Business Problem

Hollywood, a multibillion-dollar industry, still produces films with poor ratings to this day, costing hundreds of millions of dollars in lost profits per film. Is there a way to predict how a film will perform before it is released in theaters?

Background

Hollywood is a large industry that employs many thousands of actors. As with any other business, they rely on marketing to popularize their product. However, the very nature of movie releases means that the opinions of critics can affect profit margins (their opinions influence large numbers of people to see a movie or avoid it altogether). Thus, movies rated poorly by critics cost hundreds of millions in lost revenue. Hollywood does not know how a movie will be rated ahead of their releases; this makes movie production a high-risk endeavor. In the interest of commercial profitability, an algorithm to predict which films will get high critic scores becomes necessary.

This project aims to eliminate this risk in movie production by predicting how critics will rate a movie before it is released. Can this actually be done? The short answer is yes. A machine learning model will draw on past movie data to predict how well future movies will be rated on the Internet Movie Database (IMDb). This will allow Hollywood to prioritize working on movies that are predicted to have good ratings.

Data Explanation

The data used is a CSV file containing information on more than 5,000 movies from 1916-2016. Each row represents one movie and each column represents a feature of the movie, such as runtime, budget, director, actors, Facebook likes, etc. Many columns had missing data represented as “NA” in place of the data. To “clean” the data for machine learning, all the rows containing NA values were removed. This could be done because the dataset was sufficiently large.

Methods/Analysis

Construction of the machine learning model will consist of three parts: Cleaning the raw data, performing exploratory data analysis (EDA), then constructing a random forest regression model. This model was chosen as the dataset is very large and the target variable is IMDb score, which is a floating-point value out of 10 describing how well a movie performed.

The first step, the data preparation, involved two parts. First, any rows that had missing or incomplete values (labeled as “NA”) were removed from the dataset entirely so the model’s predictions would be accurate. Second, duplicate movies were removed so that some data would not be over-represented. The last part, feature selection to reduce the features to only a few key ones for analysis, was done after EDA to keep some columns for graphing purposes. Enough rows remained to continue with analysis.

With clean data, EDA was conducted to gain insight into the data and find out what the model might predict. This involved constructing graphs. First, a histogram of the numerical variables showed any skewness. Next, a graph of IMDb score per director for the 10 highest scorers was made to reveal which directors might end up in the predictions. Once these were constructed, a correlation heatmap was made to reveal which features were correlated. Any features that were correlated would be removed from the dataset to avoid introducing biases. Finally, IMDb score per genre and number of movies per genre were graphed to find out the most popular genres.

With EDA complete, the random forest regression model was constructed. This type of model works by building several decision trees to each process a test input. The average of each tree’s prediction is output as the random forest prediction (see Figure 1), making it suitable for large datasets. To build the model, the cleaned data was split into training and testing sets using the *train_test_split* function, with 20% allocated for testing. Because the target variable is numerical, the model’s performance was evaluated with three metrics: mean average error (MAE), mean squared error (MSE), and root mean squared error (RMSE). For the random forest model, these values were 0.61, 0.65, and 0.81 respectively (see Figure 2). A lower error value indicates a regression model that outputs IMDb scores closer to the actual values. These values were all <1, indicating the model is very accurate in predicting movies’ scores.

Conclusion

This project's goal was to accurately predict the IMDb scores of future films, and it appears to have succeeded. The calculated errors were below 1, indicating fairly accurate results. This project could be an important advancement for the entertainment industry if implemented correctly, as it allows movie producers to view how well their movies are doing before they are released. This effectively becomes a "safety net" for filmmakers if the model is run early in a film's life cycle. If a movie has a low predicted IMDb score, producers can simply cease production before it is marketed and shift focus to other films. This will ultimately result in lowered risk and increased profits.

Assumptions

Not all information regarding the movie industry and previous movie predictors were found. To proceed with this project, a few assumptions had to be made. First and foremost, it was assumed that the style of filmmaking would remain the same as it has been for the past decade. This would allow the predictive algorithm to be used without frequent major updates. Second, because the dataset contains many countries' films, it was assumed that the algorithm could be used by any movie sector, not just Hollywood. Third, it was assumed that the resulting errors of <1 are the lowest achievable errors.

