Linear Algebrea: wk5 projection

Bill Chung

February 09, 2023

```
library(far)
library(MASS)
library(pracma)
library(expm)
```

CLT

Sample Satistics

- page 96

What does the following notation mean?

$$T_{(n)} = h_{(n)}(X_1, X_2, ...X_n)$$

where $h_{(n)} : \mathbb{R}^n \to \mathbb{R}, \forall n \in \ltimes$

Convergence in Probability

- see page 99

$$T_{(n)} \xrightarrow{p} c$$

Weak Law of large numbers

- see page 100 and read first two lines of page 101

$$\bar{X}_{(n)} \xrightarrow{p} E[X]$$

Law of Large Numbers (LLN)

$$\lim_{n\to\infty} \sum_{i=1}^{n} \frac{X_i}{n} = E[X]$$

CLT

- See page 109

Suppose $\{X_1, X_2, ..X_n\}$ is a sequence of iid rv with $\mathrm{E}[X_i] = \mu$ and $\mathrm{V}[X_i] = \sigma^2$. Then as $n \to \infty$, random variable $\sqrt{n}(\bar{X}_n - \mu)$ converges in distribution with a normal distribution with mean 0 and variance σ^2

$$\sqrt{n}(\bar{X}_n - \mu) \xrightarrow{d} N(0, \sigma^2)$$

More about invertible matrix

Given: Suppose $A \in R^{nxn}$ and A^{-1} exist, then the following can be said

- The columns of A is the basis of R^n
- rank A = n
- $NulA = \{\vec{0}\}$
- dim $NulA = 0$
- $A^{-1}A = I$
- $AA^{-1} = I$
- The Linear transformation $\vec{x} \mapsto A\vec{x}$ is one-to-one
- A^T is an invertible matrix

Change of basis

Given: $\vec{y} \notin C(A)$, and Rank of $A = 2$, and $\vec{y} \in R^3$

Problem 1

- Let $\hat{\vec{y}} \subset C(A)$ where $\vec{C}_1$ and $\vec{C}_2$ are the basis of $C(A)$
- Find $\hat{\vec{y}}$ that minimizes $||\vec{y} - \hat{\vec{y}}||$

Solution:

- let C and N be the matrix that contains the basis of $C(A)$ and $N(A^T)$
- Since: $C\vec{x} = \hat{\vec{y}}$ and $C\vec{x} + N\vec{z} = \vec{y}$
- Simplify the expression

$$\begin{aligned} C^T C\vec{x} &= C^T\vec{y} \\ \vec{x} &= (C^T C)^{-1} C^T \vec{y} \end{aligned}$$

- Then,

$$C(C^T C)^{-1} C^T \vec{y} = \hat{\vec{y}}$$

- $C(C^T C)^{-1} C^T$ is called **projection matrix***

About projection matrix

$$\mathbb{I} = \mathbb{P} + \mathbb{B}$$
$$\vec{y} = \mathbb{P}\vec{y} + \mathbb{B}\vec{y}$$

- where P and B are the `projection matrices` for $C(A)$ and $N(A^T)$

DOT Product

$$\hat{\vec{y}} = P_{\vec{u}}^{\vec{y}} = \frac{\vec{y} \cdot \vec{u}}{\vec{u} \cdot \vec{u}}\vec{u}$$

where

$\vec{y} \cdot \vec{u}$ and $\vec{u} \cdot \vec{u}$ are scalar quantity.

Projection tells you the `length` of the `projected vector`, $\hat{\vec{y}}$ in terms of the vector that is `being projected onto` $\vec{u}$

```
# y will be projected onto u
y <- matrix(c(7,6),nrow=2)
u <- matrix(c(4,2),nrow=2)
u0 <- matrix(c(16,8),nrow=2)
```

Example

```
b <- c(1,5,3)

c1 <- c(1,2,3)
c2 <- c(1,2.1,3.2)
c3 <- c(0,1,-2)

A <- cbind(c1,c2,c3)
print(A)
```

```
##      c1  c2 c3
## [1,]  1 1.0  0
## [2,]  2 2.1  1
## [3,]  3 3.2 -2
```

```
Rank(A)
```

```
## [1] 3
```

```
#get x coordinate
x <- inv(A)%*%b
A%*%x
```

```
##      [,1]
## [1,]    1
## [2,]    5
## [3,]    3
```

example of projection matrix

```
b <- c(1,5,3)

c1 <- c(1,2,3)
c2 <- c(1,2.1,3.2)

A <- cbind(c1,c2)
print(A)
```

```
##      c1  c2
## [1,]  1 1.0
## [2,]  2 2.1
## [3,]  3 3.2
```

```
Ab <- cbind(A,b)
Rank(A)
```

```
## [1] 2
```

```
Rank(Ab)
```

```
## [1] 3
```

```
#get x coordinate
G <- t(A)%*%A

x <- inv(G)%*%t(A)%*%b

b_hat <- A%*%x

#projection matrix to C(A)
P <- A%*%inv(G)%*%t(A)

b_hat <- P%*%b

residual <- b- b_hat
print(residual)
```

```
##      [,1]
## [1,]   -1
## [2,]    2
## [3,]   -1
```

```
#projection matrix to left nullspace
N<- diag(3) - P
N%*%b
```

```
##      [,1]
## [1,]   -1
## [2,]    2
## [3,]   -1
```

