An Investigation in predictive model of Revenue of movies

Chunbo Wang

G22987927

Introduction

This study is aimed to draw a prediction of the rating of a movie based on its properties. With this model applied to commercial movies shall one be able to give an estimation on the revenue those movies may generate. Considering the study’s predictive nature, a regression analysis is to be expected.  Consequently, the tasks lie in finding out the potential factors that contribute to the rating of a movie and building a regression model that predicts the outcome(revenue) as satisfactory as possible when compared to the targets of test inputs.

To better analyze the correlation between properties and the revenue of a movie, a preprocessing stage was performed and recorded in the second part of the paper. Among the given properties, the ones that give out the most accurate outputs were recognized as inputs that contributes to the revenue.

Then, the dataset was split into training and testing groups for generating and assessing a predictive model with the method of supervised learning.

As results, the output values were compared to target values and the difference was set to error used to assess the model.

Data Preprocessing

The selected Dataset consists of about 1000 variables(movies) along with Genre, Description, Director, Actors, Runtime, Rating, Votes, Revenue and Metascore of each individual. Properties were selected as input(Votes, Revenue and Metascore) and output(Rating). The csv file was read via **pandas** library. By browsing the dataset, numerous missing values were found in input, so imputation was applied to fill in those values with mean value of that column.

Since all variables were numerical values, **arrays** was used to store all data and carry operations.

And as a general approach, the values were standardized using **sklearn.standardizer** to convert into a  scope of  0 to 1.

As for inputs, Runtime, Rating, Votes and Metascore were chosen as candidates. To verify the reliability of these inputs, hypothesis tests were chosen from **statsmodels.stats.** The null hypothesis was set to be no correlation between input(x) and output(y) while the alternative being there is correlation; With the significance of 0.05, all inputs were tested to be valid. The following table records the P-values in hypothesis test for each input that were filtered out.

A screenshot of a cell phone

Description automatically generated

Regression Analysis

In the beginning of the analysis, each training input candidate was fit to a linear model with the training output and the regression was plotted; nonetheless, all cases were assessed to be non-linear according plots. Thus, **MLPREGRESSOR** was ingratiated in model training. And the test group was input into the model, giving results for error analysis.

For the purpose of optimizing the model, different combinations of input variables were designed and tested for building the model and comparing the errors in result. First, 2 variables were chosen for building the model. Based on combinatorics, there were 4 combinations that could compose to inputs{(x1,x2),(x2,x3),(x1,x3),(x1,x2,x3)}. Each result was recorded for computing accuracy. Then, all 3 variables were set as inputs, and the result recorded and compared to all others.

4.1 Case I

In this case, we consider runtime and votes as input features. And the variation of

output values and error for test sample is presented in Figure 1and 2.

It was seen that there was a good match between output and the target, and the mean square difference was only about 0.58A screenshot of a social media post

Description automatically generated

4.2 Case II

In this case, we consider runtime and metascore as input features. And the variation

of output values and error for test sample is presented in Figure 3 and 4.

**A screenshot of a cell phone

Description automatically generated**A close up of a logo

Description automatically generated

4.3 Case III

In this case, we consider votes and metascore as input features. And the variation

of output values and error for test sample is presented in Figure 5 and 6.

A screenshot of a cell phone

Description automatically generated

4.4 Case IV

In this case, we consider runtime, votes, and metascore as input features. And the

variation of output values and error for test sample is presented in Figure 7 and 8.

A screenshot of a cell phone

Description automatically generated

All the MSEs were recoded in Table2

As a result, the efficiency of the regression for prediction was the values of mean squared error in Case IV, which has the lowest value of mean squared error. Therefore, the performance of prediction for rating of rating is the best while considering three input features including runtime,votes, and metascore.

A screenshot of a cell phone

Description automatically generated

Conclusion

Based on the Numerical result from the proposed model, it was found that a strong correlation between a movie’s properties(runtime, metascore and votes) and its rating exists:  while all three inputs were used to fit the model, the resultant mean square error between target and test output was only 0.469, the smallest among all cases. Therefore, it was concluded that the model was relatively satisfactory when all properties (runtime, metascore and votes) are considered as factors to contribute to the rating of a movie. And the built model via MLPREGRESSOR was able to produce efficient prediction according to the extent of match between prediction and target in**.** Beyond the scope of this study is the optimization of this built model, which requires extensive search in more data as the training group, and maybe other advanced algorithms.