IMDB Movie Predictions

Philip Yoon

03/23/2019

```
library(ggplot2)
```

```
## Registered S3 methods overwritten by 'ggplot2':
##   method         from
##   [.quosures     rlang
##   c.quosures     rlang
##   print.quosures rlang
```

```
library(cowplot)
```

```
##
## ************************************************************
## Note: As of version 1.0.0, cowplot does not change the
##   default ggplot2 theme anymore. To recover the previous
##   behavior, execute:
##   theme_set(theme_cowplot())
## ************************************************************
```

```
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
```

```
library(statsr)
```

```
## Loading required package: BayesFactor
## Loading required package: coda
## Loading required package: Matrix
## ************
## Welcome to BayesFactor 0.9.12-4.2. If you have questions, please contact Richard Morey (richarddmorey
##
## Type BFManual() to open the manual.
## ************
```

```
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(knitr)
```

Dataset

```
load(url("https://stat.duke.edu/~mc301/data/movies.Rdata"))
```

The dataset contains information about movies in Rotten Tomatoes and IMDB. There are 651 randomly sampled movies produced and released before 2016. There are 32 available variables. With this dataset and for the purpose of this project it is only possible to do an observational study and no causal analysis is done.The study can be generalized to movies produced and released before 2016.

We considered that some of the variables are irrelevant to the purpose of identifying the popularity of a movie: the Link to IMDB page for the movie and the Link to Rotten Tomatoes page for the movie.

Research Question

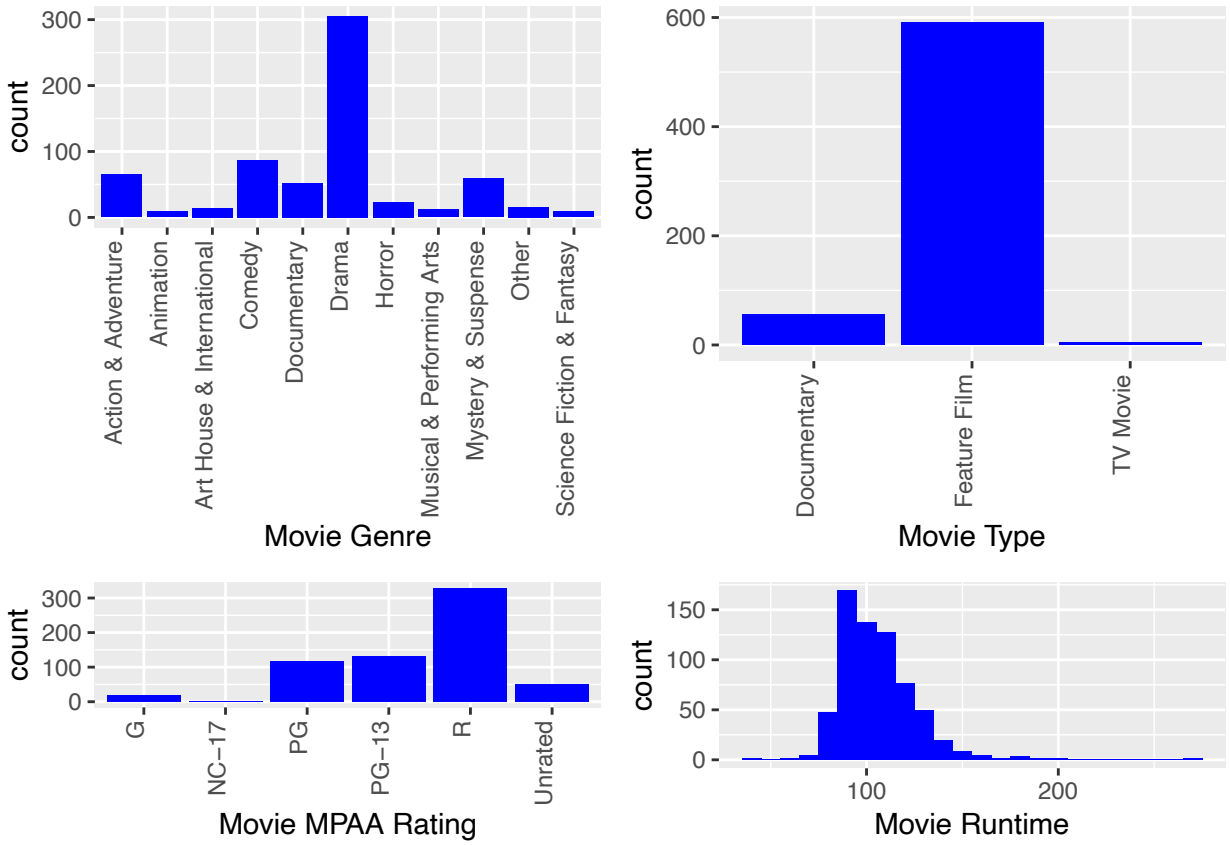
Is it possible to predict the popularity of a movie prior to its release based on certain characteristics of the movie, to be specific are variables such as movie genre, MPAA rating, run length, etc. good predictors of a popular movie?

