Predict Movie Rating Using ML Methods
(CS 5785) AML Final Writeup - Volodymyr Kuleshov

Yi-Ru Pei (yp329), Annie Hsin-Yun Tsai (ht469), and Sakshi Agarwal Mittal (sa875)

December 12, 2022

Abstract

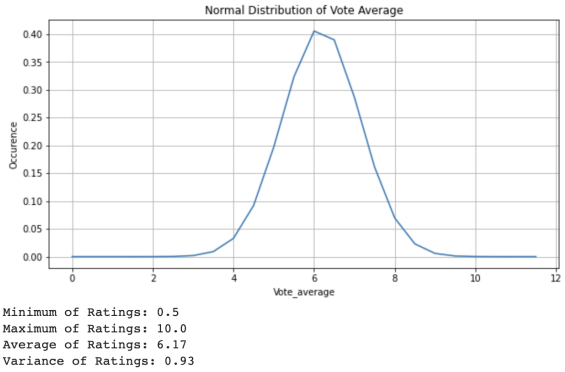
This report presents a machine-learning approach to predicting movie ratings. We train a regression model on a movie rating dataset and its corresponding features and evaluate its performance on a test set. In addition, this report presents a machine-learning method for predicting movie ratings. On a dataset of movie reviews and the features that correspond to them, we trained a regression model, and we then assessed its performance on a test set. Our model performed well, which suggests that movie studios and distributors can use it to inform their choices and raise the likelihood that they will be successful. We will use several machine learning models in this report: Linear Regression, Ridge Regression, Support Vector Machine, and Gradient Boosting Regressor to confirm the model's effectiveness based on its attributes.

1 Introduction

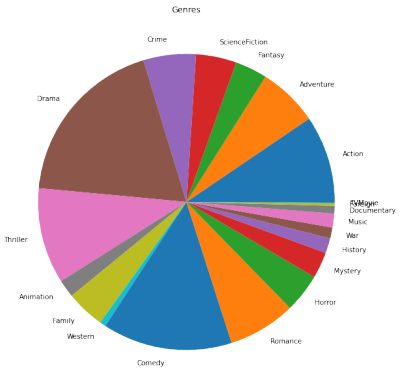
Machine learning methods can be applied to the task of predicting movie reviews. We can create a model that can make accurate predictions about new movies by training it on a dataset of movie ratings and other relevant information about the movies, such as their genre, actors, and plot. There are several reasons for using a machine learning model to predict movie ratings. First, accurate predictions of movie ratings can help movie studios and distributors make informed decisions about their productions and release plans. By knowing how a movie is likely to be received by audiences, studios and distributors can decide whether to invest more resources in promoting the movie or focus their efforts on other projects. Second, predictions of movie ratings can also help moviegoers make better-informed decisions about which movies to watch. By providing users with an estimated rating for a movie, they can decide whether it will likely be worth their time and money. Finally, predicting movie ratings is also an exciting and challenging problem for machine learning researchers. By developing and refining models for this task, researchers can improve their understanding of how to build effective regression models and apply them to real-world problems. In this report, we apply several machine-learning models to evaluate the models' performance.

2 Background

We use TMDB 5000 Movie Dataset [1] on Kaggle, a collection of over 5,000 movies, along with a variety of information and metadata about each movie. This dataset contains the film's title, budget, revenue, release date, runtime, genres, a brief plot summary, and cast and crew information. It also contains user ratings and reviews for each film. This dataset can be used to build recommendation systems, analyze movie industry trends, and train machine learning models to predict movie ratings. It is an excellent resource for anyone interested in the film industry or applying machine learning techniques to films.



(a) Normal distribution of Vote Average



(b) Pie Chart for Genres

3 Data Preprocessing

Our dataset initially had a few important categories with values in JSON format instead of single list values. So we first converted JSON values of categories: genres, keywords, production companies, cast, and crew to lists format to have a clear dataset to run our experiments. Next, we dropped the columns that did not add value to our experiment and acted as noise instead. Hence, for our experiment, we only used the columns id, original title, genres, cast, crew, production companies, keywords, and vote average. First, we merged the two CSV files and removed the null values in the vote_average column. Second, we encoded cast, crew, genres, and production_companies using MultiLabelBinarizer() and only the top 10 influentials were kept. Third, we dropped the unused features and cleaned up null values in the remaining dataset. Last, we normalized the dataset using MinMaxScaler() to rescale the values between 0 to 1 before building the model.

4 Observation

4.1 Vote Average Visualization

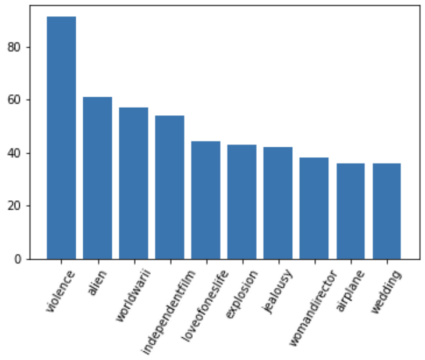
After finishing data preprocessing, we further looked into the dataset to have a better understanding of the data. Firstly, we would like to focus on the "Vote average" column to see the ratings of the movies. We generated a normal distribution plot shown below (Fig. 1a) to visualize the rating. Since, according to the property of normal distribution, 68% of the observations fall within one standard deviation from the mean value, from the plot, we could see that 68% of the ratings are between 5.20 and 7.14 (the mean is around 6.17 and the standard deviation is around 0.97).

