Models on predicting movie rating

C.Y., M.S., G.T., P.C.

STA 9750 Final Project

Introduction

This project aims to predict movies' ratings released on or before July 2017 by using various variables including budget, adult, runtime, release date, genre and production company. At the end of the report, we also explore the relationships between rating and revenue or popularity. Our data was downloaded from kaggle.

The movies dataset was collected from 45,464 movies with 26,024,289 ratings. Each movie was rated by many users, so we used the median rating to represent the rating of the movie. Then we combine the rating file with the movie metadata file via the link file. Hereby, we are left with 45,228 movies.

We also notice that there are some movies which are not in English. To keep the consistency, we filtered out 13,549 movies whose original language are not English and 11 movies whose original language are not available. In addition, we also filtered out 100 movies which don't provide clear information about release date and 7 movies whose runtime are missing.

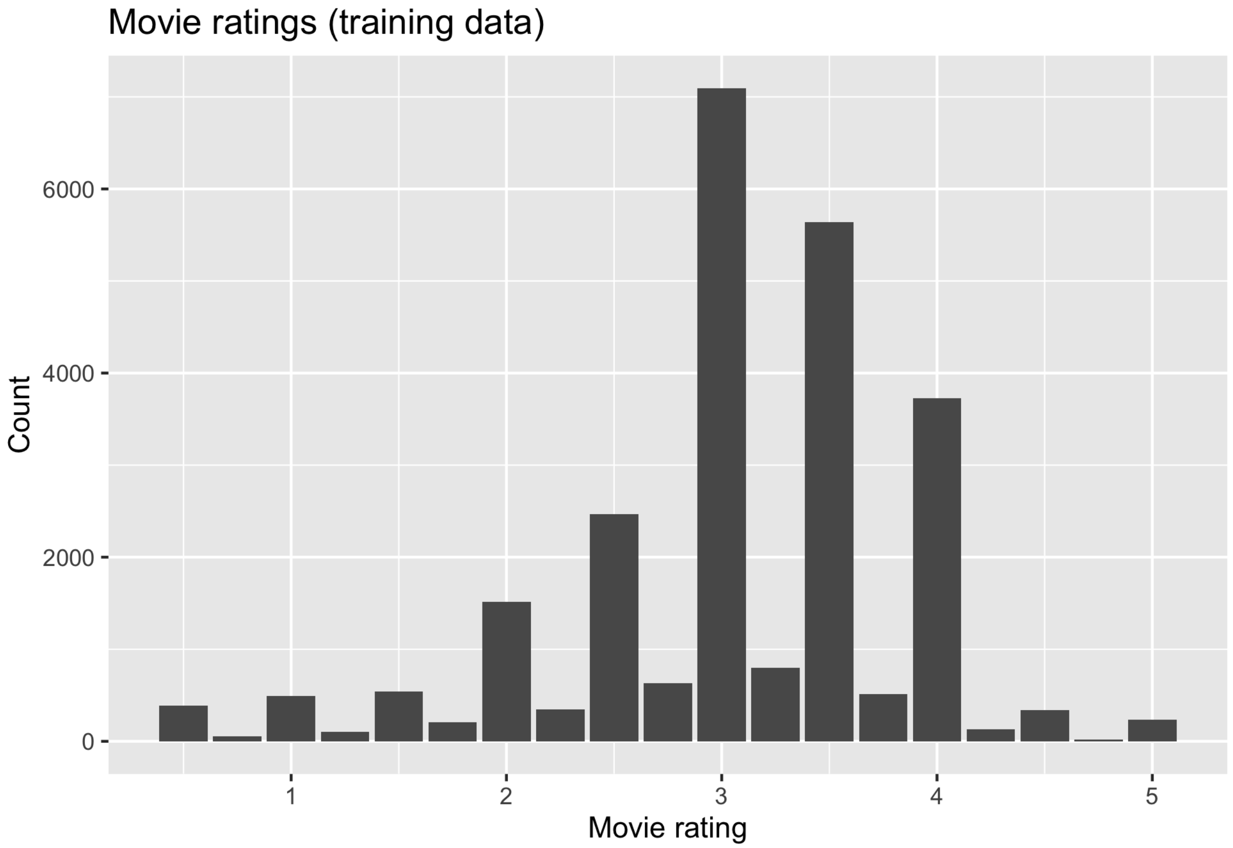
Therefore, we are left with 31,563 rows of data. The first ten rows and some columns of the data are showed below:

mrating	budget	adult	runtime	releaseYear	Animation	Comedy	WarnerBros	ParamountPictures
4.0	3.0e+07	FALSE	81	1995	TRUE	TRUE	FALSE	FALSE
3.0	6.5e+07	FALSE	104	1995	FALSE	FALSE	FALSE	FALSE
3.0	0.0e+00	FALSE	101	1995	FALSE	TRUE	TRUE	FALSE
3.0	1.6e+07	FALSE	127	1995	FALSE	TRUE	FALSE	FALSE
3.0	0.0e+00	FALSE	106	1995	FALSE	TRUE	FALSE	FALSE
4.0	6.0e+07	FALSE	170	1995	FALSE	FALSE	TRUE	FALSE
3.0	5.8e+07	FALSE	127	1995	FALSE	TRUE	FALSE	TRUE
3.0	0.0e+00	FALSE	97	1995	FALSE	FALSE	FALSE	FALSE
3.0	3.5e+07	FALSE	106	1995	FALSE	FALSE	FALSE	FALSE
3.5	5.8e+07	FALSE	130	1995	FALSE	FALSE	FALSE	FALSE

Next we split the data with 80% training data and 20% test data. We will fit our models on the training data and validate the models on the test data.

