# Final Project: Analyzing Your Scraped Data

MATH/CS 215: Intro to Data Science

Sarah Onstad-Hawes

**Load Packages**

```
library(tidyverse)
library(tidymodels)
library(openintro)
library(robotstxt)
library(rvest)
library(readr)
```

Research Question: Looking at the Top_Movies dataset which contains rankings of the top 220 movies, what is the best model for predicting the rank of a movie?

**Link to Data** https://www.imdb.com/list/ls064721857/

This scraped data is from IMDB and it contains a random persons personal rankings of their top 220 movies.

I'm using linear models to predict the rank of movies, looking at simple linear regression and multiple linear regression models to find the best model to predict the rank of a movie based on the length (in minutes) of the movie, its starscore (scale of 1-10), its genre, and the rating of the movie.

**Dataset**

```
Top_Movies <- read_csv("~/Math 215 - Fall 2021/Final Project/Top_Movies.csv")
glimpse(Top_Movies)
```

```
## Rows: 220
## Columns: 7
## $ ...1      <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 1~
## $ Title     <chr> "12 Angry Men", "The Green Mile", "The Lord of the Rings: Th~
## $ Rank      <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 1~
## $ Length    <dbl> 96, 189, 201, 142, 113, 127, 152, 93, 148, 165, 119, 138, 15~
## $ Genre     <chr> "Crime", "Crime", "Action", "Drama", "Mystery", "Crime", "Ac~
## $ Rating    <chr> "Approved", "R", "PG-13", "R", "R", "R", "PG-13", "R", "PG-1~
## $ Starscore <dbl> 9.0, 8.6, 8.9, 9.3, 8.4, 8.6, 9.0, 8.1, 8.8, 8.5, 8.5, 8.2, ~
```

```
view(Top_Movies)
```

**Variables in Dataset**

```
summary(Top_Movies)
```

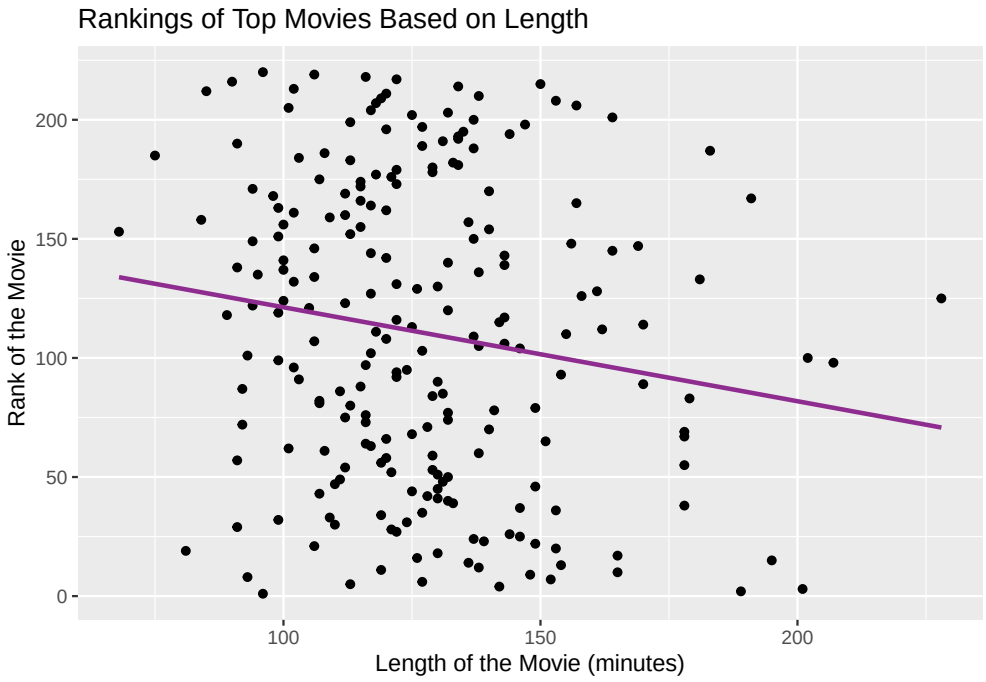
```
##       ...1           Title                Rank           Length
##  Min.   :  1.00   Length:220         Min.   :  1.00   Min.   : 68.0
##  1st Qu.: 55.75   Class :character   1st Qu.: 55.75   1st Qu.:110.8
##  Median :110.50   Mode  :character   Median :110.50   Median :124.5
##  Mean   :110.50                      Mean   :110.50   Mean   :127.3
##  3rd Qu.:165.25                      3rd Qu.:165.25   3rd Qu.:140.0
```

```
##  Max.   :220.00                      Max.   :220.00   Max.   :228.0
##     Genre              Rating            Starscore
##  Length:220         Length:220         Min.   :6.70
##  Class :character   Class :character   1st Qu.:7.70
##  Mode  :character   Mode  :character   Median :8.10
##                                        Mean   :8.05
##                                        3rd Qu.:8.30
##                                        Max.   :9.30
```