Exploratory Data Analysis

There are in total 651 movies in the dataset.The following charts show a breakdown of the type of movies included in the sample.

```
# Create histograms of some of the key movie characteristic data.
p1 <- ggplot(data=movies, aes(x=genre)) +
      geom_bar(fill="blue") +
      xlab("Movie Genre") +
      theme(axis.text.x=element_text(angle=90, hjust=1, vjust=0))
p2 <- ggplot(data=movies, aes(x=title_type)) +
      geom_bar(fill="blue") +
      xlab("Movie Type") +
      theme(axis.text.x=element_text(angle=90, hjust=1, vjust=0))
p3 <- ggplot(data=movies, aes(x=mpaa_rating)) +
      geom_bar(fill="blue") +
      xlab("Movie MPAA Rating") +
      theme(axis.text.x=element_text(angle=90, hjust=1, vjust=0))
p4 <- ggplot(data=movies, aes(x=runtime)) +
      geom_histogram(binwidth=10, fill="blue") +
      xlab("Movie Runtime")
plot_grid(p1, p2, p3, p4, align = "v", nrow = 2, rel_heights = c(2, 1, 1, 1))
```

```
## Warning: Removed 1 rows containing non-finite values (stat_bin).
```




```
#grid.arrange(p2, p3, p1, p4, nrow=2, top="Movie Characteristics")
```

There are 60 movies in the raw data that are type "Documentary" or "TV Movie". I will remove these as they will not likely be shown in a movie theater. Also, there are 52 movies with MPAA ratings of NC-17 or are unrated. These, as well, would not likely be shown in a typical movie theater and will be excluded from the analysis.

```
movies <- movies %>% filter(title_type=="Feature Film") %>%
          filter(!(mpaa_rating %in% c("NC-17", "Unrated")))
```

Looking at the summary statistics for different movie ratings:

```
summary(movies$audience_score)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   11.00   44.00   62.00   60.17   77.00   97.00
```

```
summary(movies$critics_score)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   31.00   56.00   54.12   79.00  100.00
```

```
summary(movies$imdb_rating)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.900   5.800   6.500   6.371   7.100   9.000
```

The median critics score was 56 and the median audience score was 62. The audience score ranged from 11 to 97 while the critic score ranged from 1 to 100.

