netflix.Rmd

1) Data loading:

The file NetflixData.csv contains data on # 930 TV shows and movies.

```
setwd("~/Desktop")
netflix <- read.csv("NetflixData.csv")
```

2) Data preparation:

Extracted the value of year from Sys.time() and saved it in a new object called current_year. Inspected the class of current_year and converted it to numeric.

```
current_year <- format(Sys.Date(), "%Y")
class(current_year)
```

```
## [1] "character"
```

```
current_year <- as.numeric(current_year)
```

Created a new column named time_since_release and assigned it the value of the current year minus the release_year. This variable gives the number of years since the release of the tv show/movie.

```
netflix$time_since_release <- current_year - netflix$release_year
```

Created a new column named title_length and assigned it the value of number of characters in the title of each of the tv show/movie.

```
netflix$title_length <- nchar(netflix$title)
```

Inspected the class of each column in the dataframe with a for loop.

```
for (i in 1:ncol(netflix)){
  print(class(netflix[,i]))
}
```

```
## [1] "integer"
## [1] "character"
## [1] "character"
## [1] "character"
## [1] "character"
## [1] "integer"
## [1] "character"
## [1] "integer"
## [1] "numeric"
## [1] "numeric"
## [1] "integer"
```

3) Data summary table:

Created a function that takes two inputs: input_data (a dataframe) and id (a column id). The goal of the function is to compute the descriptive statistics (mean, median, min, max) for the variable in the column id of the dataframe input_data if the column is numeric or integer.

```
summ <- function(input_data, id){
  if(class(input_data[,id]) == "numeric" | class(input_data[,id]) == "integer"){
    summary <- data.frame(cbind(var = colnames(input_data)[id],
                                mean = round(mean(input_data[,id]),2),
                              median = median(input_data[,id]),
                                min = min(input_data[,id]),
                                max = max(input_data[,id])))
    print(summary)
  }
}
```

4) Data analysis:

Invoked the function for variable time_since_release and saved the output of the function in a new object called output_data.

```
# get column id of variable time_since_release
id <- which(colnames(netflix) == "time_since_release")
# invoke the function for netflix data and column id = id (i.e., 10)
output_data <- summ(netflix, id)
```

```
##                  var  mean median min max
## 1 time_since_release 10.14      7   3  82
```

```
class(output_data)
```

```
## [1] "data.frame"
```

```
# it is a data.frame
```

Invoked the function for variable title_length and duration_min_season. Appended the output for these to output_data.

```
# get column id of variable title_length
id2 <- which(colnames(netflix) == "title_length")
# invoke the function for netflix data and column id = id2 (i.e., 11)
output_data <- rbind(output_data, summ(netflix, id2))
```

```
##            var  mean median min max
## 1 title_length 17.24     15   1  73
```

```
# repeat the process for duration_min_season
# get column id of variable duration_min_season
id3 <- which(colnames(netflix) == "duration_min_season")
# invoke the function for netflix data and column id = id3 (i.e., 8)
output_data <- rbind(output_data, summ(netflix, id3))
```

```
##                   var  mean median min max
## 1 duration_min_season 71.07     90   1 312
```

```
write.csv(output_data, "output_data.csv")
print(output_data)
```

```
##                   var  mean median min max
## 1  time_since_release 10.14      7   3  82
## 2        title_length 17.24     15   1  73
## 3 duration_min_season 71.07     90   1 312
```

Split the data into two dataframes named "tv_shows" and "movies" and repeated steps above to create two summary tables for tv_shows and movies separately. Included the same variables, i.e., time_since_release, title_length, and duration_min_season.

```
# create data for tv_shows
tv_shows <- netflix[which(netflix$type == "TV Show"),]
# create data for movies
movies <- netflix[which(netflix$type == "Movie"),]

output_data_tv_shows <- summ(tv_shows, id)
```

```
##                  var mean median min max
## 1 time_since_release 7.71      6   3  61
```

```
output_data_tv_shows <- rbind(output_data_tv_shows, summ(tv_shows, id2))
```

```
##            var  mean median min max
## 1 title_length 17.03     15   1  44
```

```
output_data_tv_shows <- rbind(output_data_tv_shows, summ(tv_shows, id3))
```

```
##                   var mean median min max
## 1 duration_min_season 1.84      1   1  15
```

```
print(output_data_tv_shows)
```

```
##                   var  mean median min max
## 1  time_since_release  7.71      6   3  61
## 2        title_length 17.03     15   1  44
## 3 duration_min_season  1.84      1   1  15
```

```
output_data_movies <- summ(movies, id)
```

```
##                  var  mean median min max
## 1 time_since_release 11.16      8   3  82
```

```
output_data_movies <- rbind(output_data_movies, summ(movies, id2))
```

```
##            var  mean median min max
## 1 title_length 17.34     15   2  73
```

```
output_data_movies <- rbind(output_data_movies, summ(movies, id3))
```

```
##                   var   mean median min max
## 1 duration_min_season 100.24     99  14 312
```

```
print(output_data_movies)
```

```
##                   var   mean median min max
## 1  time_since_release  11.16      8   3  82
## 2        title_length  17.34     15   2  73
## 3 duration_min_season 100.24     99  14 312
```

```
# Movies have 9.16 years since release compared with 5.71 for tv shows.
# They seem similar on title length with approx. 17 characters.
```