- Title of Movie
- Rank of Movie (scale from 1 to 220)
- Length of Movie (minutes): in dataset ranging from 68 to 228 minutes
- Genre (11 categories): Action, Adventure, Animation, Biography, Comedy, Crime, Drama, Family, Horror, Mystery, and Western
- Rating (7 categories): Approved, G, Not Rated, Passed, PG, PG-13, and R
- Starscore (scale of 1-10): Critic score of movies ranging from 6.7 to 9.3 in the dataset

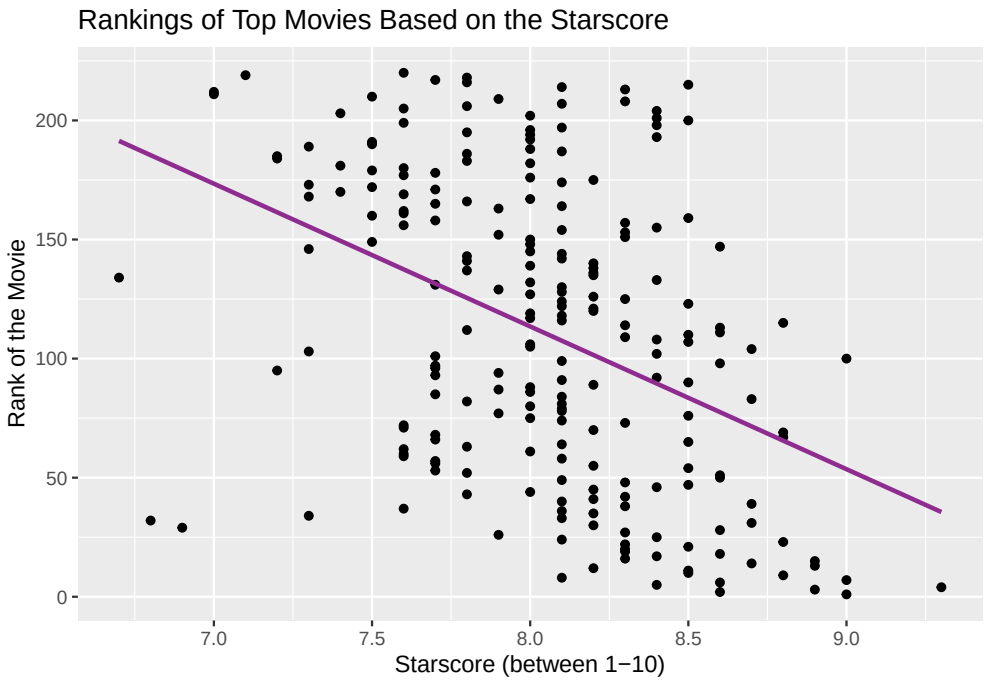
### Visualization of Rank by Different Variables

```
#visualizing length of movie and rank with a scatterplot
ggplot(data = Top_Movies, aes(x = Length, y = Rank)) +
  geom_point() +
  geom_smooth(method = "lm", color = "#8E2C90", se = FALSE) +
  labs(
    title = "Rankings of Top Movies Based on Length",
    x = "Length of the Movie (minutes)",
    y = "Rank of the Movie"
  )
```



