Introduction

This project uses a movie industry dataset from Kaggle, it has 7668 observations and 15 columns (variables). Each observation is a movie that came out between the years 1980 and 2020. The variables of interest are name, rating (PG, PG-13, R, etc.), genre, score, votes, budget, and gross.

The goals of this project are:

1. Run regression to figure out which variables are the most helpful predictors of gross (revenue).
2. Build an algorithm to classify movies as successful or unsuccessful based on revenue, reviews, and perhaps other factors that are found to be useful.
3. Create a predictive model to predict the scores (IMDb ratings) of movies.

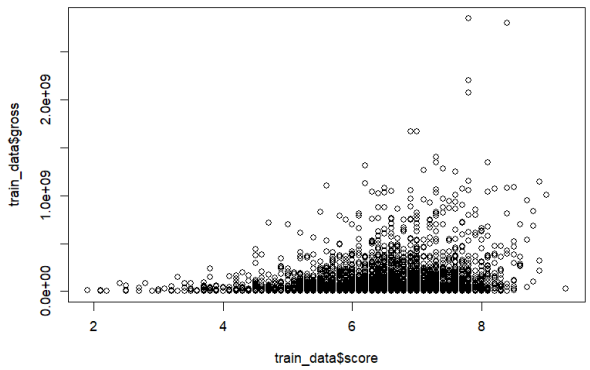
Goal #1

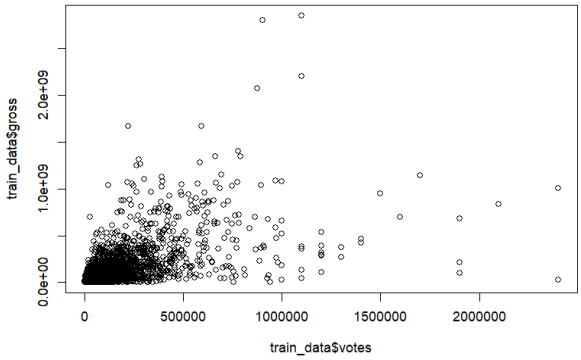
I have started by loading the dataset into R and using the na.omit() function to remove all of the missing values, this significantly reduced the number of observations to 5435 rows. I then split the data into training and test sets using a 70/30 split. I created a regression model using the training data. I had to hand select which variables to test, I did not have the luxury of testing every predictor variable since it would take too much computation time. I chose to test the variables rating, genre, score, votes, budget, and runtime on gross revenue.

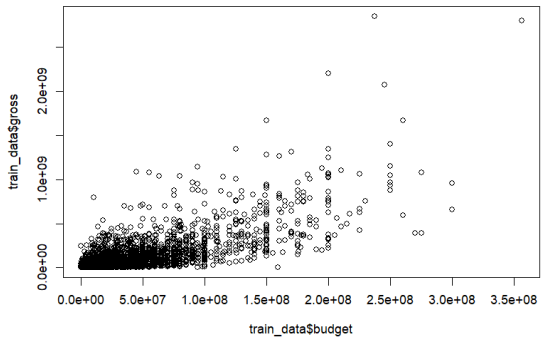
The four assumptions for this regression model are as follows:

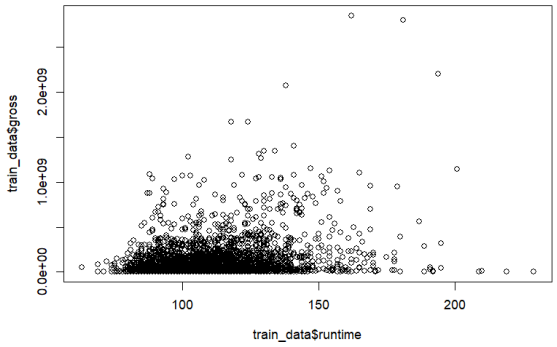
- Linearity
- Independence: Observations are independent of each other
- Normality: Random error follows a normal distribution
- Equal-Variance: Random error has the same variance

For linearity, I have to create a lot of plots since I have a good amount of possible predictor variables. I decided to not check the categorical variables since I would have to use boxplots and there isn't really a good way to check them. IMDb ratings, votes, and runtime are not linear. We can see that budget is the only slightly linear predictor.

