Homework Assignment 10

Armen Mkrtumyan

4/17/2023

Exercise 1 (35 points / 5 each)

1.1. Read *movies3.csv* to R. See the structure of the dataframe, what are the observations and what are the features of the dataframe. Print the head of the dataset.

```
data = read.csv("movies3.csv")
print(str(data)) #We can see 2912 observations and 33 features(title, genre_first...)
```

```
## 'data.frame':    2912 obs. of  33 variables:
##  $ title              : chr  "Zoom" "Zoolander 2" "Zookeeper" "Zombieland" ...
##  $ genre_first        : chr  "Action" "Comedy" "Comedy" "Adventure" ...
##  $ year               : int  2006 2016 2011 2009 2007 1998 2012 2005 2008 1998 ...
##  $ duration           : int  83 102 102 88 162 116 157 101 101 119 ...
##  $ gross_adjusted     : num  14142117 29451448 87570655 86365946 39077724 ...
##  $ budget_adjusted    : int  42555556 51065177 87177413 26964263 76858659 7519018 42704949 81581157 1
##  $ gross              : int  11631245 28837115 80360866 75590286 33048353 1980338 95720716 28045540 3
##  $ budget             : int  35000000 50000000 80000000 23600000 65000000 5000000 40000000 65000000 1
##  $ cast_facebook_likes: int  5022 24107 5392 28011 36928 1209 2759 32232 638 17768 ...
##  $ reviews            : int  176 376 305 998 966 232 1198 338 490 709 ...
##  $ index              : num  1.22 1.02 1.09 1.14 1.18 ...
##  $ Rated              : chr  "PG" "PG-13" "PG" "R" ...
##  $ Genre              : chr  "Action, Adventure, Comedy" "Comedy" "Comedy, Family, Romance" "Adventu
##  $ Director           : chr  "Peter Hewitt" "Ben Stiller" "Frank Coraci" "Ruben Fleischer" ...
##  $ Writer             : chr  "Adam Rifkin (screenplay), David Berenbaum (screenplay), Adam Rifkin (s
##  $ Actors             : chr  "Tim Allen, Courteney Cox, Chevy Chase, Spencer Breslin" "Justin Bieber
##  $ Plot               : chr  "Former superhero Jack is called back to work to transform an unlikely
##  $ Language           : chr  "English" "English, Italian, Spanish" "English" "English" ...
##  $ Country            : chr  "USA" "USA" "USA" "USA" ...
##  $ Awards             : chr  "4 wins & 8 nominations." "7 wins & 17 nominations." "1 win & 2 nominat
##  $ Metascore          : int  26 34 30 73 78 NA 95 67 56 57 ...
##  $ imdbRating         : num  4.3 4.7 5.2 7.7 7.7 7 7.4 6.1 6.6 6.6 ...
##  $ imdbVotes          : chr  "16367" "53943" "49098" "426786" ...
##  $ Production         : chr  "Sony Pictures Entertainment" "Paramount Pictures" "Columbia Pictures"
##  $ DVD                : chr  "9/2/2007" "5/24/2016" "10/11/2011" "2/2/2010" ...
##  $ Release            : chr  "8/11/2006" "2/12/2016" "7/8/2011" "10/2/2009" ...
##  $ Release_Month      : int  8 2 7 10 3 1 1 11 10 12 ...
##  $ Release_Day        : int  11 12 8 2 2 30 11 11 31 18 ...
##  $ Release_year       : int  2006 2016 2011 2009 2007 1998 2013 2005 2008 1998 ...
##  $ OscarWon           : int  0 0 0 0 0 0 1 0 0 0 ...
##  $ OtherWin           : int  4 7 1 9 2 0 87 2 2 5 ...
##  $ OscarNom           : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ OtherNom           : int  8 17 2 28 67 1 171 3 4 7 ...
## NULL
```

```
print(head(data))
```

```