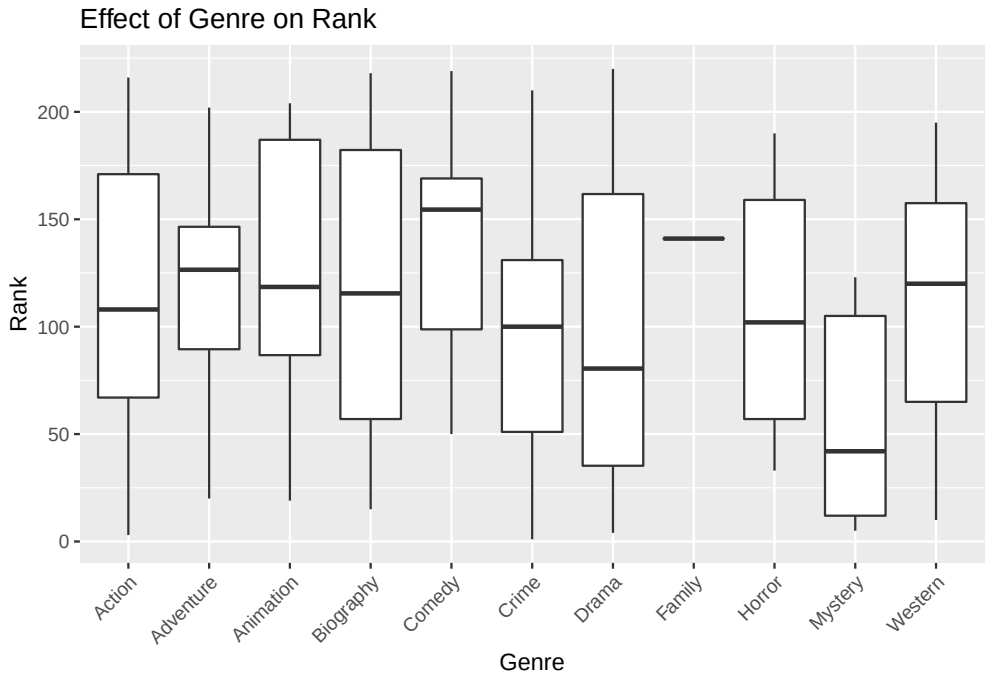

Based on the scatterplot above, there appears to potentially be a slight linear relationship between length and rank. The line suggests that as the length of the movie increases, the rank of the movie decreases (is better in rank).

```r
#visualizing starscore and rank with a scatterplot
ggplot(data = Top_Movies, aes(x = Starscore, y = Rank)) +
  geom_point() +
  geom_smooth(method = "lm", color = "#8E2C90", se = FALSE) +
  labs(
    title = "Rankings of Top Movies Based on the Starscore",
    x = "Starscore (between 1-10)",
    y = "Rank of the Movie"
  )
```



