Success of Movies

Torrey Capobianco

Bellevue University

Fall 2019

Introduction

Movies often base their success off of revenue generated in the opening week. However, revenue cannot speak to the success of a movie, as there are often times that a the budget or money spent on the creation of the movie is higher than the revenue. In this analysis, we will look at what contributes to the success of a movie based on the film's profit.

The Data

The original data was retrieved from Kaggle, compiled from The Movie Database. The data contained 20 variables, reporting 4,803 cases. Through the data cleaning process, the data set was narrowed down to 8 variables, to include the variables of interest: ID, Genres, Production Companies, Revenue, Budget, Runtime, Release Date, and Voting Average. These variables were chosen for the following:

- Genres – Does the genre of a movie have an effect on success?
- Production Companies – Do specific production companies have more success than others?
- Budget – Does a higher budget add to the success of a movie?
- Runtime – If a movie is in theaters longer, does it contribute to higher profit?
- Release date – Does the time of year matter in its success?
- Voting – Does a movie's viewer feedback add to the success of a film?

Several new variables were created to aid in the analysis of the research question. To look at the profitability of each movie, a new variable was needed to be created, titled "profit." This calculated the difference between the budget and revenue variables.
One feature that was to be looked at is if the time of year a movie was released plays a part in the success of a movie. The month from the release date was broken out into its own column. From here, the months were grouped by seasons.

- Winter: November – January
- Spring: February – April
- Summer: May – July
- Fall: August – October

By looking at the summary of the data frame, the top 5 most frequent genres in the set was revealed. A new variable was created to look at the top 5 which are in order: action, adventure, comedy, drama, and horror.

Insights

The data that was observed was non-parametric. From performing a correlation matrix between budget, runtime, vote average, revenue, and profit, it is interesting to note the findings between the variables. Runtime does not have a strong positive correlation with revenue, profit, or vote average. As well, the budget of a movie is not strongly correlated to the profit of a movie. According to the correlation matrix, as more money is spent, it doesn't necessarily mean that they will have a high profit. This can be visually viewed by the scatter plots, as some points dip into the negative numbers on the profit axis as the budget increases. Profit does show a strong correlation with revenue, as one would believe would account to the success of a movie.

From viewing scatterplots, the regression lines all indicate that budget, runtime, and vote average have a positive relationship, however, with vote average likely being very minimal contribution to the profit due to the very low sloped line.

When looking at budget vs profit by seasons, it is interesting to note that fall seems to be the least profitable. Summer and Winter reflect very similar regression lines, indicating that both those seasons are comparable to the higher profit of a movie when the budget increases. When looking at genre's regression line, action followed by adventure have a stronger positive linear regression line with horror being the weakest. Lastly, looking at budget vs. profit by production company, Paramount comes in with the strongest linear regression line.

By completing simple linear regression models, it is concluded that budget accounts for the most of the contribution to success (profit) of a movie. Results of x predicting profit:
Budget: 32.3% of variation in profit

- Run time: 4.9% of variation in profit
- Vote average: 4.8% of variation in profit
- Season: 3.3% of variation in profit
- Genre: 8.2% of variation in profit

When predicting profit based on production company, it resulted in a negative adjusted r-squared. It did not have any affect on the profit of a movie.

In a multiple linear regression model, the best fit model was predicting profit from budget, run time, vote average, and summer as the season. When adding the highest the genre, action, or production company, Paramount Pictures, to the multiple linear regression model, it did not improve the fit of the model. The best fit model produced a multiple R-squared of 0.3638 presenting that the variables accounted for 36.4% of the variability in predicting the profit of a movie. Although this is a low percentage, it was the highest that was fitted based on the variables that were observed. When splitting the data for a training model and a testing model, the root mean square error was high for both cases. Although the test model produced a lower RMSE number than the training model, indicating that the model was not overfitted, it is still concluded that this multiple linear regression model is not the best model to predict the profit of a movie.

Concluding Remarks to the Target Audience

Through this analysis, it is observed that as the budget increases, there is the likelihood that the profit will increase as well. To have a successful movie, one can plan to release their film either in the summer months or winter, as that reveals to be the most successful time of year. The length of time the movie is in theaters does not account for a large variation in the increase of profit as well as the viewership voting. The most successful genre appears to be action, however, when all variables are accounting toward what predicts the profitability of a movie, genre does not seem to have any impact in increasing the profit. Additional factors that were not researched have a large contribution to the profit a movie makes. However, based on this research, in terms of the production company, all have a fair game in creating a successful movie.

Limitations

With the data at hand, a multiple linear regression model of 36.4% accounted in predicting the profit of a movie. There is 63.6% of explanation that is not accounted for in this analysis. This percentage can include how much advertising for the movie was done and the impact it had on the viewership. Lead actors, directors, producers, as well as the franchise could also be factors that account for this variability. Since movies also grab to emotions, there is a possibility that there is not a perfectly fit model to the prediction of the success, as emotion is an anomaly that is always in flux. Overall, there is other research beyond this analysis that needs to be completed to look at the success of a movie.