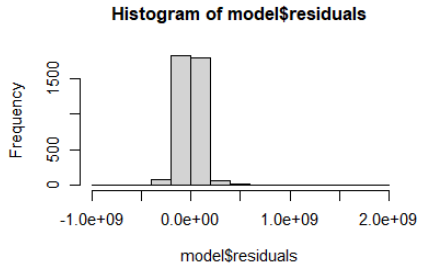


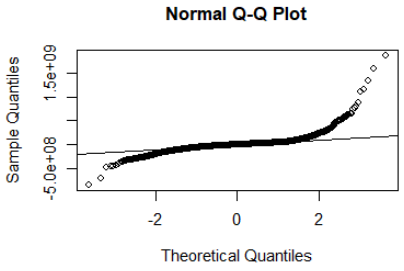


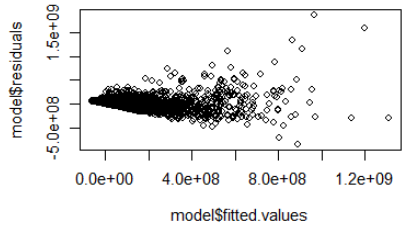




Whether or not the movies are independent of each other is arguable due to industry trends, but for the purposes of this report, they are independent.







We can see from the above figures on the previous page that the normality assumption is not met.

I checked the equal variance assumption by calculating the residuals of the dataset and then plotting them. The plot should have scattered data points and no distinct pattern, but the points on this residual plot are clustered together. Therefore, the assumption of equal variance is not met. A lot of the assumptions have not been met, which is fine since the tests are super sensitive, but it is something to keep in mind.

I will now evaluate the model's performance by calculating mean squared error (MSE) and R^2. The test MSE turned out to be 1.094193e+16, which is extremely large and indicative of poor model performance. On the bright side, the R-squared value is 0.6832952, which means that approximately 68.33% of variability in gross revenue is explained by the model.

HYPOTHESES
beta1 = regression coefficient for rating, beta2 = regression coefficient for genre, beta3 = regression coefficient for budget, beta4 = regression coefficient for score, beta5 = regression coefficient for votes, and beta6 = regression coefficient for runtime
Null hypotheses: beta1 = 0, beta2 = 0, beta3 = 0, beta4 = 0, beta5 = 0, and beta6 = 0
Alternative hypotheses: beta1 =/= 0, beta2 =/= 0, beta3 =/= 0, beta4 =/= 0, beta5 =/= 0, and beta6 =/= 0
Alpha = 0.05

RESULTS
Reject null hypothesis for betas 1 through 5 and fail to reject the null for beta 6. Rating, genre, budget, score, and votes are helpful predictors of gross revenue, but runtime is not.

```
Call:
lm(formula = gross ~ budget + votes + rating_factor + score +
    genre_factor, data = train_data)

Residuals:
      Min        1Q    Median        3Q       Max
-858937423  -36808994    -327378   29591516 1877992455

Coefficients:
                        Estimate Std. Error t value Pr(>|t|)
(Intercept)           -8.106e+07  3.818e+07  -2.123 0.033831 *
budget                 2.517e+00  5.517e-02  45.625  < 2e-16 ***
votes                  3.581e+02  1.241e+01  28.843  < 2e-16 ***
rating_factorG         2.853e+06  3.769e+07   0.076 0.939679
rating_factorNC-17    -2.176e+06  5.219e+07  -0.042 0.966752
rating_factorNot Rated 2.580e+07  4.038e+07   0.639 0.522896
```

```
rating_factorPG         2.827e+07  3.527e+07   0.801 0.422963
rating_factorPG-13      1.761e+07  3.510e+07   0.502 0.615883
rating_factorR         -1.430e+06  3.495e+07  -0.041 0.967369
rating_factorTV-MA      3.086e+08  8.517e+07   3.623 0.000295 ***
rating_factorUnrated    2.702e+07  4.927e+07   0.548 0.583487
score                   5.415e+06  2.330e+06   2.325 0.020144 *
genre_factorAdventure   4.670e+06  8.588e+06   0.544 0.586585
genre_factorAnimation   6.235e+07  1.081e+07   5.769 8.61e-09 ***
genre_factorBiography  -1.707e+07  8.606e+06  -1.984 0.047327 *
genre_factorComedy      1.167e+07  5.178e+06   2.254 0.024252 *
genre_factorCrime      -3.725e+06  7.751e+06  -0.481 0.630898
genre_factorDrama      -2.411e+06  6.019e+06  -0.401 0.688747
genre_factorFamily      4.679e+08  6.375e+07   7.339 2.63e-13 ***
genre_factorFantasy     1.598e+07  2.117e+07   0.755 0.450251
genre_factorHorror      4.009e+07  9.040e+06   4.435 9.47e-06 ***
genre_factorMystery    -1.333e+06  3.073e+07  -0.043 0.965411
genre_factorRomance    -2.751e+07  1.101e+08  -0.250 0.802706
genre_factorSci-Fi     -3.052e+06  4.932e+07  -0.062 0.950667
genre_factorThriller    3.652e+07  4.506e+07   0.810 0.417733
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 109900000 on 3782 degrees of freedom
Multiple R-squared:  0.6595,    Adjusted R-squared:  0.6574
F-statistic: 305.3 on 24 and 3782 DF,  p-value: < 2.2e-16
```