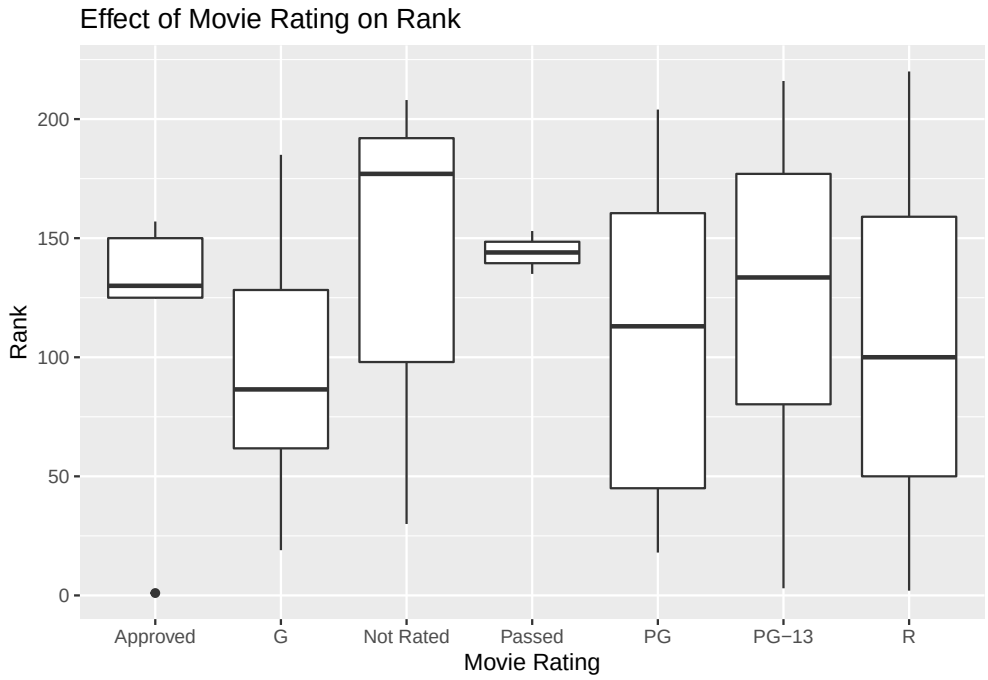

Based on the scatterplot above, there appears to be a negative linear relationship between starscore and rank. The line suggests that as the starscore of the movie increases, the rank of the movie decreases (is better in rank).

```r
#Visualizing genres effect on rank through boxplots
ggplot(data = Top_Movies, aes(x = Genre, y = Rank)) +
  geom_boxplot() +
  labs(
    title = "Effect of Genre on Rank",
    x = "Genre",
    y = "Rank") +
    theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1))
```




Looking at the boxplots above, it is hard to tell if there is a substantial difference between the medians of the different categories of genre. The `Mystery` category appears to have a lower median compared to the other genre categories and could potentially be substantially different from the others. The ranges, other than the `Family` genre appear fairly spread out.

```
#Visualizing effect of movie rating on rank through boxplots
ggplot(data = Top_Movies, aes(x = Rating, y = Rank)) +
  geom_boxplot() +
  labs(
    title = "Effect of Movie Rating on Rank",
    x = "Movie Rating",
    y = "Rank"
  )
```




The ranges of the different movie ratings appears fairly similar, however it is much more condensed for the rating `Passed` and `Approved`. The medians also appear to be pretty similar typically ranging between 100 and 150. The `G` rating has a lower median which could be a substantial difference and the `Not Rated` median is higher than that range which could also be relevant.

### Simple Linear Regression

The next step I took was exploring the different simple linear regression models from the four different variables to predict the rank of a movie.

```
#created a SLR model with length as a predictor of rank
model_length <- linear_reg() %>%
  set_engine("lm") %>%
  fit(Rank ~ Length, data = Top_Movies)
tidy(model_length)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)  161.       21.4        7.51 1.49e-12
## 2 Length        -0.395     0.165     -2.40 1.75e- 2
```

```
glance(model_length)$adj.r.squared
```

```
## [1] 0.02117216
```

**Model:**

$$\widehat{rank} = 160.74 - 0.39 \times length$$

**Slope:**

For each additional minute the movie is in length, the rank is expected to be lower, on average, by 0.39 points.

**Intercept:**

Movies that are zero minutes long are expected to have a rank on average of 160.74. This intercept doesn't make sense because no movie would be zero minutes long.