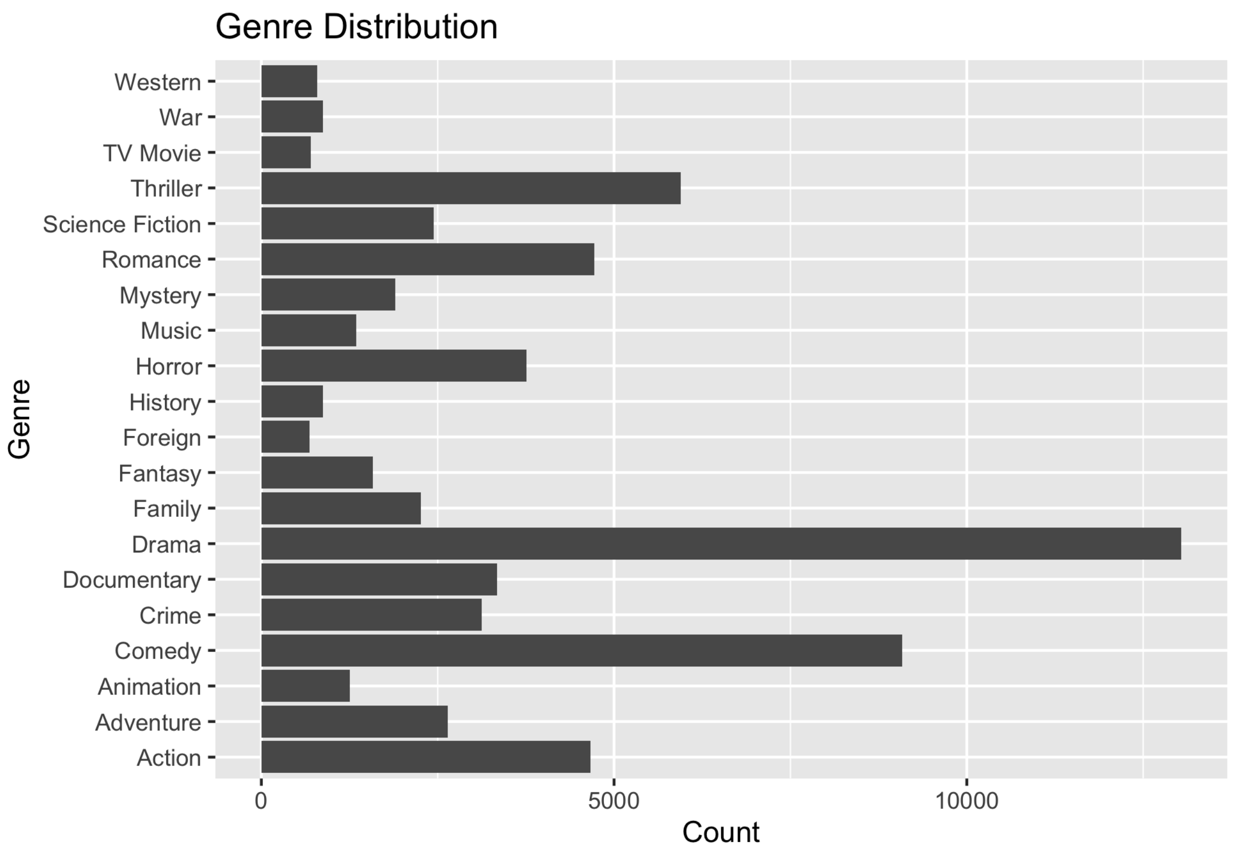
Summary Statistics

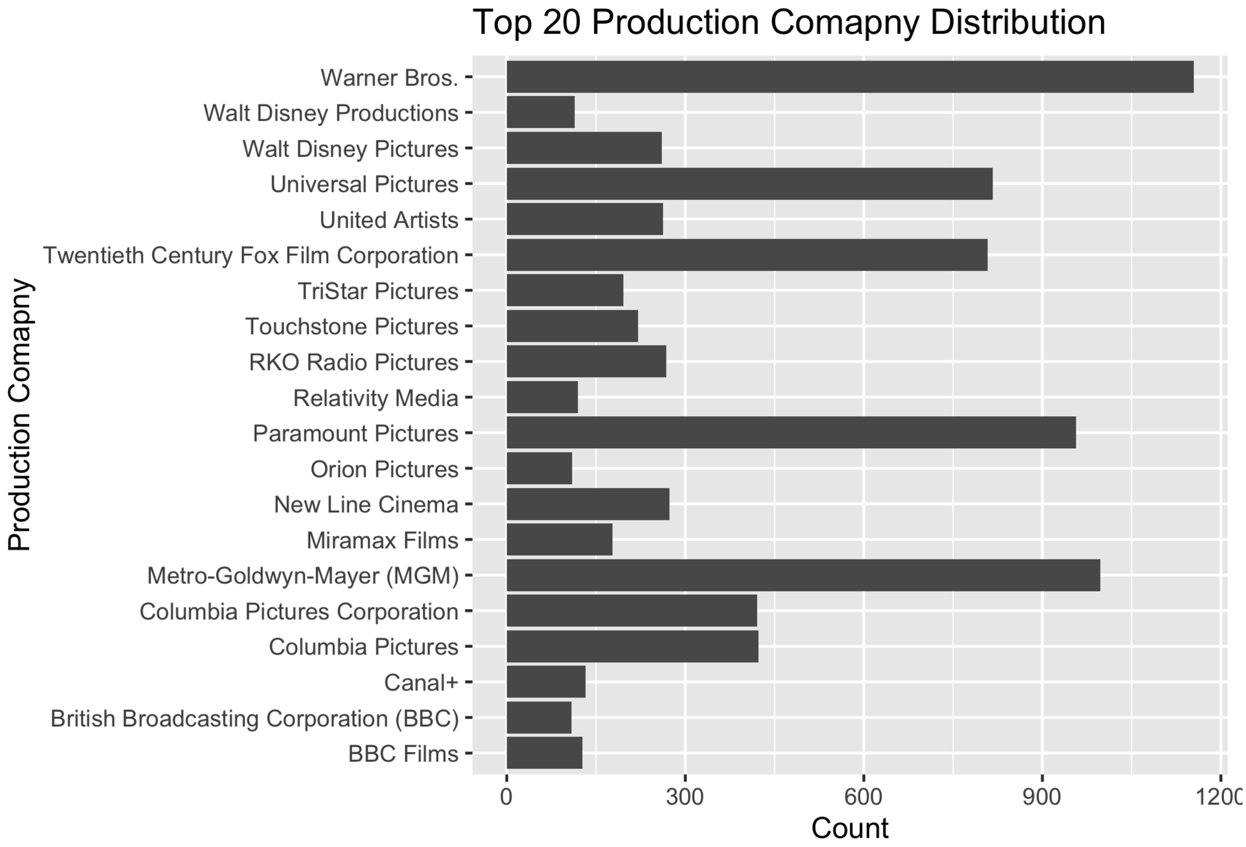
Below is the bar chart of all ratings in the training data:



