Assignment 1

2024-02-11

Question 1 - Print the structure of your dataset

```
str(netflix)
```

```
## 'data.frame':      8790 obs. of  10 variables:
##  $ show_id     : chr  "s1" "s3" "s6" "s14" ...
##  $ type        : chr  "Movie" "TV Show" "TV Show" "Movie" ...
##  $ title       : chr  "Dick Johnson Is Dead" "Ganglands" "Midnight Mass" "Confessions of an Invisibl
##  $ director    : chr  "Kirsten Johnson" "Julien Leclercq" "Mike Flanagan" "Bruno Garotti" ...
##  $ country     : chr  "United States" "France" "United States" "Brazil" ...
##  $ date_added  : chr  "9/25/2021" "9/24/2021" "9/24/2021" "9/22/2021" ...
##  $ release_year: int  2020 2021 2021 2021 1993 2021 2021 2019 2021 2013 ...
##  $ rating      : chr  "PG-13" "TV-MA" "TV-MA" "TV-PG" ...
##  $ duration    : chr  "90 min" "1 Season" "1 Season" "91 min" ...
##  $ listed_in   : chr  "Documentaries" "Crime TV Shows, International TV Shows, TV Action & Adventure
```

Question 2 - List the variables in your dataset

```
names(netflix)
```

```
##  [1] "show_id"      "type"         "title"        "director"     "country"
##  [6] "date_added"   "release_year" "rating"       "duration"     "listed_in"
```

Question 3 - Print the top 15 rows of your dataset

```
head(netflix, 15)
```

```
##    show_id    type                              title
## 1       s1   Movie               Dick Johnson Is Dead
## 2       s3 TV Show                          Ganglands
## 3       s6 TV Show                      Midnight Mass
## 4      s14   Movie  Confessions of an Invisible Girl
## 5       s8   Movie                            Sankofa
## 6       s9 TV Show     The Great British Baking Show
## 7      s10   Movie                       The Starling
## 8     s939   Movie  Motu Patlu in the Game of Zones
## 9      s13   Movie                       Je Suis Karl
## 10    s940   Movie         Motu Patlu in Wonderland
## 11    s941   Movie      Motu Patlu: Deep Sea Adventure
## 12    s942   Movie           Motu Patlu: Mission Moon
## 13    s852   Movie                   99 Songs (Tamil)
## 14    s471   Movie        Bridgerton - The Afterparty
## 15      s730     Movie      Bling Empire - The Afterparty
##                          director        country date_added release_year
## 1                 Kirsten Johnson United States  9/25/2021         2020
## 2                 Julien Leclercq         France  9/24/2021         2021
## 3                   Mike Flanagan United States  9/24/2021         2021
## 4                   Bruno Garotti         Brazil  9/22/2021         2021
## 5                    Haile Gerima United States  9/24/2021         1993
## 6                 Andy Devonshire United Kingdom  9/24/2021         2021
## 7                 Theodore Melfi United States  9/24/2021         2021
## 8                     Suhas Kadav          India 05-01-2021         2019
## 9             Christian Schwochow        Germany  9/23/2021         2021
## 10                    Suhas Kadav          India 05-01-2021         2013
## 11                    Suhas Kadav          India 05-01-2021         2014
## 12                    Suhas Kadav          India 05-01-2021         2013
## 13                      Not Given       Pakistan  5/21/2021         2021
## 14 Krysia Plonka, Kristian Mercado United States  7/13/2021         2021
## 15 Krysia Plonka, Kristian Mercado United States 06-12-2021         2021
##    rating  duration
## 1   PG-13    90 min
## 2   TV-MA  1 Season
## 3   TV-MA  1 Season
## 4   TV-PG    91 min
## 5   TV-MA   125 min
## 6   TV-14 9 Seasons
## 7   PG-13   104 min
## 8   TV-Y7    87 min
## 9   TV-MA   127 min
## 10  TV-Y7    76 min
## 11  TV-Y7    76 min
## 12  TV-Y7    71 min
## 13  TV-14   131 min
## 14  TV-14    39 min
## 15  TV-MA    36 min
##                                                        listed_in
## 1                                                  Documentaries
## 2  Crime TV Shows, International TV Shows, TV Action & Adventure
## 3                             TV Dramas, TV Horror, TV Mysteries
## 4                               Children & Family Movies, Comedies
## 5                 Dramas, Independent Movies, International Movies
## 6                                     British TV Shows, Reality TV
## 7                                                 Comedies, Dramas
## 8            Children & Family Movies, Comedies, Music & Musicals
## 9                                     Dramas, International Movies
## 10                     Children & Family Movies, Music & Musicals
## 11                              Children & Family Movies, Comedies
## 12                              Children & Family Movies, Comedies
## 13              Dramas, International Movies, Music & Musicals
## 14                                                          Movies
## 15                                                          Movies
```

