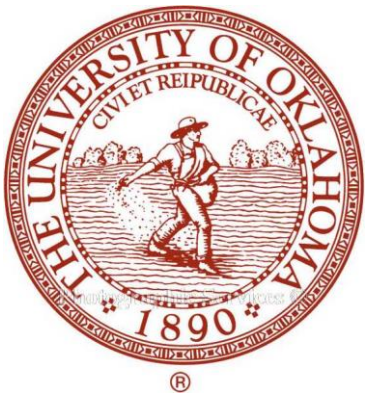
# REGRESSION OF USER MOVIE RATING- IMDb

**SAI TEJA KANNEGANTI**
**SARVESH PRATTIPATI**




**UNIVERSITY OF OKLAHOMA**

## EXECUTIVE SUMMARY

### Problem Statement

The goal of this project is to make a model which, for a movie, predicts the rating of a movie based on the existing data from IMDB.com (an online database of information related to movies). In this project, efforts are put in to predict the rating of a movie and the greatness associated with it.

### Major Concerns

The rating of a movie can't be predicted with a set of predetermined factors. The factors associated with the rating of a movie change from one to other. In this dataset, the predictors are some common means of portraying the fondness towards a movie. However, there may be some movies, which, in spite of high expectations, fail miserably at the box office.

From the data point of view, the dataset has the majority of categorical variables and very few of them are numerical variables. Also, there are plenty of missing values for which imputations or transformations need to be done to improve the analysis. One of the major concern is some of the predictors having more than 2500 factor levels.

Apart from that, Data Aggregation and Selection strategy need to be given proper weight in order to effectively evaluate the prediction analysis.

### Summary of Findings

Multiple data mining models are performed on this dataset: Linear regression, Elastic net, Decision tree, random forest, gradient tree boosting and Neural networks. To sum up, out of these models, Random forest gives the best results by far, with the test metrics lower than every other model. This is followed by gradient boosting, linear regression and elastic net.

Also, the attributes that highly affect the IMDB score are num_voted_users, duration, budget, num_users_for_reviews, gross.

### Recommendations

- Imputing categorical variables like director name, actor1 degrade the quality of dataset. So, we have done the modeling considering the complete cases.
- Many of the categorical variables in the dataset have a high number of factors, so some of the best modeling techniques like Random Forest fails on this dataset & bringing more than 2500 factor levels to an optimum number of factors deteriorate the meaning of the variable. So, we have considered numerical variables for modeling.

# 1. Project Understanding

There are different ways by which greatness of a movie can be determined. Some people rely on critics and some use their own instincts. Another way of judging a movie is by its IMDb score. In this project we analyze a dataset having information of more than 5000 movies released in various languages from 1916 to 2016. The overall goal of the project is to predict the IMDb score of the movie that is released based on some of the predictors like movie likes, the number of critics etc. Later, Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) are used as metrics for evaluating the execution of data mining models in this project.

# 2. Data Understanding

## 2.1. Data Description

For this project, the dataset is sourced by kaggle.com, extracted from a website imdb.com (Internet movie database). The dataset contains details of movies over a period of 100 years (1916-2016) in 66 countries. There are around 2400 unique director names and thousands of actors. Several other variables include budget, gross, the number of likes etc.

It consists of 5043 rows of data for 28 predictor variables out of which 12 are categorical variables and 16 are numerical variables. The description of variables is given below.

## 2.2. Data Visualization

Visualization of data is the first step in any modeling problem. In order to understand the hidden patterns in the data, the dependency between different predictor variables, visualizations are a great rescue.