From the bar chart, we can easily tell most of the movie ratings fall into the range between 3 and 4. From which, 3 has the greatest count, followed by 3.5 and 4. Looking at the table below, we can verify that the median movie rating is 3 and the mean movie rating is 3.07. Also, we create a table to summarize the variables containing mrating, budget, adult, runtime, and releaseYear. In addition, we make two bar charts below to show all the genres distribution and the first twenty production companies distribution.

mrating	budget	adult	runtime	releaseYear
Min. :0.500	Min. : 0	Mode :logical	Min. : 0.00	Min. :1918
1st Qu.:2.750	1st Qu.: 0	FALSE:25246	1st Qu.: 85.00	1st Qu.:1980
Median :3.000	Median : 0	TRUE :4	Median : 93.00	Median :2001
Mean :3.071	Mean : 5741609	NA	Mean : 93.58	Mean :1992
3rd Qu.:3.500	3rd Qu.: 0	NA	3rd Qu.: 105.00	3rd Qu.:2010
Max. :5.000	Max. :380000000	NA	Max. :1140.00	Max. :2017


Genre Distribution
Western
War
TV Movie
Thriller
Science Fiction
Romance
Mystery
Music
Horror
History
Foreign
Fantasy
Family
Drama
Documentary
Crime
Comedy
Animation
Adventure
Action
Genre
0
5000
10000
Count



Models and Analysis

Random Forest Model We begin the analysis by fitting the Random Forest Model with all the independent variables: adult, runtime, budget, releaseYear, twenty genres, as well as top fifty production companies to know the most important variables which affect predicting movie ratings mostly.

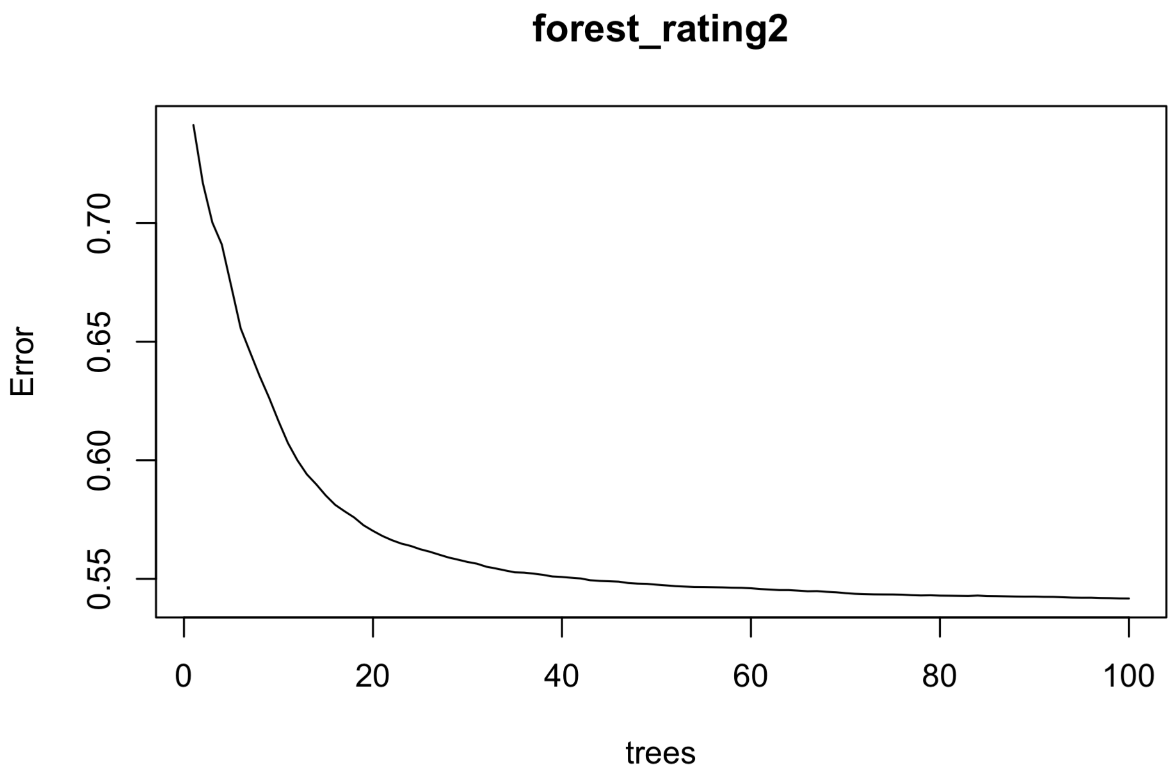
The importance table is listed below. From which, we can easily tell runtime, releaseYear, budget, and genres are important variables.

	%IncMSE
Horror	51.36145
runtime	50.16879
Documentary	48.63631
Drama	39.96644
releaseYear	38.27363
budget	35.56617

Therefore, we drop the two variables (adult and production company) and then fit another random forest model using the remaining important variables only and below is a summary and plot of the model.

```
##
## Call:
##  randomForest(formula = mrating ~ ., data = all_train %>% select(mrating,      runtime, budget, relea
##                Type of random forest: regression
##                      Number of trees: 100
## No. of variables tried at each split: 7
```

