R-HW ASSIGNMENT

Shazman Khan MSBA-4

NOVEMBER 25, 2019

R-COURSE

Prof. Thomas Kurnicki

#Chapter 3.4: Exercises: 7 and 8

#Create a user defined function named studentfunction that transposes a numeric matrix (columns become rows) and subsets the matrix in a way so that only the first 5 rows remain in the dataset.

```
studentfunction<-function(mat){
  mat<-t(mat)
  if(nrow(mat) > 5){
    mat<- mat[1:5,]
  }
  return(mat)
}
```

#a) Use the studentfunction function to transform the matrix created in Exercise 7 from chapter 2.

```
sz_matrix<-matrix(c(10,11,9,15,19, 52, 19, 7, 10, 22, 28, 40, 6, 99, 33, 35, 26, 5, 87, 91, 0, 12, 16, 81,200),byrow=TRUE,nrow=5)
sz_matrix
studentfunction(sz_matrix)
```

Output:

```
> sz_matrix<-matrix(c(10,11,9,15,19, 52, 19, 7, 10, 22, 28, 40, 6, 99, 33, 35, 26, 5, 87, 91, 0, 12, 16, 81,200),byrow=TRUE,nrow=5)
> sz_matrix
     [,1] [,2] [,3] [,4] [,5]
[1,]   10   11    9   15   19
[2,]   52   19    7   10   22
[3,]   28   40    6   99   33
[4,]   35   26    5   87   91
[5,]    0   12   16   81  200
> studentfunction(sz_matrix)
     [,1] [,2] [,3] [,4] [,5]
[1,]   10   52   28   35    0
[2,]   11   19   40   26   12
[3,]    9    7    6    5   16
[4,]   15   10   99   87   81
[5,]   19   22   33   91  200
```

#b) Use the studentfunction function to transform the matrix that was given as an example in chapter 2.1

```
xf <- matrix(1:9, byrow = TRUE, nrow = 3)

xf

studentfunction(xf)
```

Output:

```
> xf <- matrix(1:9, byrow = TRUE, nrow = 3)
> xf
     [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
[3,]    7    8    9
> studentfunction(xf)
     [,1] [,2] [,3]
[1,]    1    4    7
[2,]    2    5    8
[3,]    3    6    9
```

#8.Create a user defined function named transformmatrix that takes the diagonal of a matrix and calculates a vector with two elements. Element one is the mean of the diagonal and element two is the median.

```
transformmatrix<-function(trans_matrix){

 v1<-c(mean(diag(trans_matrix)),median(diag(trans_matrix)))

 return(v1)

}
```

#a) Use the transformmatrix function to transform the matrix created in Exercise 7 from chapter 2.

```
sz_matrix<-matrix(c(10,11,9,15,19, 52, 19, 7, 10, 22, 28, 40, 6, 99, 33, 35, 26, 5, 87, 91, 0, 12, 16, 81,200),byrow=TRUE,nrow=5)

transformmatrix(sz_matrix)
```

Output:

```
sz_matrix<-matrix(c(10,11,9,15,19, 52, 19, 7, 10, 22, 28, 40, 6, 99, 33, 35, 26, 5, 87, 91, 0, 12, 16, 81,200),byrow=TRUE,nrow=5)
> transformmatrix(sz_matrix)
```

```
[1] 64.4 19.0
```

#b) Use the transformmatrix function to transform the matrix that was given as an example in chapter 2.1

xf<- matrix(1:9, byrow = TRUE, nrow = 3)

transformmatrix(xf)

Output:

```
xf<- matrix(1:9, byrow = TRUE, nrow = 3)
> transformmatrix(xf)
[1] 5 5
```

#9. For the iris dataset (no need to call a function, iris is part of the base R) create for loop that does the following to each observation:

#a) changes the Species column from a character type to numeric. Assign 1 for setosa, 2 for virginica, and 3 for versicolor,

```
n_iris<-iris

for (i in 1:nrow(n_iris)){

  if (n_iris$Species[i]=="setosa"){n_iris$Species<-gsub("setosa","1",n_iris$Species)}

  else if(n_iris$Species[i]=="verginica"){n_iris$Species<-gsub("verginica","2",n_iris$Species)}

  else {n_iris$Species<-gsub("versicolor","3",n_iris$Species)}

}

n_iris$Species<-as.numeric(n_iris$Species)

n_iris
```

Output:

```
> n_iris
    Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1            5.1         3.5          1.4         0.2       1
2            4.9         3.0          1.4         0.2       1
3            4.7         3.2          1.3         0.2       1
4            4.6         3.1          1.5         0.2       1
5            5.0         3.6          1.4         0.2       1
6            5.4         3.9          1.7         0.4       1
7            4.6         3.4          1.4         0.3       1
8            5.0         3.4          1.5         0.2       1
9            4.4         2.9          1.4         0.2       1
10           4.9         3.1          1.5         0.1       1
11           5.4         3.7          1.5         0.2       1
12           4.8         3.4          1.6         0.2       1
13           4.8         3.0          1.4         0.1       1
14           4.3         3.0          1.1         0.1       1
15           5.8         4.0          1.2         0.2       1
```

