CMPT 353 Final Project

Matthew (301430325), Danny (301449063), Chanson (301429611)

Problem Addressed

This analysis investigates the relationship between various factors and the average rating of movies. Specifically, the analysis aims to determine whether there is a significant difference in average rating between genres, whether the average rating of movies tends to increase or decrease over time, and whether other factors such as runtime and year of release can predict the average rating of a movie.

The goal of this analysis is to evaluate the relationship between various factors and the budget of movies. Specifically, the analysis aims to determine whether there is a significant difference in budget between different genres.

Contributions

This analysis and report are split into three parts: Collection, Analysis, and Prediction. Each person completed one part of the analysis and provided their findings for the report.

Matthew	Danny	Chanson
Collection	*Analysis*	*Prediction*

Collection

Collection was split into two parts: Conversion and Filtering.

IMDb Data

The data used in this analysis comes from IMDb's Non-Commercial Datasets, which are a few TSV (Tab-Separated Values) files which contain information on different IMDb listings. The entire dataset is split among several files, which are grouped into listing type: Title and Name. This distinguishes listings from being a piece of media or a person.

Converting Data

Thankfully, Pandas provides the option to specify a "separator value" when reading CSV files, so importing a TSV file was a matter of providing an additional argument in the standard *read_csv* function.

As an additional measure, we created a script to convert the TSV files into CSV files. Through testing, there wasn't a considerable time difference in reading a TSV versus CSV. However, we had concerns over potential reading/string errors when continuously reading the TSV files. The script only generates the converted CSV files if it doesn't detect the existence of it already. Additionally, we provide the CSV files for download in addition to the TSV files so the runtime of the data collection could be optionally reduced.

Filtering Data for Movies

Besides some of the data being invalid for our analysis, the considerable size of the dataset needed to be minimized to increase efficiency/runtime.

Because of the nature of IMDb, many listings were not movies; the dataset contained TV shows, web shows, etc. Thankfully each of the listings had *titleType* tag which let us filter out anything that wasn't a movie.

Removing Invalid Listings

Though each listed movie had a title and information, there were several movies that didn't have data for specific value such as date or runtime. Though not all every attribute of a listing was necessary for our analysis, we still needed to trim the data from its large size so we removed these listings as well.

IDMb format their dataset so any unknown/invalid value is listed as "\N". Removing the data was a matter of finding any row that contained the value in as their *startYear*, *runtimeMinutes*, or *genres* attribute.

Additionally, some movies didn't have ratings (which were stored on another TSV file). As a necessary attribute to our analysis, this was a matter of removing *NaN* values.

Reducing Data Size

Even after removing invalid data, our dataset was still too large. So, to reduce the size even more we established parameters that filtered out some movies.

Our collection only contains movies released from *1995* until now. Although this removes several classic movies, the rating score and other attributes are more accurate/consistent for listings closer to the present.

We made sure to only include movies whose ratings were based off *50* or more votes. This was a important filter regardless of data size, as the rating score for these listings are likely biased or skewed due to low sample size.

Produced Collections

Our collection script produces two CSV files, a 90/10 split of the filtered.

tconst	primaryTitle	startYear	runtimeMinutes	genres	averageRating	numVotes	directors
tt7561364	Bittersweet Symphony	2019	80	['Drama']	3.7	165.0	['nm2199177']
tt3089978	Sorrow and Joy	2013	107	['Drama']	7.0	1270.0	['nm0540295']
tt1249414	Must Read After My Death	2007	73	['Documentary']	6.5	193.0	['nm1840452']
tt2009436	Black Block	2011	76	['Documentary']	7.6	95.0	['nm4571634']
tt7594568	Your Mother Should Know	2018	92	['Comedy', 'Drama', 'Family']	6.0	551.0	['nm4643680']
...	...	...	...	...	...	...	...
tt3685586	The Midnight Man	2016	105	['Comedy', 'Crime', 'Thriller']	5.1	854.0	['nm2952421']
tt1180311	The Flower of Kim Jong Il	2009	75	['Documentary', 'Drama']	6.7	891.0	['nm2912984']
tt3916354	Tutto molto bello	2014	90	['Comedy']	2.7	429.0	['nm0749303']
tt1649763	Hemo	2011	83	['Drama', 'Horror']	3.5	56.0	['nm3597735']
tt1488015	One Nation Under God	2009	91	['Documentary']	6.4	79.0	['nm3559783']

Example display of filtered dataset produced.