Conclusion

After removing runtime from the model, the test MSE slightly decreased to 1.09332e+16 and the R^2 slightly increased to 0.6835481. We can see from the output that budget, votes, and score are still helpful predictors. We also observe that whether or not a movie is rated TV-MA is a useful predictor of gross revenue. Whether or not a given movie is a family movie, an animation, a biography, a comedy, or a horror movie can also determine how much gross revenue will be made.

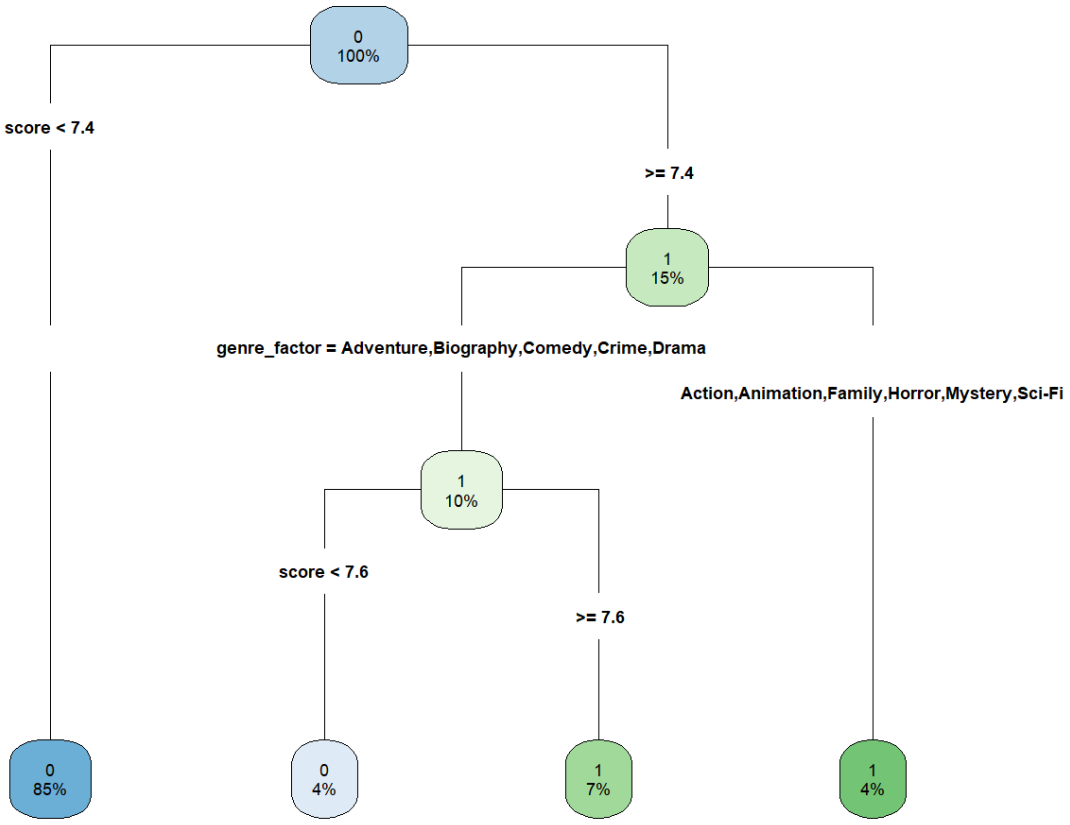
Goal #2

First, I had to define the outcome variable to determine whether a movie was considered successful or unsuccessful. I chose to evaluate gross revenue over IMDb score since scores tend to be subjective and dichotomous. The criteria for a successful movie is that it has to bring in at least 3x its production budget. I chose genre, budget, rating, and score as my predictor variables. I split the data using a 70/30 split and converted genre and rating to factors with levels. I then trained the model and created the following confusion matrix.

