Write-up for TMDB Box Office Prediction

--Can you predict a movie's worldwide box office revenue?

Author:: Mianchun Lu, Cijun Sun, Yuting Gong, Guangxu Luo

Executive Summary:

In this TMDB Box Office Prediction project, We aimed at predicting the revenue for movies using some attributes of movies. We tried several techniques including random forest, text mining, extreme gradient boosting, clustering, neural network, and SVM. Finally, we learned that xgboost with text mining is the best one.

We found that the most important variables that influence a movie's revenue are budget, popularity, number of casts & crews and sentiments in a movie overview description. In conclusion part, we recommended several ways for movie producers to generate higher revenue.

1. Statement of the problem or question(s) being addressed

Nowadays where movies made nearly $12 billion in 2018, the film industry is more popular than ever. But what movies make the most money at the box office? How much does a director matter? Or the budget? Or the genres? These are the questions we would like to answer.

For this project, we studied metadata on over 7,000 past films and applied EDA and predictive analysis. Some of our variables are numeric, and some are texts.

The goal of our project is to find out the important factors that would influence a movie's revenue, and build a predictive model to forecast a movie's revenue. The error measure we used is rmsle. Generally speaking, smaller rmsle means smaller error and better accuracy. With the help of our analysis, the production companies will be able to enhance a movie's revenue by making adjustment to relevant factors, such as release data, and control the risk of producing movies.

2. Reasons behind the choice of analytical technique(s). Explain why the technique is suitable for the question(s) being addressed.

In our analysis, we applied some analytical techniques we learnt in this course, such as text mining, clustering analysis and neural network. We also combined them with techniques we learnt before to better fit our model.

1) Text Mining (Sentiment analysis)

In our dataset, the overview of each movie conveys important information about a movie' content. We looked forward to finding out if the emotion in these movies, as well as the frequency of a word have impact on movies' revenue. For this purpose, sentiment analysis can be a good choice to help us discover and extract meaningful patterns and relationships from text. To be more specific, firstly, we counted the overview length in words and in sentences separately. Then, we used "nrc" lexicon to count the number of words that show certain sentiment, such as positive, negative, anger, etc., and tried "afinn" lexicon to scores the sentiment of each overview. Besides, we created and prepared a corpus which contained all the meaningful words in the overviews of train and test data, and found out the words appeared most frequently.

2) Random Forest

Random forests has two advantages. Firstly, reduction in over fitting: by averaging several trees, there is a significantly lower risk of over fitting. Secondly, less variance: by using multiple trees, we can reduce the chance of stumbling across a classifier that doesn't perform well because of the relationship between the train and test data. In our case, we have several numerical variables and have converted characters to several dummy variables. We thought random forest might be worthy of trying.

3) Advanced Decision Tree(Extreme Gradient Boosting)

A gradient boosting machine, much like a random forest, is a machine-learning techniques based on ensembling weak prediction models, generally decision trees. It can improve machine-learning model by iteratively training new models that specialize in addressing the weak points of the previous models. Applied to decision trees, the use of the gradient boosting techniques results in models that strictly outperform random forests most of the time, while having similar properties.[1] Based on these characteristics of gradient boosting, we decided to try this algorithm to see if it can better fit our data. We chose Extreme Gradient Boosting model for two reasons. Firstly, it is really fast when comparing with other gradient boosting models so that we were able to find a group of parameters that fit the data best in a limited time period. Secondly, it has great performance when comparing with others. We tried hundreds of groups of parameters with the help of grid search, and rmsle was the loss function we used to compare the performance of each model.

