4.MatrixInversion

April 15, 2025

1 STAT 207: Linear Regression and Matrix Inversion

Zhe Fei (zhe.fei@ucr.edu)

- NAS Chapter 7

1.1 Linear Regressions

- Linear regression is the most commonly applied procedure in statistics.
- Solving linear least squares problems quickly and reliably

Four methods for solving linear least squares problems:

- Sweeping, uses the symmetry of matrices and is conceptual simple
- Cholesky decomposition, a lower triangular square root of a positive definite matrix
- Modified Gram-Schmidt procedure, numerically more stable
- Orthogonalization by Householder reflections

1.1.1 The Sweep Operator

The popular statistical software SAS uses sweep operator for linear regression and matrix inversion.

Motivation:

A random vector $X \in \mathbb{R}^p$ with mean vector μ, covariance matrix Σ, and density

$$(2\pi)^{-p/2}|\Sigma|^{-1/2}e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)}$$

is said to follow a multivariate normal distribution.

The sweep operator permits straightforward calculation of the quadratic form $(x-\mu)^T\Sigma^{-1}(x-\mu)$ and the determinant of Σ. If we partition X and its mean and covariance so that

$$X = \begin{bmatrix} Y \\ Z \end{bmatrix}, \qquad \mu = \begin{bmatrix} \mu_Y \\ \mu_Z \end{bmatrix}, \qquad \Sigma = \begin{bmatrix} \Sigma_Y & \Sigma_{YZ} \\ \Sigma_{ZY} & \Sigma_Z \end{bmatrix},$$

then conditional on the event $Y = y$, the subvector Z follows a multivariate normal density with conditional mean and variance

$$E(Z|Y=y) = \mu_Z + \Sigma_{ZY}\Sigma_Y^{-1}(y-\mu_Y),$$
$$\mathrm{Var}(Z|Y=y) = \Sigma_Z - \Sigma_{ZY}\Sigma_Y^{-1}\Sigma_{YZ}.$$

These quantities and the conditional density of Z given $Y = y$ can all be easily evaluated via the sweep operator.

Definition:

Suppose A is an $m \times m$ symmetric matrix.

Sweep on the kth diagonal entry $a_{kk} \neq 0$ of A yields a new symmetric matrix $\widehat{A} = (\hat{a}_{ij})$ with entries

$$\begin{aligned}
\hat{a}_{kk} &= -\frac{1}{a_{kk}} \\
\hat{a}_{ik} &= \frac{a_{ik}}{a_{kk}}, \quad i \neq k \\
\hat{a}_{kj} &= \frac{a_{kj}}{a_{kk}}, \quad j \neq k \\
\hat{a}_{ij} &= a_{ij} - \frac{a_{ik}a_{kj}}{a_{kk}}, \quad i \neq k, j \neq k.
\end{aligned}$$

Inverse sweep sends A to $\check{A} = (\check{a}_{ij})$ with entries

$$\begin{aligned}
\check{a}_{kk} &= -\frac{1}{a_{kk}} \\
\check{a}_{ik} &= -\frac{a_{ik}}{a_{kk}}, \quad i \neq k \\
\check{a}_{kj} &= -\frac{a_{kj}}{a_{kk}}, \quad j \neq k \\
\check{a}_{ij} &= a_{ij} - \frac{a_{ik}a_{kj}}{a_{kk}}, \quad i \neq k, j \neq k.
\end{aligned}$$

- $\check{\hat{A}} = A$
- Successively sweeping all diagonal entries of A yields $-A^{-1}$
- Exercise: Invert the 2×2 matrix using the sweep operator:

$$A = \begin{pmatrix} 4 & 3 \\ 3 & 2 \end{pmatrix}$$

Block form of sweep:
Let the symmetric matrix A be partitioned as

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$$

If possible, sweeping on the diagonal entries of A_{11} yields

$$\hat{A} = \begin{pmatrix} -A_{11}^{-1} & A_{11}^{-1}A_{12} \\ A_{21}A_{11}^{-1} & A_{22} - A_{21}A_{11}^{-1}A_{12} \end{pmatrix}$$