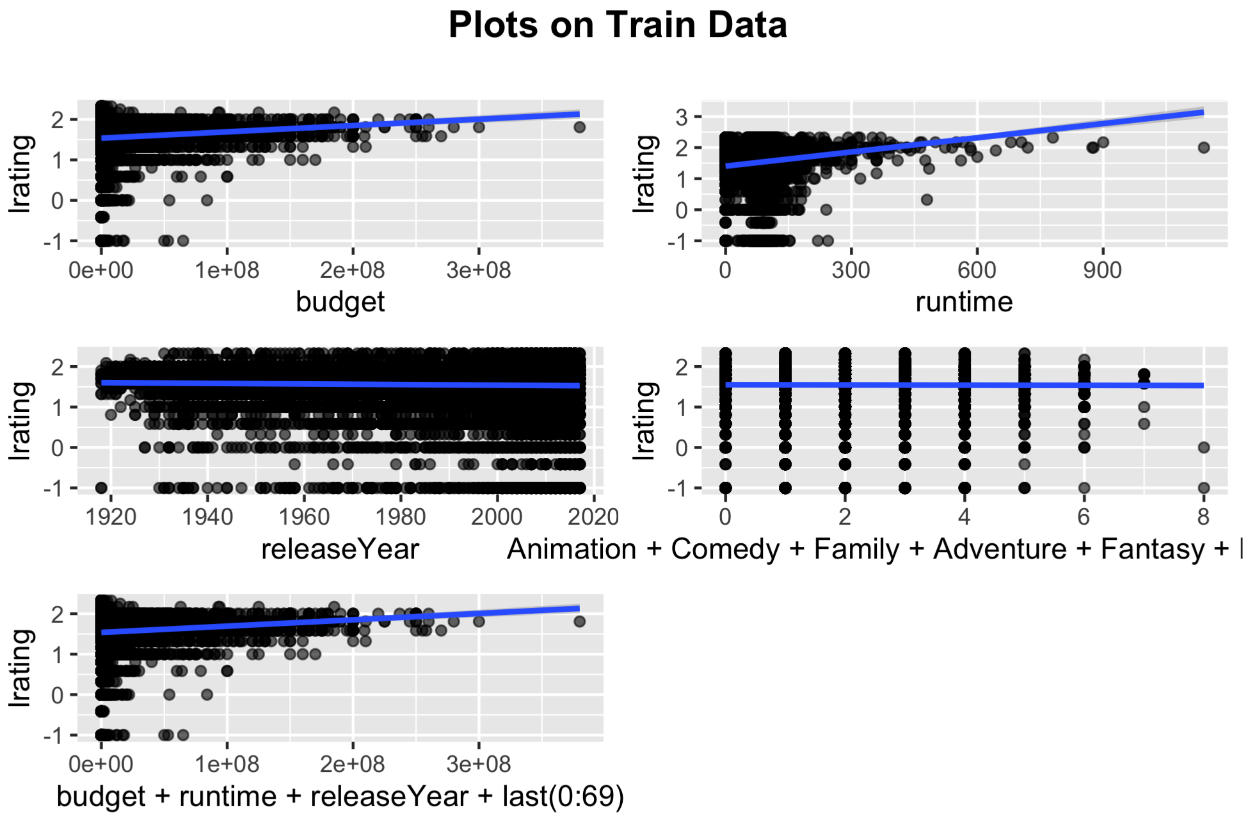
```
##
##           Mean of squared residuals: 0.5417295
##                     % Var explained: 15.59
```



Linear Regression Model Now we fit Linear Regression Model by using the important variables mentioned above.

First we start to predict movie ratings by each of the important independent variables: budget, runtime, releaseYear, and genres respectively. Then we consider the linear regression model with all the important variables. Here, we use lrating, which is the log transformation of mrating, as the dependent variable.

Then we make some plots to see whether the models fit the data well.



Here, we also calculate the Akaike information criterion (AIC) for the linear regression models. AIC is an estimator of out-of-sample prediction error and an estimator of the relative amount of information lost by a given model. So when comparing models fitted to the same data, the smaller the AIC, the better the model fit. The AIC for our models with budget, that with runtime, that with releaseYear, that with genres, and that with all the important variables are 38,720, 38,538, 38,786, 36,388, and 35,983 respectively.

It is easy to find that the linear regression model with all the important variables has the lowest AIC, which means the model fits the data best among all these linear regression models. This is also fairly reasonable in reality. The model is shown below:

```
##
## Call:
## lm(formula = lrating ~ ., data = all_train %>% select(lrating,
##     budget, runtime, releaseYear, last_col(50:69)))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.85852 -0.11946  0.08757  0.27390  1.19462
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          3.796e+00  2.822e-01  13.452  < 2e-16 ***
## budget               2.254e-09  1.657e-10  13.598  < 2e-16 ***
## runtime              1.092e-03  9.278e-05  11.768  < 2e-16 ***
## releaseYear         -1.215e-03  1.419e-04  -8.563  < 2e-16 ***
## TVMovieTRUE          3.573e-02  2.110e-02   1.693  0.09042 .
## WesternTRUE          3.637e-02  2.093e-02   1.738  0.08220 .
## ForeignTRUE        -2.925e-02  2.152e-02  -1.359  0.17415
## DocumentaryTRUE     2.975e-01  1.207e-02  24.650  < 2e-16 ***
## MusicTRUE           4.972e-02  1.558e-02   3.191  0.00142 **
## WarTRUE             4.561e-02  1.958e-02   2.329  0.01985 *
## MysteryTRUE         6.424e-02  1.397e-02   4.599 4.28e-06 ***
## ScienceFictionTRUE -8.545e-02  1.225e-02  -6.974 3.15e-12 ***
## HistoryTRUE         7.008e-02  1.964e-02   3.568  0.00036 ***
## HorrorTRUE         -2.548e-01  1.074e-02 -23.734  < 2e-16 ***
## ThrillerTRUE        5.607e-03  9.349e-03   0.600  0.54868
## CrimeTRUE           1.910e-02  1.131e-02   1.690  0.09112 .
## ActionTRUE         -1.213e-01  1.004e-02 -12.087  < 2e-16 ***
## DramaTRUE           1.239e-01  7.463e-03  16.602  < 2e-16 ***
## RomanceTRUE         2.363e-02  9.337e-03   2.531  0.01139 *
## FantasyTRUE        -2.925e-03  1.491e-02  -0.196  0.84453
## AdventureTRUE       1.797e-04  1.261e-02   0.014  0.98863
## FamilyTRUE         -4.496e-02  1.373e-02  -3.275  0.00106 **
## ComedyTRUE          3.488e-02  7.764e-03   4.493 7.07e-06 ***
## AnimationTRUE       1.276e-01  1.797e-02   7.100 1.28e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4932 on 25226 degrees of freedom
## Multiple R-squared:  0.1077, Adjusted R-squared:  0.1069
## F-statistic: 132.3 on 23 and 25226 DF,  p-value: < 2.2e-16
```

However, we observe that the R-squared is relatively low for this model even though it performs best among all the linear regression models. Therefore, we may draw two hypothesis here.