The p-value is significant for length as a predictor of rank at the alpha 0.05 level with the p-value of 1.75e-02. However, length only accounts for approximately 2% of the variation of rank with an R^2 score of 0.02, signifying that this model is not very good at predicting rank.

```
#created a SLR model with starscore as a predictor of rank
model_starscore <- linear_reg() %>%
  set_engine("lm") %>%
  fit(Rank ~ Starscore, data = Top_Movies)
tidy(model_starscore)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)    593.      71.1       8.35 8.14e-15
## 2 Starscore      -59.9      8.81     -6.80 9.87e-11
```

```
glance(model_starscore)$adj.r.squared
```

```
## [1] 0.1712292
```

**Model:**

$$\widehat{rank} = 592.98 - 59.94 \times starscore$$

**Slope:**

For each additional increase in one point of starscore of the movie, the rank is expected to be lower, on average, by 59.94 points.

**Intercept:**

Movies that have a zero starscore are expected to have a rank on average of 529.98. This intercept doesn't make sense because there isn't a possible rating of zero. The intercept for this model is nearly 600 because in the dataset starscore values range from 6.7 to 9.3, however the model takes into account potential values for the starscore of 1 to 10.

The p-value is significant for starscore as a predictor of rank at the alpha 0.05 level with the p-value of 9.87e-11. This model only accounts for approximately 17% of the variation of rank with an R^2 score of 0.17, signifying that this model is not very good at predicting rank, however it is higher than the simple linear model with length as the predictor.

```
#created a SLR model with genre as a predictor of rank
model_genre <- linear_reg() %>%
  set_engine("lm") %>%
  fit(Rank ~ Genre, data = Top_Movies)
tidy(model_genre)
```

```
## # A tibble: 11 x 5
##    term           estimate std.error statistic  p.value
##    <chr>             <dbl>     <dbl>     <dbl>    <dbl>
##  1 (Intercept)      114.        7.54    15.1   2.73e-35
##  2 GenreAdventure     5.93     16.1      0.368 7.13e- 1
##  3 GenreAnimation    12.6      23.7      0.532 5.95e- 1
##  4 GenreBiography     7.28     16.8      0.434 6.65e- 1
##  5 GenreComedy       25.2      18.6      1.36  1.76e- 1
```

```
##  6 GenreCrime       -18.7      14.8     -1.27  2.07e- 1
##  7 GenreDrama       -13.2      11.7     -1.12  2.63e- 1
##  8 GenreFamily       27.2      64.0      0.426 6.71e- 1
##  9 GenreHorror       -5.57     29.4     -0.190 8.50e- 1
## 10 GenreMystery     -56.4      29.4     -1.92  5.65e- 2
## 11 GenreWestern      -5.44     37.4     -0.145 8.85e- 1
```

```
glance(model_genre)$adj.r.squared
```

```
## [1] 0.003808029
```

**Relationship between genre and rank**

- **Action Genre** movies are expected, on average, to have a ranking of 113.77.
- **Adventure Genre** movies are expected, on average, to be ranked 5.93 points higher than *Action* movies.
- **Animation Genre** movies are expected, on average, to be ranked 12.6 points higher than *Action* movies.
- **Biography Genre** movies are expected, on average, to be ranked 7.28 points higher than *Action* movies.
- **Comedy Genre** movies are expected, on average, to be ranked 25.23 points higher than *Action* movies.
- **Crime Genre** movies are expected, on average, to be ranked 18.7 points lower than *Action* movies.
- **Drama Genre** movies are expected, on average, to be ranked 13.2 points lower than *Action* movies.
- **Family Genre** movies are expected, on average, to be ranked 27.23 points higher than *Action* movies.
- **Horror Genre** movies are expected, on average, to be ranked 5.6 points lower than *Action* movies.
- **Mystery Genre** movies are expected, on average, to be ranked 56.4 points lower than *Action* movies.
- **Western Genre** movies are expected, on average, to be ranked 5.4 points lower than *Action* movies.