Analysis

We divided the analysis into two scripts.

IMDb Script

The script uses the IDMb data frame (the data collected), which contains seven columns of attributes; *title*, *startYear*, *runtimeMinutes*, *genres*, *averageRating*, *numVotes* and *directors*. The purpose of this analysis is to investigate the relationship between the various attributes and the rating of each movie, in order to determine if movie ratings are affected by genre, release year, etc. Specifically, the analysis aims to determine whether there is a significant difference in average rating between different genres, whether the average rating of movies tends to increase or decrease over time, and whether other factors such as runtime and year of release can predict the average rating of a movie.

TMDb Script

We also collected additional data which contained additional information. We analyzed this TMDb data with the goal of investigating the relationship between the genre and the budget, to see if a more popular genre tends to have more budget/investment into it compared to less popular genres. Specifically, the analysis aims to determine whether there is a significant difference in budget between different genres.

Techniques Used

The *genres* in the TMDb data were converted from genre IDs to genres names using a *convert_genres* function. The data was then cleaned by dropping unrelated columns.

We then analyzed the data by performed several tests on both our datasets, using variety of statistical techniques.

- A normality test was performed on the data to determine which variables were normally distributed.
- An ANOVA test was then performed to determine whether there was a significant difference in *averageRating* between different *genres*.
- A pairwise Tukey's HSD test was used to compare *averageRatings* of different *genres*.
- Performing regression between *averageRating* and *startYear* to determine whether the average rating of movies tends to increase or decrease over time, and whether the year of release can affect the average rating of a movie.
- A U-test and a chi-square test were also performed on the data that was proven to be not normally distributed. To be more specific, we want to see whether certain directors are more likely to work on movies of certain genres.
- In the TMDB data, two statistical techniques were used to analyze the data. A normality test was performed on the data to determine whether the budget variable was normally distributed, then we did an ANOVA test to determine whether there was a significant difference in budget between different genres.