Hypothesis The first hypothesis is that linear regression model is not a good model for this dataset. Thus, it can be explained why the R-squared is low even with the best linear regression model. We may verify this hypothesis in the later validation on the test data part.

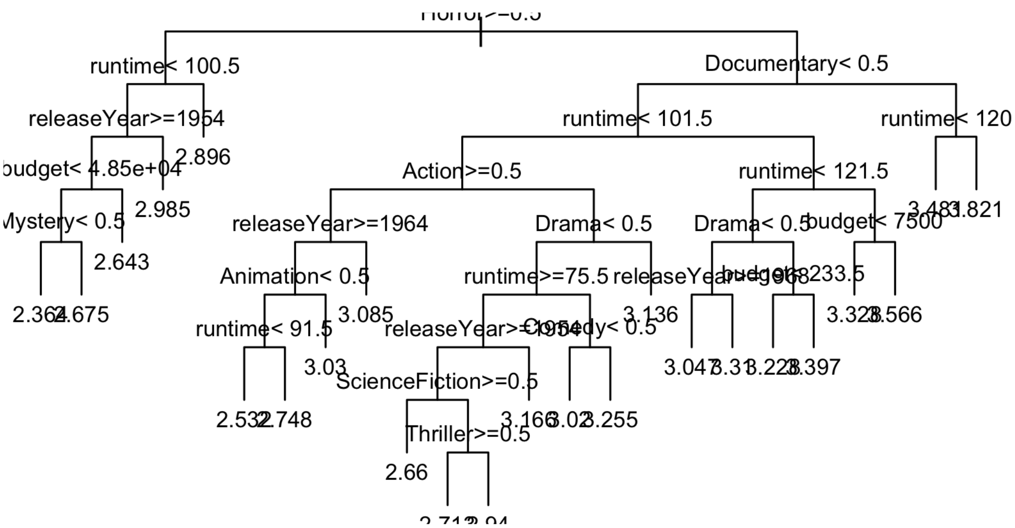
The second hypothesis is that the current variables are not sufficient to predict movie ratings efficiently. In order to have a better performance, we may need more variables to be involved in the model. We may confirm this hypothesis in the later conclusion part.

Now we will introduce another tree- based model to check the model performance.

Regression Tree Model We decide to contain all the important variables to predict movie ratings for the regression tree model. First, we fit five trees of increasing complexity on all important variables on train data.

Then we show the second tree model plot as a representation as follows. We may find Horror and Documentary in the genre category, runtime, releaseYear, budget are all important predictive variables, which verifies what we drive from the random forest model.

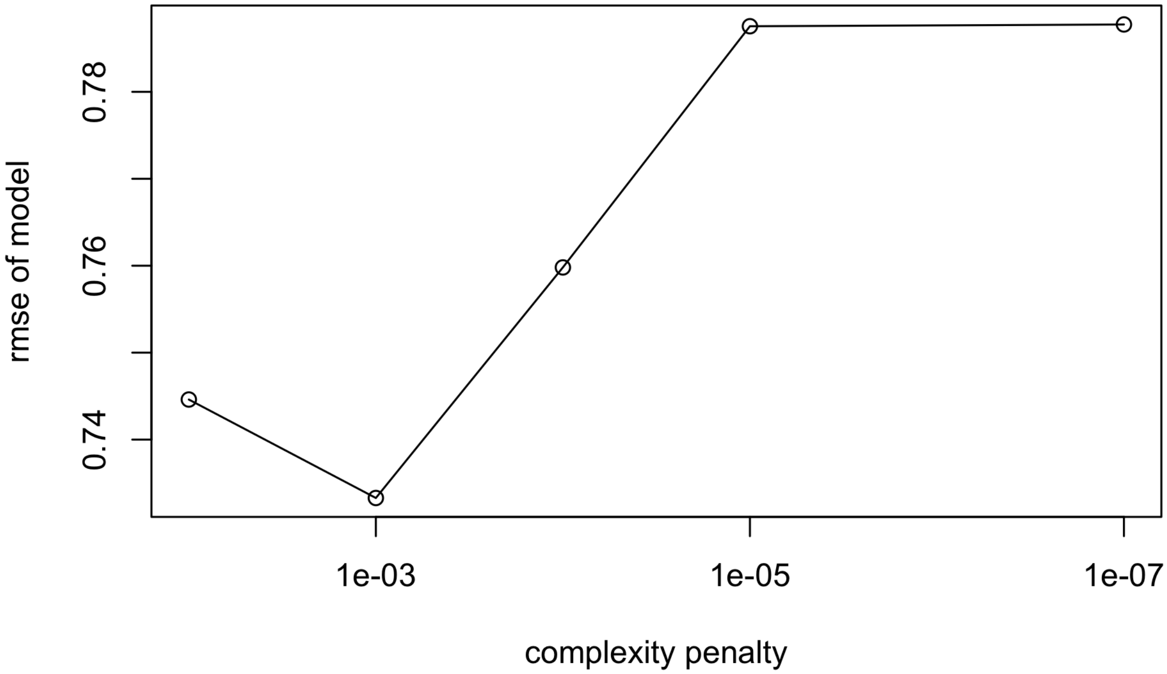
```
## n= 25250
##
## node), split, n, deviance, yval
##       * denotes terminal node
##
##   1) root 25250 16205.22000 3.071069
##     2) Horror>=0.5 3009  2138.24300 2.546278
##       4) runtime< 100.5 2571  1856.17500 2.486776
##         8) releaseYear>=1953.5 2456  1780.53300 2.463457
##          16) budget< 48500 1817  1399.85700 2.400248
##            32) Mystery< 0.5 1603  1267.83600 2.363537 *
##            33) Mystery>=0.5 214   113.67870 2.675234 *
##          17) budget>=48500 639   352.77290 2.643192 *
##         9) releaseYear< 1953.5 115    45.78587 2.984783 *
##       5) runtime>=100.5 438   219.53380 2.895548 *
##     3) Horror< 0.5 22241 13126.16000 3.142069
##       6) Documentary< 0.5 19625 11611.67000 3.093911
##        12) runtime< 101.5 12827  8391.13500 2.995673
##          24) Action>=0.5 2086  1387.45100 2.723873
##            48) releaseYear>=1963.5 1824  1221.09400 2.672012
##              96) Animation< 0.5 1708  1129.93300 2.647687
##               192) runtime< 91.5 791   606.33490 2.531606 *
##               193) runtime>=91.5 917   503.74560 2.747819 *
##              97) Animation>=0.5 116    75.26940 3.030172 *
##            49) releaseYear< 1963.5 262   127.29790 3.084924 *
##          25) Action< 0.5 10741  6819.65200 3.048459
##            50) Drama< 0.5 6017  4333.29700 2.979807
##             100) runtime>=75.5 4256  2886.58600 2.925928
##               200) releaseYear>=1954.5 3727  2601.67400 2.891870
##                 400) ScienceFiction>=0.5 329   201.62230 2.659574 *
##                 401) ScienceFiction< 0.5 3398  2380.57900 2.914361
##                   802) Thriller>=0.5 385   292.91010 2.712987 *
##                   803) Thriller< 0.5 3013  2070.06200 2.940093 *
##               201) releaseYear< 1954.5 529   250.13160 3.165879 *
##             101) runtime< 75.5 1761  1404.49600 3.110023
##               202) Comedy< 0.5 1088   945.68010 3.020221 *
##               203) Comedy>=0.5 673   435.85680 3.255201 *
##            51) Drama>=0.5 4724  2421.87600 3.135902 *
##        13) runtime>=101.5 6798  2863.17500 3.279273
##          26) runtime< 121.5 4797  1998.67600 3.212112
##            52) Drama< 0.5 1734   794.63700 3.091407
##             104) releaseYear>=1967.5 1439   656.50550 3.046560 *
##             105) releaseYear< 1967.5 295   121.11950 3.310169 *
##            53) Drama>=0.5 3063  1164.47400 3.280444
##             106) budget< 233.5 2114   877.10220 3.228004 *
##             107) budget>=233.5 949   268.60790 3.397260 *
##          27) runtime>=121.5 2001   790.98840 3.440280
##            54) budget< 7500 1057   560.51360 3.327578 *
##            55) budget>=7500 944   202.01640 3.566472 *
##       7) Documentary>=0.5 2616  1127.53300 3.503345
##        14) runtime< 120.5 2447  1051.40400 3.481406 *
##        15) runtime>=120.5 169    57.89793 3.821006 *
```



