Predicting Box Office Hits

Group Three: Micaela Geiman, Jacob Dineen, Ruben Suzara,
Sam Harvey, and Fiona Erickson

Objective 1

Understand the relationship between features in our dataset against box office performance.

Objective 2

Understand how a prospective movie concept relates to previously collected data using a nearest-neighbor approach.

Objective 3

Use visualizations and correlation matrices to better understand who and what drives box office performance.

Process for Analysis and Prediction

We'll use an open source data set of **5000** films (including variables such as budget, gross, ratings, titles, etc) to evaluate what learnings a movie studio could take away from this data and how it could help them make business decisions moving forward.

Tools needed: Python, Weka, R, Excel, XLStat

Machine Learning Techniques: Logit Models, Neural Nets, Natural Language Processing/Text Mining, Clustering, Association Rules/Recommendation Engines

Resources: StackOverflow, R/Py documentation, Course Materials

Data found here. A popular data cleaning script was used to 'normalize' the dataset, reverting it to its pre-scraped format - The script focused on pulling features from within data dictionaries (Json).

Research Plan

Subprojects will contain elements of machine learning (descriptive/predictive/prescriptive analytics), statistical analysis, marketing recommendations and ad-hoc visual analysis pertaining to each subproject goal. Marketing recommendations/solutions will stem from said analysis in a variety of different forms. Some potential ideas:

1. Content Based Recommendation Engine based on plot keywords, actors, directors, popularity/voting score.
2. Analyze which factors are significant in predicting profitability or box office gross.
3. Create visualizations on movies released by year, trends in popularity across genre and time.
4. Correlation analysis to understand which features share relationships, and exposing multicollinearity -
5. Standard time series analysis to understand and visualize the market over time.

Descriptive Visualizations

Before digging too deep into statistical methodology, it is beneficial for us to analyze and visualize some basic descriptive measurements of our data, and graphical portray information.