Question 4 - Write a user defined function using any of the variables from the data set.

```r
year_with_highest_frequency <- function(data) {
  year_counts <- table(data$release_year)
  year_with_highest <- names(year_counts)[which.max(year_counts)]
  return(year_with_highest)
}
year_highest_frequency <- year_with_highest_frequency(netflix)
```

Question 5 - Use data manipulation techniques and filter rows based on any logical criteria that exist in your dataset.

```r
tv_ma_shows <- subset(netflix, rating == "TV-MA")
```

Question 6 - Identify the dependent & independent variables and use reshaping techniques and create a new data frame by joining those variables from your dataset.

```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(tidyr)
rating_counts <- netflix %>%
    group_by(release_year, rating) %>%
    summarise(count = n()) %>%
    ungroup()
```

```
## `summarise()` has grouped output by 'release_year'. You can override using the
## `.groups` argument.
```

```r
rating_counts_wide <- rating_counts %>%
    pivot_wider(names_from = rating, values_from = count, values_fill = 0)
print(rating_counts_wide)
```

```
## # A tibble: 74 x 15
##    release_year `TV-14` `TV-PG` `TV-MA` `TV-G` `PG-13`     G    NR     R    PG
##           <int>   <int>   <int>   <int>  <int>   <int> <int> <int> <int> <int>
##  1         1925       1       0       0      0       0     0     0     0     0
##  2         1942       2       0       0      0       0     0     0     0     0
##  3         1943       0       3       0      0       0     0     0     0     0
##  4         1944       2       1       0      0       0     0     0     0     0
##  5         1945       2       0       2      0       0     0     0     0     0
##  6         1946       1       1       0      0       0     0     0     0     0
##  7     1947    0    1    0    0    0    0    0    0    0
##  8     1954    1    0    0    1    0    0    0    0    0
##  9     1955    1    1    0    0    1    0    0    0    0
## 10     1956    1    0    0    0    0    1    0    0    0
## # i 64 more rows
## # i 5 more variables: UR <int>, ‘TV-Y7‘ <int>, ‘TV-Y‘ <int>, ‘TV-Y7-FV‘ <int>,
## #   ‘NC-17‘ <int>
```

Question 7 - Remove missing values in your dataset.

```
netflix_clean <- na.omit(netflix)
```

Question 8 - Identify and remove duplicated data in your dataset

```
duplicated_rows <- duplicated(netflix)
netflix_unique <- netflix[!duplicated_rows, ]
```

Question 9 - Reorder multiple rows in descending order

```
netflix_ordered <- netflix[order(netflix$release_year, decreasing = TRUE), ]
```

Question 10 - Rename some of the column names in your dataset

```
names(netflix)[names(netflix) == "listed_in"] <- "Category"
```

Question 11 - Add new variables in your data frame by using a mathematical function (for e.g. – multiply an existing column by 2 and add it as a new variable to your data frame)

```
netflix$release_year_double <- netflix$release_year * 2
```

Question 12 - Create a training set using random number generator engine

```
set.seed(123)
train_indices <- sample(nrow(netflix), 0.8 * nrow(netflix))
train_set <- netflix[train_indices, ]
```

