Homework 2 (17 points)

INF 511

Purnabhishek Sripathi Sateesh Nuthalapati Mounika Maddi

You can complete this assignment in teams of **one to three students**. Different from HW-01, you will create your group by signing up in the “Groups” section of BbLearn. That way only one submission per group is necessary. Even if you are working alone, you must sign up as a “Group”. The **names of all team members** who participated on the assignment must be included in the `author` section of the `YAML`. All team members receiving the same score.

You must submit this assignment as a `.qmd` file rendered as a `.pdf`. Submit both the `.qmd` and the `.pdf` to Bblearn. **Any assignment that does not have both files will lose points.**

NOTE: All homeworks will be scaled to 100 points so that each homework is equally weighted in your grade.

1 Matrix calculations

Here are three matrices:

```
A<- matrix(c(2,3,5,4,
              1,5,7,8),
            nrow=2,ncol=4, byrow=TRUE)
B<- matrix(c(6,9,3,1),
            nrow=1,ncol=4)
C<- matrix(c(3,8,5,2,
              8,6,1,4),
            nrow=2,ncol=4, byrow=TRUE)
```

1.1 Matrix subsetting (1 point)

Use R to extract the element from matrix A in the first row and third column.

```
third_column <- A[,3]
third_column
```

```
[1] 5 7
```

1.2 Matrix subsetting (1 point)

Use R to extract the second column from matrix A but maintain that as a `matrix` object (i.e., it should be a column vector, and the `is.matrix()` function should be `TRUE`).

```
second_column <- A[,2,drop = FALSE]
second_column
```

```
     [,1]
```

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

```
is.matrix(second_column)
```

```
[1] TRUE
```

1.3 Matrix algebra (1 point)

Show the result of $A + C$.

```
matrix_algebra <- A+C
matrix_algebra
```

```
     [,1] [,2] [,3] [,4]
[1,]    5   11   10    6
[2,]    9   11    8   12
```

1.4 Matrix algebra (1 point)

Show the result of AB^T.

```
#product of two matrices
matrix_algebra2 <- A*as.vector(B)

matrix_algebra2
```

```
     [,1] [,2] [,3] [,4]
[1,]   12    9   30   12
[2,]    9    5   63    8
```

```
#the result of AB^{T}
t(matrix_algebra2)
```

```
     [,1] [,2]
[1,]   12    9
[2,]    9    5
[3,]   30   63
[4,]   12    8
```

2 Linear model matrices

An input matrix X and a column vector of observed data outputs Y are created below:

```
X<- matrix(c(1,1,1,1,1,1,4,1,2,3,3,4),ncol=2)
Y<- matrix(c(16,5,10,15,13,22),ncol=1)
ord<- order(Y)
(X<- X[ord,])
```

```
     [,1] [,2]
[1,]    1    1
[2,]    1    2
```

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

```
(Y<- Y[ord,,drop=FALSE])
```

```
     [,1]
[1,]    5
[2,]   10
[3,]   13
[4,]   15
[5,]   16
[6,]   22
```

2.1 Calculate X^TY (1 point)

```
#the X transpose y is:

X
```

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

```
Xt<- t(X)
result <- Xt*as.vector(Y)
result
```

```
     [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    5   13   16    5   13   16
[2,]   10   30   66   30   60   88
```

2.2 Calculate the crossproduct of column vector Y (1 point)

```
#crossproduct of Y:
crossprod(Y)
```

```
     [,1]
[1,] 1259
```

2.3 Use the `solve()` function to calculate $\hat{B}$ (2 points)

```
#B_hat <- t(B)
result_hat <- t(X)%*% X
inverse <- solve(result_hat)
B_hat <- inverse %*% result_hat
```

```
B_hat
```

```
     [,1]         [,2]
[1,]    1 3.552714e-15
[2,]    0 1.000000e+00
```

3 Linear modeling with `lm()`

3.1 Data frame (1 point)

Using the two data structures Y and X above, create a data frame with two columns: the output variable Y and the single input variable that is represented in matrix X.

```
#creating dataframe

d_fram <- data.frame("X" = c(X), "Y" = c(Y) )
d_fram
```

```
   X  Y
1  1  5
2  1 10
3  1 13
4  1 15
5  1 16
6  1 22
7  1  5
8  2 10
9  3 13
10 3 15
11 4 16
12 4 22
```

3.2 Simple linear regression (2 points)

Use the `lm()` function to model Y as a linear function of the single covariate in your data frame. Use the `data` option in the `lm()` function to specify where to locate the input and output variables (i.e., column names of the data frame). Report the estimated coefficients from the `lm()` results.

```
simple_lm <- lm(Y~X, data = d_fram)
simple_lm
```

```
Call:
lm(formula = Y ~ X, data = d_fram)

Coefficients:
(Intercept)            X
      9.931        1.862
```

```
simple_lmm <- lm(X~Y, data = d_fram)
simple_lmm
```

```
Call:
```