```
  pred_class
     0    1
0 1034   98
1  361  137
```

We observe that we have 98 false positives and 361 false negatives. I calculated the test error rate for the model which turned out to be approximately 0.2816.

I fit a decision tree to the data and created the following dendogram

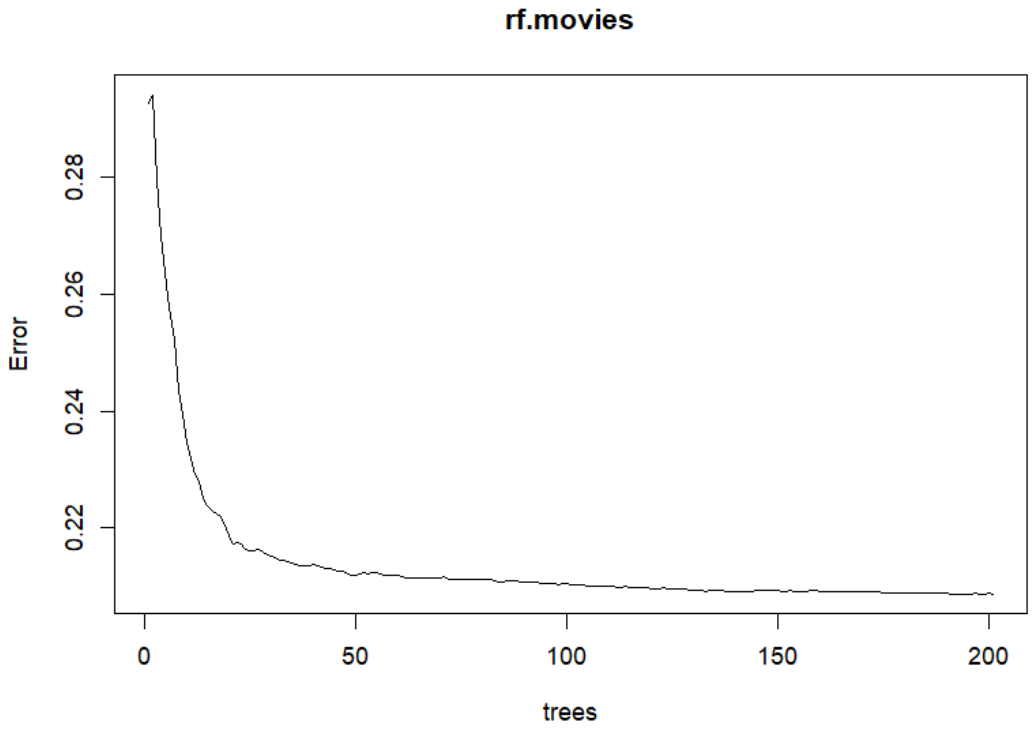


From the dendogram, we observe that movies that have IMDb scores of less than 7.4 are probably going to be classified as unsuccessful. Movies that have scores that are greater than or equal to 7.4 will probably be successful if their genre is Action, Animation, Family, Horror, Mystery, or Sci-Fi. If the movie has a score of 7.4 or higher and its genre is Adventure, Biography, Comedy, Crime, or Drama, it will probably be succesful if its score is greater than or equal to 7.6 or unsuccessful if it is less than 7.6.