Validation on Test Data We will evaluate these models by RMSE on test data respectively.

First, we directly calculate the rmse for random forest models. They are 0.726 and 0.723. Apparently, random forest 2 model has a lower rmse.

Next, we will use a plot to find the "elbow" for tree_based models, which means the error will improve very little even though the tree continues to grow at that point. This is a nice choice of model, which gives a trade-off between performance and complexity.

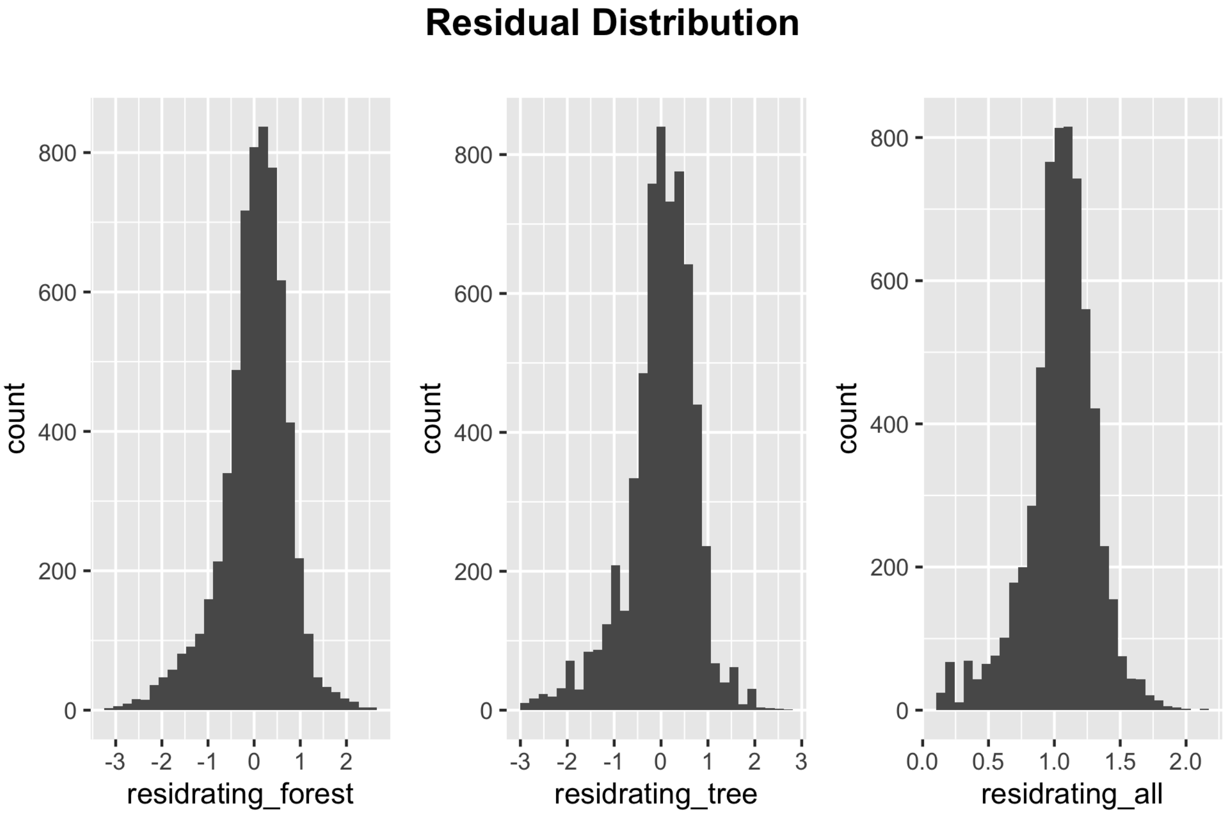


From the plot, we can tell that when cp equals 0.001, we get the lowest rmse on test data for tree_based models, which is exactly the second tree model which we show above with 0.733 rmse.

Finally, for the linear regression models, we have to mathematically calculate rmse step by step since we use lrating instead of mrating in the models.

