Machine Learning Project Report:
IMDB Movie Score Prediction

Liuzhao “Carlos” Tang

ABSTRACT

The movie’s rating score is the public's overall evaluation of the movie. Data properties and related social network data have been trusted as important factors in the evaluation process. However, the mechanisms are still a black box. This project tried to explore the mechanism with machine learning techniques and it will provide some new understandings of estimating movie quality and insights for film distributors. This study built 4 basic models(lasso regression model, support vector machine model, random forest model, and extreme gradient boosting model) and 2 ensemble models(blending ensemble model and stacking ensemble model) based on almost 5000 movies properties data and social network data to predict movie’s IMDB rating score. The study result shows blending ensemble model is the best model used to predict a movie’s rating score.

Introduction

It is difficult to explain what is a good movie but there is no doubt that the rating score of a movie is an important indicator of this question. IMDB users often make movie-watching plans and decide which movie to watch. The goal of the project is to predict the IMDA score of a movie based on its properties and social network data with machine learning techniques. This project provided a sight of estimating movie quality for industry analysts and insights for film distributors.

Task Definition

The task of the project is to predict a movie's rating score on IMDB.com based on its properties and related social data. The dependent variable of the model is the rating score that was calculated based on users' ratings with algorithms.

The independent variables have two parts: properties variables and social network variables. Properties variables are attributes of a movie such as an office box performance, duration, budget, years, aspect_ratio, content rating, casts, and so on. The social network variables are social network indicators related to a movie like Facebook like and the number of reviews.

Models will be evaluated on MSE.

Data and Variables

The raw data came from an online dataset. It had 5043 rows and 28 columns(15 numerical variables, 12 categorical variables, and 1 target variable).

Table 1 shows the descriptive statistics of numeric variables. It is clear that some variables were measured on different scales and missing values existed in some variables. The two problems would be addressed in the data pre-processing part.

Table 2 presents the missing values table. 21 variables contained missing values. The variable 'gross' had 863 missing values and it was the variable with the most missing value.

Table 1: Numeric Variable Descriptive Statistics

Column name	count	mean	std	max
num_critic_for_reviews	4993	140.1943	121.6017	813
duration	5028	107.2011	25.19744	511
director_facebook_likes	4939	686.5092	2813.329	23000
actor_3_facebook_likes	5020	645.0098	1665.042	23000
actor_1_facebook_likes	5036	6560.047	15020.76	640000
gross	4159	48468408	68452990	7.61E+08
num_voted_users	5043	83668.16	138485.3	1689764
cast_total_facebook_likes	5043	9699.064	18163.8	656730
facenumber_in_poster	5030	1.371173	2.013576	43
num_user_for_reviews	5022	272.7708	377.9829	5060
budget	4551	39752620	2.06E+08	1.22E+10
title_year	4935	2002.471	12.4746	2016
actor_2_facebook_likes	5030	1651.754	4042.439	137000
imdb_score	5043	6.442138	1.125116	9.5
aspect_ratio	4714	2.220403	1.385113	16
movie_facebook_likes	5043	7525.965	19320.45	349000

Table 2: Missing Value Table

Column_Name	Frequency	Percentage
gross	863	17.5442163
budget	485	9.859727587
aspect_ratio	326	6.627363285
content_rating	301	6.119129904
plot_keywords	152	3.090058955
title_year	106	2.154909534
director_name	102	2.073592194
director_facebook_likes	102	2.073592194
num_critic_for_reviews	49	0.996137426
actor_3_name	23	0.46757471
actor_3_facebook_likes	23	0.46757471
num_user_for_reviews	21	0.42691604
color	19	0.386257369
duration	15	0.304940028
facenumber_in_poster	13	0.264281358
actor_2_name	13	0.264281358
actor_2_facebook_likes	13	0.264281358
language	12	0.243952023

actor_1_name	7	0.142305347
actor_1_facebook_likes	7	0.142305347
country	5	0.101646676

Table 3 demonstrates the number of unique values for categorical variables. Only "color" is a binary categorical variable, and the others are high cardinality. Parts of variables included thousands of unique values.

Table 3: Unique Values in Categorical Variables

Column Name	Unique value
color	2
director_name	2398
actor_2_name	3032
genres	914
actor_1_name	2097
movie_title	4917
actor_3_name	3521
plot_keywords	4760
movie_imdb_link	4919
language	47
country	65
content_rating	18

Figure 1 shows the distribution of target variables. It is close to normal distribution. Figure 2 is the heat map of the correlation matrix. It is clear that there are strong correlations between variables so models may meet the multicollinearity problem in the training process.