Challenges/Limitations

There were some issues encountered when building the machine learning model that had to be addressed. First, there was only a limited amount of computing power available to train and test the model; as such, much time was spent waiting. Fortunately, there was enough time allocated for this, but a more powerful computer would be needed if time was short. There was also the issue of selecting the most important features for model building; this was remedied by selecting those features not correlated to each other. Correlated features unnecessarily take up memory and introduce biases. After the model was built, another issue presented itself. It was assumed movie trends would remain the same, but in reality, they actually shift constantly, requiring the model to be updated frequently if it is to be implemented correctly. For the purposes of this project, the assumption will remain that movie trends don't change.

Future Uses/Recommendations

The described algorithm is not just limited to movie score predictions. In the future, reworked algorithms based on this one can be applied to other media such as songs, video games, and TV shows. It is recommended that a new algorithm to predict TV show ratings be developed following the successful implementation of this project's algorithm. TV shows contain many features similar to movies; it would not be a stretch to retrain a similar algorithm with a dataset of shows. If the second algorithm is successful, then more can be developed in the near future for other media.

Ethical Assessment

This was one of the first steps to be conducted after business/data understanding. While movie trends are ever-changing, industry actors' demographics usually do not. This could lead to the algorithm always predicting one demographic to produce the best films, leaving out many other actors and directors. This must be acknowledged, and care must be taken to ensure diversity remains in movie industries.

Implementation Plan:

See Figure 3.

Questions

1. Why was a random forest model chosen over linear regression?
2. Would a similar algorithm have any application outside the entertainment sector?
3. Is the assumption that <1 is a safe error value founded on previous projects?
4. What is the plan to resolve the ethical issue mentioned earlier?
5. How would changing trends be accounted for if the model was deployed today?
6. On a larger data scale, how would the aforementioned power issue be resolved?
7. What types of biases would correlated features introduce in your model?
8. Could such a model be implemented using other programming languages?
9. What hyperparameters could have been tuned in this model?
10. Were there any other important features that weren't included in the dataset?

Illustrations

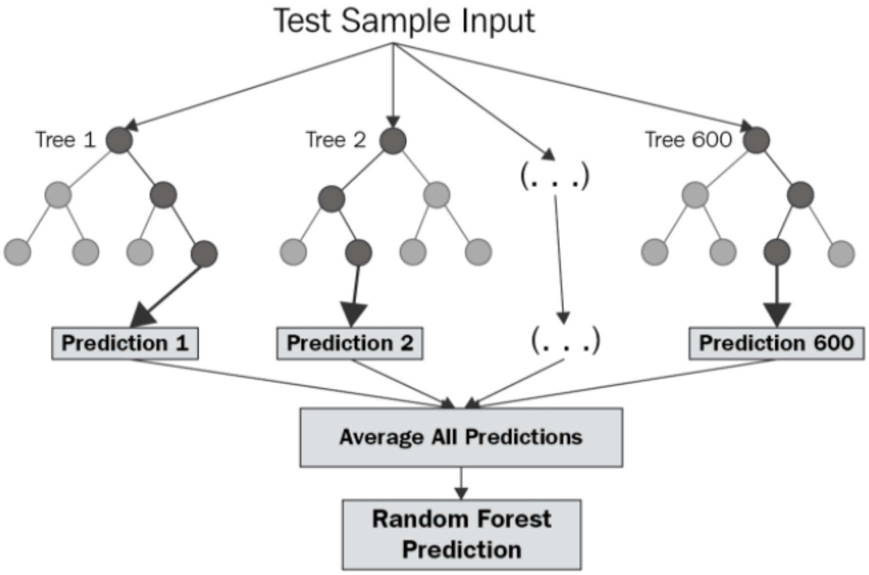
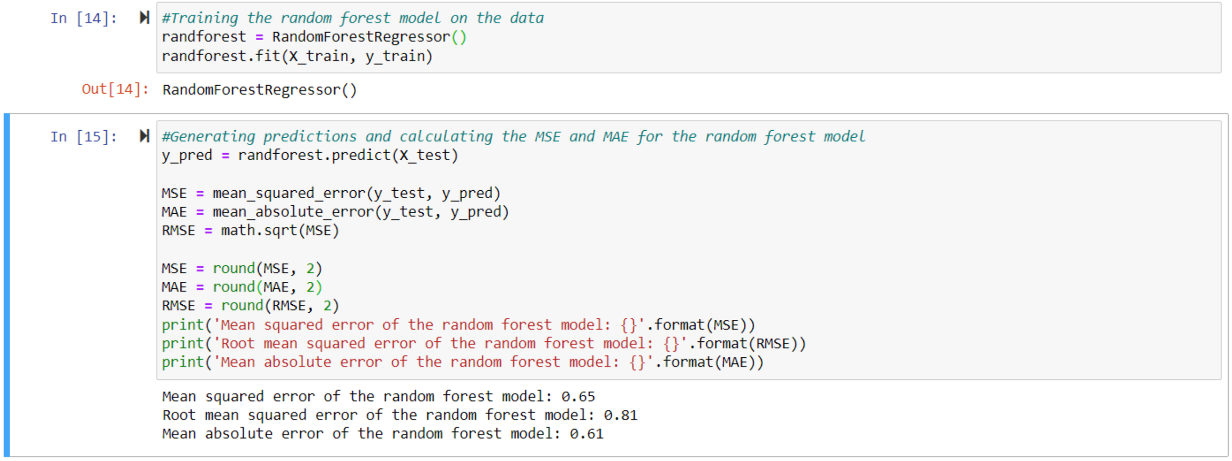