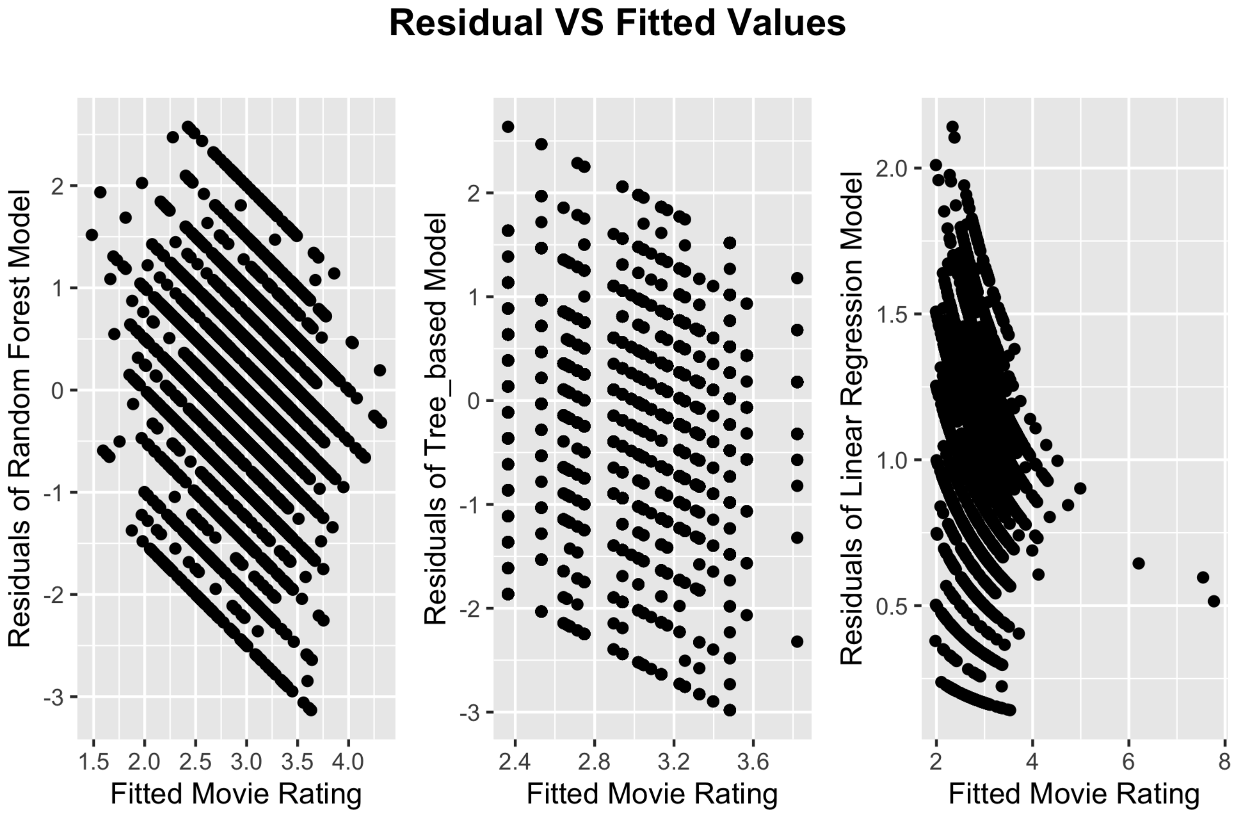
The rmse for each model are 0.806, 0.809, 0.807, 0.757, and 0.753 respectively. Clearly, the rmse of linear regression model with all the important variables are the lowest among all the linear regression models.

Now we continue to plot the residuals on the test data with the random forest model 2, the tree model 2, and the linear regression model with all the important variables, because these models have a relatively low rmse in each model category.



From the residual distribution, we can conclude that random forest model and tree_based model both have an approximate mean of 0. However, the mean of the linear regression model roughly equals to 1, which is not good. This verifies the frst one of our hypothesis that the linear regression model is not a good model given this dataset. Compared with the tree_based model, the random forest model performs better with the narrower standard deviation.

Next we proceed the predictions and residuals plot on the test data for the three models.



Apparently, the predictions and residuals plots show a similar result as before. The mean of residuals are about 0, 0, 1. However, these plots provide some additional information, which is as the ratings go up, the mean of the residuals go down, although the range of the residuals seem unchanged. That is to say, the models tend to overestimate the ratings when the ratings are low; while the models tend to underestimate the ratings when the ratings are high, which is also relatively reasonable in reality.

Conclusion

After calculation and visualization, we an easily tell that the random forest model with the independent variables containing runtime, budget, releaseYear, and genre performs best with the lowest rmse 0.723.

model	rmse
linear_model	0.7534179
forest_model	0.7232124
tree_model	0.7332849

From the importance table below, we can find runtime is the most important variable in predicting movie ratings, followed by Horror, Documentary and Drama in the genre. Variables like releaseYear and budget also play an important role to predict.