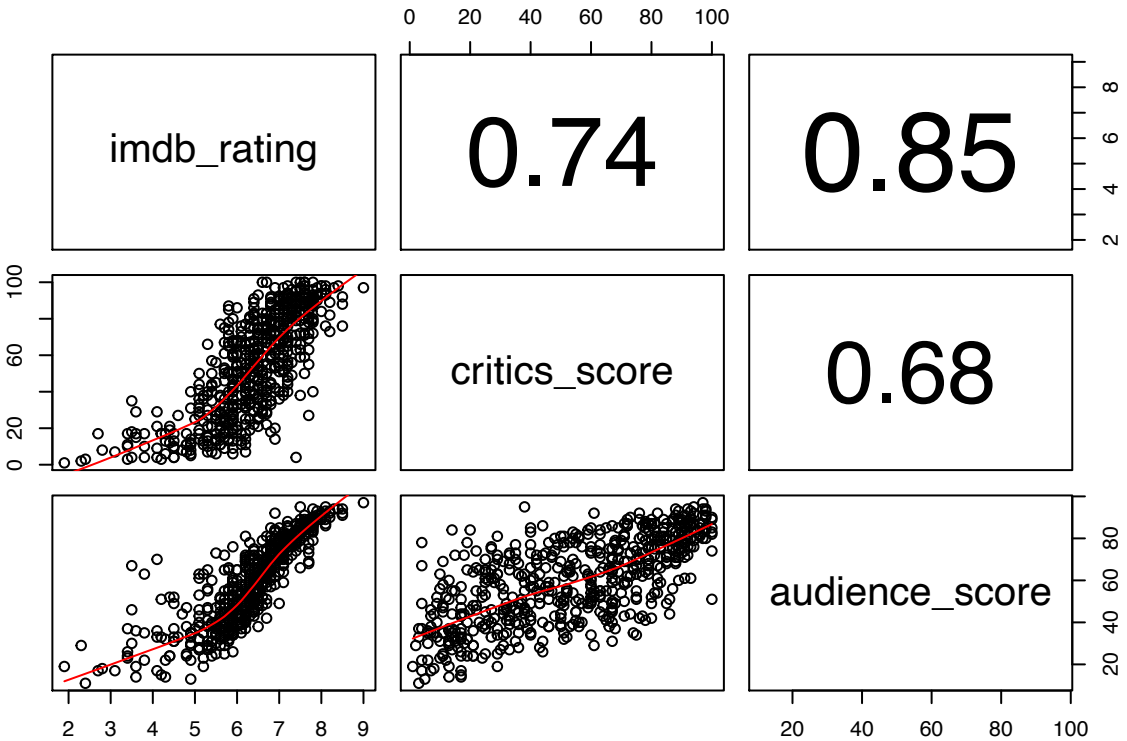
Model Development

The target response variable for the prediction model is a movie rating score, but with three to choose from, which one should be used? Two of the ratings come from the Rotten Tomatoes web site: one is an average of reviews by movie critics and the other is an average of reviews from the public (a.k.a., audience). The third rating is an average of reviews on the IMDB web site (no distinction made between critics and audience reviews).

One would expect to see a correlation between the different rating scores. The following plots show that to be the case.