I computed the following confusion matrix for the decision tree, we observe that there are 381 false positives and 77 false negatives. Its test error rate is approximately 0.2810, which is a very small improvement from the log regression's error rate.

	Reference	
Prediction	0	1
0	1055	381
1	77	117

I chose to not prune the tree since already has minimal splits, but I still did cross validation. The mean and standard deviation of the misclassifciation rates are approximately 0.3029 and 0.0252 respectively. I created a random forest using the tree and we can see the error decreased steadily as more trees were grown.



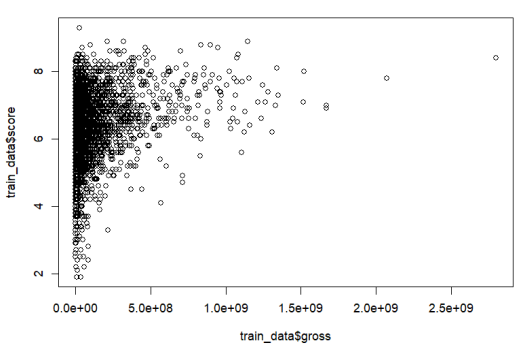
The following confusion matrix for the random forest shows that there are 297 false negatives and 169 false positives, the test error is approximately 0.2859.

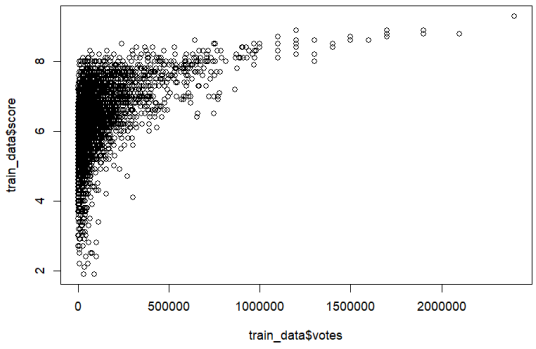
	rfpred_class	
	0	1
0	963	169
1	297	201

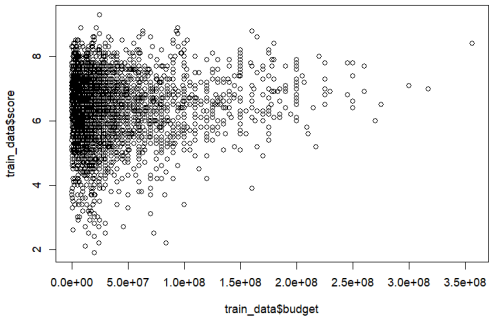
I used the importance() function to figure out that the most important variable in the random forest is budget. I calculated the forest's sensitivity and specificity which turned out to be approximately 0.4036 and 0.8507; it seems like the model is better at identifying true negatives than it is with true positives.

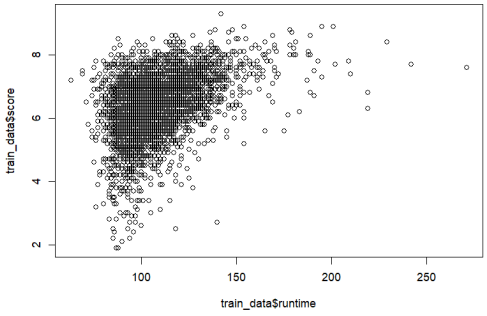
Goal #3

I started by doing a 70/30 split on the data to create training and test splits. I then checked the four assumptions: linearity, independence, normality, and equal variance.



