Computational Linear Algebra

Feng Li
feng.li@cufe.edu.cn

School of Statistics and Mathematics
Central University of Finance and Economics

Revised on August 15, 2018

Today we are going to learn...

1. Basic linear algebra
2. QR Decomposition
3. Singular Value Decomposition
4. SVD in Image Recognition
5. SVD in Linear Squares

The determinant of a matrix

- The definition of a matrix determinant

$$\det(A) = \sum_{\sigma \in S_n} sgn(\sigma) \prod_{i=1}^{n} a_{i,\sigma(i)}$$

- The calculation becomes more complicated. It can be found in R using the `det()` function.
- The `determinant()` function can calculate the log determinant.
- Verify the following properties
  - $\det(A^T) = \det(A)$
  - $\det(A^{-1}) = \frac{1}{\det(A)} = \det(A)^{-1}$
  - For square matrices A and B of equal size, $\det(AB) = \det(A)\det(B)$.
  - $\det(cA) = c^n \det(A)$ for an $n \times n$ matrix.

Eigenvalues and eigenvectors

- An eigenvector of a square matrix A is a non-zero vector v that, when the matrix is multiplied by v, yields a constant multiple of v, the multiplier being commonly denoted by λ. That is: $Av = \lambda v$.
- If A is also positive-definite, positive-semidefinite, negative-definite, or negative-semidefinite every eigenvalue is positive, non-negative, negative, or non-positive respectively.
- Eigenvalues and eigenvectors can be computed using the function `eigen()`
- Eigenvalues can be used to check if a matrix is invertible.
- The determinant of A is the product of all eigenvalues: $\det(A) = \prod_{i=1}^{n} \lambda_i = \lambda_1 \lambda_2 \cdots \lambda_n$
- The trace of A, defined as the sum of its diagonal elements, is also the sum of all eigenvalues: $\mathrm{tr}(A) = \sum_{i=1}^{n} A_{ii} = \sum_{i=1}^{n} \lambda_i = \lambda_1 + \lambda_2 + \cdots + \lambda_n$

Triangular matrices

- The functions `lower.tri()` and `upper.tri()` can be used to obtain the lower and upper triangular parts of matrices.

Outer Products

- The function `outer()` is sometimes useful in statistical calculations. It can be used to perform an operation on all possible pairs of elements coming from two vectors.
- Example

```
> x1 <- seq(1, 5)
> outer(x1, x1, "/")
> y <- seq(5, 10)
> outer(x1, y, "+")
```

Kronecker product

- If A is an $m \times n$ matrix and B is a $p \times q$ matrix, then the Kronecker product $A \otimes B$ is the $mp \times nq$ block matrix:

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & \cdots & a_{1n}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{m1}\mathbf{B} & \cdots & a_{mn}\mathbf{B} \end{bmatrix}$$

- The function `kronecker()` can be used to compute the Kronecker product of two matrices and other more general products.

The matrix inverse

- The usual definition of `square matrix inverse` is $A^{-1}A = AA^{-1} = I$ which is not commonly used in practice. The solution is **unique** when A is not singular.
- **The general inverse** for $A_{m \times n}$
  - The definition: $A^{+}A = I$
  - Notice that $(A'A)^{-1}A'A = I$, the above definition actually means $A^{+} = (A'A)^{-1}A'$.
  - The general inverse in **not unique** but useful.

QR decomposition

- QR decomposition is a decomposition of a matrix A into a product $A = QR$ of an orthogonal matrix Q and an upper triangular matrix R.
- Compared to the direct matrix inverse, inverse solutions using QR decomposition are more numerically stable as evidenced by their reduced condition numbers.
- To solve the underdetermined ($m < n$) linear problem $Ax = b$ where the matrix A has dimensions $m \times n$ and rank m
    - first find the QR factorization of the transpose of A: $A^T = QR$, where Q is an orthogonal matrix (i.e. $Q^T = Q^{-1}$), and R has a special form: $R = \begin{bmatrix} R_1 \\ 0 \end{bmatrix}$. Here R_1 is a square $m \times m$ right triangular matrix, and the zero matrix has dimension $(n - m) \times m$.
    - it can be shown that a solution to the inverse problem can be expressed as: $x = Q \begin{bmatrix} (R_1^T)^{-1} b \\ 0 \end{bmatrix} = [Q_1, Q_2] \begin{bmatrix} (R_1^T)^{-1} b \\ 0 \end{bmatrix} = Q_1 (R_1^T)^{-1} b$