Analysis Process

Packages

```r
library(jsonlite)
library(dplyr)
library(tidyverse)
library(ggplot2)
library(lubridate)
library(car)
library(caTools)
```

Data

```r
movies <- read_csv("tmdb_5000_movies.csv")
```

Data Cleaning

Parsing JSON

```r
genres <- movies %>%
  filter(nchar(genres)>2) %>%
  mutate(js = lapply(genres, fromJSON)) %>%
  unnest(js, .name_repair = "unique") %>%
  select(id, title, genres=name) %>%
  mutate_if(is.character, factor)

production_companies <- movies %>%
  filter(nchar(production_companies)>2) %>%
  mutate(js = lapply(production_companies, fromJSON)) %>%
  unnest(js, .name_repair = "unique") %>%
  select(id, title, production_companies=name) %>%
  mutate_if(is.character, factor)

m1 <-
  genres %>%
  group_by(id) %>%
  filter(row_number()==1)

m2 <-
  production_companies %>%
  group_by(id) %>%
  filter(row_number()==1)

movies2 <- merge(movies,m1, by="id")

movies3 <- merge(movies2, m2, by="id")
```

Select Variables

```r
moviesdf <- movies3[,c("id", "genres.y", "production_companies.y", "revenue",
"budget", "runtime", "vote_average", "release_date")]
```

```r
summary(moviesdf)
```

```
##        id                 genres.y
##  Min.   :     5   Drama    :1095
##  1st Qu.:  8587   Comedy   : 958
##  Median : 13536   Action   : 727
##  Mean   : 53186   Adventure: 334
##  3rd Qu.: 50837   Horror   : 279
##  Max.   :459488   Crime    : 188
##                   (Other)  : 868
##                              production_companies.y    revenue
##  Paramount Pictures                     : 281       Min.   :0.000e+00
##  Universal Pictures                     : 260       1st Qu.:0.000e+00
##  Columbia Pictures                      : 200       Median :2.522e+07
##  Twentieth Century Fox Film Corporation: 177       Mean   :8.875e+07
##  New Line Cinema                        : 157       3rd Qu.:1.009e+08
##  Walt Disney Pictures                   : 114       Max.   :2.788e+09
##  (Other)                                :3260
##      budget              runtime        vote_average     release_date
##  Min.   :        0   Min.   :  0.0   Min.   : 0.000   Min.   :1916-09-04
##  1st Qu.:  2500000   1st Qu.: 94.0   1st Qu.: 5.600   1st Qu.:1999-02-26
##  Median : 17000000   Median :104.0   Median : 6.300   Median :2005-09-05
##  Mean   : 31273706   Mean   :108.2   Mean   : 6.176   Mean   :2002-09-24
##  3rd Qu.: 41000000   3rd Qu.:118.0   3rd Qu.: 6.800   3rd Qu.:2010-12-23
##  Max.   :380000000   Max.   :338.0   Max.   :10.000   Max.   :2016-09-16
##                      NA's   :2
```

Uncovering New Information

Profit Variable

```r
moviesdf$profit <- moviesdf$revenue - moviesdf$budget
moviesdf <- na.omit(moviesdf)
```

Seasons

```r
moviesdf$month <- month(ymd(moviesdf$release_date))

moviesdf <- mutate(moviesdf, season = ifelse(month %in% 2:4, "spring",
                                  ifelse(month %in% 5:7, "summer",
                                         ifelse(month %in% 8:10, "fall", "wint
er"))))

moviesdf$season <- as.factor(moviesdf$season)

moviesdf$summer <-
  grepl("summer", moviesdf$season) %>%
  as.factor()
```

Genres

Top 5 most frequent genres in the data frame.

```
moviesdf <- mutate(moviesdf, genre = ifelse(genres.y %in% "Action", "action",
                                            ifelse(genres.y %in% "Adventure",
"adventure",
                                                   ifelse(genres.y %in% "Dram
a", "drama",
                                                          ifelse(genres.y %in
% "Comedy", "comedy",
                                                                 ifelse(genre
s.y %in% "Horror", "horror", "other"))))))

moviesdf$genre <- as.factor(moviesdf$genre)

moviesdf$action <-
  grepl("Action", moviesdf$genres.y) %>%
  as.factor()
```

Production Companies

Top 5 most frequent production companies in the data frame.