5) Regression:

Ran two different regression models separately for tv_shows and movies, to examine the relationship between sales and other variables. The dependent variable is sales. The independent variables are time_since_release, title_length, duration_min_season. Also included country and rating and interpreted the regression coefficients.

```
r1 <- lm(sales ~ time_since_release + title_length + duration_min_season, data = tv_shows)
r2 <- lm(sales ~ time_since_release + title_length + duration_min_season, data = movies)
summary(r1)
```

```
##
## Call:
## lm(formula = sales ~ time_since_release + title_length + duration_min_season,
##     data = tv_shows)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.1851 -0.8260  0.0220  0.8634  4.4232
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         10.471339   0.221780  47.215  < 2e-16 ***
## time_since_release   0.110952   0.013725   8.084 2.07e-14 ***
## title_length         0.003825   0.009590   0.399    0.690
## duration_min_season -0.041260   0.049880  -0.827    0.409
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.327 on 272 degrees of freedom
## Multiple R-squared:  0.2023, Adjusted R-squared:  0.1935
## F-statistic: 22.99 on 3 and 272 DF,  p-value: 2.705e-13
```

```r
# Older the movies, higher the sales (p < 0.001)
summary(r2)
```

```
##
## Call:
## lm(formula = sales ~ time_since_release + title_length + duration_min_season,
##     data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.1131 -1.0228  0.0509  0.9879  3.7867
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         17.641127   0.248772  70.913   <2e-16 ***
## time_since_release  -0.095526   0.005733 -16.664   <2e-16 ***
## title_length        -0.009509   0.005580  -1.704   0.0889 .
## duration_min_season  0.497798   0.001964 253.509   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.417 on 651 degrees of freedom
## Multiple R-squared:  0.991,  Adjusted R-squared:  0.991
## F-statistic: 2.388e+04 on 3 and 651 DF,  p-value: < 2.2e-16
```

```r
# Newer the movies, higher the sales (p < 0.001)
# Longer the duration, higher the sales (p < 0.001)

# Next, include country and rating
r1b <- lm(sales ~ time_since_release + title_length + duration_min_season + as.factor(country) + as.fac
r2b <- lm(sales ~ time_since_release + title_length + duration_min_season + as.factor(country) + as.fac

summary(r1b)
```

```
##
## Call:
## lm(formula = sales ~ time_since_release + title_length + duration_min_season +
##     as.factor(country) + as.factor(rating), data = tv_shows)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.9568 -0.7011  0.0000  0.7414  4.6182
##
## Coefficients:
##                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     10.8845675  1.5007435   7.253 6.06e-12 ***
## time_since_release               0.1126505  0.0150598   7.480 1.52e-12 ***
## title_length                     0.0006994  0.0105762   0.066   0.9473
## duration_min_season             -0.0335500  0.0545726  -0.615   0.5393
## as.factor(country)Argentina     -0.5652755  1.1369133  -0.497   0.6195
## as.factor(country)Australia     -1.0935898  0.9126535  -1.198   0.2320
## as.factor(country)Belgium       -0.6486392  1.4808798  -0.438   0.6618
```

```
## as.factor(country)Brazil         -0.6559955  0.8645632  -0.759   0.4488
## as.factor(country)Canada         -0.6366348  0.7412025  -0.859   0.3913
## as.factor(country)China          -1.4990808  0.9012412  -1.663   0.0976 .