4) Clustering

Clustering is an unsupervised learning that groups data based on similarities. We saw an example in class where "clustering then predict" helps to improve model accuracy. We wanted to test it out and see if clustering movies into different groups based on similarities could help our model to predict revenue better. Specifically, we adopted K-means clustering method because it is one of the most useful and popular clustering methods. Then, we run both "Total within sum of squares Plot" and "Silhouette Plot" to decide which k (number of clusters) makes the most sense for our data. Both plots suggest a k=2 clusters would reduce Total within sum of squares and give the biggest drop in Silhouette width. Therefore, we decided to build a k-means clustering model for 2 clusters and then built 2 xgboost models and combine them to calculate the total rmsle. Before the k-means clustering, we noticed that our data had both categorical and numerical data. Clustering cannot handle both types of data.Therefore, we used dummyVars in caret package to dummy categorical

[1] Chollet, Francois. (2018). *Deep Learning with Python*. Shelter Island, NY: Manning Publications.

numerics and used fullRank=True to have one less variable than the number of categories present to avoid perfect collinearity.

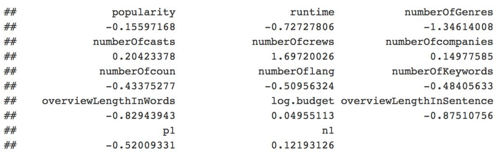
5) Neural Network

A neural network is a mathematical model that attempts to mimic the neurons found in the human brain. I tried two different library “nnet” and “keras”. The library nnet contains a basic neural network function, nnet(), which fist single-hidden-layer neural network, possibly with skip-layer connections. For the nnet model, here the following options have been chosen in fitting the net model:

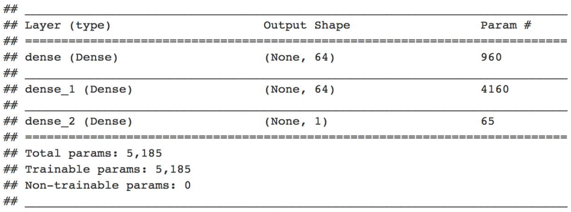
- 10 units for the hidden layer
- linear output (necessary for regression problems using nnet) (linout = T)
- use a neural network with skip layer units (skip = T)

Since these models are highly parameterized and thus will tend to overfit the training data. Cross-validation is therefore critical to make sure that the predictive performance of the neural network model is adequate.

For keras model, first we normalized the data (although the model might converge without feature normalization, it makes training more difficult, and it makes the resulting model more dependant on the choice of units used in the input) then we used a sequential model with two densely connected hidden layers, and an output layer that returns a single, continuous value.



```
##              popularity         runtime           numberOfGenres
##             -0.15597168     -0.72727806              -1.34614008
##            numberOfcasts    numberOfcrews        numberOfcompanies
##              0.20423378      1.69720026               0.14977585
##             numberOfcoun    numberOflang         numberOfKeywords
##             -0.43375277     -0.50956324              -0.48405633
##   overviewLengthInWords     log.budget overviewLengthInSentence
##             -0.82943943      0.04955113              -0.87510756
##                      p1              n1
##             -0.52009331      0.12193126
```



```
## ________________________________________________________________________
## Layer (type)                     Output Shape                  Param #
## ========================================================================
## dense (Dense)                    (None, 64)                    960
## ________________________________________________________________________
## dense_1 (Dense)                  (None, 64)                    4160
## ________________________________________________________________________
## dense_2 (Dense)                  (None, 1)                     65
## ========================================================================
## Total params: 5,185
## Trainable params: 5,185
## Non-trainable params: 0
## ________________________________________________________________________
```