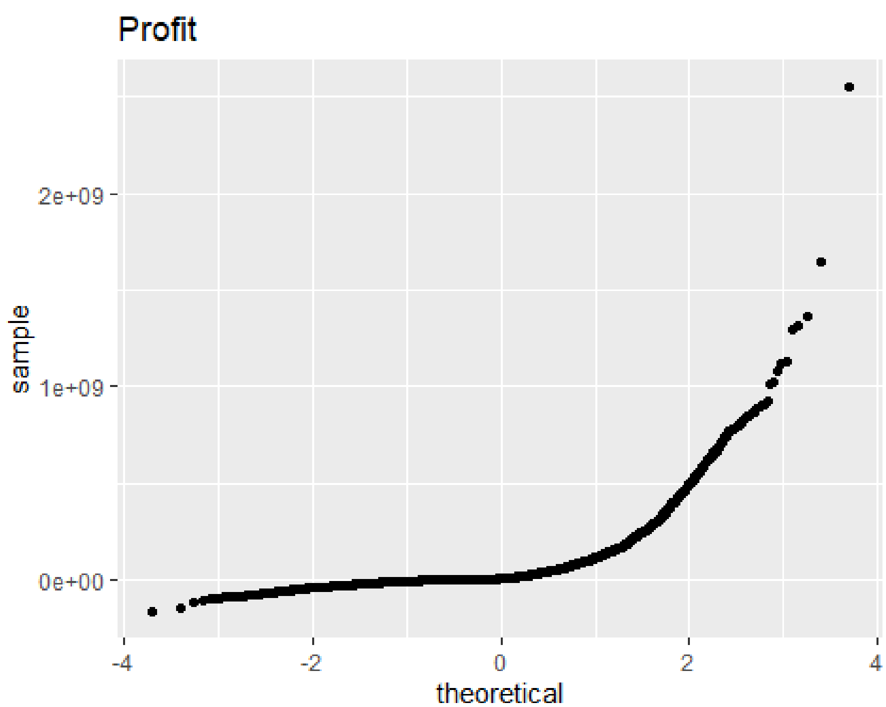
```
moviesdf <- mutate(moviesdf, company = ifelse(production_companies.y %in% "Pa
ramount Pictures", "paramount",
                                            ifelse(production_companies.y %in
% "Universal Pictures", "universal",
                                                   ifelse(production_companie
s.y %in% "Columbia Pictures", "columbia",
                                                          ifelse(production_c
ompanies.y %in% "Twentieth Century Fox Film Corporation", "fox",
                                                                 ifelse(produ
ction_companies.y %in% "Walt Disney Pictures", "disney", "other"))))))

moviesdf$company <- as.factor(moviesdf$company)

moviesdf$paramount <-
  grepl("Paramount Pictures", moviesdf$production_companies.y) %>%
  as.factor()
```

Plots

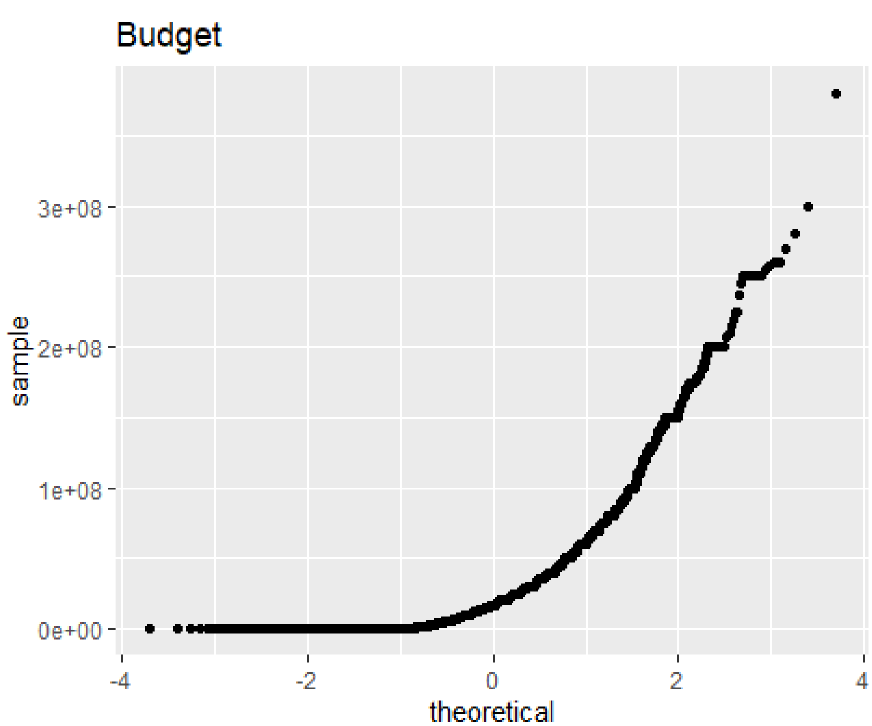
Q-Q Plots

Looking at the below graphs, all four variables are non-parametric due to the curve in the line.

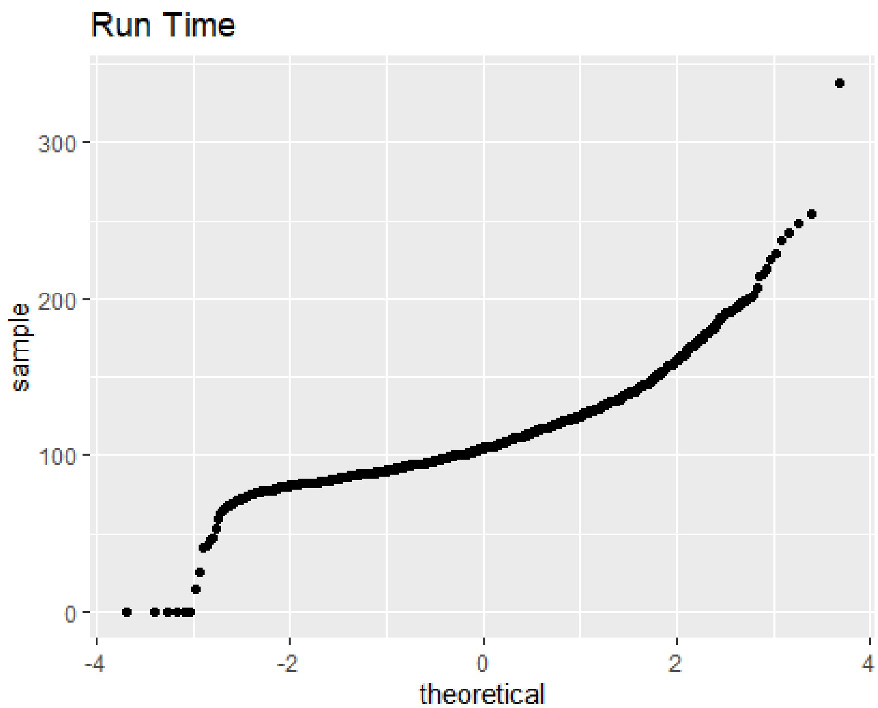
```r
qplot(sample = moviesdf$profit, stat="qq") + labs(x = "theoretical", y = "sample", title = "Profit")
```



