(f) $\{u_1, \dots, u_k, u_{k+1}, \dots, u_M\}$ is a span of $\{u_1, \dots, u_k\}$ to an orthonormal basis for $\mathbb{R}^N$.