Movie Madness

Qiang Bi
Sameena Khan
Richie Lee
Sarah Wong
Nancy Yao

THE TASK

Task

Our task is to predict the gross box office revenue of a movie, given attributes such as popularity of actors based on number of Facebook likes and total movie budget. This task is important because it will allow us to figure out what features will make or break a movie's success.

Dataset

We utilized the dataset IMDB 5000 from Kaggle. There is information in the original dataset for 5000 movies. The attributes include duration, budget, face number (number of faces in movie poster), title year, IMDB score, and aspect ratio. The dataset also includes information about the directors and lead actors, but instead of using their names as categorical attributes, we decided to use their Facebook page likes so that we could convert the attributes into numerical values.

In deciding which examples and features to consider, we chose the features that seemed most relevant (see "Selecting Features" section below) and eliminated those such as face number, which seemed arbitrary. We also limited our examples to those created in the United States, for reasons clarified in the section below. For our final dataset, our training data (70%) had 2208 examples and our test data (30%) had 947.

Selecting Features

Our original dataset included many attributes that we did not find useful for predicting box office revenue, so we pre-processed much of the data to get our finalized training and test sets. Part of this was due to inconsistencies in the original dataset. For example, not all of the gross revenue was expressed in U.S. dollars, posing a difficulty because various countries' exchange rates to dollars fluctuate differently, based on the year. For this reason, we narrowed our scope to predicting the box office revenue among movies produced in the U.S., rather than all of the examples in the original dataset.

The celebrities associated with any given movie tend to greatly impact its popularity, as in the case of "bankable stars" whose presence in a movie is supposedly able to guarantee box office success. In order to gauge the effect that the names associated with a movie had on its revenue, we needed an accurate measure of the popularity and recognizability of the lead actors and director. We scraped data from Facebook on the number of likes on each individual's page. We chose to do this because we realized that the Facebook likes initially included in our data set had a lot of missing values, and weren't the likes on the celebrities' actual Facebook pages, but those on their IMDB profile pages. Scraping the Facebook page likes of actors and directors allowed us to represent their popularity using numerical values, which can be easily interpreted by most of the machine learning techniques.

There is a possibility of error in our approach for scraping the Facebook likes. Our method searches for the given name and takes in the likes for the first page that comes up in the search results. We assume that this is accurate for most searches, as the top result would likely be the official celebrity page. However, for lesser known directors and actors, this might obtain the wrong page's number of likes if the top result is someone else with the same name. Another issue with this approach is that not all famous actors/director have a strong social media presence, and Facebook page-likes tend to undervalue the popularity of actors of the older generations.

Our final attributes were the number of critic reviews, duration, Facebook likes of the director and three leading actors, IMDB likes for the whole cast and movie, number of ratings, number of users reviews, movie budget, release year, and IMDB score.

Classifying Revenue

We decided to divide the gross revenue into a binary classification, where the threshold between classes was $30,000,000. In future work, to get a more accurate classification of revenue, we would try splitting the revenue into more classes, trying different combinations of thresholds to maximize accuracy.

Due to the distribution of our dataset, the movie revenue values were distributed non-linearly. Movies tended to either earn millions of dollars or only a few thousand. We attempted to normalize the exponential nature of these values by classifying based on the log of our revenue attribute. However, we found that this had little effect on our accuracy across all learners, so we decided to continue using the raw revenue values.

THE INVESTIGATION

	Original Data Training (10-fold CV accuracy)	New Data Training (10-fold CV accuracy)	New Data Testing (Testing accuracy)
ZeroR	**52.07%**	**51.99%**	**52.27%**
Random Forest	**82.40%**	**80.66%**	**82.58%**
Bayes Net	**76.28%**	**75.95%**	**76.35%**
Multilayer Perceptron	**77.80%**	**79.08%**	**78.25%**
Regression	**81.92%**	**80.80%**	**80.68%**
K-Nearest Neighbor	**71.64%**	**70.47%**	**72.44%**

Figure 1: Table of results obtained for different classifiers and altered dataset (where "new data" indicates data including Facebook page likes).