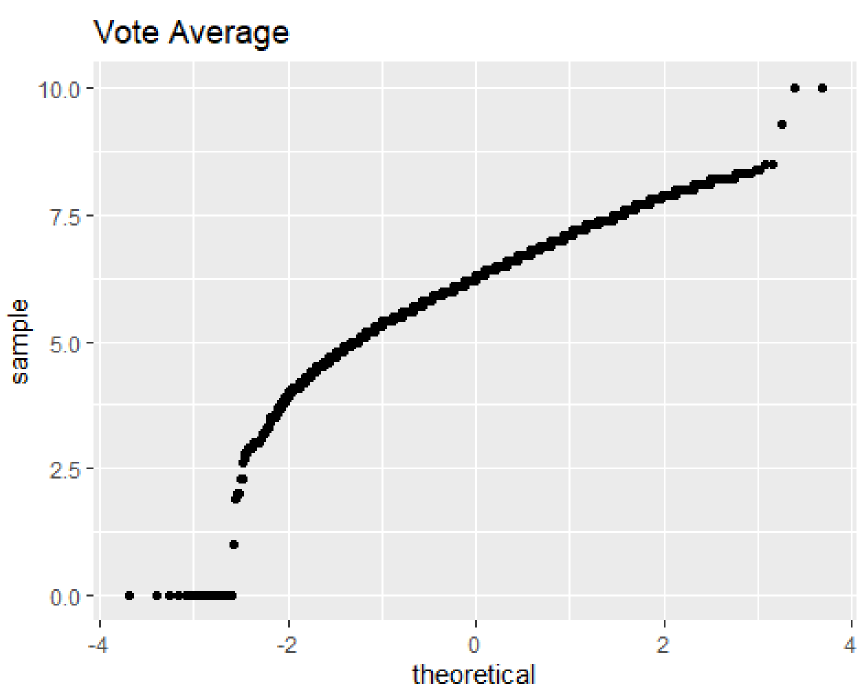

```r
qplot(sample = moviesdf$budget, stat="qq") + labs(x = "theoretical", y = "sample", title = "Budget")
```



```
qplot(sample = moviesdf$runtime, stat="qq") + labs(x = "theoretical", y = "sa
mple", title = "Run Time")
```



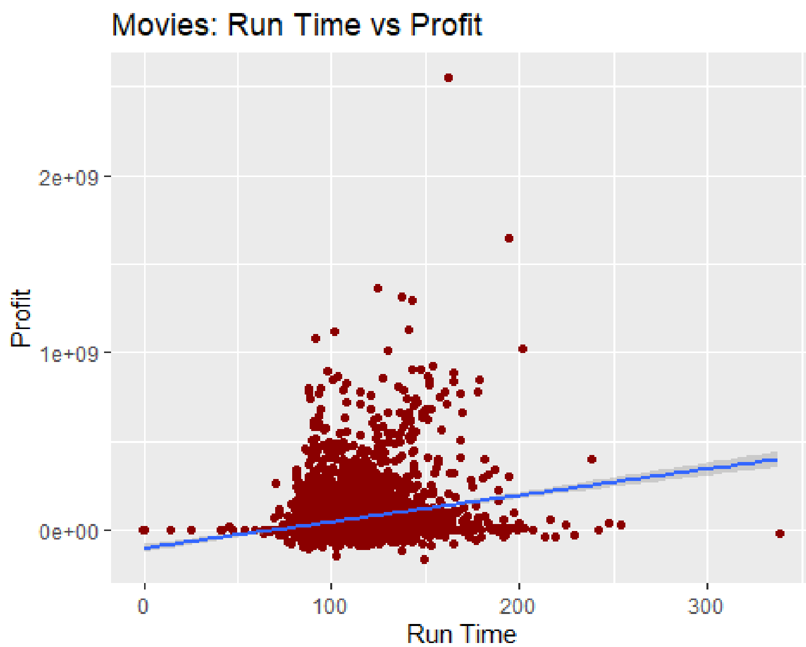

```
qplot(sample = moviesdf$vote_average, stat = "qq") + labs(x = "theoretical",
y = "sample", title = "Vote Average")
```