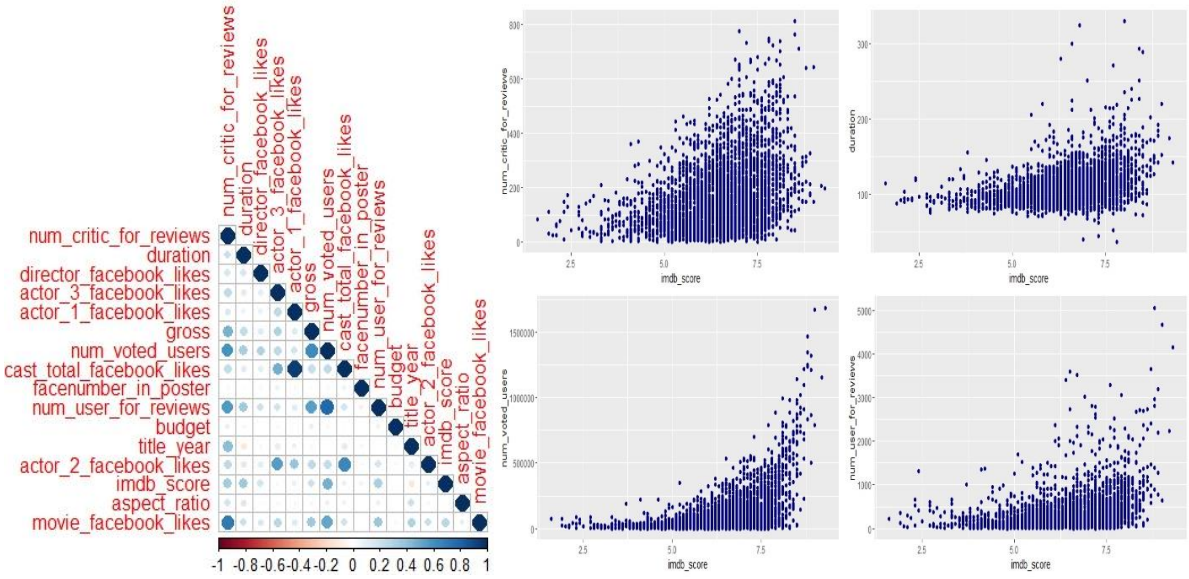



*Figure 1 Correlation plot of variables and distribution of attributes*

Figure 1 shows the correlation plot between all the numerical variables for the dataset. We observe that the pairs (cast_total_facebook_likes, actor_1_facebook_likes),

(num_users_for_reviews, num_voted_users) and (movie_facebook_likes, num_critics_for_reviews) are highly correlated. Point to be noted is that correlation takes into consideration, only the linear relationship between the variables. There may also exist nonlinear relationships between several other variables too.

# 3. Data Preparation

## 3.1. Data Pre-processing

a. Missing Values: This dataset has as many missing values as about 20%. We tried to impute the values using "mice". Since the categorical variables are proper nouns (names of persons), imputation doesn't yield meaningful results. Duplicating the genre, and many other variables don't convey meaningful information. So, considering this fact, out of the 5043 rows, we have decided to consider the 3800 complete cases for our analysis.

b. Outliers: After preliminary analysis of the dataset, we have found out that the numerical variables which we are considering have no limits. Based on the movie or persons, the figures can reach extremes. So, we didn't treat the majority of the data for outliers. Only the predictor facenumber_in_poster has an outlier with value 43 we felt that 43 faces in a poster are highly unrealistic.

c. Transformation: Raw data, upon exploration, shows that the data is highly skewed, the majority being right skewed. For these variables, we have performed logarithmic transformation of variables and the distributions appeared to be normal upon transformation.

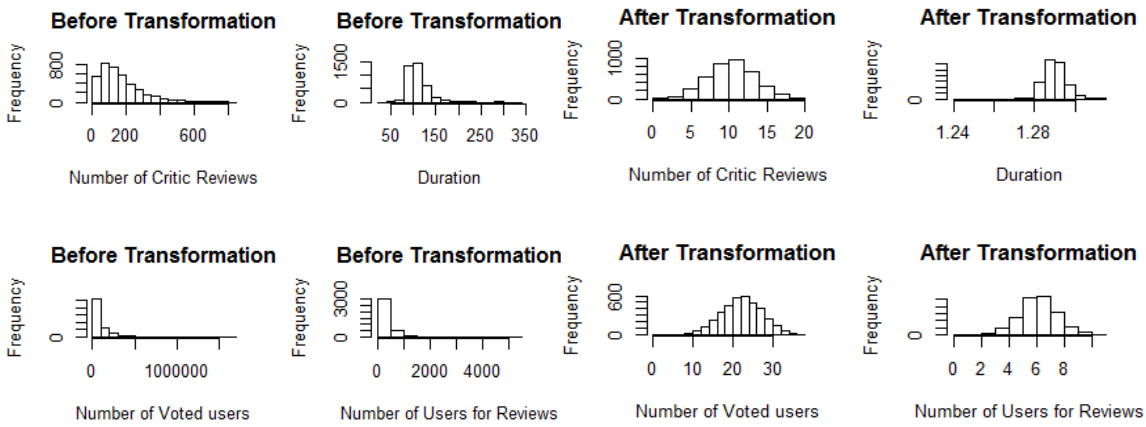



*Figure 2 Distribution of attributes before and after transformation*

d. Data Quality: The budget variable in the dataset, has varying currencies instead of Dollar. Like, Korea and Japan countries have the budget of the movie in their own currency. This value is converted to Dollar to improve data quality in the budget variable.