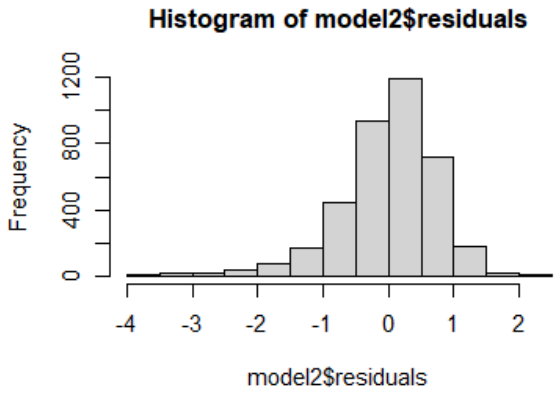


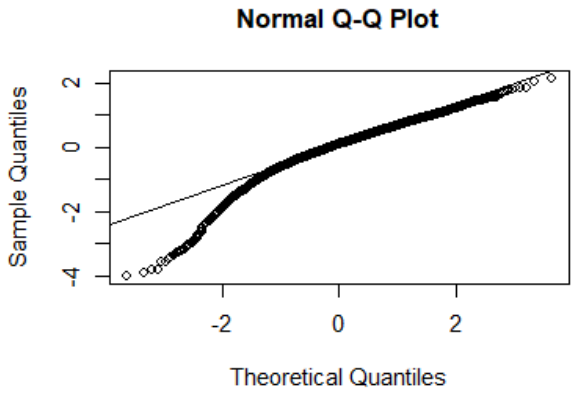


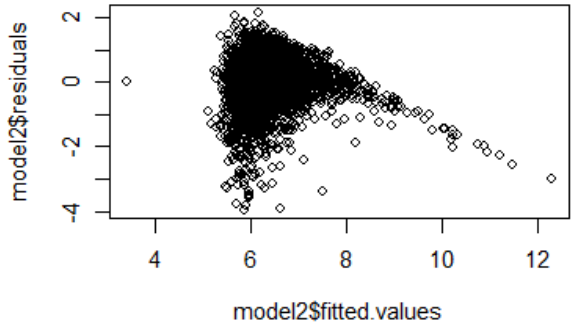


I chose gross, budget, rating, genre, votes, and runtime as my predictor variables. I plotted the quantitative variables against score and we can see that the linearity assumption is not met.

Just like before, I am assuming that the observations in this movies dataset are independent from each other.







We can see from the histogram that the residuals are approximately normally distributed, however, the Q-Q plot shows otherwise. The assumption of equal variances is not met since we can see the datapoints on the scatter plot are all clustered together.

```
Call:
lm(formula = score ~ gross + rating + genre + votes + budget +
    runtime, data = train_data)

Residuals:
    Min      1Q  Median      3Q     Max
-3.9813 -0.3610  0.0909  0.4870  2.1275

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     4.528e+00  2.526e-01  17.925  < 2e-16 ***
gross           3.506e-10  1.129e-10   3.106  0.00191 **
ratingApproved  -3.151e+00  7.837e-01  -4.021 5.91e-05 ***
ratingG         -2.184e-01  2.565e-01  -0.851  0.39465
ratingNC-17     -1.084e-01  3.432e-01  -0.316  0.75221
ratingNot Rated  2.450e-01  2.713e-01   0.903  0.36638
ratingPG        -3.490e-01  2.395e-01  -1.457  0.14515
ratingPG-13     -2.753e-01  2.381e-01  -1.156  0.24767
ratingR         -1.149e-01  2.371e-01  -0.485  0.62790
ratingTV-MA      4.047e-01  7.821e-01   0.517  0.60488
ratingUnrated    2.663e-01  3.016e-01   0.883  0.37736
ratingX          1.105e+00  7.827e-01   1.411  0.15823
```

```
genreAdventure   1.465e-01  5.731e-02   2.556  0.01062 *
genreAnimation   9.432e-01  7.291e-02  12.937  < 2e-16 ***
genreBiography   6.132e-01  5.667e-02  10.821  < 2e-16 ***
genreComedy      1.028e-01  3.566e-02   2.883  0.00396 **
genreCrime       2.940e-01  5.280e-02   5.568 2.75e-08 ***
genreDrama       3.838e-01  4.122e-02   9.310  < 2e-16 ***
genreFamily      3.749e-01  3.775e-01   0.993  0.32067
genreFantasy    -1.233e-01  1.350e-01  -0.914  0.36090
genreHorror     -2.842e-01  6.230e-02  -4.562 5.22e-06 ***
genreMystery    -4.175e-02  2.271e-01  -0.184  0.85415
genreRomance     2.661e-01  5.279e-01   0.504  0.61424
genreSci-Fi      2.755e-01  3.054e-01   0.902  0.36713
genreThriller   -3.275e-01  3.738e-01  -0.876  0.38108
genreWestern    -6.964e-01  7.457e-01  -0.934  0.35045
votes            2.203e-06  8.539e-08  25.803  < 2e-16 ***
budget          -5.404e-09  4.863e-10 -11.111  < 2e-16 ***
runtime          1.641e-02  8.161e-04  20.114  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.745 on 3777 degrees of freedom
Multiple R-squared:   0.42,      Adjusted R-squared:  0.4157
F-statistic: 97.66 on 28 and 3777 DF,  p-value: < 2.2e-16
```