Question 13 - Print the summary statistics of your dataset

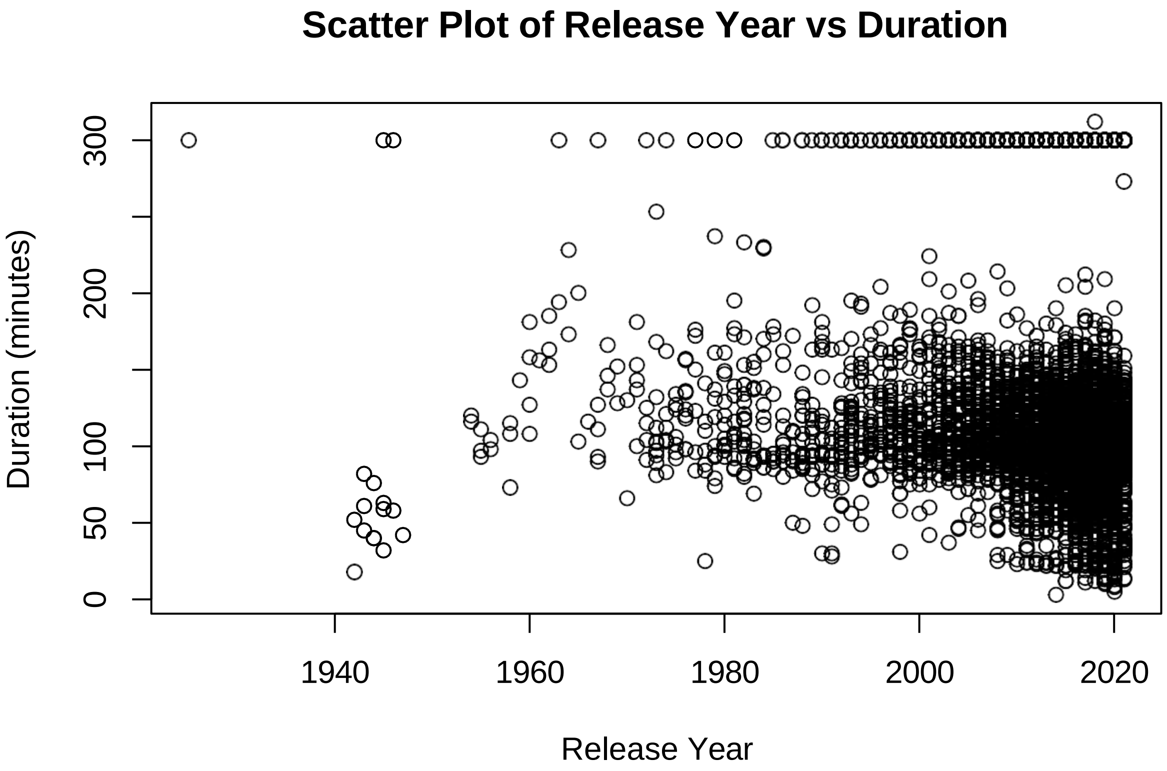
```
summary(netflix)
```

```
##    show_id              type               title              director
##  Length:8790        Length:8790        Length:8790        Length:8790
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##    country           date_added         release_year     rating
##  Length:8790        Length:8790        Min.   :1925   Length:8790
##  Class :character   Class :character   1st Qu.:2013   Class :character
##  Mode  :character   Mode  :character   Median :2017   Mode  :character
##                                        Mean   :2014
##                                        3rd Qu.:2019
##                                        Max.   :2021
##    duration           Category         release_year_double
##  Length:8790        Length:8790        Min.   :3850
##  Class :character   Class :character   1st Qu.:4026
##  Mode  :character   Mode  :character   Median :4034
##                                        Mean   :4028
##                                        3rd Qu.:4038
##                                        Max.   :4042
```