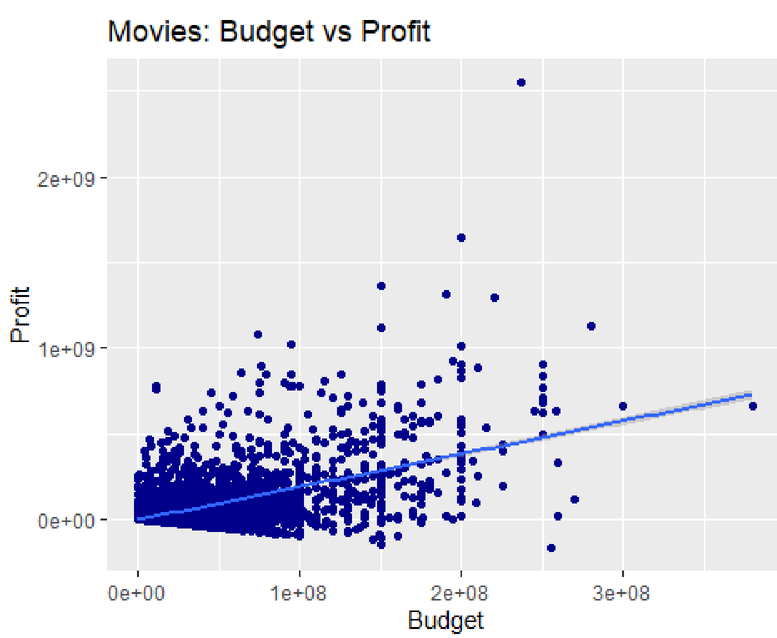
Scatter Plots

Scatter plots with regression line:

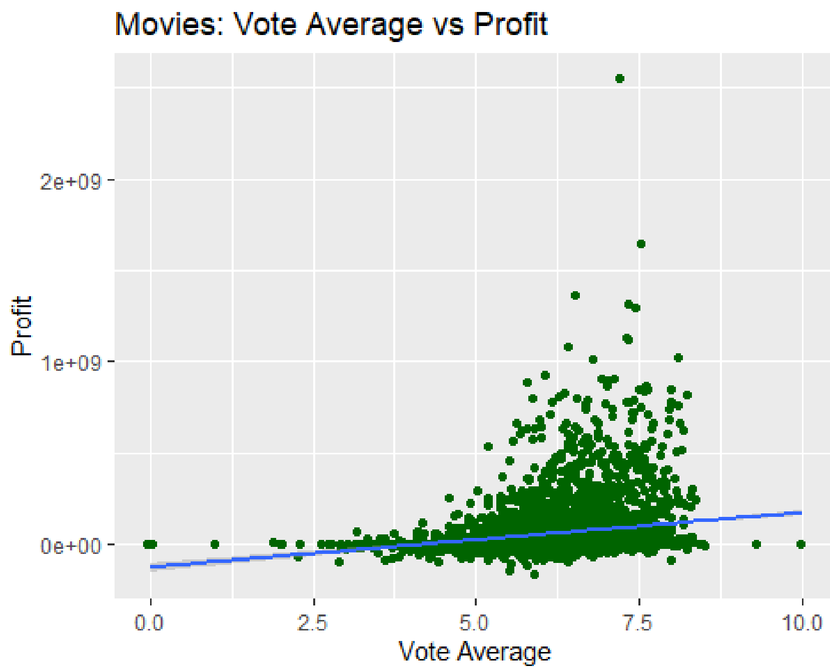
```
ggplot(moviesdf, aes(x = runtime, y = profit)) + geom_point(position = "jitte
r", color = "dark red") + geom_smooth(method = lm) + labs(x = "Run Time", y =
"Profit", title = "Movies: Run Time vs Profit")
```



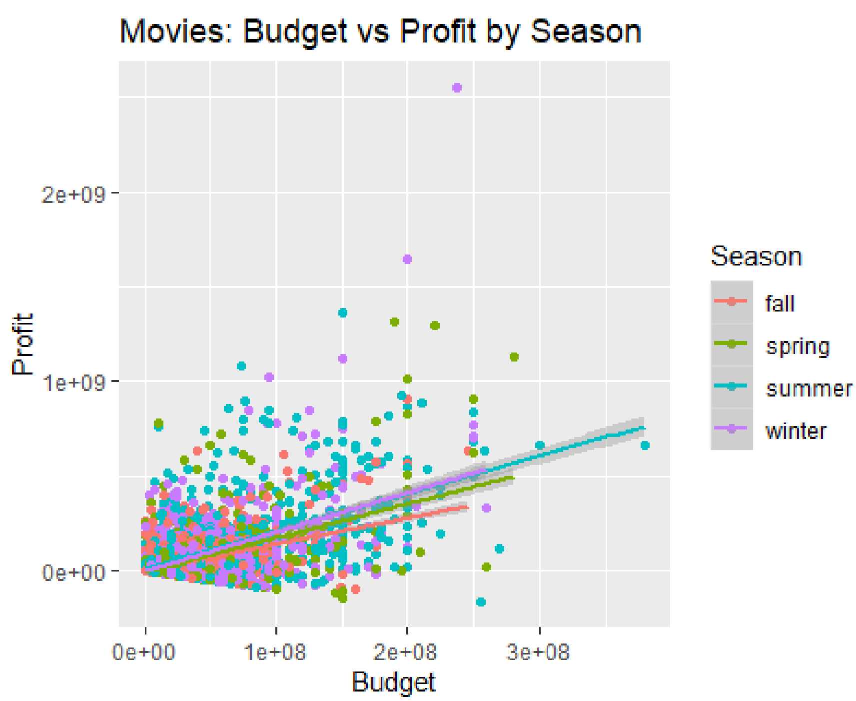

```
ggplot(moviesdf, aes(x = budget, y = profit)) + geom_point(position = "jitter
", color = "dark blue") + geom_smooth(method = lm) + labs(x = "Budget", y = "
Profit", title = "Movies: Budget vs Profit")
```