HYPOTHESES

beta1 = regression coefficient for rating, beta2 = regression coefficient for genre, beta3 = regression coefficient for budget, beta4 = regression coefficient for gross, beta5 = regression coefficient for votes, and beta6 = regression coefficient for runtime
Null hypotheses: beta1 = 0, beta2 = 0, beta3 = 0, beta4 = 0, beta5 = 0, and beta6 = 0
Alternative hypotheses: beta1 =/= 0, beta2 =/= 0, beta3 =/= 0, beta4 =/= 0, beta5 =/= 0, and beta6 =/= 0
Alpha = 0.05

RESULTS

REJECT null hypothesis for betas 1 through 6. Rating, genre, budget, gross, votes, and runtime are all useful predictors of score.

The MSE and R^2 for the model are approximately 0.5248 and 0.4017, respectively. These numbers make the model seem questionable, however, I was able to use the model to predict the IMDb score of Everything Everywhere All at Once by putting in the budget, genre, rating, runtime, votes, and gross of the movie. The dataset only includes movies from 1980-2020, so I thought it would be cool to compare the model’s predicted score and the actual IMDb score of Everything Everywhere All at Once since it came out in 2022. The model predicted the movie’s score to be 7.913085 and its actual score on the IMDb website is 7.9, which is pretty much spot on.

Summary of findings

When I created the regression model for predicting gross revenue, I found that budget, score, and votes are helpful predictors. Whether or not a given movie rated TV-MA, is a family movie, an animation, a biography, a comedy, or a horror movie can also determine how much gross revenue will be made. I was surprised to find that runtime was not a helpful predictor of revenue, I figured moviegoers would be less willing to spend their money on movies with longer runtimes, but it seems like it is dependent on other factors. Although the regression model failed almost all of the LINE assumptions, it explained approximately 68.33% of the variability in gross revenue.

I had to define the outcome variable to determine whether a movie was considered successful or unsuccessful when I created the classification algorithm. I chose to define a movie as successful if it brought in at least 3x its budget, I chose this criterion as opposed to using an average movie revenue benchmark because I felt that it would be more accurate. I think my criterion took into account the proportions of budget and revenue - sometimes movies are able to bring in a large amount of revenue even if they have a low budget. The test error rate of the algorithm turned out to be 0.2816 which is really good. I created a decision tree which had splits for score and genre, I decided to not prune the tree since it already few nodes; its mean classification rate was approximately 0.3029, which means the algorithm performed pretty well. I thought it was interesting how budget was not a split in the diagram, but it was the most important variable in the random forest. The forest's sensitivity and specificity which turned out to be approximately 0.4036 and 0.8507, respectively; this shows that the model is better at identifying true negatives than it is with true positives.

I checked the same LINE assumptions when I built the regression model to predict IMDb score, it failed almost all the assumptions, but the histogram of the residuals was approximately normally distributed. I tested to see if rating, genre, budget, gross, votes, and runtime are useful predictors, and they turned out to all be helpful in determining score. The MSE and R^2 for the model were approximately 0.5248 and 0.4017, respectively. The metrics indicate that the model didn't perform well, however, it was extremely accurate when I predicted the score of a movie that came out in 2022, I decided to test a movie that came out after 2020 since the dataset only contains movies that came out during the years of 1980-2020. I inputted all of the predictor variables of the specific movie into the predictive model and the predictive score was very close to its actual score on the IMDb website.