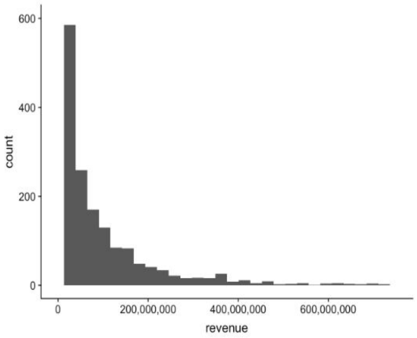
6) SVM

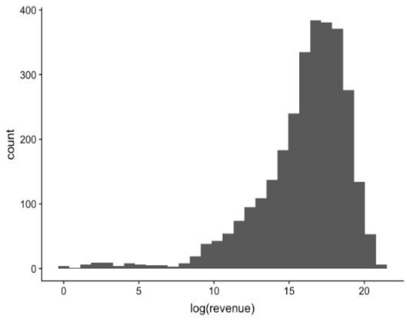
SVM is a supervised machine learning algorithm which can be used for classification or regression problems. It uses a technique called the kernel trick to transform your data and then based on these transformations it finds an optimal boundary between the possible outputs. In SVM regression, the input X is first mapped onto a m-dimensional feature space using some fixed (nonlinear) mapping, and then a linear model is constructed in this feature space. Often the data are assumed to be zero mean (this can be achieved by preprocessing), so the bias term is dropped.

3. Results from the analyses run

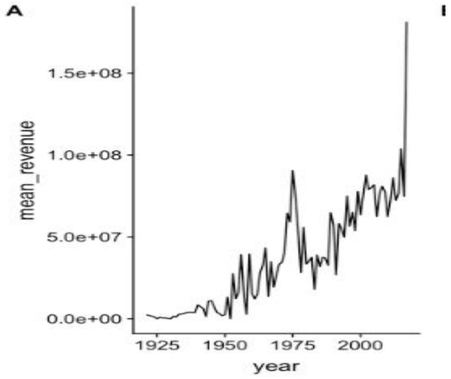
1) Exploratory Data Analysis Results

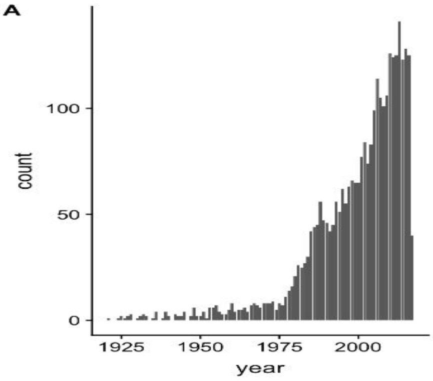
Revenue: Only a small number of movies has high revenue, whereas the majority doesn’t. The distribution of revenue is greatly positive skew. Due to the large range of revenue, we decided to do the log to revenue for a clearer distribution.



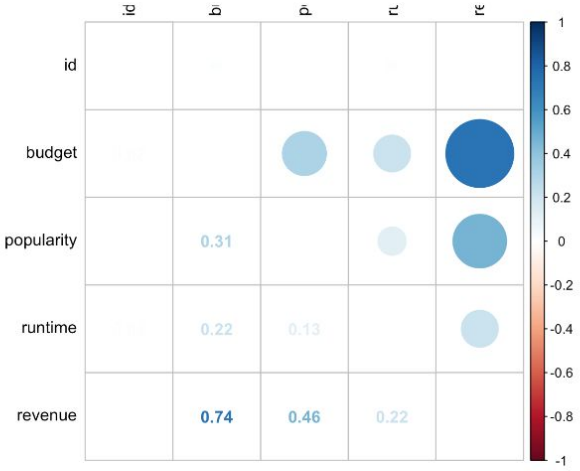


Year and revenue: Looking at the average revenue from 1921, we can find that the revenue for the film industry has been soaring. The industry is increasingly growing at a fast pace.



