wk7

Bill Chung

December 28, 2021

Solving systems of equations

- by comparing the rank of $\mathbb{A}$ and the augmented matrix, we can find out if the systems of equations have
  - exactly one solutions, (this is case when $\mathbb{A}$ is `invertible`)
  - many solutions, (the solution set exist in `hyperplane` which is `affine nullspace of $A$`)
  - no solutions (rank of the `argumented matrix` is greater than rank of A)

General approach

$$\begin{aligned}
\mathbb{A}\vec{x} &= \vec{b} \\
\mathbb{B}\vec{x}_B + \mathbb{N}\vec{x}_N &= \vec{b} \\
\mathbb{B}^T\mathbb{B}\vec{x}_B &= \mathbb{B}^T\vec{b} - \mathbb{B}^T\mathbb{N}\vec{x}_N \\
\vec{x}_B &= \left(\mathbb{B}^T\mathbb{B}\right)^{-1}\mathbb{B}^T\vec{b} - \left(\mathbb{B}^T\mathbb{B}\right)^{-1}\mathbb{B}^T\mathbb{N}\vec{x}_N
\end{aligned}$$

where $\mathbb{A} = [\mathbb{B}|\mathbb{N}]$ $\mathbb{B}$ is matrix containing independent column vectors $\mathbb{N}$ is matrix containing dependent column vectors

When RHS is not in the span of the column space

$$\hat{\vec{b}} \neq \vec{b}$$

$\hat{\vec{b}} \in C(\mathbb{A})$, where $C(\mathbb{A})$ is a column space of $\mathbb{A}$, among all the vectors in $C(\mathbb{A})$.

$$\vec{b} = B_{C(A)}\vec{x}_{C(A)} + B_{N(A^T)}\vec{x}_{N(A^T)}$$

where $B_{C(A)}$ and $B_{N(A^T)}$ are the basis of $C(A)$ and $N(A^T)$ $\vec{x}_{C(A)}$ and $\vec{x}_{N(A^T)}$ are the coordinate of the corresponding basis Subsituting $B_{C(A)}$ as $B_{C(A)}$ and $\vec{x}_{C(A)}$ as $\vec{x}_B$, the above equation can be written as the following:

$$B^T\vec{b} = B^TB\vec{x}_B$$

$$\vec{x}_B = (B^TB)^{-1}B^T\vec{b}$$

and $(B^TB)^{-1}B^T$ is called `projection matrix of C(A)` where B are the basis of $C(A)$

$$\vec{b} = B_{C(A)}\vec{x}_{C(A)} + B_{N(A^T)}\vec{x}_{N(A^T)}$$

where $B_{C(A)}$ and $B_{N(A^T)}$ are the basis of $C(A)$ and $N(A^T)$ $\vec{x}_{C(A)}$ and $\vec{x}_{N(A^T)}$ are the coordinate of the corresponding basis Subsituting $B_{C(A)}$ as $B_{C(A)}$ and $\vec{x}_{C(A)}$ as $\vec{x}_B$, the above equation can be written as the following:

$$B^T\vec{b} = B^T B\vec{x}_B$$

$$\vec{x}_B = (B^T B)^{-1}B^T\vec{b}$$

and $(B^T B)^{-1}B^T$ is called `projection matrix of C(A)` where B are the basis of $C(A)$

problem from page 2 of the text

```
#systems of equations
A <- matrix(c(2,-1,1.5,1,0,-4),nrow=2,byrow = TRUE)

#solution given in the text
x <- c(5,6.5,3)

A%*%x
```

```
##      [,1]
## [1,]    8
## [2,]   -7
```

```
A
```

```
##      [,1] [,2] [,3]
## [1,]    2   -1  1.5
## [2,]    1    0 -4.0
```

```
Rank(A)
```

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

```
#RHS
b <- c(8,-7)
```

```
#Is this Homogeneous equations or inhomogeneous equation?
H <- cbind(A,b)
H
```

```
##                b
## [1,] 2 -1  1.5  8
## [2,] 1  0 -4.0 -7
```

```r
rref(H)
```

```
##                  b
## [1,] 1 0 -4.0  -7
## [2,] 0 1 -9.5 -22
```

```r
Rank(H)
```