##         title genre_first year duration gross_adjusted budget_adjusted    gross
## 1        Zoom      Action 2006       83       14142117        42555556 11631245
## 2 Zoolander 2      Comedy 2016      102       29451448        51065177 28837115
## 3   Zookeeper      Comedy 2011      102       87570655        87177413 80360866
## 4  Zombieland   Adventure 2009       88       86365946        26964263 75590286
## 5      Zodiac       Crime 2007      162       39077724        76858659 33048353
## 6 Zero Effect      Comedy 1998      116        2978040         7519018  1980338
##     budget cast_facebook_likes reviews    index Rated                     Genre
## 1 35000000                5022     176 1.215873    PG Action, Adventure, Comedy
## 2 50000000               24107     376 1.021304 PG-13                   Comedy
## 3 80000000                5392     305 1.089718    PG   Comedy, Family, Romance
## 4 23600000               28011     998 1.142553     R Adventure, Comedy, Horror
## 5 65000000               36928     966 1.182441     R     Crime, Drama, History
## 6  5000000                1209     232 1.503804     R      Comedy, Crime, Drama
##          Director
## 1    Peter Hewitt
## 2     Ben Stiller
## 3    Frank Coraci
## 4 Ruben Fleischer
## 5   David Fincher
## 6     Jake Kasdan
##
## 1                                                                  Adam Rifkin (screenplay), David Be
## 2                              Justin Theroux, Ben Stiller, Nicholas Stoller, John Hamburg, Drake Sath
## 3 Nick Bakay (screenplay), Rock Reuben (screenplay), Kevin James (screenplay), Jay Scherick (screenp
## 4
## 5
## 6
##                                                        Actors
## 1            Tim Allen, Courteney Cox, Chevy Chase, Spencer Breslin
## 2            Justin Bieber, Jon Daly, Pen\xe9lope Cruz, Ben Stiller
## 3               Kevin James, Rosario Dawson, Leslie Bibb, Ken Jeong
## 4     Jesse Eisenberg, Woody Harrelson, Emma Stone, Abigail Breslin
## 5 Jake Gyllenhaal, Mark Ruffalo, Anthony Edwards, Robert Downey Jr.
## 6               Bill Pullman, Ben Stiller, Ryan O'Neal, Kim Dickens
##
## 1
## 2
## 3                                                 A group of zoo animals decide to break their
## 4                        A shy student trying to reach his family in Ohio, a gun-toting tough guy t
## 5                       In the late 1960s/early 1970s, a San Francisco cartoonist becomes an amateu
## 6 The world's greatest detective Daryl Zero aided by his associate Steve Arlo investigates a complex
##                    Language Country                   Awards Metascore
## 1                   English     USA  4 wins & 8 nominations.        26
## 2 English, Italian, Spanish     USA 7 wins & 17 nominations.        34
## 3                   English     USA   1 win & 2 nominations.        30
## 4                   English     USA 9 wins & 28 nominations.        73
## 5                   English     USA 2 wins & 67 nominations.        78
## 6                   English     USA            1 nomination.        NA
##   imdbRating imdbVotes                  Production        DVD   Release
## 1        4.3     16367 Sony Pictures Entertainment   9/2/2007 8/11/2006
## 2        4.7     53943          Paramount Pictures  5/24/2016 2/12/2016
```

```
## 3        5.2     49098           Columbia Pictures 10/11/2011  7/8/2011
## 4        7.7    426786      Sony/Columbia Pictures   2/2/2010 10/2/2009
## 5        7.7    353948          Paramount Pictures  7/24/2007  3/2/2007
## 6        7.0     12912           Warner Home Video   7/7/1998 1/30/1998
##   Release_Month Release_Day Release_year OscarWon OtherWin OscarNom OtherNom
## 1             8          11         2006        0        4        0        8
## 2             2          12         2016        0        7        0       17
## 3             7           8         2011        0        1        0        2
## 4            10           2         2009        0        9        0       28
## 5             3           2         2007        0        2        0       67
## 6             1          30         1998        0        0        0        1
```