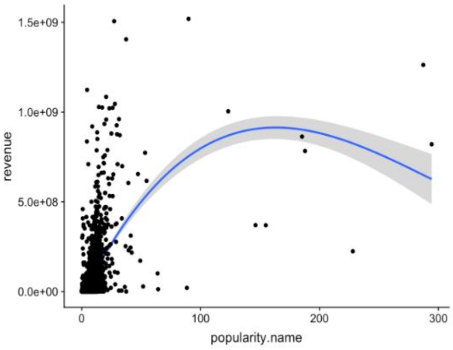


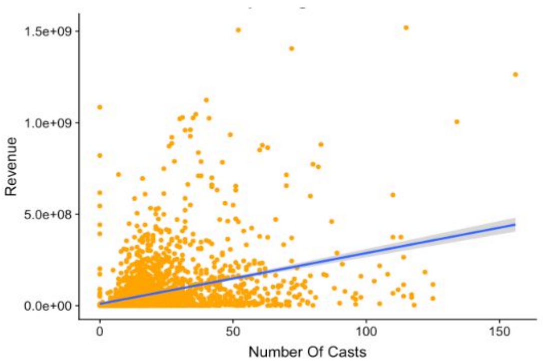
The following heatmap indicates that budget has the highest correlation with revenue. And popularity is the second one.



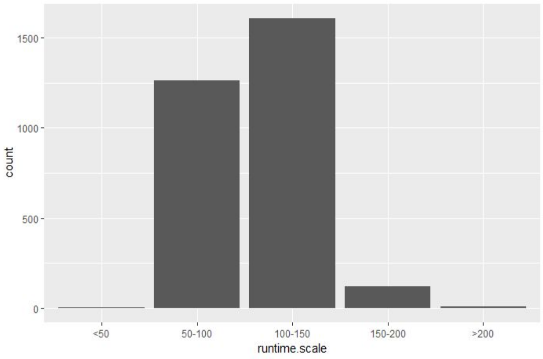
Popularity and revenue: Commonly speaking, the more popular the movie enjoys, the higher revenue it can make. While there is still some outliers, such as movies enjoy high popularity but have limited revenue. It might be the result of marketing or operation failure.

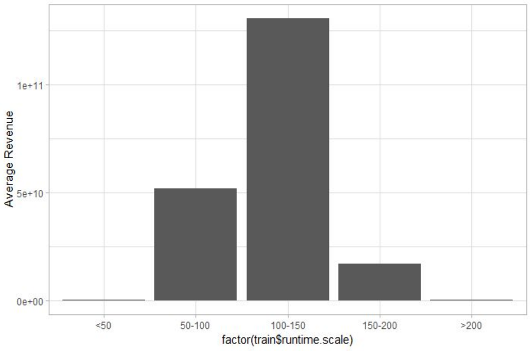
The number of casts and revenue: Normally speaking, the larger the cast, the more revenue the movie made.





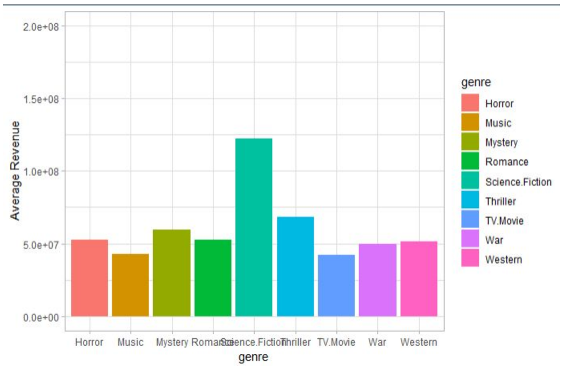
Runtime and revenue: We divided the runtime into five groups and found that the length of runtime also has great impact on the revenue. Too long(<50) and too short(>200) movies are not expected to have high revenue. Movies with runtime between 100 minutes and 150 minutes may be the most attractive choice to audience. The left chart is the distribution of movies in these five groups, and the right chart is the average revenue in these five groups.

