Homework: Coding question #1

Exercise 2 chapter 3.4

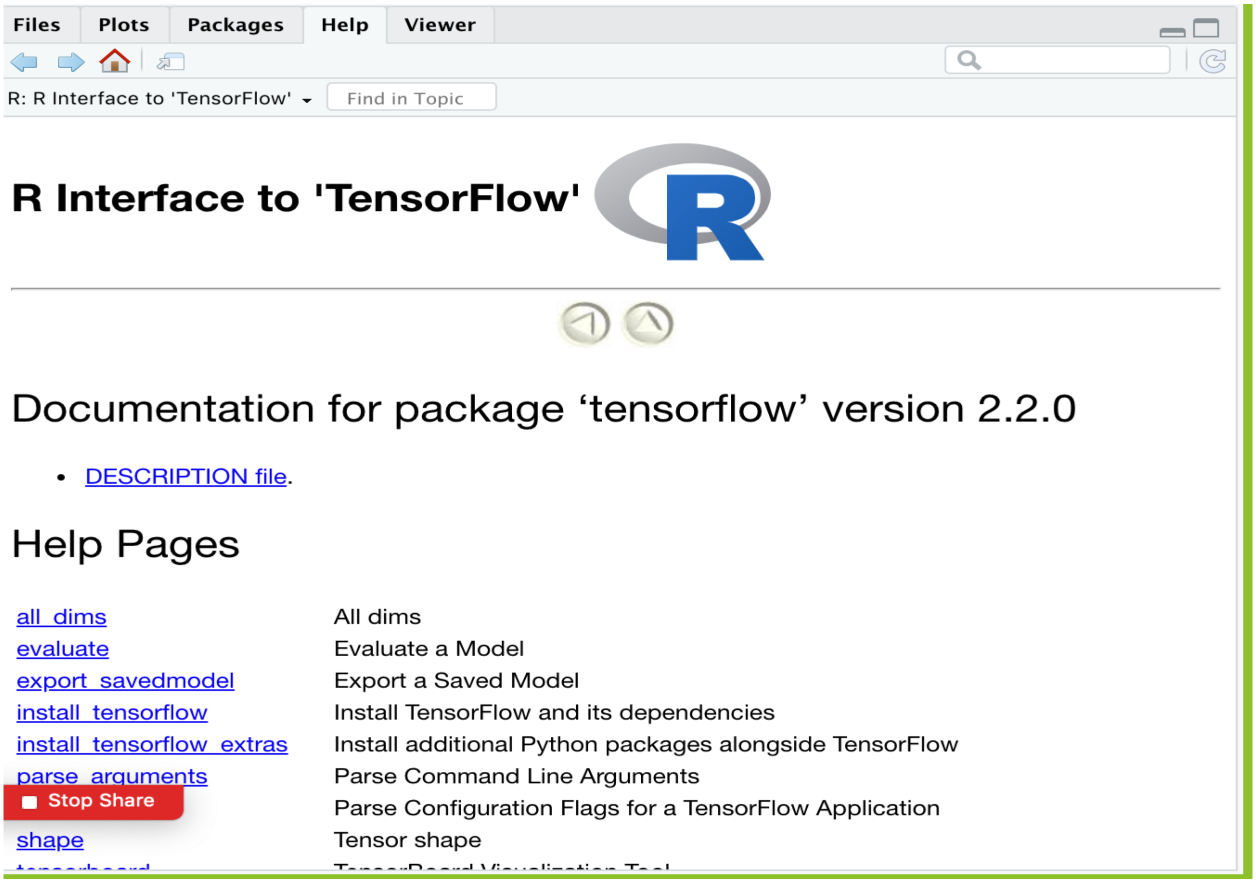
```
##       ##
### Business Card ###
##
## Business Script: Script Header ##
## Data Scientist: Tanvi Gupta ##
## Date Created: 2020 − 12 − 05 ##
## Copyright (c) Tanvi Gupta, 2020 ## Email: tangupta2019@student. hult. edu ##
##Contact: + 1 617 − 710 − 3273
##
## Additional Info:
## Pursuing  Masters in Business Analytics ## Masters in International Business
## Bachelors in Business Administration ##
## Beyond the curriculum ##
## climbed 22000 feet peak in the Himalayas##
```

Analysis-

The above script header tells about the author of the script. It provides with all the basic information of the author which makes it looks like the business card of the author. It also helps a coder to maintain recognized business profile.

Exercise 3 Chapter 3.4

```
#part a − installing library
install.packages("tensorflow")
#part b − CRAN − library documentation
help(package = tensorflow)
#part c − help document function that trains mode]
? train
```



R Interface to 'TensorFlow'

Documentation for package ‘tensorflow’ version 2.2.0

- DESCRIPTION file.

Help Pages

all_dims	All dims
evaluate	Evaluate a Model
export_savedmodel	Export a Saved Model
install_tensorflow	Install TensorFlow and its dependencies
install_tensorflow_extras	Install additional Python packages alongside TensorFlow
parse_arguments	Parse Command Line Arguments
	Parse Configuration Flags for a TensorFlow Application
shape	Tensor shape
tensorboard	TensorBoard Visualization Tool

Analysis-
The above code makes it more appealing to the user by amending the traditional way of doing programming. Neural library is quiet supple that helps in making artificial neural networks with subsist set of rules or algorithms.