## 3.2. Data Splitting

From the source of this dataset, there is no specific test dataset to check the model. As a rule of thumb, we have split the data into 80:20 ratio for train and test datasets. Prior to splitting, we sampled the data randomly so that each movie has equal weight of being included in the train or test dataset.

# 4. Modeling

## 4.1 Brief Explanation of Modelling Algorithms:

### 4.1.1 Linear Regression Models

a. Simple Linear Regression: For investigating the relationship between the response variable, IMDB Score, and other independent variables, Collinearity (removed color variable) and the normal distribution is taken care. Variables with 3 and 2 star significant (p-value<0.05) are used in formulae. The variable selection is done based on the significance level (p-value).

b. Elastic Net: In order to take care of any multicollinearity we choose this model so that lambda (shrinkage parameter) solves multicollinearity. Cross-validation approach technique is implemented, which uses the grid search to find the optimal alpha and lambda hyper-parameter for this data set. Finally, optimal parameter values are used to perform regression. The regression formulae used is same as the formulae used in the Simple linear regression. Tuning parameters are provided in Appendix A.

### 4.1.2 Tree Methods

c. Decision Trees: Unlike linear methods, trees map non-linear relations quite well. This dataset uses continuous variable decision trees (regression problems). Decision trees also help in variable importance. Decision trees have low bias and high variance, so this might cause over-fitting from the model, we calculated the complexity parameter, cp which turned out to be 0.01 from the minimum x error (x error is the cross-validation error).

d. Random forests: This is one of the bagging method which gives the regression estimate and combats overfitting (high variance). Also, it also provides with important variables. It does prediction by taking the average of outputs of different trees. Initially, the model is executed with 10000 trees for which we calculated the minimum MSE value and figured out 285 trees to reach minimum error estimate.

e. Gradient boosting method (GBM): It is an accurate and effective procedure used for both classification and regression. It combines the outputs from weak learners and creates a strong learner, thus improving the prediction power of the model. It is used to impart additional boost to accuracy.

#### 4.1.3 KNN

In this K-closest observation are defined as the ones with the smallest Euclidean distance to the data point under consideration. For prediction, in regression, it takes the weighted average of the values of the K-neighbours.

#### 4.1.4 Neural Networks

An artificial Neural Network is a non-linear regression technique designed based on the biological working of the nervous system. For this dataset, we have used the neural net package to train the neural networks using back propagation. We have enabled the linear output by switching off classification output since we have to predict the numerical output. Data used is scaled & centered. A number of hidden layers given are 5 and 3 and the fit formula used is same as that of Linear Regression.

#### 4.1.5. SVM

It is a supervised algorithm used for both classification and regression. It has kernels, which are used in non-linear separation. So, it basically does some complex transformation and finds out a process to separate the data. Involves gamma (which makes the data fit exactly, can cause overfitting) and error term C, which gives a proper trade-off between smooth decision and classifying the training points correctly.

In all, the optimal parameters are estimated based on cross-validation score to find the effective combination of these to avoid over-fitting.

### 4.2 Plan of Analysis

1) Modeling the data using the above-mentioned techniques. Models are evaluated with RMSE and MAE as the evaluation metrics.
2) Choose the best models to combine them in the stacking algorithm.
3) Using the predictions from these best models as inputs, and actual responses as output, we would train a high-level learner using a linear regression ensemble (caretStack).
4) The high-level learner would be the final model, which will be tuned further for better accuracy and performance.

### 4.3 Model Selection

1) Since IMDB Rating is a numeric and continuous variable, primarily, we will be using RMSE (Root mean square error) as the main evaluation metric. As we need to pay attention to the largest residuals than the small residuals, RMSE metric is used as a primary evaluation parameter.
2) Secondarily, we will be using MAE (mean absolute error) as the second reference to evaluation metric. This metric tells us on average, the magnitude of error for a particular model and it is easy to interpret.

3) Based on the execution time.