4.2 Genres Visualization

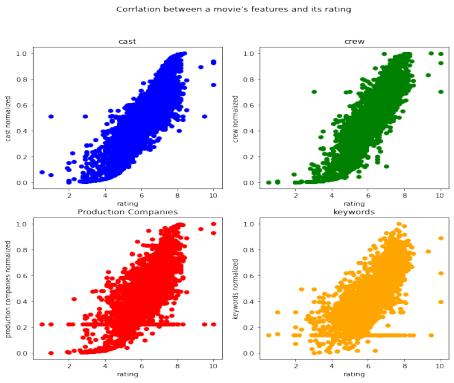
We also created a pie chart of the different genres the movies belonged to, as seen in Fig. **??**. Through the pie chart, we were able to get a clearer idea and also visually categorize them. Additionally, the pie chart helps us understand the most popular genres and those with less significance. As observed in the figure below, Drama, Comedy, and Thriller are the most popular genres for movies.

4.3 Keywords Visualization

Another column that we would like to take a closer look at is the Keywords column. We were interested in finding out the top 10 keywords used by these movies. To observe this, we needed to use the get_dummies function from pandas to count the occurrence of each keyword since some cells in the column contain multiple keywords at a time. From Fig. **??**, we were able to see the top ten keywords in the Keywords column, and the word with the highest occurrence is "violence".



(a) Top 10 Keywords



(b) Correlation between Movie's features and Rating

4.4 Correlation between Features and Ratings

One of the most important observations we made was to understand the correlation between various movie features such as cast, crew, production companies, and keywords with the movie's rating. As observed in Fig. 2b. below, the features have a positive correlation with the ratings. However, for some specific cast and crew, the ratings are high or low, but on average, they are in the middle range. However, for keywords and production companies, there are quite some variances and asymmetry.

5 Previous Works

Originally the dataset used by the project was about different streaming platforms. However, while working on further analysis of data and building onto the previous codes, the results from the analysis were invalid in the end, which led to the decision that there was a need to change the dataset used for the project. For the new dataset, some works finished prior to the final analysis were data preprocessing as well as visualizations on various data points.

6 Methods

6.1 Linear Regression

For the baseline model, we used linear regression, a commonly used predictive modeling technique. This is represented by an equation that predicts the value of a target variable based on one or more predictor variables. We are evaluating the accuracy of our model by looking at the R-Squared value in Eqn. 1, which measures the proportion of variance in the dependent variable that is explained by the independent variable(s). It is a value between 0 and 1, where a high value indicates that the model is a good fit for the data, and a low value indicates that the model is not a good fit.

$$R^2 = 1 - \frac{sum\ squared\ regression\ (SSR)}{total\ sum\ of\ squares\ (SST)} = 1 - \frac{\Sigma_{i=1}^{n}(Y_i - \hat{Y}_i)^2}{\Sigma_{i=1}^{n}(Y_i - \bar{Y}_i)^2} \tag{1}$$

6.2 Ridge Regression

The motivation for applying ridge regression is to improve the performance and stability of a linear regression model by adding a regularization term to the objective function, as shown in Eqn. **??**, where Q is the number of features. This regularization term, also known as the L2 penalty, helps to prevent overfitting by penalizing large values for the model coefficients. This can improve the interpretability of the model, as well as its ability to generalize to new data. Additionally, ridge regression can be used in cases where

there are a large number of features and a small number of training examples, as it can help to reduce the complexity of the model and improve its performance.

$$MSE\ with\ l2\ penalties = \frac{1}{n}\Sigma_{i=1}^{n}(Y_i - \hat{\lambda})^2 + \lambda * \Sigma_{q=1}^{Q} w_q^2 \tag{2}$$

6.3 Support Vector Machine

Support vector machines (SVMs) are a type of supervised learning algorithm that can be used for classification or regression. The motivation for using SVMs is to find the hyperplane in a high-dimensional space that maximally separates the data points of different classes. This is known as the maximum margin classifier. By finding the maximum margin hyperplane, the SVM is able to create a decision boundary that is robust to noise and can generalize well to new data. Additionally, SVMs can handle non-linearly separable data using the kernel trick, which allows them to operate in a higher-dimensional space without explicitly computing the coordinates of the data in that space. This makes SVMs a powerful and flexible tool for a variety of machine-learning tasks.