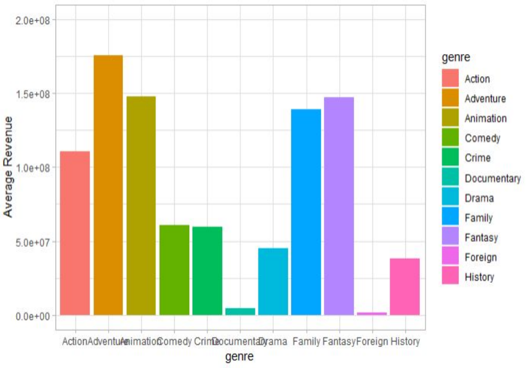


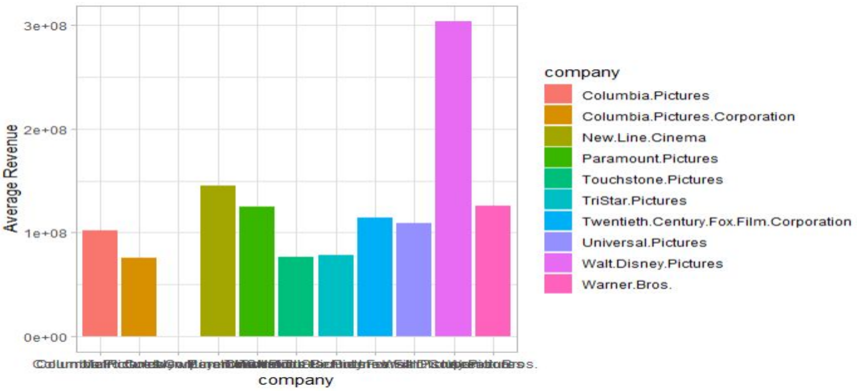


Genres/producing company and revenue: The most popular genres of movies are Drama, Comedy, Thriller, and Action. The top three movie producing companies are Warner Bros, Universal Pictures, and Paramount Pictures.

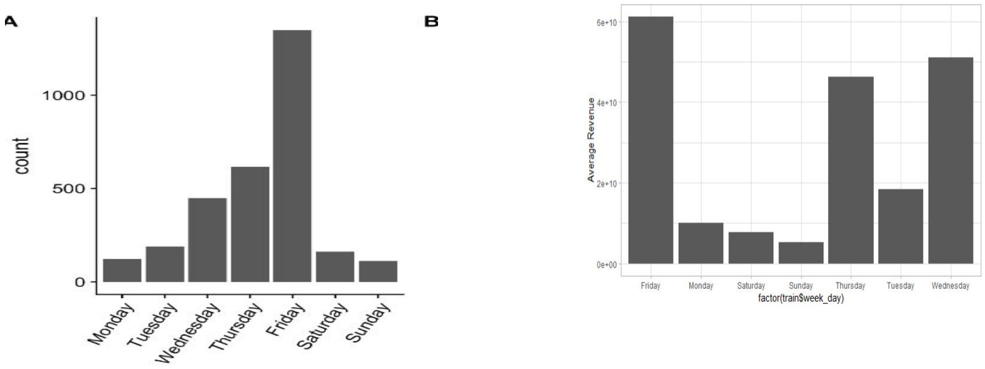
For genres, even though Adventure, Animation, Family, and Fantasy are not the most popular genres, their revenue is much higher than other types(first and second pictures). For producing companies, Walt Disney has much higher revenue than others (third one).