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

```r
#unique solution
A <- matrix(c(1,-2,-1,3), nrow =2 , byrow = TRUE)
A
```

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

```r
Rank(A)
```

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

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

x <- inv(A)%*%b

#get the solution
A%*%x
```

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

```r
#no solution

A <- matrix(c(1,-2,-1,2), nrow =2 , byrow = TRUE)
Rank(A)
```

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

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

#how can I check if we can span b or not?
```

```r
#many solution

A <- matrix(c(1,-2,-1,2), nrow =2 , byrow = TRUE)
A
```

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

```
Rank(A)
```

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

```
b <- c(-1,1.5)

H <- cbind(A,b)
#how can I check if we can span b or not?
Rank(H)
```

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

See Page 5

```
a <- c(1,-2,1,0,0,2,-8,8,-4,5,9,-9)

A <- matrix(a, nrow=3, byrow = TRUE)
Rank(A)
```

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

```
#okay I made mistake, need to get the
#RHS out
b <- A[,4]
A <- A[,c(1,2,3)]
A
```

```
##      [,1] [,2] [,3]
## [1,]    1   -2    1
## [2,]    0    2   -8
## [3,]   -4    5    9
```

```
Rank(A)
```

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

```
b
```

```
## [1]  0  8 -9
```

```
x <- inv(A)%*%b

A%*%x
```

```
##      [,1]
## [1,]    0
## [2,]    8
## [3,]   -9
```

Solution Sets to Linear Systems

See Page 45

```r
a <- c(3,5,-4,-3,-2,4,6,1,-8)

A <- matrix(a, nrow=3, byrow = TRUE)
Rank(A)
```

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

```r
b <- c(7,-1,-4)

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

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

```r
rref(H)
```

```
##                     b
## [1,] 1 0 -1.333333 -1
## [2,] 0 1  0.000000  2
## [3,] 0 0  0.000000  0
```

```r
B <- A[, c(1,2)]
N <- A[,c(3)]
```

$$
\begin{aligned}
\mathbb{A}\vec{x} &= \vec{b} \\
\mathbb{B}\vec{x}_B + \mathbb{N}\vec{x}_N &= \vec{b} \\
\mathbb{B}^T\mathbb{B}\vec{x}_B + \mathbb{B}^T\mathbb{N}\vec{x}_N &= \mathbb{B}^T\vec{b} \\
\mathbb{G}\vec{x}_B + \mathbb{B}^T\mathbb{N}\vec{x}_N &= \mathbb{B}^T\vec{b} \\
\mathbb{G}^{-1}\mathbb{G}\vec{x}_B + \mathbb{G}^{-1}\mathbb{B}^T\mathbb{N}\vec{x}_N &= \mathbb{G}^{-1}\mathbb{B}^T\vec{b} \\
\mathbb{I}\vec{x}_B + \mathbb{G}^{-1}\mathbb{B}^T\mathbb{N}\vec{x}_N &= \mathbb{G}^{-1}\mathbb{B}^T\vec{b} \\
\vec{x}_B + \mathbb{G}^{-1}\mathbb{B}^T\mathbb{N}\vec{x}_N &= \mathbb{G}^{-1}\mathbb{B}^T\vec{b}
\end{aligned}
$$

$$
\begin{aligned}
ax + b_1 &= 5 \\
d + 5c &= 12
\end{aligned}
$$

```r
B
```

```
##      [,1] [,2]
## [1,]    3    5
## [2,]   -3   -2
## [3,]    6    1
```

```r
N
```

```
## [1] -4  4 -8
```

```r
K <- t(B)%*%B

#get the solution
x_b <- inv(K)%*%t(B)%*%b

#check the solution
B%*%x_b
```

```
##      [,1]
## [1,]    7
## [2,]   -1
## [3,]   -4
```

```r
#
inv(K)%*%t(B)%*%N
```

```
##           [,1]
## [1,] -1.333333
## [2,]  0.000000
```

```r
4/3
```

```
## [1] 1.333333
```

Recall that we have selected first two columns vectors as the basis and their coordiate is given as -1 and 2. For the vector in $\mathbb{N}$, we got $\frac{4}{3}$.