#b) creates a new column that groups the Petal.Length into 3 groups: group#1 for Petal.Length from 0 to 2, group #2 from 2.01 to 4.5, and group #3 from 4.51 to 7.

```
n_iris$new_col <- c()

for (i in 1:nrow(n_iris)){

 if (n_iris$Petal.Length[i]<=2){n_iris$new_col[i]<-"group#1"}

 else if (n_iris$Petal.Length[i]>2.01 && n_iris$Petal.Length[i]<=4.5){n_iris$new_col[i]<-"group#2"}

 else{n_iris$new_col[i]<-"group#3"}

}

n_iris
```

Output:

```
> n_iris
    Sepal.Length Sepal.Width Petal.Length Petal.Width Species new_col
1            5.1         3.5          1.4         0.2       1 group#1
2            4.9         3.0          1.4         0.2       1 group#1
3            4.7         3.2          1.3         0.2       1 group#1
4            4.6         3.1          1.5         0.2       1 group#1
5            5.0         3.6          1.4         0.2       1 group#1
6            5.4         3.9          1.7         0.4       1 group#1
7            4.6         3.4          1.4         0.3       1 group#1
8            5.0         3.4          1.5         0.2       1 group#1
9            4.4         2.9          1.4         0.2       1 group#1
10           4.9         3.1          1.5         0.1       1 group#1
11           5.4         3.7          1.5         0.2       1 group#1
12           4.8         3.4          1.6         0.2       1 group#1
13           4.8         3.0          1.4         0.1       1 group#1
14           4.3         3.0          1.1         0.1       1 group#1
15           5.8         4.0          1.2         0.2       1 group#1
16           5.7         4.4          1.5         0.4       1 group#1
17           5.4         3.9          1.3         0.4       1 group#1
18           5.1         3.5          1.4         0.3       1 group#1
19           5.7         3.8          1.7         0.3       1 group#1
20           5.1         3.8          1.5         0.3       1 gro.....
```

#3.Using the iris dataset:

#a) combine the Setosa and Versicolor into group “0” and label the Virginica to “1”. Create a new variable called iris$Group with the 0 or 1 labels,

```
sziris<-iris

mziris$iris_Group<-c()

for (i in 1:nrow(sziris)){

 if (sziris$Species[i]=="setosa" ||sziris$Species[i]=="versicolor"){sziris$iris_Group[i]<-"0"}

 else{sziris$iris_Group[i]<-"1"}

}

sziris
```

Output:

```
sziris
    Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
1            5.1         3.5          1.4         0.2     setosa
2            4.9         3.0          1.4         0.2     setosa
3            4.7         3.2          1.3         0.2     setosa
4            4.6         3.1          1.5         0.2     setosa
5            5.0         3.6          1.4         0.2     setosa
```

6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5.0	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa
10	4.9	3.1	1.5	0.1	setosa
11	5.4	3.7	1.5	0.2	setosa
12	4.8	3.4	1.6	0.2	setosa
13	4.8	3.0	1.4	0.1	setosa
14	4.3	3.0	1.1		

	iris_Group
1	0
2	0
3	0
4	0
5	0
6	0
7	0
8	0
9	0
10	0
11	0
12	0
13	0
14	0
15	0
16	0
17	0
18	0
19	0
20	0
21	0
22	0
23	0
24	0
25	0
26	0
27	0
28	0
29	0
30	0
31	0
32	0
33	0
34	0
35	0
36	0
37	0
38	0
39	0
40	0
41	0
42	

#b) build a logistic regression model using any available data that will predict the observation being Virginica (value of 1 in Group variable),

```
sziris$iris_Group<-as.numeric(sziris$iris_Group)

virginica_pred<-glm(iris_Group~Sepal.Length+Sepal.Width+Petal.Length+Petal.Width,data=sziris,family = "binomial")

summary(virginica_pred)
```

Output:

```
Call:
glm(formula = iris_Group ~ Sepal.Length + Sepal.Width + Petal.Length +
    Petal.Width, family = "binomial", data = sziris)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.01105  -0.00065   0.00000   0.00048   1.78065

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)   -42.638     25.708  -1.659   0.0972 .
Sepal.Length   -2.465      2.394  -1.030   0.3032
Sepal.Width    -6.681      4.480  -1.491   0.1359
Petal.Length    9.429      4.737   1.990   0.0465 *
Petal.Width    18.286      9.743   1.877   0.0605 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 190.954  on 149  degrees of freedom
Residual deviance:  11.899  on 145  degrees of freedom
AIC: 21.899

Number of Fisher Scoring iterations: 12
```

#only use the significant variables which are Petal.Length and Petal.Wodth

#c) calculate the probability of a new plant being a Virginica for the following parameters:

#Sepal.Width =5 Petal.Length =10 Petal.Width =7 Sepal.Length=9

```
np<- data.frame(Sepal.Length=9, Sepal.Width=5,Petal.Length=10,Petal.Width=7)

np_pr <- predict(virginica_pred, new_plant, type='response')
```

np_pr

Output:

```
np_pr <- predict(virginica_pred, np, type='response')
> np_pr
1
1
```