1.2. Calculate the average of the column budget.

```
mean_budget = mean(data$budget)
print(mean_budget)
```

```
## [1] 40183590
```

1.3. Subset the dataframe to have only columns gross_adjusted, budget_adjusted, gross, budget (store in variable movies_sub).

```
movies_sub = data[,c("gross_adjusted", "budget_adjusted", "gross", "budget")]
```

1.4. Calculate the mean of each column of the subsetted dataframe by *apply*.

```
apply(movies_sub, MARGIN = 2, mean)
```

```
##  gross_adjusted budget_adjusted          gross         budget
##        84532189        51922575        57613345        40183590
```

1.5. Find the movie title with the minimum badget according to the dataset.

```
index = which.min(data$budget)
data[index, "title"]
```

```
## [1] "Tarnation"
```

1.6. Calculate how many movies are there for which the budget of the movie is smaller than the mean budget and when number of reviews is greater than 200.

```
sum(data$budget < mean_budget & data$reviews > 200)
```

```
## [1] 1304
```

1.7. Add a new column to dataframe *Years_after_prod* that will be equal to *Release_year* - *year*. Look at the summary of the new column.

```
data$Years_after_prod <- data$Release_year - data$year
summary(data$Years_after_prod)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##  0.0000  0.0000  0.0000  0.1334  0.0000 14.0000     445
```

Exercise 2 (35 points / 5 each)

2.1. For the following part, import *countries.csv* dataset to R by reading it. Then print first 4 observations in the dataset. Next state how many observations and how many features the dataframe has? Are all features numeric ones?

```
countries = read.csv("countries.csv")
print(head(countries, n = 4))
```

```
##          Country               Region Population Area_sqm Popullation_Density
## 1    Afghanistan ASIA (EX. NEAR EAST)   31056997   647500                48.0
## 2        Albania       EASTERN EUROPE    3581655    28748               124.6
## 3        Algeria      NORTHERN AFRICA   32930091  2381740                13.8
## 4 American Samoa              OCEANIA      57794      199               290.4
##   Coast_Area_Ratio Net_Migration Infant_Mortaility_Per_1000Birth GDP_Per_Capita
## 1             0.00         23.06                          163.07            700
## 2             1.26         -4.93                           21.52           4500
## 3             0.04         -0.39                           31.00           6000
## 4            58.29        -20.71                            9.27           8000
##   Literacy Phones_Per_1000 Arable Crops Other Climate Birthrate Deathrate
## 1     36.0             3.2  12.13  0.22 87.65       1     46.60     20.34
## 2     86.5            71.2  21.09  4.42 74.49       3     15.11      5.22
## 3     70.0            78.1   3.22  0.25 96.53       1     17.14      4.61
## 4     97.0           259.5  10.00 15.00 75.00       2     22.46      3.27
##   Agriculture Industry Service
## 1       0.380    0.240   0.380
## 2       0.232    0.188   0.579
## 3       0.101    0.600   0.298
## 4          NA       NA      NA
```

```
print(str(countries)) #NO, all the features are not numeric, we also have chr, int
```

```
## 'data.frame':    227 obs. of  20 variables:
##  $ Country                       : chr  "Afghanistan" "Albania" "Algeria" "American Samoa" ...
##  $ Region                        : chr  "ASIA (EX. NEAR EAST)" "EASTERN EUROPE" "NORTHERN AFRICA" "
##  $ Population                    : int  31056997 3581655 32930091 57794 71201 12127071 13477 69108 
##  $ Area_sqm                      : int  647500 28748 2381740 199 468 1246700 102 443 2766890 29800 
##  $ Popullation_Density           : num  48 124.6 13.8 290.4 152.1 ...
##  $ Coast_Area_Ratio              : num  0 1.26 0.04 58.29 0 ...
##  $ Net_Migration                 : num  23.06 -4.93 -0.39 -20.71 6.6 ...
##  $ Infant_Mortaility_Per_1000Birth: num  163.07 21.52 31 9.27 4.05 ...
##  $ GDP_Per_Capita                : int  700 4500 6000 8000 19000 1900 8600 11000 11200 3500 ...
##  $ Literacy                      : num  36 86.5 70 97 100 42 95 89 97.1 98.6 ...
##  $ Phones_Per_1000               : num  3.2 71.2 78.1 259.5 497.2 ...
##  $ Arable                        : num  12.13 21.09 3.22 10 2.22 ...
##  $ Crops                         : num  0.22 4.42 0.25 15 0 0.24 0 4.55 0.48 2.3 ...
##  $ Other                         : num  87.7 74.5 96.5 75 97.8 ...
##  $ Climate                       : num  1 3 1 2 3 NA 2 2 3 4 ...
##  $ Birthrate                     : num  46.6 15.11 17.14 22.46 8.71 ...
##  $ Deathrate                     : num  20.34 5.22 4.61 3.27 6.25 ...
##  $ Agriculture                   : num  0.38 0.232 0.101 NA NA 0.096 0.04 0.038 0.095 0.239 ...
##  $ Industry                      : num  0.24 0.188 0.6 NA NA 0.658 0.18 0.22 0.358 0.343 ...
##  $ Service                       : num  0.38 0.579 0.298 NA NA 0.246 0.78 0.743 0.547 0.418 ...
## NULL
```

2.2. The column Population shows the population for all the countries. Find the maximum and minimum population values in the dataset.

```
minimum_pop = min(countries$Population)
maximum_pop = max(countries$Population)
paste("Minimum: ", minimum_pop)
```

```
## [1] "Minimum:  7026"
```

```
paste("Maximum: ", maximum_pop)
```

```
## [1] "Maximum:  1313973713"
```