```r
print(A)
```

```
##      [,1] [,2] [,3]
## [1,]    3    5   -4
## [2,]   -3   -2    4
## [3,]    6    1   -8
```

```r
print(x_b)
```

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

Additional information shown in rref(A)

- rref(A) tells you the solution to homogeneous systems of equations (See page 43)

```r
rref(A)
```

```
##      [,1] [,2]      [,3]
## [1,]    1    0 -1.333333
## [2,]    0    1  0.000000
## [3,]    0    0  0.000000
```

- rref(A) tells you the relationship between the basis and dependent vectors in expressing the solution vector [-1,2,0]

$$
\begin{aligned}
x_1 \quad - \quad \frac{4}{3}x_3 &= -1 \\
x_2 \qquad &= \;\; 2 \\
0 &= \;\; 0
\end{aligned}
$$

Parametric description of solution sets

- `free variables` act as parameters.
  - Can anyone define `parameter`?
- See the example of parametric description of solution sets

```r
r1 <- c(1,6,2,-5,-2,-4)
r2 <- c(0,0,2,-8,-1,3)
r3 <- c(0,0,0,0,1,7)

A <- rbind(r1,r2,r3)

rref(A)
```

```
##    [,1] [,2] [,3] [,4] [,5] [,6]
## r1    1    6    0    3    0    0
## r2    0    0    1   -4    0    5
## r3    0    0    0    0    1    7
```

$$
\begin{aligned}
x_1 &= -6x_2 - 3x_4 \\
x_2 &= \text{free} \\
x_3 &= 5 + 4x_4 \\
x_4 &= \text{free} \\
x_5 &= 7
\end{aligned}
$$

- The above example has ∞ number of solutions

Homogeneous Linear Systems

```
r1 <- c(3,5,-4,0)
r2 <- c(-3,-2,4,0)
r3 <- c(6,1,-8,0)

A <- rbind(r1,r2,r3)

rref(A)
```

```
##    [,1] [,2]      [,3] [,4]
## r1    1    0 -1.333333    0
## r2    0    1  0.000000    0
## r3    0    0  0.000000    0
```

- Now rewrite each row as equations and `free variables` needs to go to the RHS.

$\vec{x} = [x_1, x_2, x_3] = [\frac{4}{3}x_3, 0, x_3]$

- Now, let's get the constant out.

$\vec{v} = [\frac{4}{3}, 0, 1]$

- I have not yet explained $\vec{p}$ is, but the idea is that solution set (i.e., hyperplane) can be expressed as `parametric vector equation` of the plane that has the following form.

$$\vec{x} = s\vec{p} + t\vec{v}$$

- People say, the solution is in the `parametric vector form`

```
r1 <- c(3,5,-4)
r2 <- c(-3,-2,4)
r3 <- c(6,1,-8)

b <- c(7,-1,-4)

A <- rbind(r1,r2,r3)
Ab <- cbind(A,b)
rref(Ab)
```

```
##                     b
## r1 1 0 -1.333333 -1
## r2 0 1  0.000000  2
## r3 0 0  0.000000  0
```

$\vec{x} = [x_1, x_2, x_3]^T = [-1, 2, 0]^T + [\frac{4}{3}x_3, 0, x_3]^T = [-1, 2, 0]^T + x_3[\frac{4}{3}, 0, 1]$

- Recall that $\mathbb{A}\vec{x} = \vec{0}$ has the parametric vector solution.
- The solution to $\mathbb{A}\vec{x} = \vec{b}$ can be found by shifting the solution to the $\mathbb{A}\vec{x} = \vec{0}$, which is a subspace, by a constant vector $\vec{p}$. The resulting solution set is `hyperplane`

More examples

Given:

$$x_1 + 6x_2 + 3x_4 = 0$$
$$3x_3 - 4x_4 = 5$$
$$x_5 = 7$$

