RPart1_Homework

NYC Data Science Academy

Question #1

Compound interest can be computed using the formula

$$A = P(1 + \frac{r}{100})^n$$

where P is the original money lent, A is what it amounts to in n years at R percent per year interest. Write an R command to calculate a vector of numbers indicating the amount of money owed (A) after n years, where n ranges from 1 to 15 in yearly increments. The original amount lent is 5000 dollars (P) and the interest rate remains constant throughout the period at 11.5% (r).

```
n = 1:15
P = 5000
r = .115
A = P*((1+ r)^n)
```

Question #2

Assume that we have collected the heights and weights of four people. The heights (in cm) are 180, 165, 160, 193; the respective weights (in kg) are 87, 58, 65, 100. Create two vectors, `height` and `weight`, using the data. Body mass index (BMI) is defined as

$$BMI = \frac{mass_{kg}}{height_m^2}$$

Write an R command to make a vector calculating the BMI values for the four people. Be careful of the units! As a challenge, use the height vector to make a boolean vector named `tall` of the heights above 6 feet.

```
height = c(180, 165, 160, 193)
weight = c(87, 58, 65, 100)

BMI =  (weight)/(height/100)

tall = height > 182.88
```

Question #3:

1. From your RStudio, import the built-in cars dataset by running `data(cars)`.
2. Print the first 5 lines from cars.
3. Randomly generate a vector as long as the the number of rows in cars containing elements NY, CA or CT. Call the vector `state`. Run the code `set.seed(0)` on the line above your vector. This makes your results reproducible (anybody who runs the code `set.seed(0)` on their randomized vector will end up with the same random vector you generated).
4. Add state to the data frame cars as a new column. Again name the column `state`.

5. Create a new column `ratio` whose value is the ratio `dist/speed`. Then compute the average and standard deviation of that column.

```
#1.
data(cars)

#2.
head(cars, n =5)
```

```
##   speed dist
## 1     4    2
## 2     4   10
## 3     7    4
## 4     7   22
## 5     8   16
```

```
#3
set.seed(0)
state = sample(x = c('NY', 'CA', 'CT'), size = length(cars$speed), replace = TRUE)


#4.
cars$state = sample(x = c('NY', 'CA', 'CT'), size = length(cars$speed), replace = TRUE)

#5.

cars$ratio = (cars$dist/cars$speed)

mean(cars$ratio)
```

```
## [1] 2.632496
```

```
sd(cars$ratio)
```

```
## [1] 1.068006
```

Question #4:

Read the `TimesSquareSignage.csv` and import it into R as `ts_data`. Then check the following features of the dataset:

1. The number of observations and the number of variables.
2. The type (`class`) of each variable.
3. How many missing values are there in the dataset?
4. Which rows (people) have missing values? Which columns (variables) include missing values?

```
ts_data = read.csv('https://s3.amazonaws.com/graderdata/TimesSquareSignage.csv', stringsAsFactors=FALSE)

# 1.
nrow(ts_data)
```

```
## [1] 184
```

```
#2.
sapply(ts_data,class)
```

```
## Screen.Name..LED...Vinyl.Signs.                Building.Address
##                       "character"                     "character"
##             Location.Description                        Location
##                       "character"                     "character"
##                            Height                            Type
##                       "character"                     "character"
##                                X.                           Width
##                         "integer"                       "integer"
##                         X__Height                              SF
##                         "integer"                       "integer"
##                        Note.Photo                             X_.
##                       "character"                       "integer"
##                           X_Width                        X_Height
##                         "integer"                       "integer"
##                              X_SF                           TOTAL
##                         "integer"                       "integer"
##                          TOTAL.SF                  TOTAL.BY.TYPE
##                         "integer"                     "character"
```

```
#3.
sum(is.na(ts_data))
```

```
## [1] 520
```

```
#4.
View(ts_data)

which(is.na(ts_data$Screen.Name..LED...Vinyl.Signs.))
```

```
## integer(0)
```

```
colSums(is.na(ts_data))
```

```
## Screen.Name..LED...Vinyl.Signs.                Building.Address
##                               0                               0
##             Location.Description                        Location
##                               0                               0
##                          Height                            Type
##                               0                               0
##                              X.                           Width
##                               0                               9
##                         X__Height                              SF
##                               9                               0
##                        Note.Photo                             X_.
##                               0                             167
##                          X_Width                        X_Height
```

```
##                     167                     167
##                    X_SF                   TOTAL
##                       1                       0
##                TOTAL.SF          TOTAL.BY.TYPE
##                       0                       0
```

Question #5:

From the Time Square dataset, we'd like to extract specific information about advertising in Midtown Manhattan. Obtain the following data frames and save them in a subfolder named `data` in your current directory as CSV files:

1. Observations from Upper Broadway. Save as `UpperBway.csv`.
2. Observations with greater-than-average square footage. Save as `SF.csv`.
3. The name, address, and location of the ten signs with the largest total square footage. Save as `TopTen.csv`.

```
#1.
ts_data = data.frame(ts_data)
UpperBway <- ts_data[ts_data$Location    == "Upper Bway",]
write.csv(UpperBway,'UpperBway.csv',row.names=FALSE)

#2.
mean_sf = mean(ts_data$SF)
SF <- ts_data[ts_data$SF>mean_sf,]
write.csv(SF, 'SF.csv',row.names=FALSE)

#.3
TopTen = ts_data[order(-ts_data$TOTAL.SF),][1:10,c(1,2,4)]
write.csv(TopTen, 'TopTen.csv',row.names=FALSE)
```