	New Data Training using Gross Revenue (10-fold CV accuracy)	New Data Training using Log of Gross Revenue (10-fold CV accuracy)
ZeroR	**51.99%**	**50.05%**
Random Forest	**80.66%**	**81.48%**
Bayes Net	**75.95%**	**75.32%**
Multilayer Perceptron	**79.08%**	**77.54%**
Regression	**80.80%**	**80.12%**
K-Nearest Neighbor	**70.47%**	**71.70%**

Figure 2: Table of results obtained for different classifiers and altered dataset (where "new data" indicates data including Facebook page likes) when predicting the gross revenue and the log of the gross revenue

K-Nearest Neighbor
This method of classifying based on the distance between an example and its various features showed the lowest accuracy. We found that this was an ineffective learning method for our data. When we ran our test set on our learner, we only yielded a 72.439% accuracy. The reason why KNN was not very effective in classifying our dataset likely lies in the fact that movies are successful for different reasons. A movie could have a really famous director and some not-so-famous actors and have a large box office, while another movie could have a not-so-famous director but some famous actors and still have a large box office.

Random Forest
We used Random Forest as a decision tree method for training and testing on our data sets. This algorithm is an ensemble learning method, which is a weak learner that trains on several decision trees to output a classification while avoiding overfitting. This approach tended to do well on our data set, giving us a final training value of 80.66% accuracy and testing value of 82.58% accuracy, on a ZeroR of 51.99% and 52.27% respectively. We think the reason that Random Forest gave us the best results was due to its ability to prune on attributes that were irrelevant, which was more effective than our attempts to do this manually by removing each attribute and checking the cross-validation accuracies.

Multilayer Perceptron
We used multilayer perceptron as our neural network model for testing our data set. This model consists of layers of connected perceptron nodes that map weighted input data to outputs. This approach performed decently, giving us 77.80% training accuracy (in correctly classified instances) on the dataset including IMDB Facebook likes, with a slightly improved 79.08% training accuracy after real Facebook likes were included. The test accuracy with this data was 78.25%.

Overall, we expected multilayer perceptron to perform well because the hidden layers make the model more powerful and able to handle non-linear patterns. The model performed about 27% better than ZeroR.

Linear Regression

We attempted utilizing linear regression in order to predict gross revenues for movies. In this case, we did not use binary classifications, because linear regression does not use categorical outputs and instead uses continuous ones.

With the data that we had, the linear regression model did not yield high accuracy - we got a root relative squared error of 63.4741% and a correlation coefficient of 0.7728 when performing ten-fold cross validation on the training data.

One way we attempted to increase accuracy was by predicting the log value of the gross revenue, as opposed to simply predicting the gross value. We hoped that this would create data with a stronger linear relationship that could be modelled by the linear regression model. However, doing this resulted in a poorer result - both a lower correlation coefficient of 0.6389 and a higher root relative squared error of 76.9379. Since a larger correlation coefficient value and a smaller error indicates a better model, using the log of the gross revenue did not succeed in ensuring a higher accuracy model. Both of these models were however better than the baseline of ZeroR, which gave a 100% root relative squared error and a low correlation coefficient of -0.0396 when training on the gross data, and also gave a 100% root relative squared error and a correlation coefficient of -0.0374 when training on the log data.

Although we found subpar results with linear regression in our original dataset, we identified one possible problem: that the attributes that displayed the popularity of directors and actors (essentially, the attributes with Facebook likes for those directors and actors) could be misleading. The dataset utilized the likes that IMDB's pages for those directors and actors received, as opposed to the pages of the directors and actors themselves. We then obtained the number of likes these directors and actors received on their own pages, and trained and tested on that data.

Performing a linear regression on the log of the gross revenue gave a result of a correlation coefficient of 0.5833, and a root relative squared error of 81.2148%. Performing a linear regression on the gross revenue itself gave a higher correlation coefficient of 0.7569 and a lower root relative squared error of 65.3166. This indicates that the linear regression performed better on the gross revenue itself as opposed to the log of the revenue. Both of these models were also better than the baseline of ZeroR, which gave a 100% root relative squared error and a low correlation coefficient of -0.0806 when training on the gross data, and also gave a 100% root relative squared error and a correlation coefficient of -0.0387 when training on the log data.