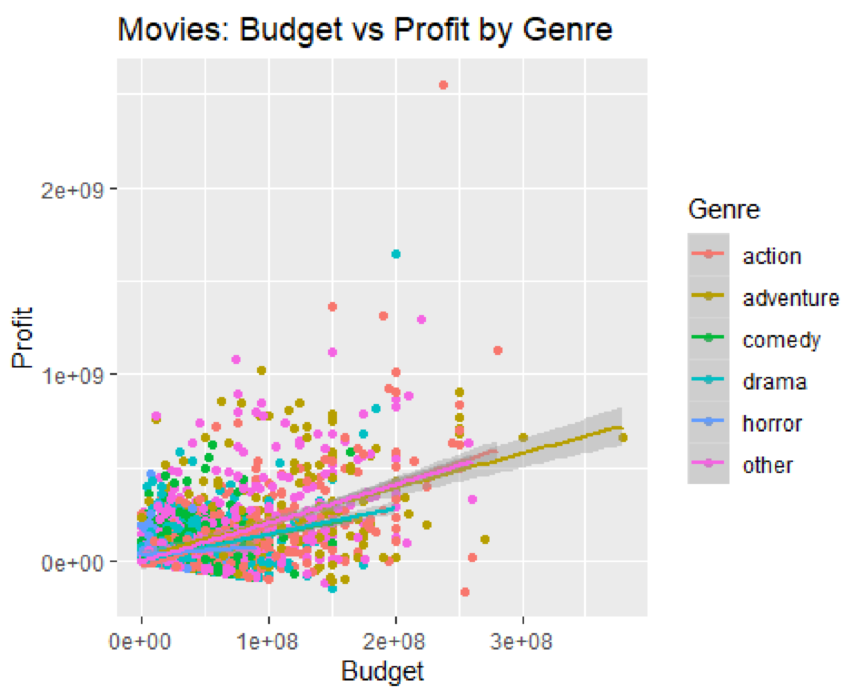
```r
ggplot(moviesdf, aes(x = vote_average, y = profit)) + geom_point(position = "
jitter", color = "dark green") + geom_smooth(method = lm) + labs(x = "Vote Av
erage", y = "Profit", title = "Movies: Vote Average vs Profit")
```



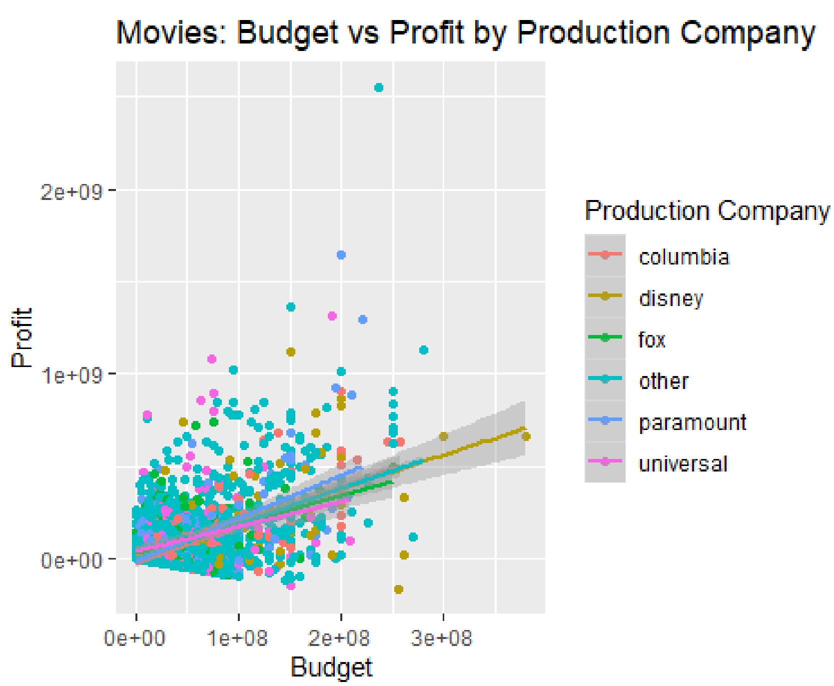

```r
ggplot(moviesdf, aes(x = budget, y = profit, color = season)) + geom_point(po
sition = "jitter") +geom_smooth(method = lm) + labs(x = "Budget", y = "Profit
", title = "Movies: Budget vs Profit by Season", color = "Season")
```



```r
ggplot(moviesdf, aes(x = budget, y = profit, color = genre)) + geom_point(position = "jitter") +geom_smooth(method = lm) + labs(x = "Budget", y = "Profit", title = "Movies: Budget vs Profit by Genre", color = "Genre")
```




```r
ggplot(moviesdf, aes(x = budget, y = profit, color = company)) + geom_point(position = "jitter") +geom_smooth(method = lm) + labs(x = "Budget", y = "Profit", title = "Movies: Budget vs Profit by Production Company", color = "Production Company")
```



Correlation Table

```r
cor(moviesdf[, c("revenue", "budget", "runtime", "vote_average", "profit")],
method = "kendall")
```

```
##                revenue      budget   runtime vote_average    profit
## revenue      1.0000000 0.569396790 0.2005507  0.149019299 0.6783035
## budget       0.5693968 1.000000000 0.2063774  0.006237209 0.2166101
## runtime      0.2005507 0.206377440 1.0000000  0.278303944 0.1242596
## vote_average 0.1490193 0.006237209 0.2783039  1.000000000 0.1873257
## profit       0.6783035 0.216610061 0.1242596  0.187325735 1.0000000
```