Figure 1: Diagram showing a random forest regression algorithm, which is made of several decision tree algorithms. The input is processed by several decision trees and each outputs a prediction. The average of all the predictions becomes the output of the random forest.



```
In [14]:  #Training the random forest model on the data
          randforest = RandomForestRegressor()
          randforest.fit(X_train, y_train)

Out[14]:  RandomForestRegressor()
```

```
In [15]:  #Generating predictions and calculating the MSE and MAE for the random forest model
          y_pred = randforest.predict(X_test)

          MSE = mean_squared_error(y_test, y_pred)
          MAE = mean_absolute_error(y_test, y_pred)
          RMSE = math.sqrt(MSE)

          MSE = round(MSE, 2)
          MAE = round(MAE, 2)
          RMSE = round(RMSE, 2)
          print('Mean squared error of the random forest model: {}'.format(MSE))
          print('Root mean squared error of the random forest model: {}'.format(RMSE))
          print('Mean absolute error of the random forest model: {}'.format(MAE))

          Mean squared error of the random forest model: 0.65
          Root mean squared error of the random forest model: 0.81
          Mean absolute error of the random forest model: 0.61
```

Figure 2: The random forest model and the three error values used to evaluate its performance. They are all lower than 1, meaning this particular model is accurate for predicting current films' popularity.