```
a1 <- c(1,6,0,3,0,0)
a2 <- c(0,0,1,-4,0,5)
a3 <-  c(0,0,0,0,1,7)
a <- c(a1,a2,a3)

A <- matrix(a,nrow=3,byrow=T)
print(A)
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    1    6    0    3    0    0
## [2,]    0    0    1   -4    0    5
## [3,]    0    0    0    0    1    7
```

```
print(rref(A))
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    1    6    0    3    0    0
## [2,]    0    0    1   -4    0    5
## [3,]    0    0    0    0    1    7
```

This is parametric description of solution set. Affine subspace or hyperplane.

$$x_1 = -6x_2 - 3x_3$$
$$x_2 \text{ is free}$$
$$x_3 = 5 + 4x_4$$
$$x_4 \text{ is free}$$
$$x_5 = 7$$

Network example

```
a1 <- c(1,1,0,0,0,800)
a2 <- c(0,1,-1,1,0,300)
a3 <-  c(0,0,0,1,1,500)
a4 <-  c(1,0,0,0,1,600)
a5 <-  c(0,0,1,0,0,400)
a <- c(a1,a2,a3,a4,a5)

A <- matrix(a,nrow=5,byrow=T)
print(A)
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    1    1    0    0    0  800
## [2,]    0    1   -1    1    0  300
## [3,]    0    0    0    1    1  500
## [4,]    1    0    0    0    1  600
## [5,]    0    0    1    0    0  400
```

```
rref(A)
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    1    0    0    0    1  600
## [2,]    0    1    0    0   -1  200
## [3,]    0    0    1    0    0  400
## [4,]    0    0    0    1    1  500
## [5,]    0    0    0    0    0    0
```

```
B <- A[,1:4]
b <- A[,c(6)]
print(B)
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1    1    0    0
## [2,]    0    1   -1    1
## [3,]    0    0    0    1
## [4,]    1    0    0    0
## [5,]    0    0    1    0
```

```
print(b)
```

```
## [1] 800 300 500 600 400
```

```
print(Rank(B))
```

```
## [1] 4
```

```
print(Rank(A))
```

```
## [1] 4
```

```
inv(t(B)%*%B)%*%t(B)%*%b
```

```
##      [,1]
## [1,]  600
## [2,]  200
## [3,]  400
## [4,]  500
```

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
- Rouche-Capelli Theorem $<$
  - Tells you the number of solutions in the systems of equations
  - Alfredo Capelli (5 August 1855 – 28 January 1910)

Change of basis

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

`Problem:`

- 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

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

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

- Then, $C(C^TC)^{-1}C^T\vec{y} = \hat{\vec{y}}$
- $C(C^TC)^{-1}C^T$ is called `projection matrix`

Projection matrix

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

- 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` $\vec{u}$

Using Projection matrix

```
# 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)

## using projection matrix
P <- u%*%(solve(t(u)%*%u)%*%t(u))
print(P)
```

```
##      [,1] [,2]
## [1,]  0.8  0.4
## [2,]  0.4  0.2
```

```
print(P%*%y)
```

```
##      [,1]
## [1,]    8
## [2,]    4
```

`Projection formula` on to $\vec{u}$

```
print(drop((t(y)%*%u)/(t(u)%*%u))*u)
```

```
##      [,1]
## [1,]    8
## [2,]    4
```

```
print(drop((t(y)%*%u)/(t(u)%*%u)))
```

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

Using `Projection formula` on to $\vec{u}_0$

```
print(drop((t(y)%*%u0)/(t(u0)%*%u0))*u0)
```

```
##      [,1]
## [1,]    8
## [2,]    4
```

```
print(drop((t(y)%*%u0)/(t(u0)%*%u0)))
```

```
## [1] 0.5
```

Orthonormal basis

```
mybasis <- matrix(c(1,2,3,4,5,6),nrow=3)
print(mybasis)
```

```
##      [,1] [,2]
## [1,]    1    4
## [2,]    2    5
## [3,]    3    6
```

```
print(orthonormalization(mybasis))
```

```
##           [,1]       [,2]       [,3]
## [1,] 0.2672612  0.8728716  0.4082483
## [2,] 0.5345225  0.2182179 -0.8164966
## [3,] 0.8017837 -0.4364358  0.4082483
```