Linear Models

Simple Linear Regression

With budget predicting profit, budget accounts for 32.3% of variation in profit. When the budget increases by $1, the profit increases by $1.92.

```r
profitlm1 <- lm(profit ~ budget, data = moviesdf)
summary(profitlm1)
```

```
##
## Call:
## lm(formula = profit ~ budget, data = moviesdf)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -653730922  -40284894     -57418   11585527 2097583542
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.702e+06  2.169e+06  -1.246    0.213
## budget       1.924e+00  4.174e-02  46.104   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 115500000 on 4445 degrees of freedom
## Multiple R-squared:  0.3235, Adjusted R-squared:  0.3233
## F-statistic:  2126 on 1 and 4445 DF,  p-value: < 2.2e-16
```

When run time predicts profit, it accounts for 4.9% of variation in profit.

```r
profitlm2 <- lm(profit ~ runtime, data = moviesdf)
summary(profitlm2)
```

```
##
## Call:
## lm(formula = profit ~ runtime, data = moviesdf)
##
## Residuals:
```

```
##        Min         1Q     Median         3Q        Max 
## -417129109  -60222573  -34760139    8269447 2413249328 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -103705320   10789174  -9.612   <2e-16 ***
## runtime        1490254      97917  15.220   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 136900000 on 4445 degrees of freedom
## Multiple R-squared:  0.04953,    Adjusted R-squared:  0.04932 
## F-statistic: 231.6 on 1 and 4445 DF,  p-value: < 2.2e-16
```

When vote average predicts profit, it accounts for 4.8% of variation in profit..

```
profitlm3 <- lm(profit ~ vote_average, data = moviesdf)
summary(profitlm3)
```

```
## 
## Call:
## lm(formula = profit ~ vote_average, data = moviesdf)
## 
## Residuals:
##        Min         1Q     Median         3Q        Max 
## -214932685  -67177261  -38730151   10938197 2462840715 
## 
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  -127331623   12511312  -10.18   <2e-16 ***
## vote_average   29924444    1998076   14.98   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 1.37e+08 on 4445 degrees of freedom
## Multiple R-squared:  0.04804,    Adjusted R-squared:  0.04782 
## F-statistic: 224.3 on 1 and 4445 DF,  p-value: < 2.2e-16
```

When season predicts profit, it accounts for 3.3% of variation in profit.

```
profitlm4 <- lm(profit ~ season, data = moviesdf)
summary(profitlm4)
```

```
## 
## Call:
## lm(formula = profit ~ season, data = moviesdf)
## 
## Residuals:
##        Min         1Q     Median         3Q        Max 
## -264132873  -64465101  -32940604    9888099 2486499986 
## 
```

```
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  30240604    3724500   8.119 6.01e-16 ***
## seasonspring 14203002    5795027   2.451   0.0143 *
## seasonsummer 68182179    5669753  12.026  < 2e-16 ***
## seasonwinter 34224497    5639351   6.069 1.40e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 138100000 on 4443 degrees of freedom
## Multiple R-squared:  0.03404,    Adjusted R-squared:  0.03339
## F-statistic: 52.19 on 3 and 4443 DF,  p-value: < 2.2e-16
```

When genre predicts profit, it accounts for 8.2% of variation in profit.

```
profitlm5 <- lm(profit ~ genres.y, data = moviesdf)
summary(profitlm5)

##
## Call:
## lm(formula = profit ~ genres.y, data = moviesdf)
##
## Residuals:
##        Min          1Q     Median          3Q        Max
## -258512154  -49410367  -30280402   10154483 2473962899
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)              77002188    4994579  15.417  < 2e-16 ***
## genres.yAdventure        70502724    8901914   7.920 2.98e-15 ***
## genres.yAnimation       100176058   13222213   7.576 4.30e-14 ***
## genres.yComedy          -41153393    6623936  -6.213 5.68e-10 ***
## genres.yCrime           -49292633   11018705  -4.474 7.89e-06 ***
## genres.yDocumentary     -66165662   19160564  -3.453 0.000559 ***
## genres.yDrama           -46721786    6443847  -7.251 4.87e-13 ***
## genres.yFamily           48700601   19331519   2.519 0.011796 *
## genres.yFantasy          19670781   13514689   1.456 0.145599
## genres.yForeign         -77115888  134761119  -0.572 0.567188
## genres.yHistory         -26557947   27939155  -0.951 0.341878
## genres.yHorror          -41657743    9484093  -4.392 1.15e-05 ***
## genres.yMusic           -60812225   23629351  -2.574 0.010097 *
## genres.yMystery         -28914757   21870898  -1.322 0.186215
## genres.yRomance         -29087092   14363215  -2.025 0.042916 *
## genres.yScience Fiction  43502172   14691733   2.961 0.003083 **
## genres.yThriller        -38429476   11211567  -3.428 0.000614 ***
## genres.yTV Movie        -77335521   77911169  -0.993 0.320954
## genres.yWar             -46276759   29142613  -1.588 0.112371
## genres.yWestern         -50614218   26393848  -1.918 0.055219 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 134700000 on 4427 degrees of freedom
## Multiple R-squared:  0.0842, Adjusted R-squared:  0.08027
## F-statistic: 21.42 on 19 and 4427 DF,  p-value: < 2.2e-16
```