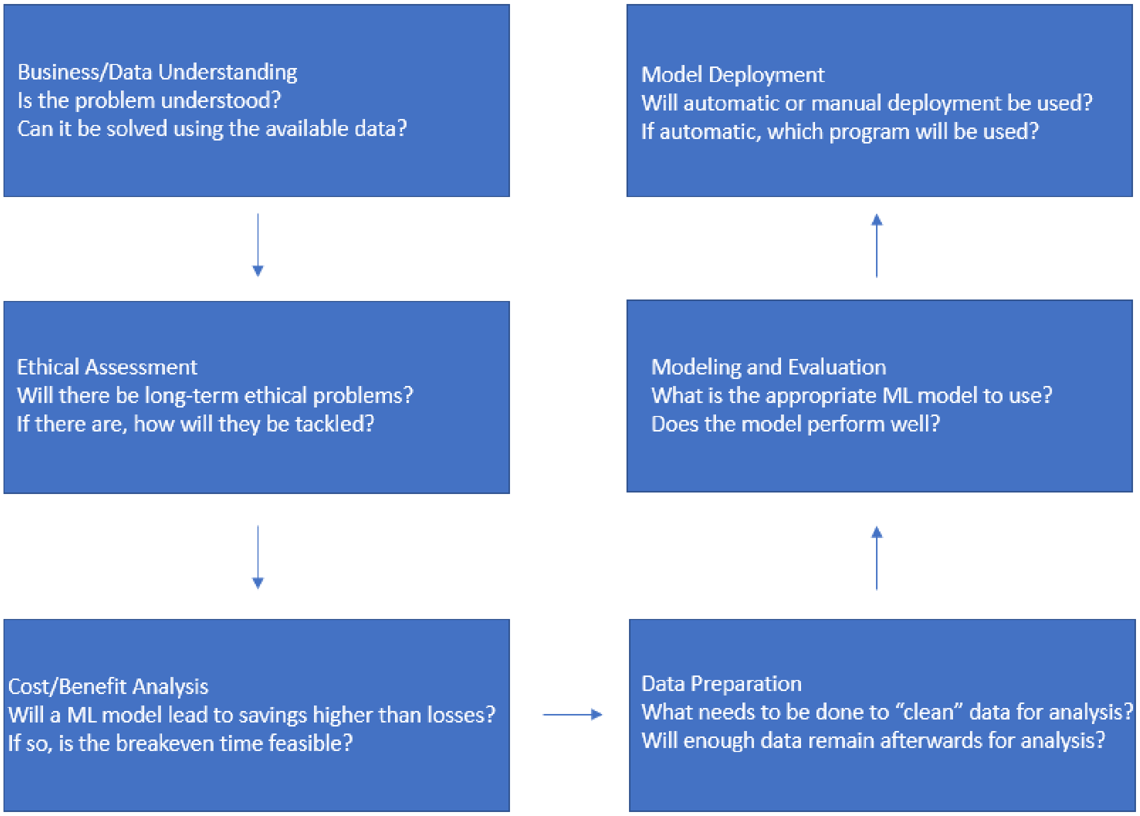

Business/Data Understanding
Is the problem understood?
Can it be solved using the available data?
Ethical Assessment
Will there be long-term ethical problems?
If there are, how will they be tackled?
Cost/Benefit Analysis
Will a ML model lead to savings higher than losses?
If so, is the breakeven time feasible?
Data Preparation
What needs to be done to "clean" data for analysis?
Will enough data remain afterwards for analysis?
Modeling and Evaluation
What is the appropriate ML model to use?
Does the model perform well?
Model Deployment
Will automatic or manual deployment be used?
If automatic, which program will be used?

Figure 3: The implementation plan for this project, displaying a few of the questions that must be answered at each step of the process.

References

Chakraborty, P., Rahman, S., & Zahid, Z. (2019). Movie Success Prediction using Historical and Current Data Mining. International Journal of Computer Applications. Retrieved on April 3, 2022 from https://www.researchgate.net/publication/335878983_Movie_Success_Prediction_using_Historical_and_Current_Data_Mining#:~:text=Identifying%20the%20right%20factors%20can,views%2C%20trailer%20views%20etc...

Gleeson, P. P. D. (2018). Statistics on People Getting Famous in Acting. Work - Chron.com. Retrieved on April 3, 2022 from https://work.chron.com/statistics-people-getting-famous-acting-23946.html.

Mayo, A. (2021). Hollywood is losing out on $10 billion in revenue every year by underfunding work by Black creators. Business Insider. Retrieved on April 3, 2022 from https://www.businessinsider.com/hollywood-could-boost-revenue-by-10-billion-2021-3.

Raj, A. (2020). A Quick and Dirty Guide to Random Forest Regression. Towards Data Science. Retrieved on April 3, 2022 from https://towardsdatascience.com/a-quick-and-dirty-guide-to-random-forest-regression-52ca0af157f8.

Sakoui, A. (2019). Hollywood Employs More Workers Than Mining and Farming, MPAA Says. Bloomberg. Retrieved on April 3, 2022 from https://www.bloomberg.com/news/articles/2019-03-18/hollywood-tops-mining-crop-production-in-employment-mpaa-says.

Watson, A. (2020). Film Industry - statistics & facts. Statista. Retrieved on April 3, 2022 from https://www.statista.com/topics/964/film/.