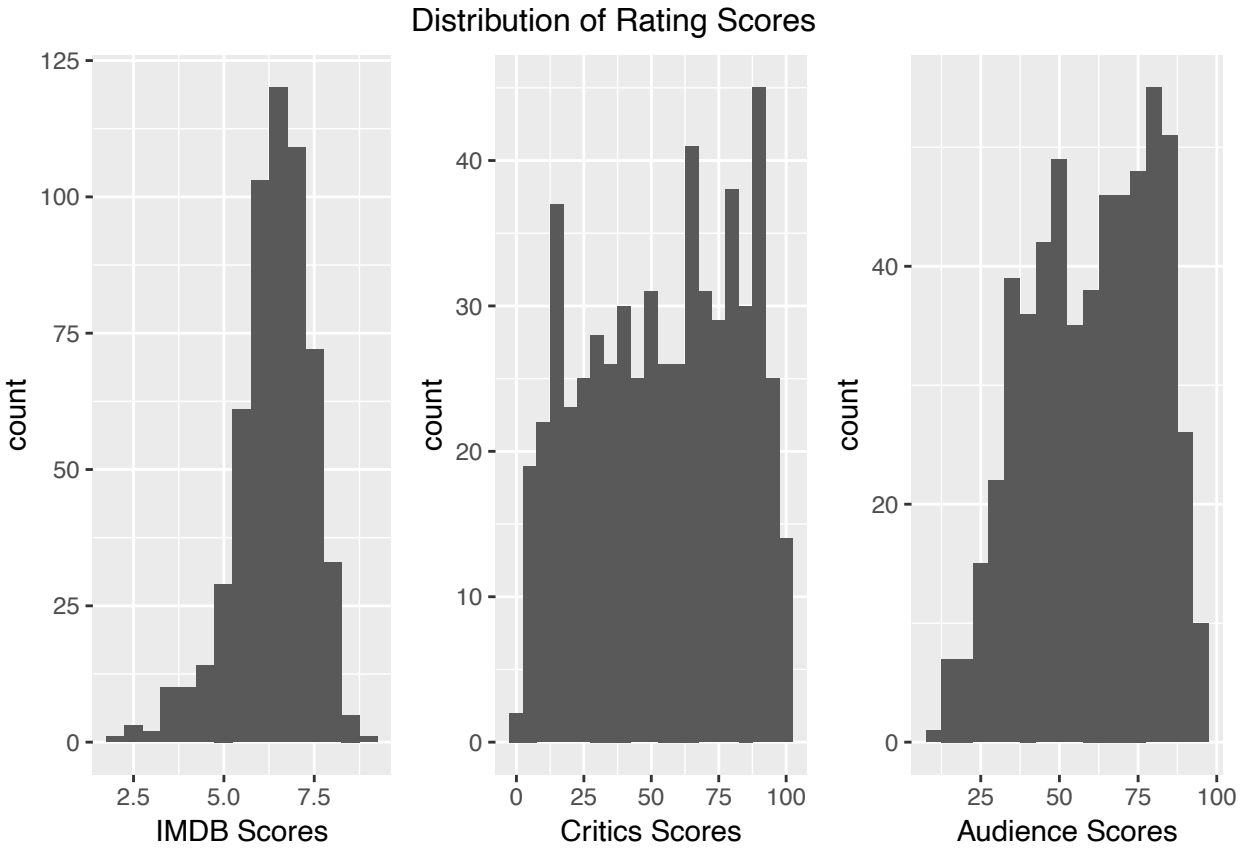
```
# Helper function for adding correlation coeficient values to a pairwise plot
# (taken from pairs() help page).
panel.cor <- function(x, y, digits = 2, prefix = "", cex.cor, ...)
{
    usr <- par("usr"); on.exit(par(usr))
    par(usr = c(0, 1, 0, 1))
    r <- abs(cor(x, y))
    txt <- format(c(r, 0.123456789), digits = digits)[1]
    txt <- paste0(prefix, txt)
    if(missing(cex.cor)) cex.cor <- 0.8/strwidth(txt)
    text(0.5, 0.5, txt, cex = cex.cor * r)
}

# Create pairwise plots of the movie rating scores to test for collinearity.
# Using the helper function above, the linear correlation R value is included
# on the chart.
pairs(~ imdb_rating + critics_score + audience_score,
      data=movies, lower.panel=panel.smooth, upper.panel=panel.cor)
```



Due to the above correlations, only one of the ratings are to be selected as the response variable. I will look at their distributions to help make the correct decision.

```
p1 <- ggplot(data=movies, aes(x=imdb_rating)) +
      geom_histogram(binwidth=0.5) +
      xlab("IMDB Scores")
p2 <- ggplot(data=movies, aes(x=critics_score)) +
      geom_histogram(binwidth=5) +
      xlab("Critics Scores")
p3 <- ggplot(data=movies, aes(x=audience_score)) +
      geom_histogram(binwidth=5) +
      xlab("Audience Scores")
grid.arrange(p1, p2, p3, nrow=1,
             top="Distribution of Rating Scores")
```



Contrary to the two Rotten Tomatoe scores, the IMDB scores show a mostly normal distribution centered around a mean of 6.37 with somewhat of a left-side skew. Given its distribution and the fact that it has the highest pairwise correlation with the other scores, the IMDB rating will be the response variable.