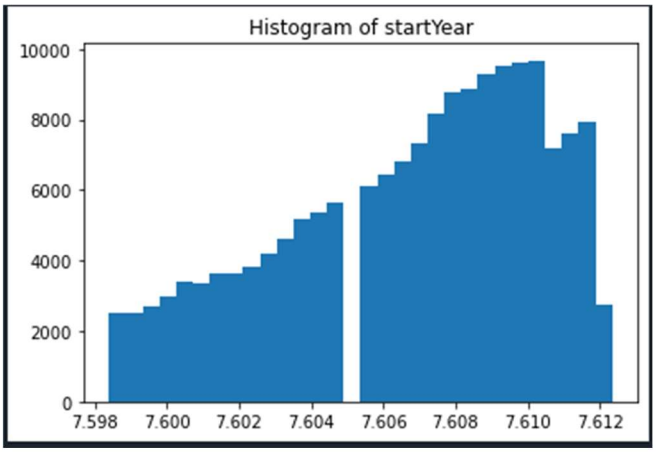
OLS Regression Results

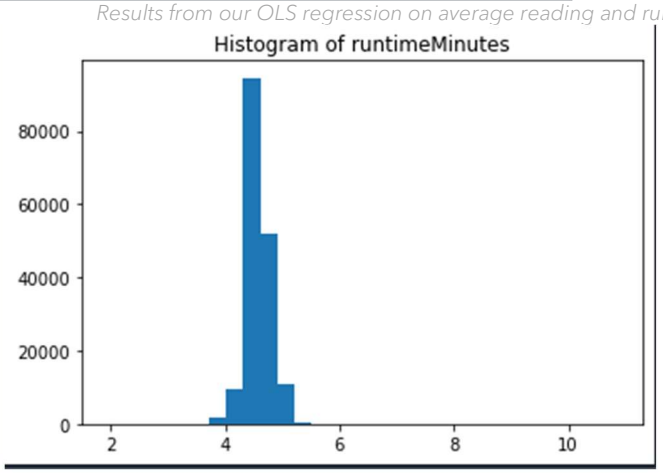
Dep. Variable:	y	R-squared:	0.000
Model:	OLS	Adj. R-squared:	0.000
Method:	Least Squares	F-statistic:	14.29
Date:	Fri, 04 Aug 2023	Prob (F-statistic):	0.000157
Time:	14:28:43	Log-Likelihood:	-2.9075e+05
No. Observations:	169638	AIC:	5.815e+05
Df Residuals:	169636	BIC:	5.815e+05
Df Model:	1		
Covariance Type:	nonrobust		

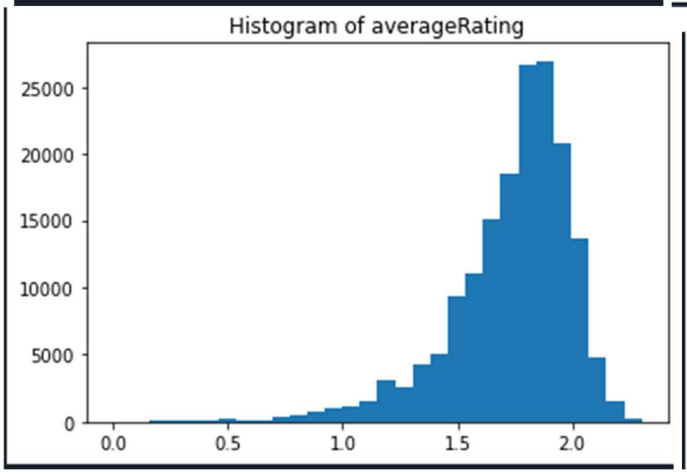
	coef	std err	t	P>|t|	[0.025	0.975]
x	-0.0017	0.000	-3.781	0.000	-0.003	-0.001
one	9.2408	0.888	10.405	0.000	7.500	10.982

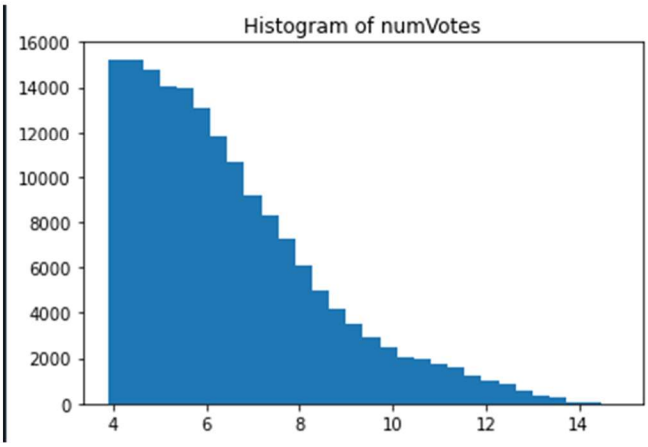
Omnibus:	6262.711	Durbin-Watson:	1.053
Prob(Omnibus):	0.000	Jarque-Bera (JB):	6975.755
Skew:	-0.492	Prob(JB):	0.00
Kurtosis:	3.139	Cond. No.	5.48e+05

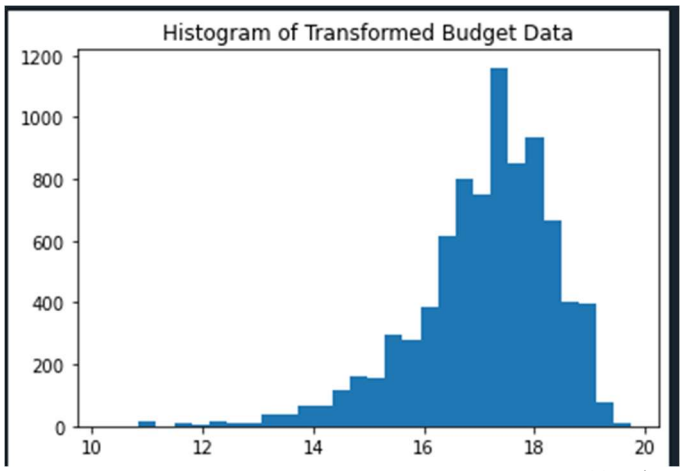
Results from our OLS regression on average reading and runtime.