2.3. Now when you have the minimum and maximum values of the population you can subset the dataset, so you can see which are the countries with these populations. Find those countries.

```
print(subset(countries, Population == minimum_pop))
```

```
##                   Country           Region Population Area_sqm
## 175 St Pierre & Miquelon NORTHERN AMERICA       7026      242
##     Popullation_Density Coast_Area_Ratio Net_Migration
## 175                  29            49.59         -4.86
##     Infant_Mortaility_Per_1000Birth GDP_Per_Capita Literacy Phones_Per_1000
## 175                            7.54           6900       99           683.2
##     Arable Crops Other Climate Birthrate Deathrate Agriculture Industry Service
## 175  13.04     0 86.96      NA     13.52      6.83          NA       NA      NA
```

```
print(subset(countries, Population == maximum_pop))
```

```
##    Country               Region Population Area_sqm Popullation_Density
## 43   China ASIA (EX. NEAR EAST) 1313973713  9596960               136.9
##    Coast_Area_Ratio Net_Migration Infant_Mortaility_Per_1000Birth
## 43             0.15          -0.4                           24.18
##    GDP_Per_Capita Literacy Phones_Per_1000 Arable Crops Other Climate Birthrate
## 43           5000     90.9           266.7   15.4  1.25 83.35     1.5     13.25
##    Deathrate Agriculture Industry Service
## 43      6.97       0.125    0.473   0.403
```

2.4. Now suppose we want to consider only those countries, which are in the region *C.W. OF IND. STATES*. Subset the dataframe as follows. Name the new dataframe *CIS_countries*. These are countries that are members of Commonwealth of Independent States (CIS) and one of them is Armenia. How many countries are there that belong to the CIS.

```
CIS_countries <- subset(countries, countries$Region == "C.W. OF IND. STATES")
print(nrow(CIS_countries)) #There are 12 countries which belong to CIS
```

```
## [1] 12
```

2.5. Consider the climate of the CIS countries. Calculate the mean and standard deviation of the feature. What can you say about the results? Hint: If you are getting NA after running functions, one reason can be that the variable has NA inside. Look at the help of the functions by typing *?function_name*, specifically for the argument *na.rm*.

```
#The standard deviation of 2.55 and mean of 1.25 indicate that there is large amount of variability in
sd(CIS_countries$Climate, na.rm = T)
```

```
## [1] 1.257201
```

```
mean(CIS_countries$Climate, na.rm = T)
```

```
## [1] 2.55
```

2.6. The difference between the birth rate and the death rate of a country or place is called the natural increase. The natural increase is calculated by subtracting the death rate from the birth rate. *Natural increase = birth rate - death rate.* Calculate Natural Increase and keep it in the dataframe under a column *NaturalIncrease*. Then find which countries have the highest and lowest natural increase. Are they the same as those countries with minimum and maximum Population that you found in one of previous exercises. (6

points)
Hint: For adding new feature, recall how you calculated number of lost games during the lectures, when you had only wins and draws. The steps for min and max should be the same as for finding the countries as they were in the case of Population. If you obtain NAs, consider what have you done while calculating mean and standard deviation to avoid the problem.

```
countries$NaturalIncrease <- countries$Birthrate - countries$Deathrate
minimum =  min(countries$NaturalIncrease, na.rm = T)
maximum = max(countries$NaturalIncrease, na.rm = T)
paste("Minimum natural increase: ", subset(countries, NaturalIncrease == minimum)$Country)
```

```
## [1] "Minimum natural increase:  Botswana"
```