Since the goal is to predict the popularity of a movie prior to its release, the prediction model uses only variables from the data set that could be known ahead of time. Thus, variables such as DVD release date, number of IMDB votes, best picture nomination/win, etc. are excluded in the model. Variables such as studio name, actor/director names, URLs, etc are excluded as well beause they are not useful in our goal.

I will be using a backward elimination method of stepwise regression. I will start with all the variables in other words the full model and remove variables to create a model with as few predictors as possible. The initial variables are:

1. genre
2. runtime
3. mpaa_rating
4. thtr_rel_month

5. best_actor_win
6. best_actress win
7. best_dir_win

Theater release month is included assuming that movies released at certain times of the year may be more popular than others. Release year is discarded as being irrelevant and release day is insignificant given release month is already included.

The initial model therefore is:

```
intial_model <- lm(imdb_rating ~ genre + runtime + mpaa_rating + thtr_rel_month +
              best_actor_win + best_actress_win + best_dir_win, data=movies)
summary(intial_model)
```

```
##
## Call:
## lm(formula = imdb_rating ~ genre + runtime + mpaa_rating + thtr_rel_month +
##     best_actor_win + best_actress_win + best_dir_win, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8670 -0.5307  0.0430  0.6227  1.9998
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     5.033861   0.376917  13.355  < 2e-16 ***
## genreAnimation                 -0.306365   0.373898  -0.819 0.412921
## genreArt House & International  0.386649   0.321918   1.201 0.230233
## genreComedy                    -0.071803   0.158142  -0.454 0.649977
## genreDocumentary                0.790707   0.675727   1.170 0.242441
## genreDrama                      0.593064   0.134232   4.418 1.20e-05 ***
## genreHorror                    -0.113862   0.241935  -0.471 0.638089
## genreMusical & Performing Arts  0.922245   0.353553   2.609 0.009339 **
## genreMystery & Suspense         0.384125   0.175588   2.188 0.029113 *
## genreOther                      0.736434   0.279093   2.639 0.008558 **
## genreScience Fiction & Fantasy -0.266850   0.333834  -0.799 0.424431
## runtime                         0.015921   0.002688   5.923 5.56e-09 ***
## mpaa_ratingPG                  -0.790634   0.284121  -2.783 0.005574 **
## mpaa_ratingPG-13               -1.057501   0.289001  -3.659 0.000277 ***
## mpaa_ratingR                   -0.715151   0.282431  -2.532 0.011613 *
## thtr_rel_month                  0.006145   0.011526   0.533 0.594129
## best_actor_winyes              -0.044683   0.114659  -0.390 0.696908
## best_actress_winyes             0.065389   0.124816   0.524 0.600569
## best_dir_winyes                 0.358104   0.155371   2.305 0.021545 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9365 on 554 degrees of freedom
## Multiple R-squared:  0.248,  Adjusted R-squared:  0.2236
## F-statistic: 10.15 on 18 and 554 DF,  p-value: < 2.2e-16
```

```
anova(intial_model)
```

```
## Analysis of Variance Table
##
## Response: imdb_rating
```

```
##                   Df Sum Sq Mean Sq F value    Pr(>F)
## genre             10  95.24   9.524 10.8585 < 2.2e-16 ***
## runtime            1  41.21  41.211 46.9851 1.913e-11 ***
## mpaa_rating        3  18.54   6.181  7.0468 0.0001178 ***
## thtr_rel_month     1   0.25   0.252  0.2869 0.5924429
## best_actor_win     1   0.10   0.096  0.1093 0.7410594
## best_actress_win   1   0.27   0.272  0.3101 0.5778658
## best_dir_win       1   4.66   4.659  5.3122 0.0215453 *
## Residuals        554 485.92   0.877
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The adjusted R-square value is 0.2589, so there is plenty of room for improvement in the model.

The next procedure was to one-by-one remove insignificant variables based on p values, eliminating the variable with the highest p value each time, until all remaining variables were significant. It seems like a possible case of overfitting to create a model with an inflated R-squared value due to incuding statistically insignificant predictors.

The result is a model using only the variables for genre, runtime, MPAA rating, and whether the director ever won an Oscar as predictors. The model results are summarized below.