Predicting profit based production companies resulted in a negative adjusted R-squared.

```
profitlm6 <- lm(profit ~ production_companies.y, data = moviesdf)
summary(profitlm6)

##
## Call:
## lm(formula = profit ~ production_companies.y, data = moviesdf)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -540597108  -46827861    -163933    3071558 2315676020
##
## Coefficients:
##
Estimate
## (Intercept)
16982922
## production_companies.y101st Street Films
-16982922
## production_companies.y1492 Pictures
576153913
## production_companies.y1818
-9750294
## production_companies.y19 Entertainment
-24060756
## production_companies.y21 Laps Entertainment
-12628311
## production_companies.yXYZ Films
## production_companies.yYari Film Group
## production_companies.yYash Raj Films
## production_companies.yYeah
## production_companies.yYoung Medium
## production_companies.yYounggu-Art Movies
## production_companies.yYouth House Productions
## production_companies.yZentropa Entertainments
## production_companies.yZephyr Films
## production_companies.yZininsa Film Production
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.43e+08 on 3139 degrees of freedom
## Multiple R-squared:  0.2682, Adjusted R-squared:  -0.03654
## F-statistic: 0.8801 on 1307 and 3139 DF,  p-value: 0.9967
```

Multiple Linear Regression Model

```
multi_lm1 <- lm(profit ~ budget + runtime + vote_average + summer, data = mov
iesdf)
summary(multi_lm1)
```

```
##
## Call:
## lm(formula = profit ~ budget + runtime + vote_average + summer,
##     data = moviesdf)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -648988354  -44327992   -8807124   24565633 2089659497
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.704e+08  1.164e+07 -14.645  < 2e-16 ***
## budget        1.841e+00  4.261e-02  43.203  < 2e-16 ***
## runtime       8.735e+04  8.887e+04   0.983    0.326
## vote_average  2.519e+07  1.751e+06  14.387  < 2e-16 ***
## summerTRUE    2.272e+07  4.024e+06   5.646 1.75e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 112100000 on 4442 degrees of freedom
## Multiple R-squared:  0.3638, Adjusted R-squared:  0.3632
## F-statistic: 635.1 on 4 and 4442 DF,  p-value: < 2.2e-16
```

```
multi_lm2 <- lm(profit ~ budget + runtime + vote_average + summer + action, d
ata = moviesdf)
summary(multi_lm2)
```

```
##
## Call:
## lm(formula = profit ~ budget + runtime + vote_average + summer +
##     action, data = moviesdf)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -644003294  -44166815   -8959809   24627277 2095689791
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.667e+08  1.172e+07 -14.219  < 2e-16 ***
## budget        1.861e+00  4.331e-02  42.960  < 2e-16 ***
## runtime       9.073e+04  8.883e+04   1.021   0.3071
## vote_average  2.473e+07  1.759e+06  14.059  < 2e-16 ***
## summerTRUE    2.285e+07  4.021e+06   5.682 1.42e-08 ***
## actionTRUE   -1.177e+07  4.654e+06  -2.529   0.0115 *
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.12e+08 on 4441 degrees of freedom
## Multiple R-squared:  0.3647, Adjusted R-squared:  0.364
## F-statistic:   510 on 5 and 4441 DF,  p-value: < 2.2e-16
```

Model Fit

Comparing the fit of both models, adding the strongest regression genre action to the model did not significantly improve the fit from the first model. The first model will be used.

```
anova(multi_lm1, multi_lm2)

## Analysis of Variance Table
##
## Model 1: profit ~ budget + runtime + vote_average + summer
## Model 2: profit ~ budget + runtime + vote_average + summer + action
##   Res.Df        RSS Df  Sum of Sq      F  Pr(>F)
## 1   4442 5.5772e+19
## 2   4441 5.5692e+19  1 8.0198e+16 6.3951 0.01148 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Multicollinearity Test

There seems to be no collinearity within the data.

```
vif(multi_lm1)

##       budget      runtime vote_average       summer
##     1.107185     1.229956     1.147844     1.029423

1/vif(multi_lm1)

##       budget      runtime vote_average       summer
##    0.9031912    0.8130373    0.8711985    0.9714180

mean(vif(multi_lm1))

## [1] 1.128602
```

Machine Learning

```
set.seed(42)
rows <- sample(nrow(moviesdf))
shuffled_moviesdf <- moviesdf[rows, ]

split <- sample.split(shuffled_moviesdf, SplitRatio = .8)
train_data <- subset(shuffled_moviesdf, split == "TRUE")
test_data <- subset(shuffled_moviesdf, split == "FALSE")
```

```r
train_model <- lm(profit ~ budget + runtime + vote_average + summer, data = t
rain_data)

predict_model <- predict(train_model, test_data)


p <- predict(multi_lm1, moviesdf)
error_p <- p - moviesdf[["profit"]]
sqrt(mean(error_p^2))

## [1] 111989137

error_test <- predict_model - test_data[["profit"]]
sqrt(mean(error_test^2))

## [1] 100866891
```

Source

The Movie Database(TMDb) (2017). TMDB 5000 Movie Dataset. Kaggle.
https://www.kaggle.com/tmdb/tmdb-movie-metadata#tmdb_5000_movies.csv