The linear model with QR

- For a linear model $y = X\hat{\beta}$, the QR approach is slightly slower than $(X'X)^{-1}Xy$ but more accurate.
- The procedure is to compute the **QR decomposition** of X to get R and $Q'Y$, solve $R\beta = Q'y$.
  - As $X = QR$, the linear regression is now written As

$$\hat{Y} = X\beta = QR\beta$$

  - which yields

$$\begin{aligned}
\hat{\beta} &= (X'X)^{-1}X'y = X^{+}y \\
&= ((QR)'QR)^{-1}(QR)'y \\
&= (R'Q'QR)^{-1}(QR)'y \\
&= (R'R)^{-1}R'Q'y \\
&= R^{-1}Q^{T}y
\end{aligned}$$

The linear model with QR: Example

Singular Value Decomposition

- In linear algebra, the singular value decomposition (SVD) is a factorization of a real or complex matrix, with many useful applications in signal processing and statistics.
- Formally, the singular value decomposition of an $m \times n$ real or complex matrix M is a factorization of the form

$$\mathbf{M}_{m\times n} = \mathbf{U}_{m\times n}\mathbf{\Sigma}_{n\times n}\mathbf{V}'_{n\times n}$$

- where U is a $m \times n$ real or complex unitary matrix, Σ is an $m \times n$ rectangular diagonal matrix with nonnegative real numbers on the diagonal, and
- V' (the conjugate transpose of V, or simply the transpose of V if V is real) is an $n \times n$ real or complex unitary matrix.
- The diagonal entries $\Sigma_{i,i}$ of Σ are known as the singular values of M. The n columns of U and the n columns of V are called the left-singular vectors and right-singular vectors of M, respectively.

Some properties of SVD I

- In statistics, dependent variable are said to be **orthogonal** if they are **uncorrelated**
- U is orthogonal (its columns are eigenvectors of MM')
- V is orthogonal (its columns are eigenvectors of $M'M$)
- Non-negative real values $\Sigma_{i,i}$ called singular values. It's square is an eigenvalue of $M'M$)
- The rank of a matrix is equal to the number of non-zero singular values.
- If M is a $n \times n$ nonsingular matrix, then its inverse is given by

$$M^{-1} = V\Sigma^{-1}U^{T}$$

- If M is singular or ill-conditioned, then we can use SVD to approximate its inverse (**pseudo-inverse**) by the following matrix

$$(U\Sigma V')^{-1} \approx V\Sigma_0^{-1}U^{T}$$

where

$$\Sigma_0^{-1} = \begin{cases} 1/\Sigma_{ii} & \Sigma_{ii} > \epsilon \\ 0, & Otherwise \end{cases}$$

Some properties of SVD II

- **The condition number** measures the degree of singularity of $M'M$ (the larger this value is, the closer $M'M$ is to being singular)

$$k = \frac{\text{Max eigenvalue}}{\text{Minimal eigenvalue}}$$

and the conditional index $\sqrt{k}$ which connects to the Σ_{ii}.

Classification of Handwritten Digits

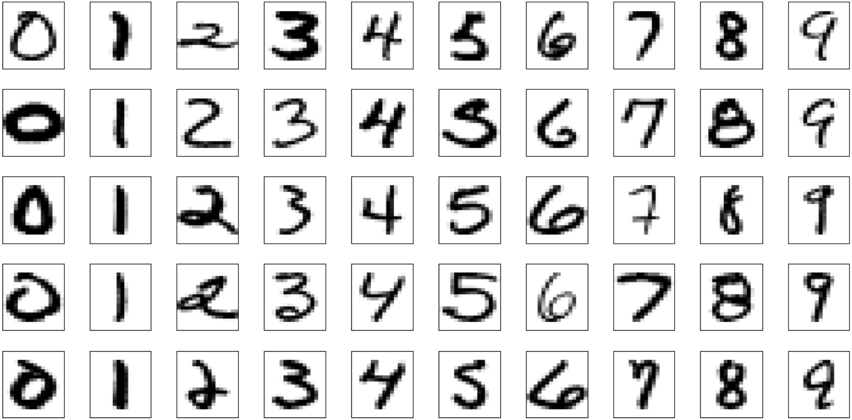


FIGURE 11.9. *Examples of training cases from ZIP code data. Each image is a* 16×16 *8-bit grayscale representation of a handwritten digit.*

The naive method

- The naive method is to check the distance from each test image to the mean of training image.