```
Z <- (orthonormalization(mybasis)) # z is orthonormal basis of codomain (I called it output space)
```

```
A <- matrix(c(4,3,5,6,8,10,5,12,13),nrow=3, byrow=T)
print(A)
```

```
##      [,1] [,2] [,3]
## [1,]    4    3    5
## [2,]    6    8   10
## [3,]    5   12   13
```

```
c(Norm(A[1,]),Norm(A[2,]),Norm(A[3,])) #norm of each row vectors in A (i.e., sample)
```

```
## [1]  7.071068 14.142136 18.384776
```

```
B <- A%*%Z
print(B)
```

```
##           [,1]     [,2]          [,3]
## [1,]  6.681531 1.963961  1.224745e+00
## [2,] 13.897585 2.618615 -6.217249e-15
## [3,] 18.173764 1.309307 -2.449490e+00
```

```
print(Z)
```

```
##           [,1]       [,2]       [,3]
## [1,] 0.2672612  0.8728716  0.4082483
## [2,] 0.5345225  0.2182179 -0.8164966
## [3,] 0.8017837 -0.4364358  0.4082483
```

Change of basis

```
a <- c(2/3,-2/3,1/3,2/3,1/3,-2/3)
A <- matrix(a,nrow=3,ncol=2,byrow=T)
print(fractions(A))
```

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

```
k <- rnorm(10000,5,5)
myData <- matrix(k,nrow=2,ncol=5000,byrow=T)
c_A <- A%*%myData

C = orthonormalization(A)
GS_A<- cbind(C[,1],C[,2])  #orthonormalized basis spanning C(A)
E <- GS_A%*%myData
```

- $\vec{b} \notin C(A)$ and suppose I want to flip $\vec{b}$ with respect to $C(A)$
- How would you develop the transformation matrix that flips $\vec{b}_0$?

```
b0 <- matrix(c(10,10,10),nrow=3)
B <- cbind(A,b0)
print(rref(B))
```

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

- Let D transformation matrix that is expressed in terms of the basis spanning $C(A)$ and $N(A^T)$
- Then,

$$D = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \tag{1}$$

- Let $\vec{b}_{1C}$ be the flipped vector whose coordiate is expressed in terms of basis spanning $C(A)$ and $N(A^T)$
- $T\vec{b}_0 = \vec{b}_{1\text{standard basis}}$ and $D\vec{b}_{0C} = \vec{b}_{1C}$
- Then, using
- $\vec{b}_{0C} = [0\ 0\ 1]^T$ and
- $\vec{b}_0 = [10\ 10\ 10]^T$ and
- C and D, we can get T

Find $\vec{b}_{\text{1standard basis}}$

```
D <- matrix(c(1,0,0,0,1,0,0,0,-1), nrow=3, byrow=T)
dim(C)
```

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

```
print(D)
```

```
##      [,1] [,2] [,3]
## [1,]    1    0    0
## [2,]    0    1    0
## [3,]    0    0   -1
```

```
T <- C%*%D%*%inv(C)
print(T%*%b0)
```

```
##           [,1]
## [1,] 15.714286
## [2,]  7.142857
## [3,]  1.428571
```

Change of basis is useful

- Can you explain why?
- Developing transformation matrix
- Evaluating long term behavior of the transformation matrix
- Understanding how the initial state will evolve over time
- Determining the influence of each basis of $C(A^T)$ to the $C(A)$

Question

- Explains what happens when you project $\vec{x}$ onto
  - basis that are not orthogonal
    - Try add the projected ones and compare with the original
  - basis that are orthogonal, but not normal
  - basis that are **orthonormal** and add the projected vector.