```
final_model <- lm(imdb_rating ~ genre + runtime + mpaa_rating +
                  best_dir_win, data=movies)
summary(final_model)
```

```
##
## Call:
## lm(formula = imdb_rating ~ genre + runtime + mpaa_rating + best_dir_win,
##     data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8689 -0.5436  0.0404  0.6056  2.0432
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     5.035253   0.368299  13.672  < 2e-16 ***
## genreAnimation                 -0.290502   0.371977  -0.781 0.435153
## genreArt House & International  0.392689   0.320509   1.225 0.221017
## genreComedy                    -0.059992   0.156688  -0.383 0.701958
## genreDocumentary                0.779294   0.673510   1.157 0.247742
## genreDrama                      0.599714   0.132672   4.520 7.55e-06 ***
## genreHorror                    -0.107019   0.241248  -0.444 0.657499
## genreMusical & Performing Arts  0.924963   0.352691   2.623 0.008965 **
## genreMystery & Suspense         0.382923   0.173083   2.212 0.027347 *
## genreOther                      0.731349   0.277479   2.636 0.008631 **
## genreScience Fiction & Fantasy -0.263124   0.332956  -0.790 0.429709
## runtime                         0.016264   0.002488   6.537 1.42e-10 ***
## mpaa_ratingPG                  -0.790039   0.283350  -2.788 0.005481 **
## mpaa_ratingPG-13               -1.060774   0.288355  -3.679 0.000257 ***
## mpaa_ratingR                   -0.715358   0.281770  -2.539 0.011394 *
## best_dir_winyes                 0.359052   0.155015   2.316 0.020907 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.9346 on 557 degrees of freedom
## Multiple R-squared:  0.2471, Adjusted R-squared:  0.2268
## F-statistic: 12.19 on 15 and 557 DF,  p-value: < 2.2e-16
```

```
anova(final_model)
```

```
## Analysis of Variance Table
##
## Response: imdb_rating
##               Df Sum Sq Mean Sq F value    Pr(>F)
## genre         10  95.24   9.524 10.9040 < 2.2e-16 ***
## runtime        1  41.21  41.211 47.1820 1.735e-11 ***
## mpaa_rating    3  18.54   6.181  7.0763 0.0001129 ***
## best_dir_win   1   4.69   4.686  5.3650 0.0209069 *
## Residuals    557 486.51   0.873
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The adjusted R-squared value of 0.2884 is only slightly above our old model, however in this model all the predictors are significant.

The coefficients of the model tell us a number of things. First, the genre variable is a mix of 6/11 showing statistical significance. Of those "Musicals and Performing Arts" genre are rated almost a full point higher than the base genre, which is "Action & Adventure" movies (all other predictors are held constant).

The MPAA rating predictor is similiar, although this time all the rating categories are significant and all have a negative affect relative to the base "G" rating. There is more than a full point reduction for a PG-13 rated movie (all other predictors held constant).

Movie run time appears to have a positive effect on movie rating. There is probably an upper limit to this that was not tested in this analysis, because obviously a five hour movie would not be rated four points higher than a one our movie. The relationship holds at least over the range of movie runtimes in the dataset (68 - 202 minutes).

Finally and surprisingly, the model raises the predicted movie rating if the director has ever won an Oscar (all other predictors held constant) whereas the variables for the same being true of the lead actor or actress were removed from the model as being statistically insignificant.