6.4 Gradient Boosting Regressor

The motivation for using gradient boosting regressor is to improve the performance of a regression model by combining multiple weak learners into an ensemble model. In gradient boosting, weak learners are trained sequentially, with each learner focusing on the mistakes made by the previous learners. This allows the ensemble model to make more accurate predictions than any of the individual weak learners alone. Additionally, gradient boosting uses a loss function to measure the error of the current model and determine the direction in which the model should be improved. This allows the algorithm to adaptively adjust the contribution of each weak learner to the final prediction. By combining these techniques, gradient boosting can produce highly accurate regression models that are able to make effective predictions on a wide range of data.

6.5 Cross Validation

The other method we adopted for the dataset was cross-validation. Cross-validation is a method used in machine learning to validate the model's efficiency, which utilizes the subsets of the dataset to test the unseen subsets from the input data. Among all cross-validation methods, we picked k-fold cross-validation. Furthermore, we applied it to our experiment. How k-fold cross-validation works is that we first split the dataset into k subsets. It has required us to leave one subset for the evaluation purpose of the trained model while we perform training on the rest of the subsets.

7 Experimental Analysis

7.1 Experiment Setting

First, we split the dataset into 70% of training data and 30% of testing data. The input dataset size (after preprocessing) included 4740 rows and 82 columns. Then, we use K-fold Cross Validation (K=5) on the training data to select the best available models based on Mean Squared Error (MSE), as shown in Fig. 3. Eqn. 3 describes how to calculate the MSE, where Y_i and $\hat{Y}_i$ are the ground-true value and predicted value, respectively. Mean squared error (MSE) is a standard metric used to evaluate the performance of regression models. It measures the average squared difference between predicted and actual values. A low MSE indicates that the model is making accurate predictions, while a high MSE indicates that the model is not a good fit for the data.

Finally, we employ this machine learning problem to evaluate the estimated movie ratings of all films in the test dataset with several metrics, as shown in Sec. **??**. We split the dataset first and applied K-fold

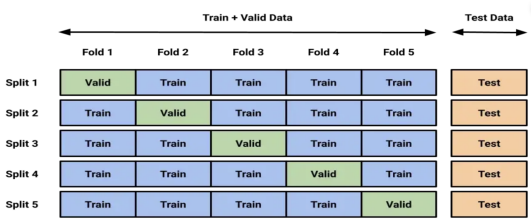


Figure 3: Visualization of K-fold Cross Validation, where K=5 [3]

because we wanted to verify the model's performance correctly. If we directly use K-fold to find the best machine learning models without holding a testing set, we cannot know the generalization error.

$$MSE = \frac{1}{n}\Sigma_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \tag{3}$$

7.2 Results and Analysis

Results	Linear Regression	L2 Regression	SVM	Gradient Boosting Regressor
KFold MSE	0.80	0.69	0.70	0.59
Mean Squared Error	0.78	0.70	0.71	0.60
r^2 score	0.17	0.27	0.25	0.37

As shown in Sec. **??** in our technical experiment, we tested various regression and classification algorithms to identify the best-performing model for our data. We evaluated the performance of these models using metrics such as mean squared error and R-Squared values. After conducting our analysis, we found that Gradient Boosting Regressor was the best-performing model we tested. It had the lowest mean squared error and the highest R-Squared value, indicating a solid fit between the test and predicted data. In order to prevent overfitting and further improve the model's performance, we applied ridge regression and observed an increase in the R-Squared score and a decrease in the MSE. We also found that Support Vector Machines (SVMs) effectively generated clear margins between classes and were well-suited for high-dimensional data. Our results showed that the SVM model performed better than the baseline model (linear regression). Overall, our findings suggest that both Gradient Boosting Regressor and SVM are strong candidates for use in predictive modeling tasks. Further experimentation and analysis may be needed to determine the optimal model for a specific application or dataset.

8 Conclusion

In conclusion, our experiment successfully demonstrated the potential use of machine learning methods for predicting movie ratings. Despite achieving an accuracy of only 37%, this level of performance may still be valuable in specific scenarios where even a rough estimate of movie reviews could be beneficial. For example, this model could be used by movie studios to help guide their decision-making processes, such as by providing an initial assessment of the potential success of a new release. In future work, we plan to continue exploring ways to improve the performance of our model, which could include experimenting with different machine learning algorithms and feature engineering techniques, as well as using more extensive and diverse datasets. We will also investigate the use of other evaluation metrics to gain a more comprehensive understanding of the capabilities and limitations of our approach.

References

[1] "TMDB 5000 Movie Dataset" https://www.kaggle.com/datasets/tmdb/tmdb-movie-metadata (accessed Dec. 2, 2022).

[2] "Predict Movie Ratings via Machine Learning" https://www.kaggle.com/code/bhsraman/predict-movie-ratings-via-machine-learning/notebook (accessed Dec. 4, 2022).

[3] "Build Training/Validation/Testing Set (Translated)" link (accessed Dec. 10, 2022).

[4] W. R. Bristi, Z. Zaman and N. Sultana, "Predicting IMDb Rating of Movies by Machine Learning Techniques," 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2019, pp. 1-5, doi: https://doi.org/10.1109/ICCCNT45670.2019.8944604.