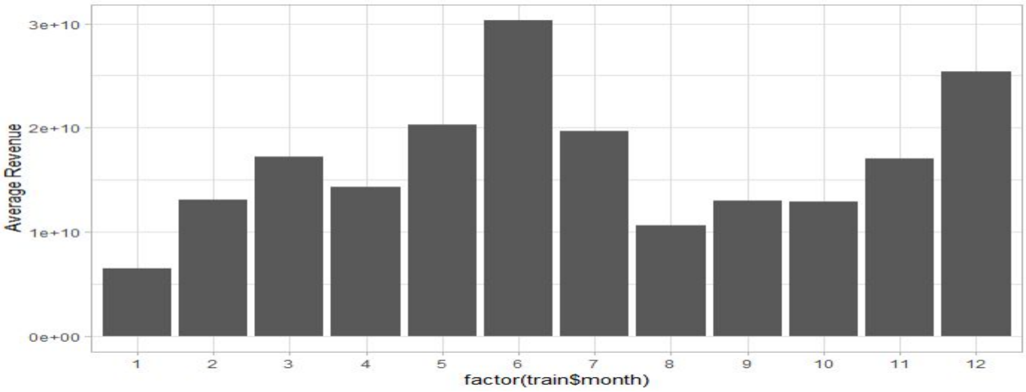






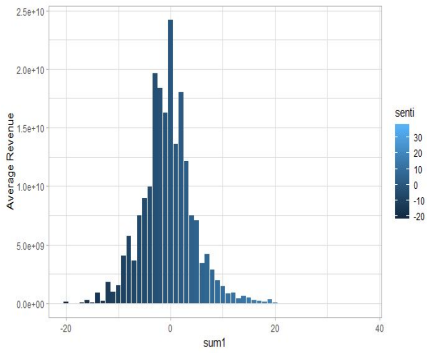
Release date and revenue: After counting the amount of movies released on each week day, we found that most movies are released on Fridays. The release date has great relationship with revenue: Movies released in June and December have higher revenue than others; Movies released on Wednesday, Thursday and Friday have higher revenue than others.

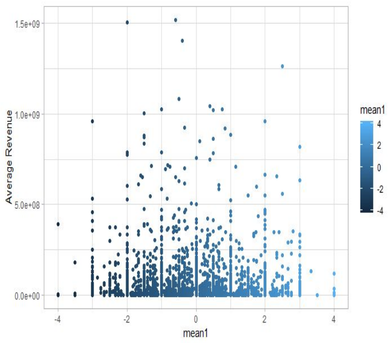


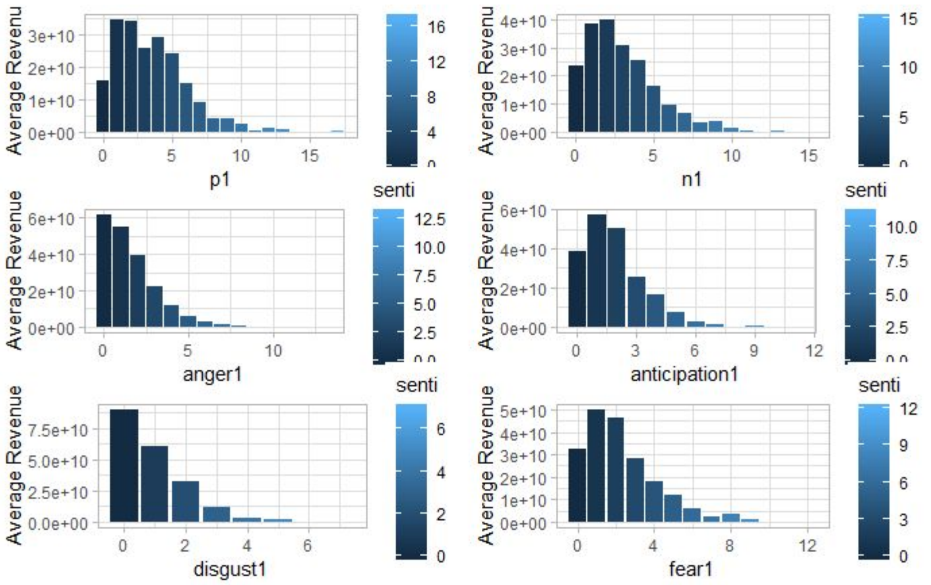


Overview and revenue:

The first two charts shows the relationship between revenue and the sentiment score of the overview. The last six charts shows the relationship between the revenue and the number of words with certain emotion. From these bar charts, a conclusion can be got that the overviews of movies with high revenue tend to include less sentimental words. The emotion of their overview is relatively neutral.







2) Predictive Analysis

Result and what we learned from xgboosting:

In order to fit the xgb model, we converted categorical variables to numeric variables before we started. After that, we tried to fit the xgb model with arbitrary parameters. Then, in order to find out the relatively best parameters, we try the grid search. But the results of grid search are not better than before. We think there might be some over-fitting problems.