All the p-values for the different genres are not statistically significant at the alpha 0.5 level. This model only accounts for approximately 0.4% of the variation of rank with an Rˆ2 score of 0.0038, signifying that this model is very bad at predicting rank.

```
#created a SLR model with movie rating as a predictor of rank
model_rating <- linear_reg() %>%
  set_engine("lm") %>%
  fit(Rank ~ Rating, data = Top_Movies)
tidy(model_rating)
```

```
## # A tibble: 7 x 5
##   term            estimate std.error statistic  p.value
##   <chr>              <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)       113.        28.4     3.96  0.000101
## 2 RatingG           -20.2       36.2    -0.558 0.577
## 3 RatingNot Rated    28.4       40.2     0.707 0.480
## 4 RatingPassed       31.4       53.2     0.591 0.555
## 5 RatingPG           -7.71      30.9    -0.249 0.803
## 6 RatingPG-13        13.5       29.7     0.454 0.650
## 7 RatingR            -8.25      29.0    -0.285 0.776
```

```
glance(model_rating)$adj.r.squared
```

```
## [1] 0.003841597
```

**Relationship between movie rating and rank**

- **Approved Ranking** movies are expected, on average, to have a rank of 112.6.
- **G Rating** movies are expected, on average, to be ranked 20.23 points lower than *Approved* movies.

- **Not Rated Rating** movies are expected, on average, to be ranked 28.4 points higher than *Approved* movies.
- **Passed Rating** movies are expected, on average, to be ranked 31.4 points higher than *Approved* movies.
- **PG Rating** movies are expected, on average, to be ranked 7.71 points lower than *Approved* movies.
- **PG-13 Rating** movies are expected, on average, to be ranked 13.5 points higher than *Approved* movies.
- **R Rating** movies are expected, on average, to be ranked 8.25 points lower than *Approved* movies.

The p-values of all of the different rating variables are not significant at the alpha 0.05 level, indicating that there is not a statistically significant difference of the different ratings from the baseline rating of approved on predicting the rank of the movie. This model only accounts for approximately 0.4% of the variation of rank with an R^2 score of 0.0038, signifying that this model is very bad at predicting rank.

### Multiple Linear Regression Models

The next step is looking at the significant predictors from the simple linear regression models `Length` and `Starscore` in a combined model.

```
#MLR model with length and starscore as predictors of rank
model_length_starscore <- linear_reg() %>%
  set_engine("lm") %>%
  fit(Rank ~  Length + Starscore, data = Top_Movies)
tidy(model_length_starscore)
```

```
## # A tibble: 3 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept) 596.        72.2       8.26  1.39e-14
## 2 Length        0.0487     0.167     0.291 7.71e- 1
## 3 Starscore   -61.1        9.74     -6.28  1.86e- 9
```

```
glance(model_length_starscore)$adj.r.squared
```

```
## [1] 0.1677345
```

**Model:**

$$\widehat{rank} = 596.39 + 0.05 \times length - 61.13 \times starscore$$

**Slope of length:**

All else held constant, for each additional one minute increase in length of the movie, we would expect on average the rank to be higher by 0.049.

**Slope of starscore:**

All else held constant, for each additional one point increase in starscore of the movie, we would expect on average the rank to be lower by 61.13.