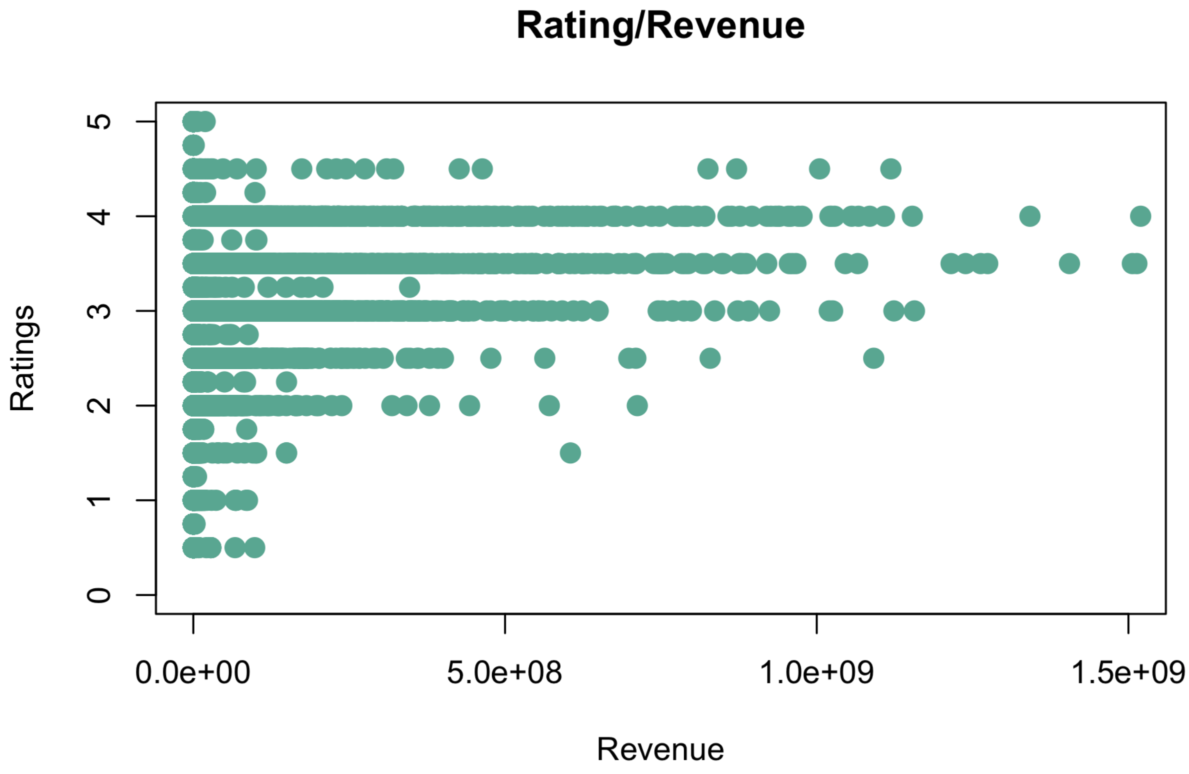
	%IncMSE
runtime	58.9271593
budget	29.8929048
releaseYear	45.9014203
TVMovie	5.5648870

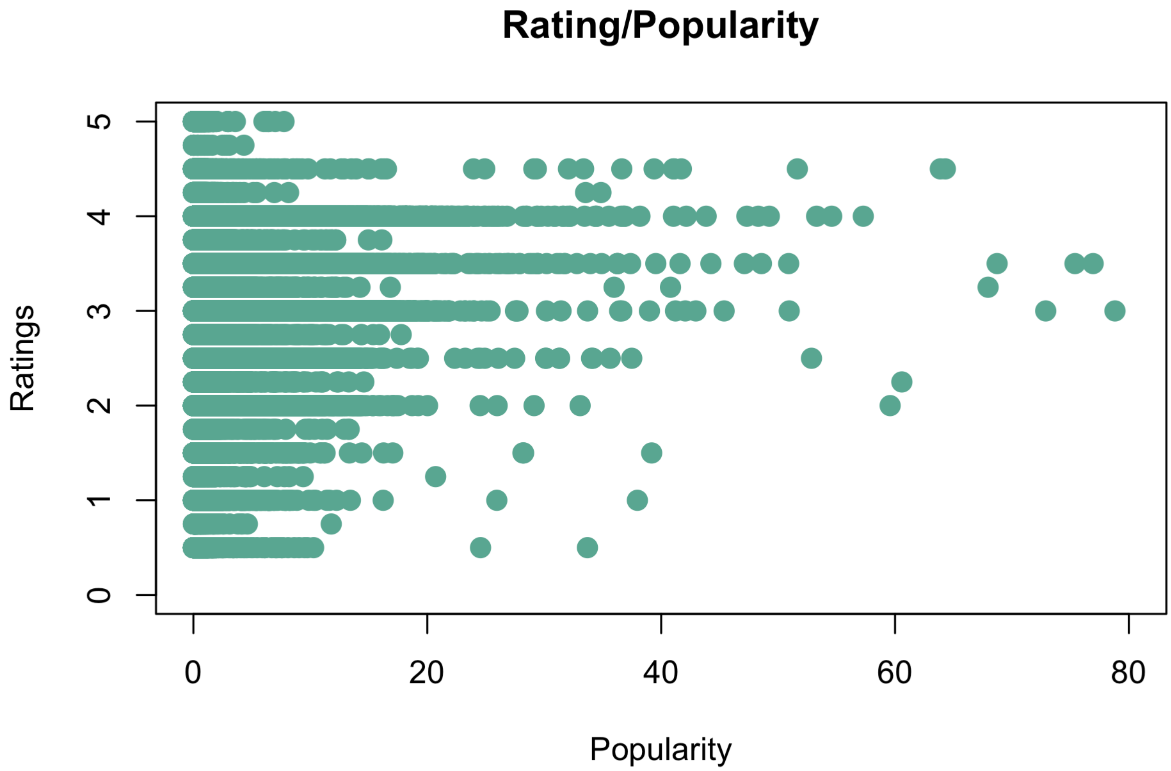
	%IncMSE
Western	13.7118194
Foreign	0.9399367
Documentary	51.4420053
Music	10.9770234
War	6.3720678
Mystery	13.9014132
ScienceFiction	17.1637523
History	5.4990320
Horror	50.4837133
Thriller	26.6106797
Crime	13.7190853
Action	25.2939337
Drama	38.2157824
Romance	12.2260529
Fantasy	11.8369290
Adventure	15.0867041
Family	14.2214324
Comedy	23.8501329
Animation	20.9825171

This makes sense as well in reality. However, the rmse for the best model is still relatively high given the movie ratings scale (0-5). Therefore, we may need to take other variables like title, overview, actors and so on into consideration in order to have a better performance of the model, which confirms our second hypothesis that our current variables are not adequate in predicting movie ratings effectively.

Further Research

We also created scatter plots to display the relationship between ratings on the Y-axis to revenue and popularity on the X-axis. The results of the initial plot displayed a strong positive correlation between rating and revenue. As the ratings of movies increases, the revenue of it tends to also increase. However, the second plot displayed a weak positive correlation between ratings and popularity. This indicates that while both variables tend to go up in response to one another, there is a lower likelihood of there being a relationship with the two variables.


Rating/Revenue
Ratings
0
1
2
3
4
5
0.0e+00
5.0e+08
1.0e+09
1.5e+09
Revenue


Rating/Popularity
Ratings
0
1
2
3
4
5
Popularity
0
20
40
60
80