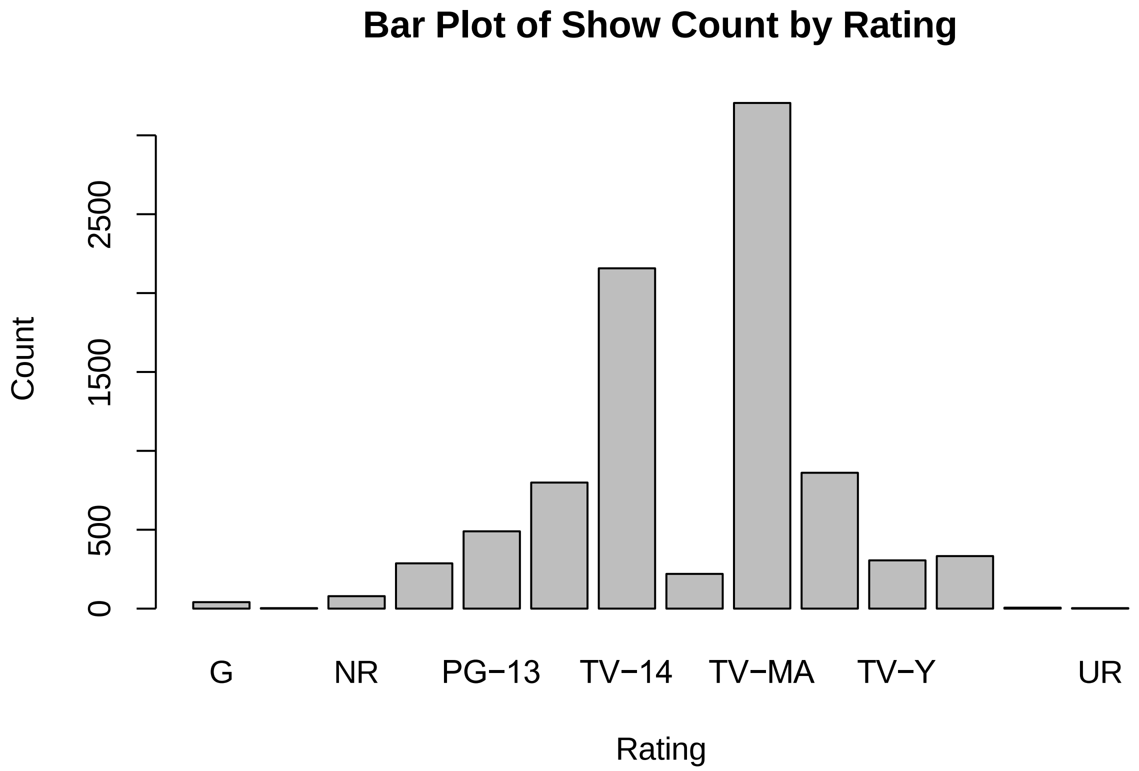
Question 14 - Use any of the numerical variables from the dataset and perform the following statistical functions Mean Median Mode Range

```
# Mean
mean_release_year <- mean(netflix$release_year, na.rm = TRUE)

# Median
median_release_year <- median(netflix$release_year, na.rm = TRUE)

# Mode
mode_release_year <- Mode(netflix$release_year)

# Range
range_release_year <- range(netflix$release_year, na.rm = TRUE)
```

Questions 15 - Plot a scatter plot for any 2 variables in your dataset



Question 16 - Plot a bar plot for any 2 variables in your dataset



Question 17 - Find the correlation between any 2 variables by applying least square linear regression model

```
model <- lm(duration_numeric ~ release_year, data = netflix)
summary(model)
```

```
##
## Call:
## lm(formula = duration_numeric ~ release_year, data = netflix)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
##  -163.47   -70.67   -46.07   131.53   264.64
##
##   Coefficients:
##                 Estimate  Std. Error  t value  Pr(>|t|)
## (Intercept)    -2661.890     229.539   -11.60    <2e-16 ***
## release_year       1.401       0.114    12.29    <2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 94.29 on 8788 degrees of freedom
## Multiple R-squared:   0.01691,    Adjusted R-squared:   0.0168
## F-statistic: 151.2 on 1 and 8788 DF,   p-value: < 2.2e-16
```