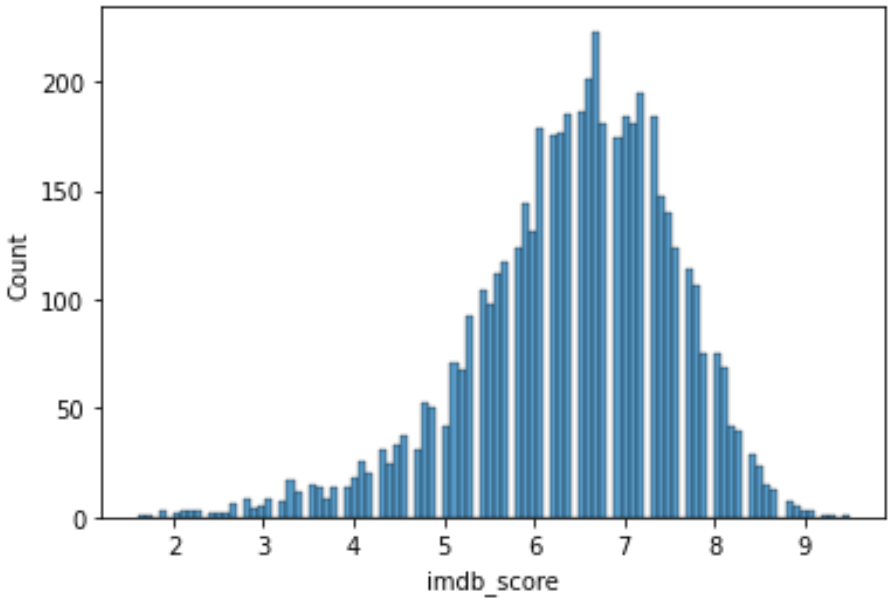

Figure 1: Distribution of imdb_score

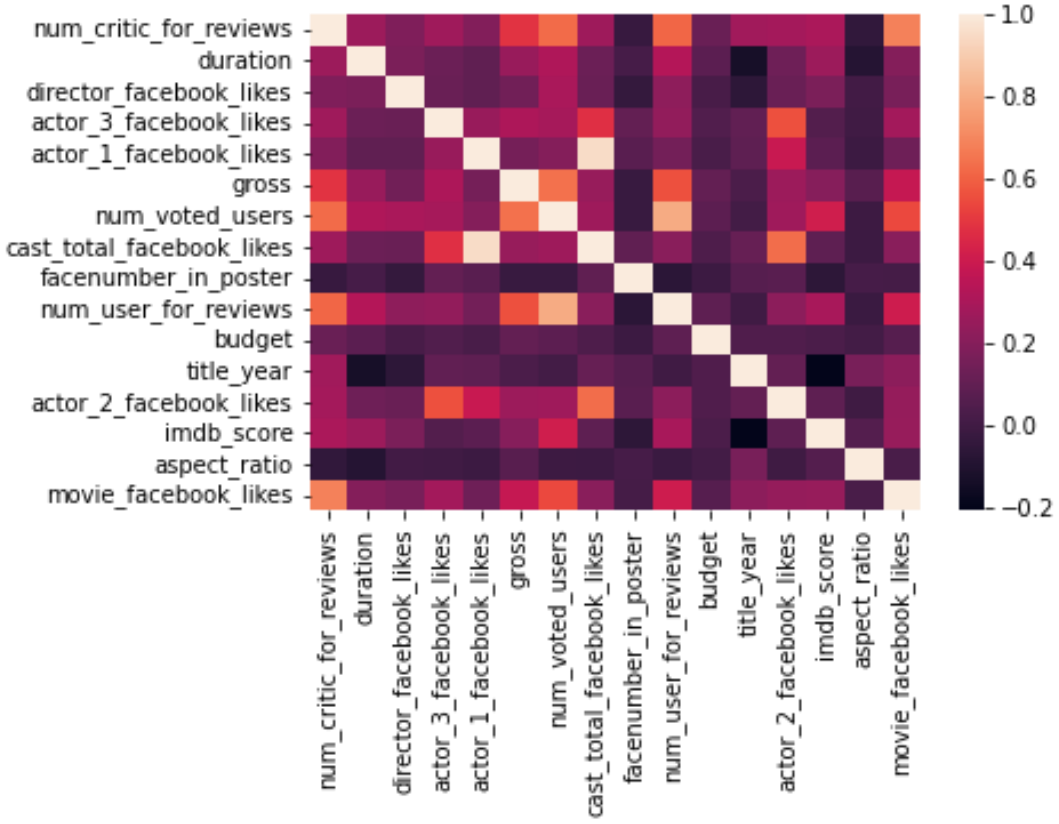

Figure 2: Correlation Matrix

Methodology

Pre-processing

For the missing value problem, I used the cinemagoer package to extract data from the IMDB database based on the IMDB number. First, fill missing values with the data in the cinemagoer set. Second, if missing values still exist, fill it with 'Other' for categorical variable and with the mean of variable for numeric variables

For the outliers problem, first, located variables that contained outliers based on EDA. Then, removed it and recheck the variable distribution: if they were a mild outlier, keet it, and if they were not mild, remove it.

In feature engineering, For categorical variables, there were two encoding strategies. If the number of unique variables is low(color), used one-hot encoding. If the number of

unique variables is high, used hashing encoding technique. Especially, for “movie_title” and “plot_keywords”, create two new features: word_num, and avg_length.

