HW1\_Multivariate

Benjan

10/21/2022

## Problem 1

### a.

The dimensions of a matrix is denoted by m times n where m is the number of rows and n is the number of columns. So, a n times 3 matrix has n rows and 3 columns. Here is an example of 2 times 3 matrix:

m1 <- matrix(c(1,2,3,4,5,6),nrow=2,byrow=TRUE)  
m1

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

### b.

We can use the same elements of m1 but change the number of rows and columns:

m2 <- matrix(c(1,2,3,4,5,6),nrow=3,byrow=TRUE)  
m2

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

## Problem 2

For a matrix multiplication A times B, the column number of A should match the row number of B. The resulting matrix is has the row number of A and column number of B.

### a.

Here A is 2 by 2 matrix and B is 2 by 3 matrix. From the statement above, the matrix multiplication is valid and we get a 2 by 3 matrix.

A\_2a <- matrix(c(2,-1,1,2),nrow=2,byrow=TRUE)  
B\_2a <- matrix(c(1,1,2,2,0,1),nrow=2,byrow=TRUE)  
A\_2a %\*% B\_2a

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

### b.

Here A is 3 by 2 matrix and B is 2 by 3 matrix. The matrix multiplication is possible and gives us 3 by 3 matrix.

A\_2b <-matrix(c(1,0,2,1,-2,1),nrow=3,byrow=TRUE)  
B\_2b <- matrix(c(1,1,2,2,0,1),nrow=2,byrow=TRUE)  
A\_2b %\*% B\_2b

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

### c.

Here A is 3 by 2 matrix and B is 1 by 3. The column number of A does not match with the row number of B and thus the matrix multiplication is not possible.

## Problem 3

A matrix is invertible (has an inverse) if its determinant is non-zero. A matrix times its inverse gives us identity matrix which is a square matrix with 1’s in diagonal and 0’s everywhere else.

## a.

Let’s calculate the determinant:

A\_3a <- matrix(c(2,1,1,2),nrow=2,byrow=TRUE)  
det(A\_3a) #determinant

## [1] 3

As the matrix has non-zero determinant, it has an inverse (the question does not asks us to calculate it). The matrix times its inverse is an identity matrix as follows:

matrix(c(1,0,0,1),nrow=2,byrow=TRUE)

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

## b.

Following the same steps as a:

B\_3a <- matrix(c(2,0,1,0),nrow=2,byrow=TRUE)  
det(B\_3a) #determinant

## [1] 0

As the matrix has zero determinant, it does not have an inverse. Hence the expression is not solvable.

## Problem 4

Lets read the files first and see the first few observations.

women <- read.csv("https://raw.githubusercontent.com/EricBrownTTU/ISQS6350/main/womens\_track.csv")  
men <- read.csv("https://raw.githubusercontent.com/EricBrownTTU/ISQS6350/main/mens\_track.csv")  
head(women)

## m100 m200 m400 m800 m1500 m3000 marathon country  
## 1 11.61 22.94 54.50 2.15 4.43 9.79 178.52 argentin  
## 2 11.20 22.35 51.08 1.98 4.13 9.08 152.37 australi  
## 3 11.43 23.09 50.62 1.99 4.22 9.34 159.37 austria  
## 4 11.41 23.04 52.00 2.00 4.14 8.88 157.85 belgium  
## 5 11.46 23.05 53.30 2.16 4.58 9.81 169.98 bermuda  
## 6 11.31 23.17 52.80 2.10 4.49 9.77 168.75 brazil

head(men)

## m100 m200 m400 m800 m1500 m3000 mystery marathon country  
## 1 10.39 20.81 46.84 1.81 3.70 14.04 29.36 137.72 argentin  
## 2 10.31 20.06 44.84 1.74 3.57 13.28 27.66 128.30 australi  
## 3 10.44 20.81 46.82 1.79 3.60 13.26 27.72 135.90 austria  
## 4 10.34 20.68 45.04 1.73 3.60 13.22 27.45 129.95 belgium  
## 5 10.28 20.58 45.91 1.80 3.75 14.68 30.55 146.62 bermuda  
## 6 10.22 20.43 45.21 1.73 3.66 13.62 28.62 133.13 brazil