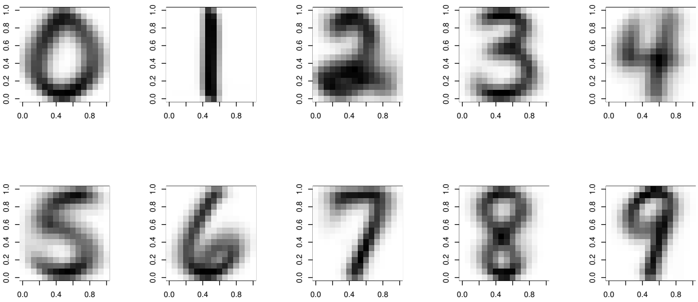


Figure: The mean of each digit from the training sample.

The naive method

- Now it is the time to check the testing sample to the mean of the training sample. We pick the first five testing digits.
- We find the first, third and the fifth are rather easy to classify by eyeballs. But the second and fourth ones are particular difficult.

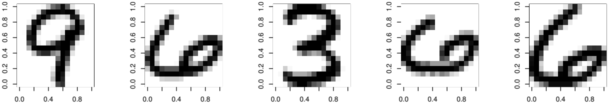


Figure: The testing image

The SVD method

- We pick the digit 9 as an example in this method and plot the first ten singular image from the SVD decomposition
- We first use four bases, which yields the correct specification as follows We also tries to classify other digits which gives robust results. But when we increase more basis function, there comes the risk of overfitting.
- It maybe not a good idea to use all the bases but one can always pick up the bases according to the first kth largest eigen values.

The SVD method

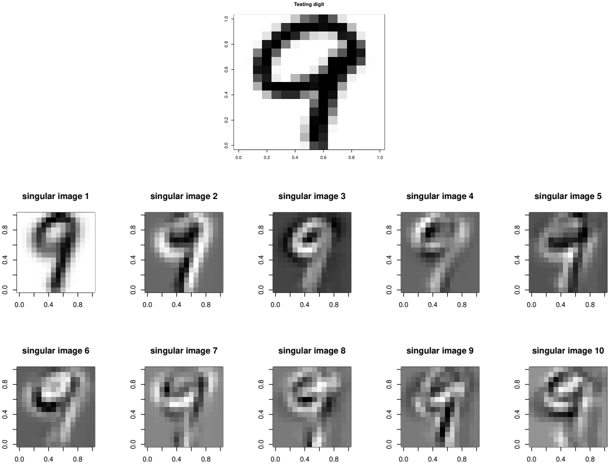
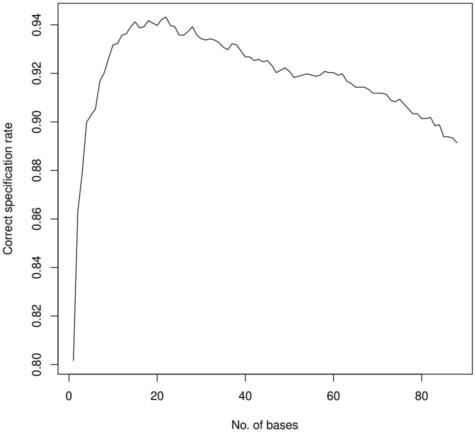


Figure: The first ten sigumar images from the training sample for digit 9.

The SVD method

- We will find out when we overfit (see the plot of classification success as a function of the number of basis vectors.)
- To see this, we loop over all testing observations and number of bases from 1 to 88, and then count the correct specification numbers.



Linear Regression

- The least square solution of $y = X\beta + \epsilon$ may not be exist due to $X'X$ is singular.
- If $X'X$ is ill-conditioned or singular, we can use SVD to obtain a least squares solution as follows:

$$\hat{\beta} = (X'X)^{-1}X'y = X^{+}y \approx V\Sigma_0^{-1}U^{T}y$$

where X^{+} is the general inverse, and

$$\Sigma_0^{-1} = \begin{cases} 1/\Sigma_{ii} & \Sigma_{ii} > \epsilon \\ 0, & \text{Otherwise} \end{cases}$$