NAS Proposition 7.5.3

- A symmetric matrix A is positive definite if and only if each diagonal entry can be swept in succession and is positive until it is swept.
- When a diagonal entry of a positive definite matrix A is swept, it becomes negative and remains negative thereafter.
- Furthermore, taking the product of the diagonal entries just before each is swept yields the determinant of A.
$$\det A = \det(A_{11}) \det(A_{22} - A_{21} A_{11}^{-1} A_{12})$$

Applications of Sweeping In **linear regression**, start with the matrix

$$\begin{bmatrix} X^T X & X^T y \\ y^T X & y^T y \end{bmatrix}$$

and sweep on the diagonal entries of $X^T X$. Then the basic theoretical ingredients

$$\begin{aligned} &\begin{bmatrix} -(X^T X)^{-1} & (X^T X)^{-1} X^T y \\ y^T X (X^T X)^{-1} & y^T y - y^T X (X^T X)^{-1} X^T y \end{bmatrix} \\ = &\begin{bmatrix} -\frac{1}{\sigma^2} \mathrm{Var}(\hat{\beta}) & \hat{\beta} \\ \hat{\beta}^T & \|y - \hat{y}\|_2^2 \end{bmatrix} \end{aligned}$$

magically emerge.

Multivariate normal: perform sweeping on the diagonal entries of Σ for the matrix

$$\begin{bmatrix} \Sigma & x - \mu \\ x^T - \mu^T & 0 \end{bmatrix},$$

we get the quadratic form $-(x-\mu)^T \Sigma^{-1} (x - \mu)$ in the lower-right block of the swept matrix.

- In the process we can also accumulate $\det \Sigma$.
- To avoid underflows and overflows, it is better to compute $\ln \det \Sigma$ by summing the logarithms of the diagonal entries as we sweep on them.

Conditional mean and variance: assume $X = (Y^T, Z^T)^T$, and sweep on the upper-left block of

$$\begin{bmatrix} \Sigma_Y & \Sigma_{YZ} & \mu_Y - y \\ \Sigma_{ZY} & \Sigma_Z & \mu_Z \\ (\mu_Y - y)^T & \mu_Z^T & 0 \end{bmatrix},$$

we get

$$\begin{aligned} E(Z|Y = y) &= \mu_Z + \Sigma_{ZY} \Sigma_Y^{-1} (y - \mu_Y), \\ \mathrm{Var}(Z|Y = y) &= \Sigma_Z - \Sigma_{ZY} \Sigma_Y^{-1} \Sigma_{YZ}. \end{aligned}$$

Exercise: implement the sweep operator, in python, it should start with:

[1]:
```
def sweep(A, k):


    return A_hat
```

1.1.2 Cholesky Decompositions

André-Louis Cholesky was a French military officer, geodesist, and mathematician.

From a collegue:

> The structure should be exploited whenever solving a problem.

Common structures include: symmetry, positive (semi)definiteness, sparsity, low rank, ...

Let A be an $m \times m$ positive definite matrix. The Cholesky decomposition L of A is a lower-triangular matrix with positive diagonal entries that serves as an asymmetric square root of A.

How to show such L **exists and is unique**? By induction.

For $m > 1$, the square root condition $A = LL^T$ can be written as

$$\begin{pmatrix} a_{11} & a^T \\ a & A_{22} \end{pmatrix} = \begin{pmatrix} \ell_{11} & 0^T \\ \ell & L_{22} \end{pmatrix} \begin{pmatrix} \ell_{11} & \ell^T \\ 0 & L_{22}^T \end{pmatrix},$$

which should satisfy

$$\begin{aligned} a_{11} &= \ell_{11}^2 \\ a &= \ell_{11}\ell \\ A_{22} &= \ell\ell^T + L_{22}L_{22}^T. \end{aligned}$$

Solving these equations gives

$$\begin{aligned} \ell_{11} &= \sqrt{a_{11}} \\ \ell &= \ell_{11}^{-1}a \\ L_{22}L_{22}^T &= A_{22} - \ell\ell^T. \end{aligned}$$

- This proof is constructive and can be easily implemented in computer code.
- Can compute $\det A$.
- Positive semidefinite matrices also possess Cholesky decompositions.