Visualization of our results from the normality test.

Results & Findings

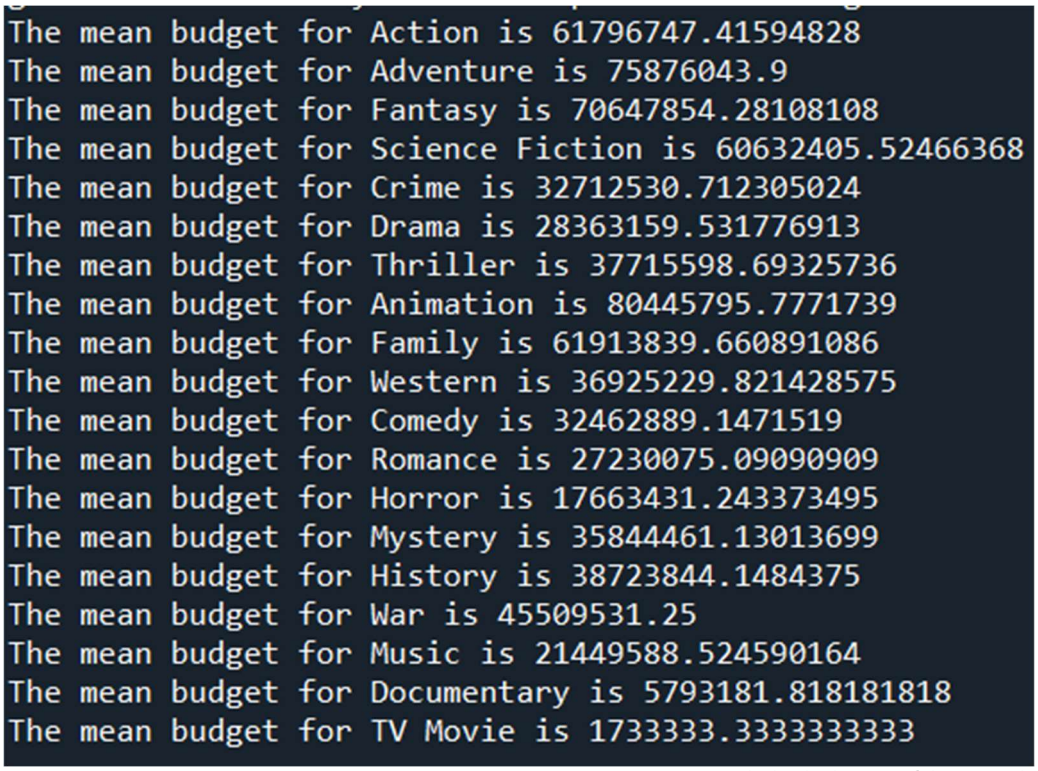
The IMDb results of the normality test showed that only the average rating variable was normally distributed. The ANOVA test found that there was a significant difference in average rating between different genres. The Tukey's HSD test provided further information about which genres had significantly different average ratings. Since there is a significant number of pairs to be compared, we focus on two of the most widely known genres: Action and Adventure. The results showed a significant difference when comparing both Action and Adventure with other genres. For example, the mean difference in average ratings between the Adventure and Animation genres is *0.4873*, and the p-value is *0.0*, indicating that there is a statistically significant difference between these two groups. Thus, according to our data, movies of the Adventure genre have significantly different average ratings compared to movies of the Animation genre. Similarly, Action movies have significantly different average ratings compared to Adventure movies. In real life, it means that there are significant differences in how audiences rate movies in both genres.