Orthogonal

- Two vectors $\vec{v_1}$ and $\vec{v_2} \in R^m$ are orthogonal, if $\vec{v_1} \cdot \vec{v_2} = 0$
- Note that the dot product produce scalar quantity 0 not $\vec{0}$
- Notice $\vec{v}_1$ is size of 3 vector and `orth( )` returns normalized $\vec{v}_1$

```
v1 <- c(3,4,5)
```

Normalizing the basis

```
c_A <- orth(v1)
print(c_A)
```

```
##           [,1]
## [1,] 0.4242641
## [2,] 0.5656854
## [3,] 0.7071068
```

```
#notice what happens when you dot v1 and c_A
print(v1%*%c_A)
```

```
##          [,1]
## [1,] 7.071068
```

```
Norm(v1)
```

```
## [1] 7.071068
```

Space, subspace, orthogonal complement subspace

- Let S be space of R^n, A is R^{mxn} matrix.
- Let $C(A)$ and $N(A^T)$ be the column space and left nullspace of A
- $C(A)$ and $N(A^T)$ are orthogonal complement subspace of each other.
- Then, any vector, $\vec{x} \in S$ but $\vec{x} \notin C(A)$ or $\vec{x} \notin N(A^T)$ can be expressed by the linear combination of basis of $C(A)$ and $N(A^T)$

Diagonal matrix

```
D1 <- diag(c(5,2,10),3,3)
print(D1)
```

```
##      [,1] [,2] [,3]
## [1,]    5    0    0
## [2,]    0    2    0
## [3,]    0    0   10
```

```
print(inv(D1)) #notice when the diagonal elements has zero in it, D1 becomes singular.
```

```
##      [,1] [,2] [,3]
## [1,]  0.2  0.0  0.0
## [2,]  0.0  0.5  0.0
## [3,]  0.0  0.0  0.1
```

```
print(D1 %^% 3) # using the function in expm
```

```
##      [,1] [,2] [,3]
## [1,]  125    0    0
## [2,]    0    8    0
## [3,]    0    0 1000
```

Orthogonal matrix

$$U^{-1} = U^T$$

- Let W be a subspace of R^n and let $\vec{y} \in R^n$ but $\vec{y} \notin W$.
- Then, $\hat{\vec{y}} \in W$ that is the closest approximation of $\vec{y}$ is the $\vec{y}$ projected onto W

Proerty of matrx that is not square, but has orthonormal basis

```
v <- matrix(c(2,1,2),nrow=3)
O <- orthonormalization(v)
print(O)
```

```
##           [,1]       [,2]       [,3]
## [1,] 0.6666667 -0.2357023 -0.7071068
## [2,] 0.3333333  0.9428090  0.0000000
## [3,] 0.6666667 -0.2357023  0.7071068
```

```
U <- cbind(O[,1],O[,2])
print(t(U)%*%U)
```

```
##      [,1] [,2]
## [1,]    1    0
## [2,]    0    1
```

Suppose C is matrix that contains orthonormal basis of W. Since there exist $\vec{y} \notin W$, C can't be square matrix.

However, the basis in C can still be `orthonormal`.

Let C be retangular matrix with orthonormal basis,

$$\vec{y} = C\vec{x}_w + N\vec{x}_N$$

where - N is the basis spanning orthogonal complement subspace of W. Then,

$$C^T\vec{y} = C^TC\vec{x}_w$$

Since C is matrix that contains orthonormal basis, C^TC becomes identify matrix.

$$C^T\vec{y} = \vec{x}_W$$

Now, the location of $\hat{\vec{y}}$ in terms of the basis in C can be expressed as below

$$C\vec{x}_W = \hat{\vec{y}}$$

Solving for $\vec{x}_W$

$$\vec{x}_W = C^T\hat{\vec{y}}$$

Sub the above expression of $\vec{x}_W$ to the following equation

$$C^T\vec{y} = C^TC\vec{x}_w$$

$$C^T\vec{y} = C^TC(C^T\hat{\vec{y}})$$

Then,

$$CC^T\vec{y} = \hat{\vec{y}}$$

Gram-Schmidt Process

- Let $\{\vec{x}_1, \vec{x}_2 ... \vec{x}_p\}$ be basis for a nonzero subspace W of R^n where $p < n$. Gram-Schimidt process converts $\{\vec{x}_1, \vec{x}_2 ... \vec{x}_p\}$ to $\{\vec{v}_1, \vec{v}_2 ... \vec{v}_p\}$ where $\{\vec{v}_1, \vec{v}_2 ... \vec{v}_p\}$ are orthogonal basis for W
- Gram-Schimit process is projecting one set of basis to another basis that is orthogonal to them.
- Notice the `orthonormalization( )` in `R` returns 3 x 3 matrix. This function in R returns the basis spanning the subspace that is orthogonal to subspace spanned by $\vec{v}_1$

```
GS <- orthonormalization(v1)
print(GS)
```

```
##           [,1]       [,2]       [,3]
## [1,] 0.4242641  0.9055385  0.0000000
## [2,] 0.5656854 -0.2650357  0.7808688
## [3,] 0.7071068 -0.3312946 -0.6246950
```

Gram-Schmidt