Regression analysis: with

$$(X, y)^T(X, y) = \begin{bmatrix} X^TX & X^Ty \\ y^TX & y^Ty \end{bmatrix} = \begin{bmatrix} L & 0 \\ \ell^T & d \end{bmatrix} \begin{bmatrix} L^T & \ell \\ 0^T & d \end{bmatrix}$$

then

$$L\ell = X^T y, \quad L^T\beta = \ell$$

and

$$d^2 =$$

- **Forward substitution** to solve $Lf = v$:

$$\begin{aligned} f_1 &= \ell_{11}^{-1} v_1 \\ f_2 &= \\ &... \end{aligned}$$

- **Backward substitution** to solve $Ub = w$:

$$\begin{aligned} b_m &= u_{mm}^{-1} w_m \\ b_{m-1} &= \\ &... \end{aligned}$$

1.1.3 Gram-Schmidt Orthogonalization

- QR decomposition of a $p \times q$ matrix X, where Q is $p \times q$ with orthonormal columns and R is a $q \times q$ invertible upper-triangular matrix.
- How to determine Q and R:

Gram-Schmidt orthogonalization takes a collection of vectors such as the columns $x_1, ..., x_q$ of the design matrix X into an orthonormal collection of vectors $u_1, ..., u_q$ spanning the same column space.

$$u_1 = \frac{1}{\|x_1\|_2} x_1.$$

Given $u_1, \dots, u_{k-1}$, the next unit vector u_k in the sequence is defined by dividing the column vector

$$v_k = x_k - \sum_{j=1}^{k-1} (u_j^T x_k) u_j$$

by its norm,

$$u_k = \frac{v_k}{\|v_k\|_2}.$$

The upper-triangular entries of the matrix R are given by the formulas

$$r_{jk} = u_j^T x_k \quad \text{for } 1 \leq j < k,$$

and

$$r_{kk} = \|v_k\|_2,$$

where $v_k = x_k - \sum_{j=1}^{k-1} r_{jk} u_j$.

1.1.4 Householder Orthogonalization

Another way to construct the QR decompostion

$$H_{q-1}...H_2H_1X = \begin{pmatrix} R \\ 0 \end{pmatrix},$$

where R is $q \times q$ upper triangular with positive diagonal entries. Let $O = H_{q-1}...H_2H_1$ the orthogonal matrix, we can derive

$$\hat{\beta} = R^{-1}r_1, X = O_q^T R.$$

1.1.5 Comparison of the Different Algorithms

The various methods are listed in order of their numerical accuracy as rated by Seber and Lee.

Method	Flop Count
Sweeping	$pq^2 + q^3$
Cholesky Decomposition	$pq^2 + \frac{1}{3}q^3$
Householder Orthogonalization	$2pq^2 - \frac{2}{3}q^3$
Modified Gram-Schmidt	$2pq^2$

1.2 Python Implementations

[2]:
```
import numpy as np

# generate a random matrix
A = np.random.randint(1, 10, size=(3, 2))
A
```

[2]:
```
array([[9, 2],
       [8, 1],
       [9, 2]])
```

[3]:
```
# compute the QR decomposition of A
Q, R = np.linalg.qr(A)

# check that Q and R satisfy the QR decomposition property
print(np.allclose(A, np.dot(Q, R)))
print(np.allclose(np.eye(2), np.dot(Q.T, Q)))
print(Q)
print(R)
```

```
True
True
[[-0.59867109  0.37628835]
 [-0.53215208 -0.84664878]
```

```
 [-0.59867109  0.37628835]]
[[-15.03329638  -2.92683646]
 [  0.           0.65850461]]
```

[4]:
```
B = A.T@A
# Compute the Cholesky decomposition
L = np.linalg.cholesky(B)

L
```

[4]:
```
array([[15.03329638,  0.        ],
       [ 2.92683646,  0.65850461]])
```

[ ]:
```

```