Our regression analysis showed that there was a relationship between average rating and the release year, as well as between average rating and the run time. Specifically, the results suggest that there is a significant but negative relationship between average rating and release date, hence we can say that movie releases in more recent years tend to have very slightly lower ratings than movies released in earlier years; movies with longer run times are more likely to have higher ratings than movie with shorter runtime. However, by doing a linear regression again on the release year and the average rating, we found that movies released in later years tend to have higher average ratings than movies released in earlier years. Based on what we found, we conclude that linear regression is more accurate.

By performing U tests on average values of Action and Adventure movies, we found that the average rating of Adventure movies is more diverse than Action Movie, which can also means that the score ofAdventure movies vary from low to high score more than Action movies.

Finally, our Chi Square tests show that certain directors are more likely to work on movies of certain genres, which we may conclude that most directors are best at working on their strong genre rather than other genres.

In the TMDB data, The ANOVA test found that there is a significant difference in budget between different genres. In other words, the genre of a movie can have an impact on its budget, since we found that more popular genres tend to have a higher budget and more investment on them.



Detailed calculations of the mean budget for each genre.

Limitations

Since the normality test failed on most of the categories in the IMDb dataset, we were not able to do more analysis on the relationship between each column. Another limitation of this analysis is that we only consider movies from a specific year range since the original data is too large. This may limit the generalizability of the results to other time periods. Additionally, the analysis only considers a limited number of variables and does not account for other factors that may influence the average rating of movies. We also have the same limitation for our TMDb dataset, in which this analysis only considers movies from two specific datasets (the TMDb and TMDb 5000 datasets). If we had access to a more detailed database, we could do more statistical analyses on the cast members, the revenues, and the average rating to see if more well-known cast usually starred in movies that have better revenue and ratings.

Prediction

For our prediction model, we only used the TMDb dataset.

Data Cleaning

With the TMDb data we decided to only include English movies and movies with at least 50 ratings. We also added 2 attributes, *profit* and *month* the movie was released. Since the data included lists in JSON, we had to convert the JSON to Python lists.

Problem Refinement

Our goal is to predict the success of a movie before the movie is released based on the details of the movie.

Before we can do any calculations, we must determine what we mean by success,

- Profit
- Ratings
- Revenue
- Number of ratings

In a perfect world where the cost of a movie ticket is the same and the percentage of user ratings per number of watches is the same, then the revenue should correlate to the number of ratings (surprise, it does not). Due to this and other factors we will use profit and ratings as a measurement of success.

Just to make sure, we check whether there is a correlation between profit and rating, which we discovered there was not.

Decision tree visualization.

Data Gathering

The relevant attributes that we have available to determine the success of a movie are: *budget*, *genres*, *keywords*, *production_companies*, *release_date*, *runtime*, and *cast*.

Since we want to predict an unreleased movies rating, we determined that *release date* is not a relevant attribute, however the month it releases can be important as events like Christmas can cause Christmas movies to be more popular.

Cast is a very important attribute in determining the success of a movie, there are a ton of movies that prioritize famous actors to attract global attention. Unfortunately, with the amount of cast members a movie can have, such as *Avatar* with 83 cast members, we would have a total of 283 different sets for only one movie alone, which is not plausible for us. We also felt like exploding the cast members would not be an accurate representation of a movie, so instead we decided to remove the attribute entirely.

Like what we did with the IMDb data, we create the power set of the genres attribute. With keywords we did not believe that different combinations of keywords would really affect the result, so we decided to explode it instead. Same goes with *production_companies*.

Preprocessing

Since machine learning models do not accept categories as input, we had to encode *genres*, *keywords*, *production_companies*, and *month*. Fortunately, the dataset includes IDs for each attribute and *month* can just be represented by its numeric counterpart. The only exception is with *genres*. Since we used a list of *genres,* we had to first convert it to a string, then encode it.