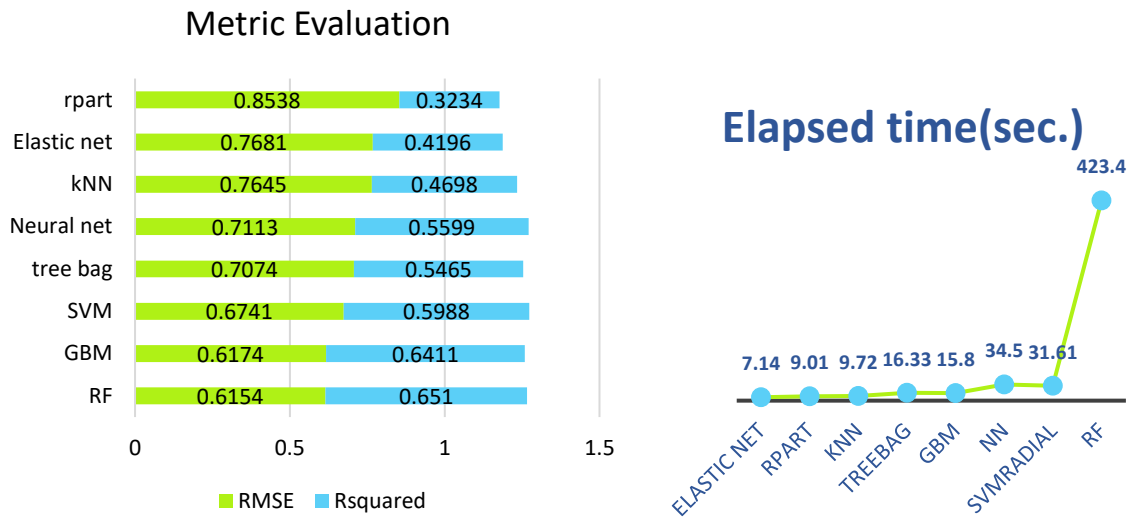



*Figure 3 RMSE and elapsed execution time for all the models*
*Right: Elapsed Time Left: Metric Evaluation*

From the above table & Graph, it is clear that RF, GBM, and SVM have done better in terms of RMSE and MAE. Now these three models will be selected to implement in stacking algorithm.

Now we will explore the space of different models for this problem, i.e, ensemble multiple regression models. Stacking is implemented because the combination of models would do better in performance than compared to individual methods. The selected three (RF, GBM, and SVM) learners build an intermediate prediction, like one prediction for each learner. Then a final model is added to learn from the intermediate predictions. This final model is stacked on top of these learners to give better performance.

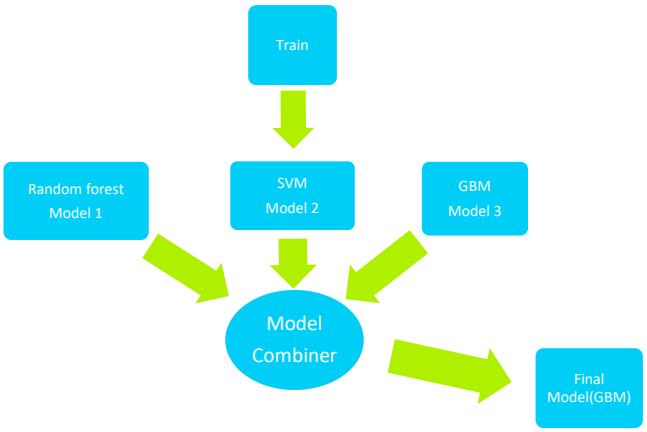



*Figure 4 Ensemble stacking method*

Here we have stacked as final model individually all the three methods to observe which performs well.

The conclusion of model selection:

1) Selected the best top three methods.
2) Used these three methods/learners predicted outputs to train the final classifier.
3) Used RF, GBM, and SVM as final classifiers, to test the performance.

**4.4 Final Model**

To select the best final model, all the three models have been stacked to check the performance. Below tables show the results.

| **Before Stacking** | **GBM** | **RF** | **SVM** |
|---|---|---|---|
| RMSE | 0.7051026 | 0.6933121 | 0.7450903 |
| MAE | 0.5558759 | 0.5759998 | 0.5102352 |

| **After Stacking** | **GBM** | **RF** | **SVM** |
|---|---|---|---|
| RMSE | 0.6741655 | 0.6858708 | 0.6757145 |
| MAE | 0.5926593 | 0.5799645 | 0.5932901 |

After stacking, SVM has improved. Likewise, GBM and RF also did show improvements. Clearly, GBM has better reduced RMSE, which is so it is considered as the final model. Going further, the GBM is tuned and again added to the stack. Tuning parameters for GBM are mentioned in Appendix A. Now the GBM model is added to the stack and the final prediction RMSE and MAE results are shown below.

| **After Tuning and stacking** | **GBM** |
|---|---|
| **RMSE** | 0.651043 |
| **MAE** | 0.4910735 |

Comparing both the above tables, after stacking and after tuning & stacking, there is an improvement in reduction of RMSE and MAE.