### a.

First thing to remember is that we calculate covariance and correlation only for numeric columns. Let’s see the covariance matrix first.

cov(women[,1:7])

## m100 m200 m400 m800 m1500 m3000  
## m100 0.20449414 0.47871354 1.0109549 0.03561313 0.10949327 0.27648539  
## m200 0.47871354 1.23445468 2.5501422 0.08706347 0.25793721 0.65016135  
## m400 1.01095492 2.55014215 7.1734877 0.26041229 0.70145286 1.71690475  
## m800 0.03561313 0.08706347 0.2604123 0.01171246 0.03243687 0.07704125  
## m1500 0.10949327 0.25793721 0.7014529 0.03243687 0.11050673 0.26558165  
## m3000 0.27648539 0.65016135 1.7169047 0.07704125 0.26558165 0.67952949  
## marathon 9.44436040 23.17862603 57.4924621 2.56637323 8.88078552 22.57166714  
## marathon  
## m100 9.444360  
## m200 23.178626  
## m400 57.492462  
## m800 2.566373  
## m1500 8.880786  
## m3000 22.571667  
## marathon 925.957204

Now, let’s see the correlation matrix.

cor(women[,1:7])

## m100 m200 m400 m800 m1500 m3000 marathon  
## m100 1.0000000 0.9527911 0.8346918 0.7276888 0.7283709 0.7416988 0.6863358  
## m200 0.9527911 1.0000000 0.8569621 0.7240597 0.6983643 0.7098710 0.6855745  
## m400 0.8346918 0.8569621 1.0000000 0.8984052 0.7878417 0.7776369 0.7054241  
## m800 0.7276888 0.7240597 0.8984052 1.0000000 0.9016138 0.8635652 0.7792922  
## m1500 0.7283709 0.6983643 0.7878417 0.9016138 1.0000000 0.9691690 0.8779334  
## m3000 0.7416988 0.7098710 0.7776369 0.8635652 0.9691690 1.0000000 0.8998374  
## marathon 0.6863358 0.6855745 0.7054241 0.7792922 0.8779334 0.8998374 1.0000000

The diagonal is insignificant. Also, a correlation matrix is symmetrical matrix, so we will only look at the elements that fall below the diagonal. As we can see the highest correlation is between 3000 meters and 1500 meters followed by between 100 meters and 200 meters. Overall, the elements closer to the diagonal have high value. The reason for this is the smaller the ratio of distances is, the bigger the correlation. For instance, a runner who can do well in 100 meters will likely do well in 200 meters but same cannot be said for a marathon .

### b.

Let’s follow the same steps for men’s record.

cov(men[,1:7])

## m100 m200 m400 m800 m1500 m3000  
## m100 0.12350249 0.20902182 0.43069956 0.016920438 0.03836684 0.17441020  
## m200 0.20902182 0.41557024 0.79905603 0.033115455 0.07788771 0.35913859  
## m400 0.43069956 0.79905603 2.12290020 0.080743131 0.18974209 0.90887976  
## m800 0.01692044 0.03311545 0.08074313 0.004055758 0.00911532 0.04406209  
## m1500 0.03836684 0.07788771 0.18974209 0.009115320 0.02430774 0.11592929  
## m3000 0.17441020 0.35913859 0.90887976 0.044062088 0.11592929 0.64185811  
## mystery 0.40184545 0.81171145 2.07341549 0.100049327 0.26343721 1.41154798  
## mystery  
## m100 0.4018455  
## m200 0.8117114  
## m400 2.0734155  
## m800 0.1000493  
## m1500 0.2634372  
## m3000 1.4115480  
## mystery 3.2678936

cor(men[,1:8])