Model Design

This study chooses 4 basic models and 2 ensemble methods: lasso regression model, support vector machine(SVM) model, random forest model, gradient boosting model, blending ensemble model and stacking ensemble model. The model with the best performance will be the model which could be useful to deploy into production.

The dataset split ratio is 4:1. It means that 80% of data would be split randomly into the train set and 20% of data would be split into the test set.

As the figure 3 shows, the model selection process are: (1) dataset split; (2)standardization; (3)gird search; (4)ensemble model generation; (5)cross validation (6)final model generation.

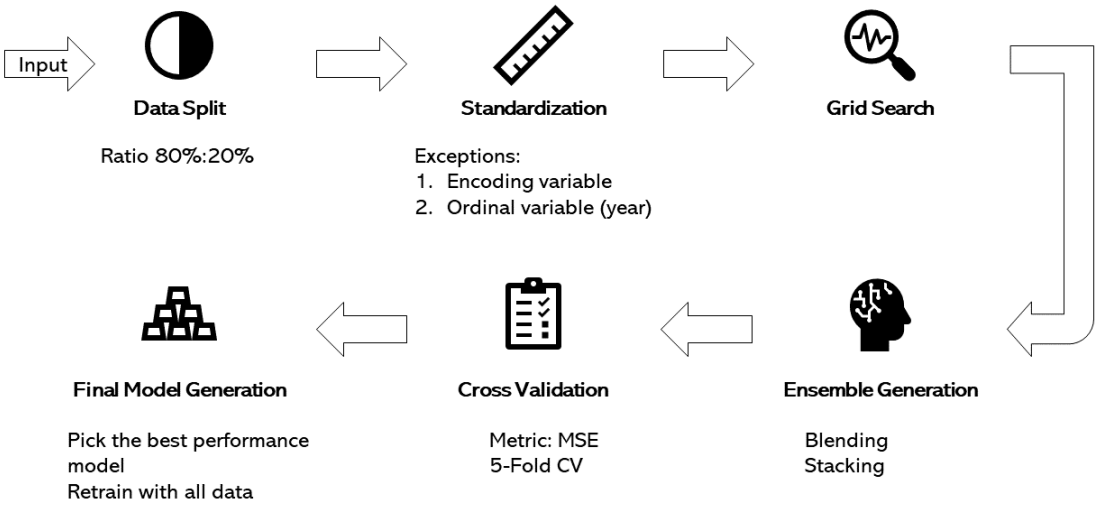

Figure 3: Model Selection Process

Result

Table 4 shows all 6 models' performances. In 4 basic models, the random forest model has the lowest MSE, so it was selected to be the second layer model in the stacking ensemble. Blending Ensemble Model had the lowest MES of all 6 models, which meant the average squared error between model prediction and the true value was 0.645. So, the blending ensemble model was the model which could be useful to deploy in production.

Table is the top 5 feature importance from random forest model. Top 5 important features are: the number of voted users, the movie duration, the movie budget, the released year of movie, and the number of users for reviews.

Table 4: Model Result

Model	MSE
Lasso Regression	0.911
SVM	1.292
Random Forest	0.722
XGB	0.769
Blending Ensemble Model	0.644
Stacking Ensemble Model	0.653

Table 5: Top 5 Feature Importance

Feature Name	Importance
num_voted_users	0.211623
duration	0.126319
budget	0.069417
title_year	0.065868
num_user_for_reviews	0.060038

Conclusion

According the model results, the mian conclusion of the study are:

[1] The blending ensemble model has the best performance score. The finding suggests to select the model to deploy into production.

[2] The effect of Social newwork data is less than the movie properties data. The number of voted users, duration, budget, released year, and the number of users for reviews have the highest importance. All of them are movie properties features.

Future Work

To improve model performance and get robust conclusions in the future research, I suggest to put attention to the following 4 points:

First, data. This data contains some missing values and incorrect data like invalid IMDB id, and its social data is collected at one time, which means the same actors/directors have the same “like” data whenever a movie is released. So, a high-quality dataset can improve models’ performance.

The second is feature engineering. A feature assumption is all numeric variables follow the normal distribution. It may be incorrect. Future work could use some transformations like box-cox transformation for features that don’t follow normal distribution before standardization. Another suggestion is about the encoding technique. Because the dataset contains several high cardinality columns, this study use hashing encoding to transform these columns into 8 features. This process loses some information, so future work would try other encoding techniques to improve performance.

Third, Model. This study only generate 6 models and lasso regression model has non-converge problem. All of this may make negative effect on final model performance. Therefore, future work might pick other models like Extra Randomized Trees(ERT) and Neural Networks and try other ensemble methods like bagging and Boosting. Also, this study didn't implement the grid search for the second layer model in stacking ensemble methods. Maybe grid search can improve stacking models’ performance.

Finally, the evaluation. This project only chooses MSE as the metric of models. Future studys could try oher types of metric to evaluate model performances.