Model Diagnostics

```
# Made into dataframe to make it easier to produce the diagnostic plots
pMod <- fortify(final_model)

# Create residuals scatter plot
p1 <- ggplot(pMod, aes(x=.fitted, y=.resid)) + geom_point() +
      geom_smooth(se=FALSE) + geom_hline(yintercept=0, col="red", linetype="dashed") +
      xlab("Fitted Values")+ylab("Residuals") +
      ggtitle("Residual vs Fitted Plot")

# The following is a bunch of extra code to get around ggplot not being able
# to automatically draw a normal distribution line on a QQ plot
# This code comes from a blog post at http://mgimond.github.io/ES218/Week06a.html
pMod$.qqnorm <- qqnorm(pMod$.stdresid, plot.it=FALSE)$x
y <- quantile(pMod$.stdresid, c(0.25, 0.75)) # Find the 1st and 3rd quartiles
x <- quantile(pMod$.qqnorm, c(0.25, 0.75))   # Find the 1st and 3rd quartiles
slope <- diff(y) / diff(x)             # Compute the line slope
```

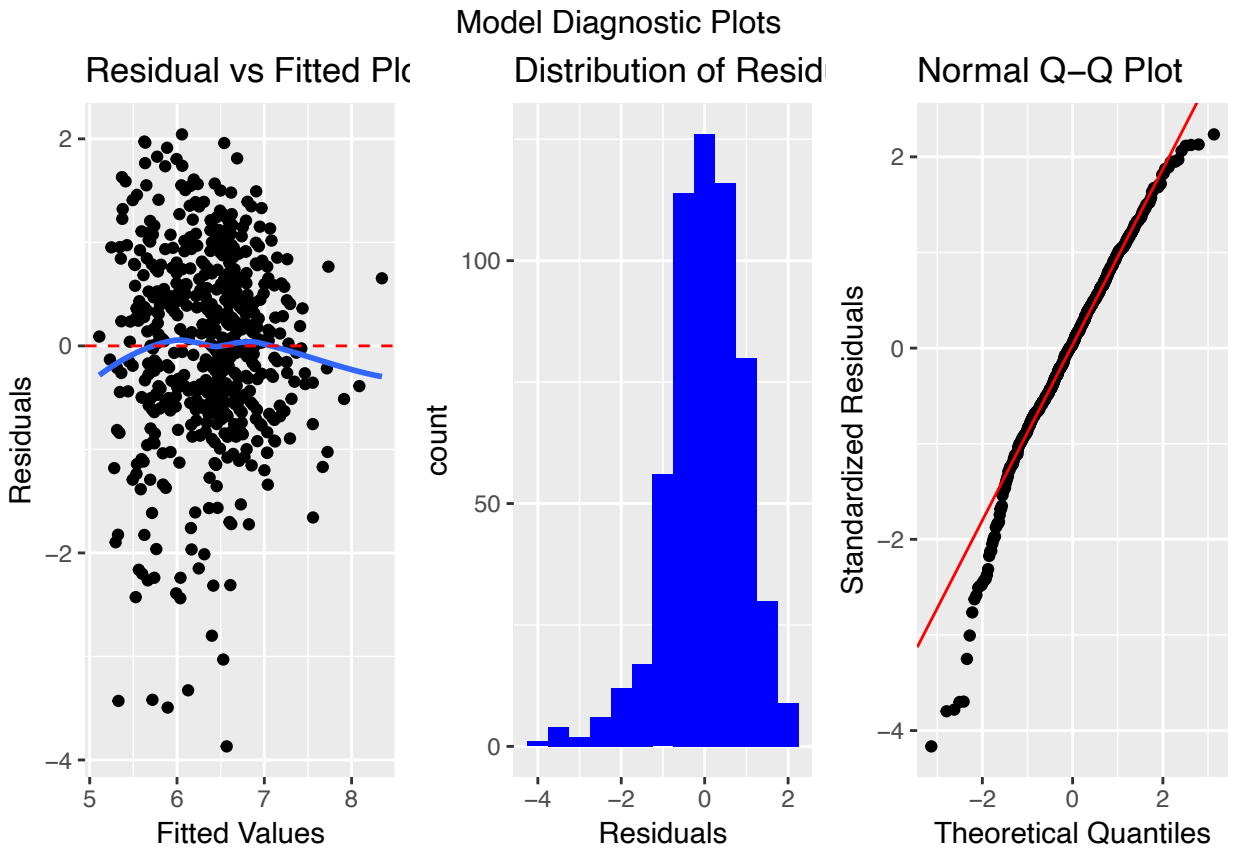
```
int <- y[1] - slope * x[1]              # Compute the line intercept

# Create residuals QQ plot
p2 <- ggplot(pMod, aes(.qqnorm, .stdresid)) +
      geom_point(na.rm = TRUE) +
      geom_abline(intercept=int, slope=slope, color="red") +
      xlab("Theoretical Quantiles")+ylab("Standardized Residuals") +
      ggtitle("Normal Q-Q Plot")

# Create residuals histogram plot
p3 <- ggplot(data=pMod, aes(x=.resid)) +
      geom_histogram(binwidth=0.5, fill="blue") +
      xlab("Residuals") +
      ggtitle("Distribution of Residuals")

grid.arrange(p1, p3, p2, nrow=1, top="Model Diagnostic Plots")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



The model diagnostic plots above show that the model is passable. There is good scatter of the residuals around zero for the range of fitted values (the mean value of the residuals is zero). The residuals QQ plot and distribution histogram show a nice normal distribution. Overall, the evidence points toward the final model being valid.

Prediction