There are two reasons we considered to the overfitting problem, the irrelevant sentiment related variables and the redundant trees in the model. For the first reason, we tried to drop some of sentiment related variables to see if the accuracy of the model could be better. The results showed that adding these variables have positive influence on model's accuracy. So in the final model, these variables will be kept. As for the second reason, we reduced the number of rounds(the number of trees) in the model, and our results became better, which means that the overfitting problem partially caused by the number of rounds.

Another way to enhance the accuracy of the xgboost model is to convert categorical variables to separated dummy variables, instead of numeric variables. Therefore, more features of the categorical variables can be saved in the model.

Result and what we learned from clustering:

After the clustering, we predicted and combined the result. We reduced the RMSLE to 2.235. However, we compared the "clustering then predict" total RMSLE with "no clustering" result, and found the "no clustering" performed better! Therefore, it is not the clustering that improved our model, but the dummy we did help the model.

From this exercise, we learned that dummy categorical variables are useful. Originally, we used as.numeric to change our categorical variables to numeric variables. However, dummy is more accurate than simply treating them as numerics. Treating them as numeric make them look like there is a linear relationship between different factors while there is no this kind relationship between them at all. Using dummy can better represent categorical variables.

We also reflect on why clustering does not help our model. Movies probably are more complicated that they should be grouped into more than 2 categories. Grouping them into 2 groups is not enough, but grouping them into more groups will reduce our train dataset and leads to low accuracy. As a result, we decided not to adopt clustering in our final model.

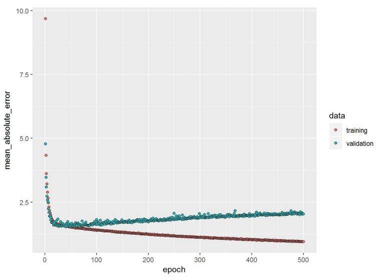
Result and what we learned from Random Forest:

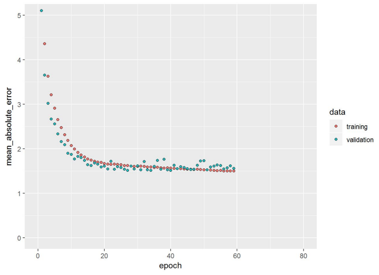
We found that random forest is not a good model in predicting the movie revenue. The result on kaggle is 2.83. We think it is because in our data, there is not many real numerical variable. The majority variables are the binary variable that was converted from categorical variables.

On the other hand, these converted numerical variables are not closely correlated with the investment of the movie. The majority of them originated from the script and overview, which fall into the category of movie design. That's why it cannot precisely predict the revenue of the movie.

Result and what we learned from Neural Network:

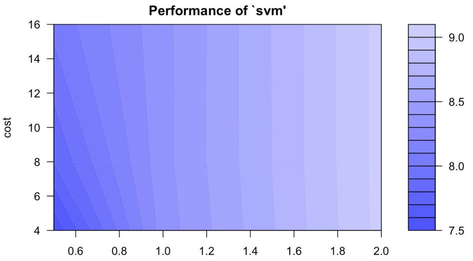
When we perform package nnet, we found out that nnet is not giving satisfying result. The result on kaggle is 2.44. Neural nets are not good models for sparse data. In sparse data, that terrain is basically flat, so as the model steps through your data trying to find optimal parameters, it is essentially wandering randomly to find the hidden nodes to reach the local optimal. we need to figure out how to transform your data to make it something more neural net compatible. However, when we tried to scale the data or using log transformation, due to the sparse variables from text analytics, there is still no improvements. Another way to transform the data is by conducting PCA. Regarding Keras, it gives us better result on Kaggle, which is 2.21.