Testing these models on the gross revenue, we got a high root relative squared error of over 300%, and another low correlation coefficient value of 0.1431. Testing these models of the log of the gross revenue, we got a correlation coefficient value of 0.1984 and a root relative squared error of 190.5455% - a much lower error than when we tested on the actual gross values. This indicates that taking the log of the gross value did not result in a very accurate prediction, but did result in a prediction that was more accurate than when we did not take the log. This is because taking the log of the gross revenue mapped our data to a more linear relationship, allowing for the model to increase its prediction accuracy.

Here are the results we received:

Linear Regression:	ZeroR Correlation Coefficient	ZeroR Root Relative Squared Error	Correlation Coefficient	Root Relative Squared Error
Original Data Training	**-0.0396**	**100%**	**0.7728**	**63.47%**
New Data Training	**-0.0806**	**100%**	**0.7569**	**65.3166**

Figure 3: Table of results obtained for linear regression technique with original and altered dataset (where "new data" indicates data including Facebook page likes)

Linear Regression:	ZeroR Correlation Coefficient	ZeroR Root Relative Squared Error	Correlation Coefficient	Root Relative Squared Error
Original Data Training with Log Classification	**-0.0374**	**100%**	**0.6389**	**76.94%**
New Data Training with Log Classification	**-0.0387**	**100%**	**0.5833**	**81.21%**
New Data Testing with Log Classification	**0**	**100%**	**0.1984**	**190.5455**

Figure 4: Table of results obtained for linear regression technique with original and altered dataset (where "new data" indicates data including Facebook page likes) when predicting the log of the gross revenue

Linear Regression	ZeroR Correlation Coefficient	ZeroR Root Relative Squared Error	Correlation Coefficient	Root Relative Squared Error
New Data Testing	**0**	**100%**	**0.1431**	**367.8479**
New Data Testing with Log Classification	**0**	**100%**	**0.1984**	**190.5455**

Figure 5: Table of results obtained for linear regression technique when testing values on altered dataset (where "new data" indicates data including Facebook page likes) and predicting gross revenue or log of the gross revenue

Classification via Regression:
This technique is able to use regression methods in order to classify data into two binary classes. After splitting out data in half into two binary classes, we noted that classification via regression gave accuracies that were much higher than ZeroR. While ZeroR gave an accuracy of roughly 50% when predicting the gross revenue and the log of the gross revenue, classification via regression was able to give an accuracy of roughly 80%, both when predicting the gross revenue and the log of the gross revenue, and both in training and testing. We believe that it performs well because Weka uses multiple regression methods for this learner - using multiple learners that may be weak can result in a high accuracy learner.

Bayes Net
This learner is a network that models the conditional dependencies between our various attributes. We thought this would be a useful method because we believe there are probabilistic dependencies between certain features, such as actor likes, budget, and gross. However, this performed among our algorithms of lower accuracy, testing with accuracy of 76.35% from a ZeroR baseline of 52.27%. This may be because our attributes did not always correlate strongly. For example, certain directors had very popular Facebook pages- Quentin Tarantino with 1,032,689 likes- and some had little to no Facebook presence at all, independent from the success of their movie.

THE RESULTS

Conclusion
We have found that a well-pruned decision tree algorithm with ensemble learning methods like Random Forest performed the best for the task of predicting box office revenue. It resulted in a 30.2% increase from ZeroR classification with an accuracy of 82.5766%. We believe that our results show promise in the binary classification of movies that will make a gross profit of $30 million. Of course, there are always improvements to be made, which are detailed as follows.

Future Development
If we were to continue iterating on our results, we would want to make a few additions to our attributes by obtaining more of our own data. We think it would be valuable to consider release month, since blockbusters tend to be released during the holiday and summer months. It would also be helpful to train on actors' and directors' net worth, as a more indicative numerical measurement of how influential certain figures are on movie success. We would also like to expand our predictions to other countries. We might want to predict the revenue of the box office's opening weekend, since a highly-anticipated blockbuster will sell more tickets than a flop within the first few days. This would help guide our problem, since movies can accumulate gross for weeks, but successful movies will have a strong showing coming out of opening weekend.

Group Member Contributions
Qiang Bi - Wrote the script to scrape Facebook likes, obtained the IMDB 5000 data set
Sameena Khan - Trained learners on training and testing set, report write-up
Richie Lee - Cleaned data for training and testing data sets
Sarah Wong - Trained learners on training and testing set, final website, report write-up
Nancy Yao - Trained learners on training and testing set, final website, report write-up