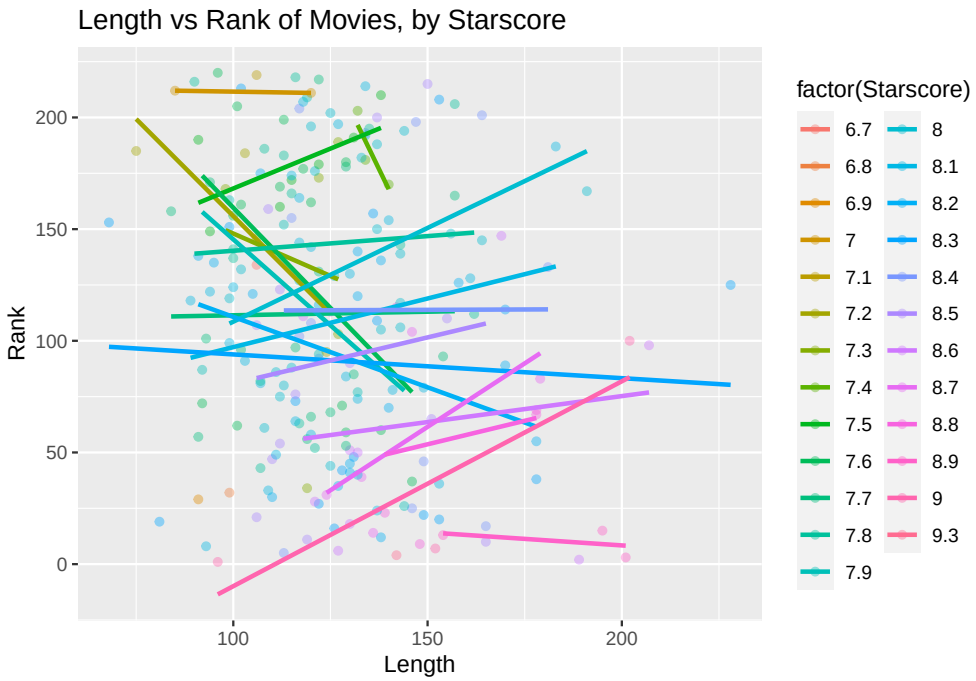
**Intercept:**

Movies with a length and starscore of 0 are expected to have a rank of 596.4, on average. This intercept does not make sense because there wouldn't be a movie with zero minutes.

The p-values of length is not statistically significant at the alpha 0.05 level with a value of 7.7e-01, however the starscore is statistically significant at the alpha 0.05 level, with a p-value of 1.86e-09. The adjusted R^2 value is 0.17, meaning that this model accounts for 17% of the variability of rank, suggesting it is not a very good model.

I then included a visualization of length versus rank, factored by starscore.

```
#Visualization of length vs rank by the starscore
ggplot(data = Top_Movies, aes(x = Length, y = Rank, color=factor(Starscore))) +
  geom_point(alpha = 0.4) +
  geom_smooth(method = "lm", se = FALSE) +
  labs(
    title = "Length vs Rank of Movies, by Starscore",
    x = "Length",
    y = "Rank"
  )
```




```
#MLR model with length and starscore as predictors with interaction term to predict rank of the movie
model_length_starscore_int <- linear_reg() %>%
  set_engine("lm") %>%
  fit(Rank ~ Length * Starscore, data = Top_Movies)
tidy(model_length_starscore_int)
```

```
## # A tibble: 4 x 5
##   term             estimate std.error statistic p.value
##   <chr>               <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)       200.      343.        0.584   0.560
## 2 Length              3.28      2.74      1.19    0.233
## 3 Starscore         -12.9      42.1      -0.306   0.760
## 4 Length:Starscore   -0.392     0.332    -1.18    0.240
```

```
glance(model_length_starscore_int)$adj.r.squared
```

```
## [1] 0.1692308
```

**Model:**

$$\widehat{rank} = 200.49 + 3.28 \times length - 12.9 \times starscore - 0.39 \times length \times starscore$$

None of the p-values of the predictors are statistically significant at the alpha 0.05 level for predicting the rank of the movie, suggesting this isn't a good model. The adjusted R^2 value is 0.17, showing that the interaction model accounts for approximately 17% of the variability of the rank.

## Conclusion

### Best Model

Overall, none of the models are very good at predicting the rank of a movie. None of the models are able to account for more than 17% of the variability of the rank, showing that all of these models aren't very good. However, the best model to answer my research question of predicting the rank of a movie is the simple linear regression model with starscore as a predictor of rank because it is the simplest model with a statistically significant predictor and accounts for the most variability of rank, accounting for approximately 17%.

**Best Model:**

$$\widehat{rank} = 592.98 - 59.94 \times starscore$$

.