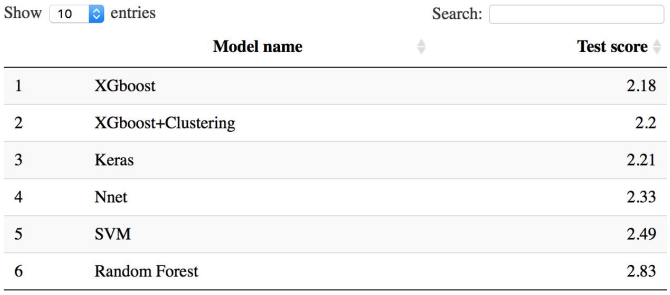




Result and what we learned from SVM:

I used "tune.svm" to tune the svm parameters using grid search method to improve model accuracy.

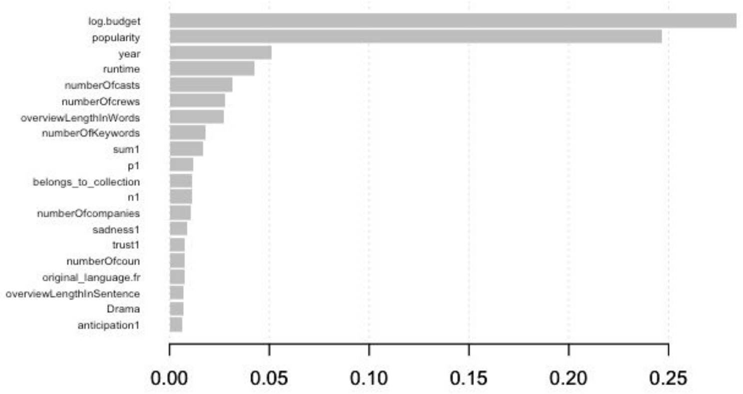




	Model name	Test score
1	XGboost	2.18
2	XGboost+Clustering	2.2
3	Keras	2.21
4	Nnet	2.33
5	SVM	2.49
6	Random Forest	2.83

4. Discuss the conclusions from the analysis and offer recommendations in a form that is simple and understandable to decision makers. Support conclusions with relevant charts.

Our final model used both Text Mining and xgboost. It has the lowest error rate with rmsle equals to 2.18. We run a variable importance plot based on the xgboost model as shown below to understand what factors influence movie revenue.



Based on our final xgboost model variable importance plot, below are some of our findings and recommendations.

Findings and Recommendations for movie producers:

- A few of the most important variables for predicting a movie's revenue are: **budget**, **number of casts** and **number of crews**. These variables suggest that a movie with bigger budget is more likely to have a higher revenue as number of casts and number of crews all indicate size of the budget.
- The second most important variable for predicting a movie's revenue is **popularity**. This could be meaningful for movie producers with small budget as it indicates that even if a movie doesn't have a high production budget, it can achieve high revenue if the movie has a high popularity before it is released. Therefore, we recommend movie producers with low budget to wisely allocate their resources to marketing to increase popularity and exposure among consumers. They can even leverage low-cast or even free marketing tools such as social media to increase movie publicity.
- We also see that **Overview Length** and its sentiment such as **p1** (the number of positive words) is another important indicator for movie revenue. We suggest movie producers to write detailed overview with proper amount of positive words(0-6) to attract consumers to movies.
- As for **Runtime**, too long(>200 min) or too short(<50 min) movies are not attractive to audience. The best runtime of a movie is between 100 minutes and 150 minutes.
- **Belong_to_collections:** Movies belongs to a collection are more likely to have higher revenue because they already have a large audience base.
- **Drama:** drama is the most common type of movies. So, it has great influence on the model results, even though it is not the most profitable movie genre.

Insights and Recommendations for investors:

- As we can see from the plot, the third most important variable that indicates movie revenue is **year**. As the year gets closer and closer to nowadays, the movie revenue goes higher and higher. This suggests that the movie industry is getting more and more popular and profitable. People loves watching movies as entertainment. We suggest investors to continue to invest in movies industry to support better movie production.
- **Budget** is the most important factor that can influence revenue. However, for those low-budget movies, a good overview, proper length of runtime, as well as the marketing campaigns before the movie is on can all help them to improve their revenue.