```
# Use the final model to generate rating predictions for Dirty Grandpa released
# in January 2016 and for Deadpool released in February 2016.
dataDG <- data.frame(genre="Comedy", runtime=102, mpaa_rating="R", best_dir_win="no")
predDG <- predict(final_model, dataDG, interval="predict")

dataDead <- data.frame(genre="Action & Adventure", runtime=108, mpaa_rating="R", best_dir_win="no")
predDead <- predict(final_model, dataDead, interval="predict")

# Show prediction results.
df <- data.frame(t=c("Dirty Grandpa", "Deadpool"),
                 p=c(sprintf("%2.1f", predDG[1]),
                     sprintf("%2.1f", predDead[1])),
                 i=c(sprintf("%2.1f - %2.1f", predDG[2], predDG[3]),
                     sprintf("%2.1f - %2.1f", predDead[2], predDead[3])),
                 r=c("6.0", "8.1"))
# Simple table generator
kable(df, col.names=c("Movie Title", "Predicted Rating", "95% Prediction Interval", "Actual Rating"))
```

Movie Title	Predicted Rating	95% Prediction Interval	Actual Rating
Dirty Grandpa	5.9	4.1 - 7.8	6.0
Deadpool	6.1	4.2 - 7.9	8.1

As can be seen, the model was very close in predicting the rating for Dirty Grandpa, but significantly off in its prediction for Deadpool; the real rating for which is even outside of the 95% confidence prediction interval.

Because the 95% confidence prediction intervals are very wide, it reflects the limited predictive capability of the model (further evidenced by its F-statistic and adjusted R-square values, F-statistic is a good indicator of whether there is a relationship between our predictor and the response variables. The further the F-statistic is from 1 the better it is).

Conclusion

We have concluded that yes, it is possible to predict the popularity of a movie based upon basic movie characteristic data. In this analysis, a valid, parsimonious, multi-variable, linear regression model was created that proved to have some capability for predicting movie popularity as indicated by IMDB movie rating score.

But, there is much room for improvement. As shown in the predictions the predictive power of the model is limited. Some further suggestions for improving the mode could be: - Start with a larger analysis sample to capture more variability in the population - data. - Use a stratified sample reflecting the true proportion of movie genres in the population rather than a simple random sample. - Create separate models for each movie genre. - Identify other movie characteristic data to add to the model; identificaton of sequels and their ratings or searching for keywords in the movie title or description, for example.