Exercise 8 chapter 3.4

```
#User defined function for diagonal
transformationmatrix <- function(x){
 my_dia <- diag(x)
 my_mean <- mean(my_dia)
 my_median <- median(my_dia)
 my_vector <- c(my_mean, my_median)
 return(my_vector)
}

#Part A Matrix - 1
my_mat_1 <-
 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),
    nrow = 5, byrow = T)
my_mat_1
#calling the function for part A
transformationmatrix(my_mat_1)

#Matrix 2 for part B
my_mat_2 < −matrix(1: 9, nrow = 3, byrow = T)
my_mat_2

#calling the function for part B
transformationmatrix(my_mat_2)
```

Output

```
> #User defined function for diagonal
> transformationmatrix < −function(x){
+  my_dia < −diag(x)
+  my_mean < −mean(my_dia)
+  my_median < −median(my_dia)
+  my_vector < −c(my_mean, my_median)
+  return(my_vector)
+ }
>
> #Part A Matrix − 1
> my_mat_1 < −
+  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),
+      nrow = 5, byrow = T)
> my_mat_1
   [,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
> #calling the function for part A
> transformationmatrix(my_mat_1)
[1] 64.4 19.0
>
> #Matrix 2 for part B
> my_mat_2 < −matrix(1: 9, nrow = 3, byrow = T)
> my_mat_2
   [,1] [,2] [,3]
[1,]   1   2   3
[2,]   4   5   6
```

```
[3,]   7   8   9
>
> #calling the function for part B
> transformationmatrix(my_mat_2)
[1] 5 5
>
```

Analysis:
By filling out the different vectors of numeric type will give us the matrix. R studio tells us about the user defined functions.
The output of first matrix 64.4.19.0
Output of second matrix is 5 5

Exercise 10 chapter 4.4

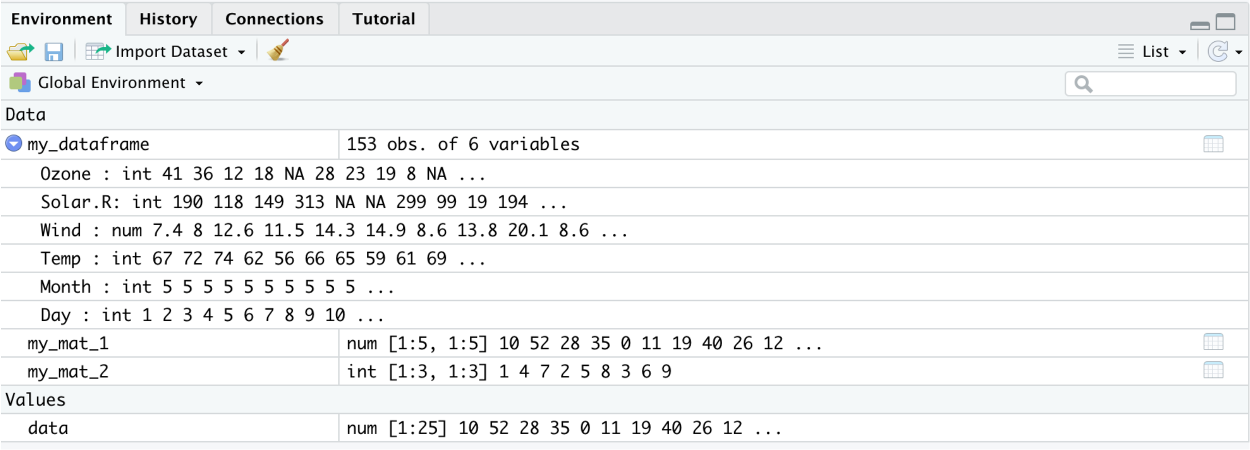
```
#Library(MASS) and Airquality dataset library(MASS)
my_dataframe  < − airquality

#Defining the function
df_cleanup  < − function(my_df, col_list)
{
 for(i in col_list){ #cat(i)
   if(sum(is.na(my_df[, i]  >  0))){
     my_df  < −my_df[−which(is.na(my_df[ , i])), ]
   }
 }#ending the i − loop

}#End of function
new_df  < − df_cleanup(my_dataframe, c(1,2))
```

Output

```
>  #Library(MASS) and Airquality dataset library(MASS)
>  my_dataframe  < − airquality
>
>  #Defining the function
>  df_cleanup  < − function(my_df, col_list)
+ {
+   for(i in col_list){ #cat(i)
+     if(sum(is.na(my_df[, i]  >  0))){
+       my_df  < −my_df[−which(is.na(my_df[ , i])), ]
+     }
+   }#ending the i − loop
+
+ }#End of function
> new_df <- df_cleanup(my_dataframe
```

Analysis:

I have created a user defined function where in I have checked the air quality data set and managed to clean the data with thw missing values and indexis and result can be seen below in the output.



```
Data
my_dataframe        153 obs. of 6 variables
   Ozone : int 41 36 12 18 NA 28 23 19 8 NA ...
   Solar.R: int 190 118 149 313 NA NA 299 99 19 194 ...
   Wind : num 7.4 8 12.6 11.5 14.3 14.9 8.6 13.8 20.1 8.6 ...
   Temp : int 67 72 74 62 56 66 65 59 61 69 ...
   Month : int 5 5 5 5 5 5 5 5 5 5 ...
   Day : int 1 2 3 4 5 6 7 8 9 10 ...
my_mat_1            num [1:5, 1:5] 10 52 28 35 0 11 19 40 26 12 ...
my_mat_2            int [1:3, 1:3] 1 4 7 2 5 8 3 6 9
Values
data                num [1:25] 10 52 28 35 0 11 19 40 26 12 ...
```