## m100 m200 m400 m800 m1500 m3000 mystery  
## m100 1.0000000 0.9226384 0.8411468 0.7560278 0.7002382 0.6194618 0.6325389  
## m200 0.9226384 1.0000000 0.8507270 0.8066265 0.7749513 0.6953770 0.6965391  
## m400 0.8411468 0.8507270 1.0000000 0.8701714 0.8352694 0.7786139 0.7872045  
## m800 0.7560278 0.8066265 0.8701714 1.0000000 0.9180442 0.8635939 0.8690489  
## m1500 0.7002382 0.7749513 0.8352694 0.9180442 1.0000000 0.9281140 0.9346970  
## m3000 0.6194618 0.6953770 0.7786139 0.8635939 0.9281140 1.0000000 0.9746354  
## mystery 0.6325389 0.6965391 0.7872045 0.8690489 0.9346970 0.9746354 1.0000000  
## marathon 0.5199490 0.5961837 0.7049905 0.8064764 0.8655492 0.9321884 0.9431763  
## marathon  
## m100 0.5199490  
## m200 0.5961837  
## m400 0.7049905  
## m800 0.8064764  
## m1500 0.8655492  
## m3000 0.9321884  
## mystery 0.9431763  
## marathon 1.0000000

The correlation matrix looks similar to the women’s one and the reasoning for it is same. The highest correlation is between 3000 meters and mystery.

### c.

Here’s the logic to be followed for this. First, we will look at what columns are common across both dataframes and make a subset with only those columns. Then in those two subsets we will select only the first five observations and merge them by rows. The ‘marathon’ column is extra in both and ‘mystery’ column is extra in ‘men’ dataframe.

merged <- rbind(women[1:5,-8],men[1:5,-c(7,9)])

Now, lets use the ‘dist’ function.

round(dist(merged,method='euclidean'),2)

## 1 2 3 4 5 6 7 8 9  
## 2 26.39   
## 3 19.55 7.06   
## 4 20.84 5.61 2.11   
## 5 8.63 17.79 10.96 12.24   
## 6 41.81 16.14 22.62 21.56 33.28   
## 7 51.37 25.34 32.02 30.89 42.81 9.69   
## 8 43.52 17.61 24.23 23.11 34.96 1.98 7.89   
## 9 49.68 23.67 30.32 29.20 41.11 8.02 1.78 6.21   
## 10 33.51 9.76 14.87 14.30 25.15 8.98 18.41 10.86 16.76

### d.

Looking at the distance matrix, observation 7 and 9 are most similar and observation 1 and 7 are most disimilar. We determined this by finding the smallest value in the distance matrix for similarity and largest value for dissimilarity.

## Problem 5.

### a.

We can convert the covariance matrix to correlation matrix by using ‘cov2cor’ function. Lets first record the covariance matrix and then use the function to convert.

cov <- matrix(c(3.8778,2.8110,3.1480,3.5062,2.8110,2.1210,2.2669,2.5690,3.1480,2.2669,2.6550,2.8341,3.5062,2.5690,2.8341,3.2352),nrow=4,byrow=TRUE)  
cov

## [,1] [,2] [,3] [,4]  
## [1,] 3.8778 2.8110 3.1480 3.5062  
## [2,] 2.8110 2.1210 2.2669 2.5690  
## [3,] 3.1480 2.2669 2.6550 2.8341  
## [4,] 3.5062 2.5690 2.8341 3.2352

Now, for the correlation matrix.

cor = cov2cor(cov)  
cor

## [,1] [,2] [,3] [,4]  
## [1,] 1.0000000 0.9801619 0.9810921 0.9899048  
## [2,] 0.9801619 1.0000000 0.9552780 0.9807159  
## [3,] 0.9810921 0.9552780 1.0000000 0.9670131  
## [4,] 0.9899048 0.9807159 0.9670131 1.0000000

### b.

We can calculate the correlation between variable a and variable b by using this formula.

So, using the above formula the correlation between the first two variables is:

2.8110 / ((3.8778^0.5)\*(2.1210^0.5))

## [1] 0.9801619