The conclusion of final Model:

1) caretStack is run using all the three selected models(RF, GBM, SVM), to check the performance.

2) Results showed GBM has better improvement in RMSE reduction.
3) Further, GBM is tuned and stacked again to obtain lowest RMSE of 0.65, the lowest RMSE for all the models for this data set.

# 5. Results and Discussion

## 5.1. Results

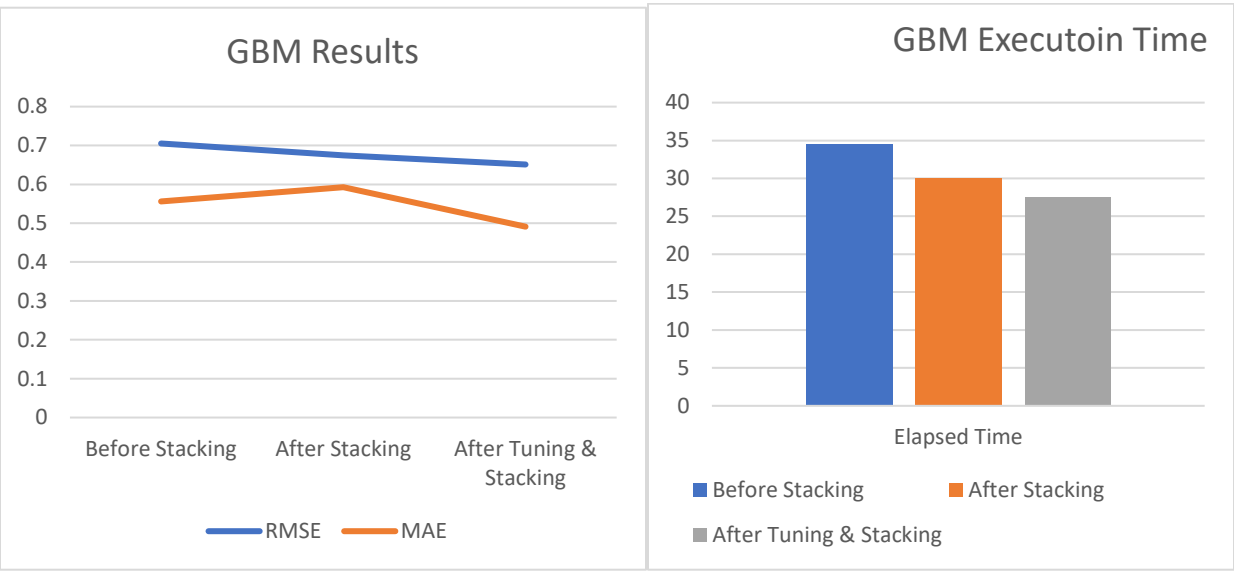
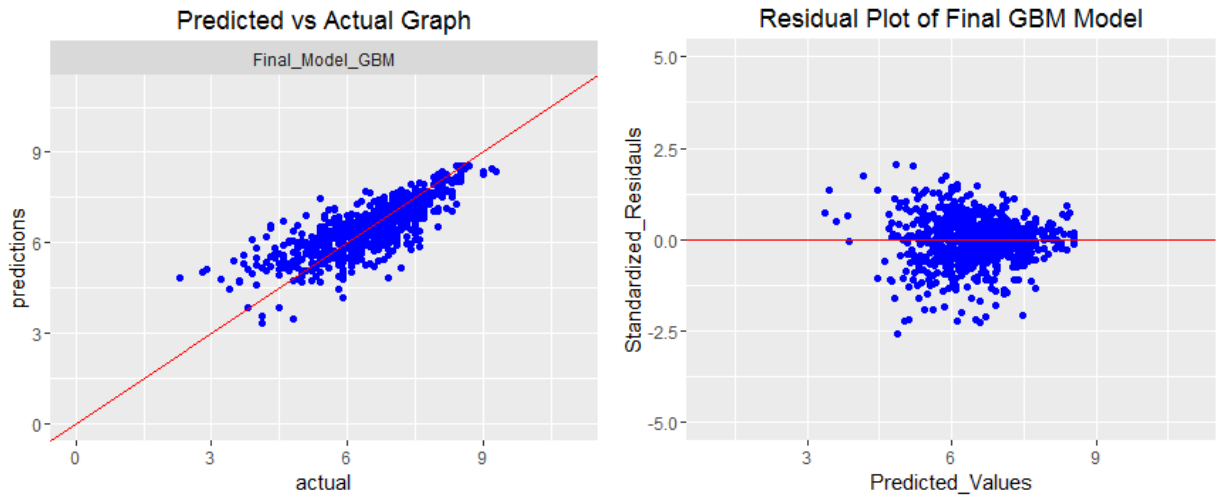