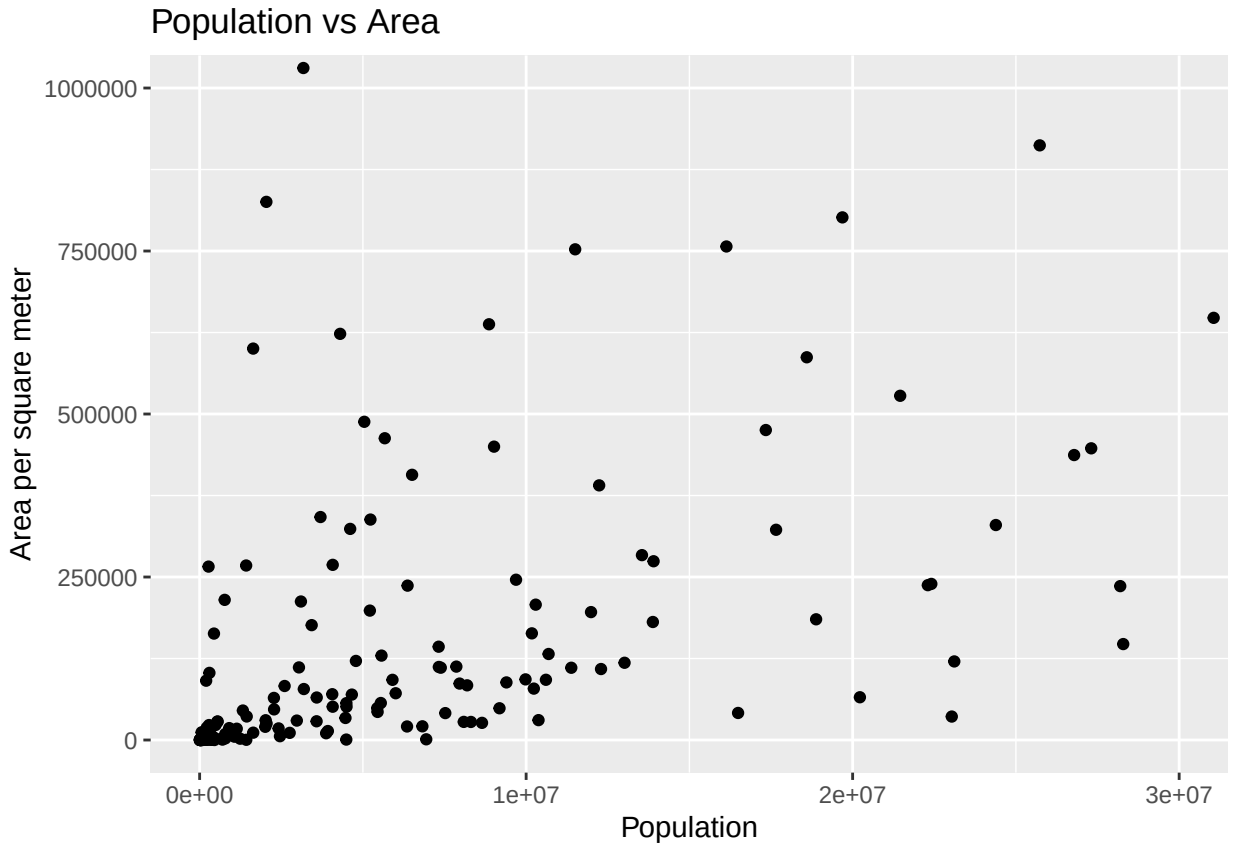
```
paste("Maximum natural increase: ", subset(countries, NaturalIncrease == maximum)$Country)
```

```
## [1] "Maximum natural increase:  Gaza Strip"
```

2.7. Plot a scatterplot using ggplot2 library between *Population* and *Area_sqm*. Do not forget give proper names to axes and to the graph itself. Interpret graph in a few words. Can you make the graph any better? Hint: ?xlim and ?ylim

```
library("ggplot2")

ggplot(data=countries, aes(x=Population, y=Area_sqm)) +geom_point() + labs(y= "Area per square meter",
```




```
#The coord_caresian is added to keep all the points and not get a warning that some points were deleted
#The graph is scattered so it does not show any linear correlation between the population per area, onl
#Some relations, when fewer the population meant smaller area
```

Exercise 3 (30 points / 10 each)

3.1. Using a for loop get the factorial of 14, and check your result with R's built-in function *factorial()*.

```
my_factorial<-function(n)
{
  if(n == 0)
    return(1)
  else
    return(n * factorial(n - 1))
}
my_factorial(12)
```

```
## [1] 479001600
```

```
factorial(12)
```

```
## [1] 479001600
```

3.2. Write a while loop that prints out standard random normal numbers (use rnorm()) but stops (breaks) if you get a number bigger than 1.

```
number <- 0
while(number <= 1)
{
  number <- rnorm(1)
  print(number)
}
```

```
## [1] -0.8496796
## [1] -0.2413364
## [1] -0.114657
## [1] -0.5972514
## [1] -1.137295
## [1] -1.204471
## [1] 0.1833554
## [1] -0.2621988
## [1] 0.4362364
## [1] 1.056705
```

```
print(number)
```

```
## [1] 1.056705
```

3.3. Write a function that will get an input n (n $>$ 1), and will return the n-th Fibonacci number.

```
my_fibonacci<-function(n)
{
  if(n < 2)
    return(n)
  else
    return(my_fibonacci(n-1) + my_fibonacci(n-2))
}
my_fibonacci(10)
```

```
## [1] 55
```