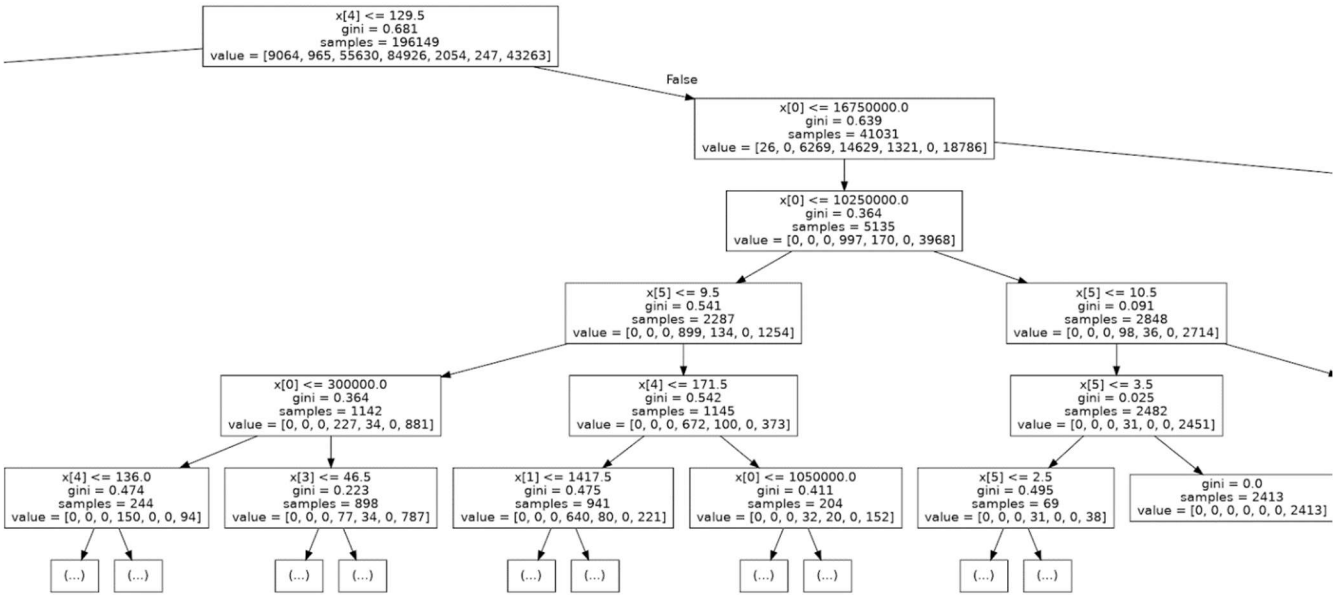
Another problem we faced was with movies without any keywords, which were shown as *NaN*, so we replaced all *NaN* values with the ID of 0 since that ID was not being used.

Profit Predictions

For our goal of predicting profit, we used 3 different regression models, K-nearest neighbors, decision trees, and random forest. The results for K-nearest neighbors resulted in the model scoring in the 75% range while decision trees and random forest had a near 100% score.

Rating Predictions

For ratings, to normalize it between 0-5, we had to categorize arbitrarily. We used the ratings of [0-5): Bad, [5-7]: Average, and (7-10]: Good. We used the same models but their classifier counterpart and for the results for K-nearest neighbors resulted in the model scoring in the 73% range while decision trees and random forest had a practically 100% score.



Decision tree visualization.

Analysis of Results

We were quite skeptical of this result. The k-nearest neighbors seemed more accurate to what we believed was true, in contrast decision trees and random forests seemed completely embellished. There were two reasons we could come up with to explain the results, the first was that in the documentation of decision trees, it states that it is "able to handle both numerical and categorical data. However, the scikit-learn implementation does not support categorical variables". I am unsure what the differences were between categorical data and categorical variables, but this might be affecting the results. The other possibility, although highly unlikely, is that it is just that simple to predict the success of a movie.

Prediction Conclusion

There were many things we could have done differently, using cast members, using different machine learning models, better data separation on attributes on lists, and obtaining more

knowledge on what we are able and unable to do. Nevertheless, I believe that with the time constraints and my current ability, I did a very good job on my data analysis.