*Figure 5: Left: GBM Results Right: GBM Execution Time*

Clearly, GBM has shown improvements in mean squared error and in the reduction of execution time, which makes it reliable and final model for this dataset.
As GBM showed to be the suitable model for this dataset, the predictions of movie ratings are calculated and plotted in the below graph.




*Final model predicted rating graph (GBM)*
*Right: Standardized Residuals  Left: Predicted Vs Actual*

The final GBM model has done a good job in terms of prediction, as we can see many of the points lie close to the linear line. Also, the many of the residuals are close to the red line. This indicates good prediction.

### 5.2. Variable Importance

The importance of variables can be selected from the above models, like, from correlation, Random Forest, Decision Tree and from variables selected from significance level in linear Regression formulae.

From Linear regression, selections are made based on the significance level(p-value) and are listed in a random order. They include num_critic_for_reviews, duration, director_facebook_likes, gross, num_voted_users, facenumber_in_poster, num_user_for_reviews, budget, title_year, movie_facebook_likes.

Also, from decision trees and random forests we get the order of variables according to weights.

The top 5 important variables from Random Forest, SVM and Gradient Boosting that influence the rating are:

1) Number of Voted users
2) Budget
3) Number of Critics for review
4) Number of users for review
5) Duration

## 6. Conclusion

The stacking ensemble method has helped GBM to reduce the mean square error and as a result providing a good prediction of movie rating.

The movie ratings have been predicted and the important variables for the movie rating prediction also have been listed above.

So if the new user rates a movie, the rating he would be giving, would be influenced by Number of already voted users, the budget of the movie, Number of critics reviewed, the number of users voted and the duration of the movie.

## 7. Appendix

Appendix A:

Variables in the dataset

*Table 1 Attributes of Dataset*

| Name | Description |
|---|---|
| Color | The color of the movie whether color or black and white. |
| director_name | Name of the director who directed the movie. |
| num_critic_for_reviews | A number of critics who reviewed the movie. |
| Duration | The length of the movie in minutes. |
| director_facebook_likes | Total count of facebook likes for the director. |
| actor_3_facebook_likes | Total count of facebook likes for actor 3. |
| actor_2_name | Name of actor 2 in the movie. |
| actor_1_facebook_likes | Total count of facebook likes for actor 1. |
| Gross | Gross earnings of the movie. |
| Genres | Category of the movie. |
| actor_1_name | Name of the lead actor in the movie. |
| movie_title | Name of the movie. |
| num_voted_users | Total count of users who voted for the movie. |
| cast_total_facebook_likes | The sum of facebook likes of all the cast of the movie. |
| actor_3_name | Name of actor 3 in the movie. |
| facenumber_in_poster | A number of faces in the poster. |
| plot_keywords | Keywords related to the plot of the movie. |
| movie_imdb_link | Link to the IMDB web page. |
| num_user_for_reviews | A number of users who reviewed the movie. |
| Language | Language in which the movie is made. |
| Country | A country where movie belongs to. |
| content_rating | A rating which specifies age group the movie relates to. |
| Budget | The money put in to complete the movie. |
| title_year | The year in which movie is released to the audience. |

| | |
|---|---|
| actor_2_facebook_likes | Total count of facebook likes for actor 2. |
| imdb_score | IMDB score for the movie. (response variable) |
| aspect_ratio | The ratio of width to height of the movie screen. |
| movie_facebook_likes | Total count of facebook likes for the movie. |

GBM Tuning Parameters

n.trees ---- 150,
interaction.depth ---- 3
shrinkage ---- 0.1
n.minobsinnode ---- 10

Elastic Net

Alpha ---- 0.1111
Lambda ---- 0.000914