## as.factor(country)Colombia       -2.1540295  1.4711609  -1.464   0.1445
## as.factor(country)Denmark         0.4839906  1.4804721   0.327   0.7440
## as.factor(country)Finland        -2.4408737  1.4818100  -1.647   0.1009
## as.factor(country)France          0.4741722  0.7611345   0.623   0.5339
## as.factor(country)Germany         0.5308010  0.9968998   0.532   0.5949
## as.factor(country)India          -0.1617089  0.7393902  -0.219   0.8271
## as.factor(country)Ireland        -0.7294564  1.4835414  -0.492   0.6234
## as.factor(country)Israel         -0.1477560  1.4797765  -0.100   0.9205
## as.factor(country)Italy          -0.0567180  1.1480545  -0.049   0.9606
## as.factor(country)Japan          -0.0142749  0.7160378  -0.020   0.9841
## as.factor(country)Lebanon        -0.8891609  1.4716753  -0.604   0.5463
## as.factor(country)Malaysia       -0.2137256  0.9966154  -0.214   0.8304
## as.factor(country)Mexico         -0.6511711  0.7643416  -0.852   0.3951
## as.factor(country)Netherlands    -1.7962104  1.4964340  -1.200   0.2312
## as.factor(country)Norway         -0.0868322  1.4910773  -0.058   0.9536
## as.factor(country)Poland          1.2450583  1.1376601   1.094   0.2749
## as.factor(country)Russia          1.1784119  1.5015316   0.785   0.4334
## as.factor(country)Singapore      -0.6119740  1.1258819  -0.544   0.5873
## as.factor(country)South Africa   -0.0741508  1.4795547  -0.050   0.9601
## as.factor(country)South Korea     0.1383327  0.6960442   0.199   0.8426
## as.factor(country)Spain          -1.4167888  0.8088916  -1.752   0.0812 .
## as.factor(country)Sweden         -1.8650753  1.1390221  -1.637   0.1029
## as.factor(country)Taiwan         -0.2973155  0.7397119  -0.402   0.6881
## as.factor(country)Thailand       -0.6986747  0.9952929  -0.702   0.4834
## as.factor(country)Turkey         -0.1748323  1.1262088  -0.155   0.8768
## as.factor(country)United Kingdom -0.2670385  0.6551171  -0.408   0.6839
## as.factor(country)United States  -0.1993425  0.6248096  -0.319   0.7500
## as.factor(rating)TV-14           -0.2484371  1.3597296  -0.183   0.8552
## as.factor(rating)TV-G            -0.2017103  1.4090380  -0.143   0.8863
## as.factor(rating)TV-MA            0.0514162  1.3580360   0.038   0.9698
## as.factor(rating)TV-PG           -0.0865257  1.3627093  -0.063   0.9494
## as.factor(rating)TV-Y            -0.2595788  1.3790049  -0.188   0.8509
## as.factor(rating)TV-Y7           -0.5532761  1.4036684  -0.394   0.6938
## as.factor(rating)TV-Y7-FV         1.0794285  1.9419844   0.556   0.5789
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.339 on 232 degrees of freedom
## Multiple R-squared:  0.3073, Adjusted R-squared:  0.1789
## F-statistic: 2.393 on 43 and 232 DF,  p-value: 1.835e-05
```

```
summary(r2b)
```

```
##
## Call:
## lm(formula = sales ~ time_since_release + title_length + duration_min_season +
##     as.factor(country) + as.factor(rating), data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.0528 -0.8745  0.0077  0.9113  3.9765
```

```
##
## Coefficients: (1 not defined because of singularities)
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     21.269423   2.082155  10.215   <2e-16 ***
## time_since_release              -0.096582   0.006152 -15.698   <2e-16 ***
## title_length                    -0.013723   0.006014  -2.282   0.0229 *
## duration_min_season              0.500260   0.002405 208.006   <2e-16 ***
## as.factor(country)Argentina     -0.036944   1.722311  -0.021   0.9829
## as.factor(country)Australia     -0.031592   1.584831  -0.020   0.9841
## as.factor(country)Austria       -0.480890   1.813013  -0.265   0.7909
## as.factor(country)Belgium       -3.921930   2.112446  -1.857   0.0639 .
## as.factor(country)Brazil        -0.379702   1.567392  -0.242   0.8087
## as.factor(country)Bulgaria      -0.790722   2.067051  -0.383   0.7022
## as.factor(country)Cambodia      -1.433127   2.070775  -0.692   0.4892
## as.factor(country)Canada        -0.349329   1.521657  -0.230   0.8185
## as.factor(country)Chile          0.404646   2.073227   0.195   0.8453
## as.factor(country)China         -1.529484   1.676535  -0.912   0.3620
## as.factor(country)Colombia      -1.931321   2.073636  -0.931   0.3520
## as.factor(country)Denmark       -3.704739   2.496485  -1.484   0.1383
## as.factor(country)Egypt         -0.213866   1.570051  -0.136   0.8917
## as.factor(country)Finland       -3.179871   2.070654  -1.536   0.1252
## as.factor(country)France        -0.462372   1.540853  -0.300   0.7642
## as.factor(country)Germany       -0.794812   1.616394  -0.492   0.6231
## as.factor(country)Ghana         -1.836005   1.815533  -1.011   0.3123
## as.factor(country)Greece         0.969841   2.064729   0.470   0.6387
## as.factor(country)Hong Kong     -0.260197   1.576185  -0.165   0.8689
## as.factor(country)India         -0.761111   1.528607  -0.498   0.6187
## as.factor(country)Indonesia     -1.148243   1.576239  -0.728   0.4666
## as.factor(country)Ireland       -0.916516   1.717097  -0.534   0.5937
## as.factor(country)Israel        -1.837470   1.716832  -1.070   0.2849
## as.factor(country)Italy         -0.590844   1.601708  -0.369   0.7123
## as.factor(country)Japan         -0.677441   1.558546  -0.435   0.6640
## as.factor(country)Kuwait        -2.054212   2.077266  -0.989   0.3231
## as.factor(country)Lebanon       -0.149511   2.065620  -0.072   0.9423
## as.factor(country)Malaysia      -1.607421   1.818178  -0.884   0.3770
## as.factor(country)Mexico        -0.759188   1.573322  -0.483   0.6296
## as.factor(country)Namibia        0.435290   2.073009   0.210   0.8338
## as.factor(country)Netherlands   -0.859335   1.678664  -0.512   0.6089
## as.factor(country)New Zealand   -2.045396   1.809385  -1.130   0.2588
## as.factor(country)Nigeria       -1.126068   1.572059  -0.716   0.4741
## as.factor(country)Norway        -0.507200   1.721121  -0.295   0.7683
## as.factor(country)Pakistan       0.179190   1.625634   0.110   0.9123
## as.factor(country)Philippines   -0.762897   1.589362  -0.480   0.6314
## as.factor(country)Saudi Arabia  -1.387413   1.619457  -0.857   0.3919
## as.factor(country)Serbia         0.555125   2.067971   0.268   0.7885
## as.factor(country)Singapore     -1.251333   2.070877  -0.604   0.5459
## as.factor(country)Slovenia      -0.119018   2.067313  -0.058   0.9541
## as.factor(country)South Africa  -0.745710   1.669704  -0.447   0.6553
## as.factor(country)South Korea   -0.648960   1.610003  -0.403   0.6870
## as.factor(country)Spain          0.245309   1.562820   0.157   0.8753
## as.factor(country)Sweden        -2.162021   2.070486  -1.044   0.2968
## as.factor(country)Taiwan        -1.881120   1.816024  -1.036   0.3007
## as.factor(country)Thailand      -2.017988   1.812355  -1.113   0.2660
## as.factor(country)Turkey        -1.453623   1.573467  -0.924   0.3559
```

```
## as.factor(country)United Kingdom -0.412159   1.523608  -0.271   0.7869
## as.factor(country)United States  -0.407710   1.505265  -0.271   0.7866
## as.factor(country)Uruguay        -0.709834   2.073396  -0.342   0.7322
## as.factor(rating)NR              -2.860948   1.485287  -1.926   0.0546 .
## as.factor(rating)PG              -3.656433   1.434041  -2.550   0.0110 *
## as.factor(rating)PG-13           -3.310065   1.430281  -2.314   0.0210 *
## as.factor(rating)R               -3.548990   1.422994  -2.494   0.0129 *
## as.factor(rating)TV-14           -3.234192   1.422158  -2.274   0.0233 *
## as.factor(rating)TV-G            -3.041819   1.455434  -2.090   0.0370 *
## as.factor(rating)TV-MA           -3.068279   1.418863  -2.162   0.0310 *
## as.factor(rating)TV-PG           -3.084497   1.427958  -2.160   0.0312 *
## as.factor(rating)TV-Y            -3.127097   1.500857  -2.084   0.0376 *
## as.factor(rating)TV-Y7           -3.363143   1.473114  -2.283   0.0228 *
## as.factor(rating)TV-Y7-FV               NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.41 on 591 degrees of freedom
## Multiple R-squared:  0.9919, Adjusted R-squared:  0.991
## F-statistic:  1148 on 63 and 591 DF,  p-value: < 2.2e-16
```

```
# results remain consistent
# i.e., signs of time_since_release + title_length + duration_min_season effects are
# still similar and robust
```