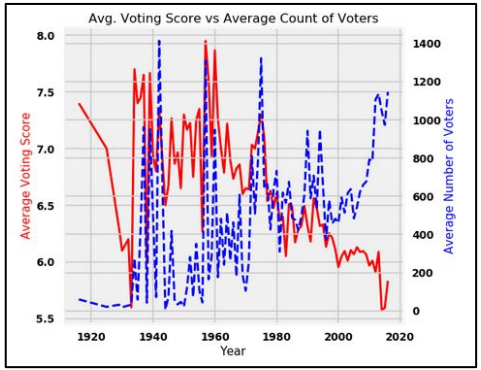


Figure 1.1

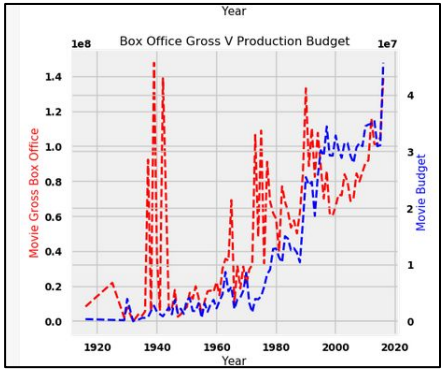


Figure 1.2

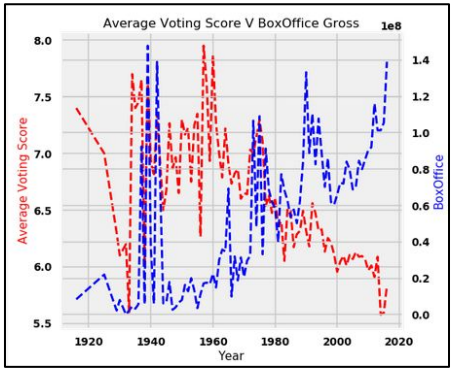


Figure 1.3

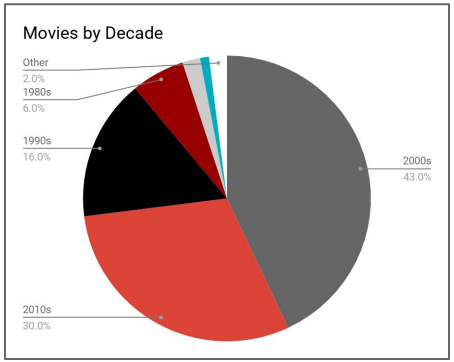


Figure 1.4

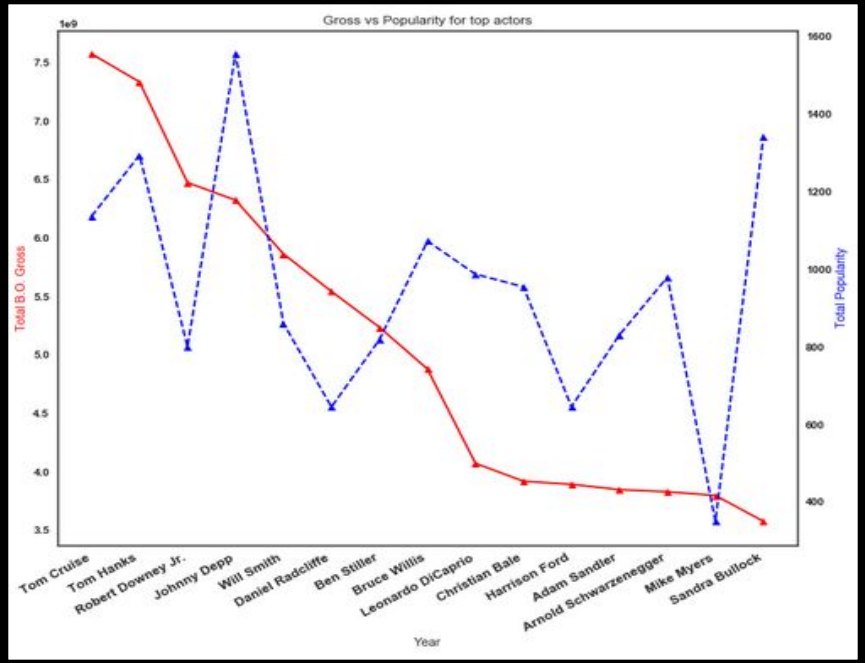


Figure 1.5

Descriptive Visualizations

Looking at the top 15 actors by total box office gross (of the 5k movies in the dataset) we can begin to see which actors might have a positive response towards their own box office performance. We can also delineate between actors who bring in money, and actors who might bring in critical acclaim as well (Johnny Depp, Sandra Bullock). This is an aggregate, however, and doesn't account for notions of central tendency.

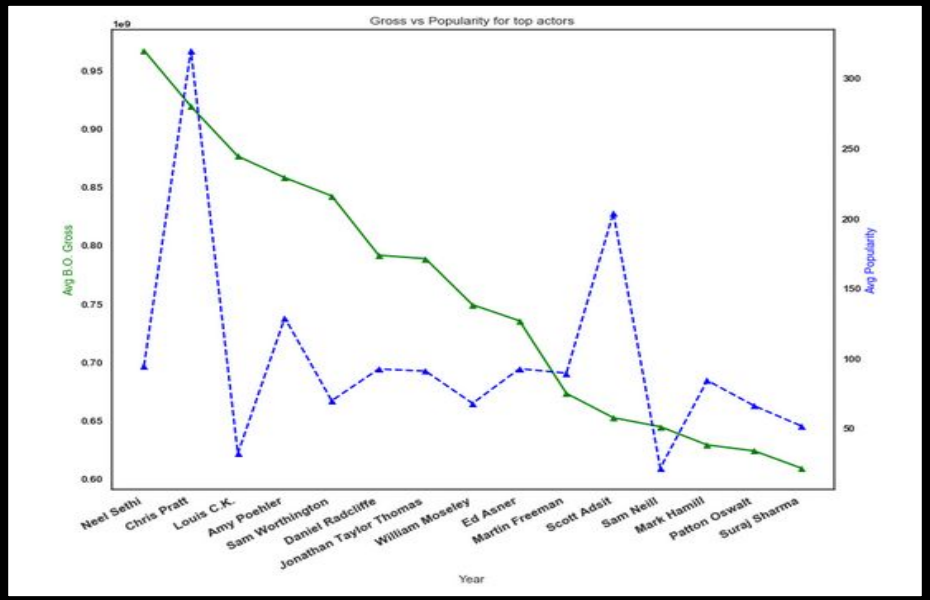


Figure 1.6

Descriptive Visualizations

It is likely better to look at averages to account for appearances in the top 5k movies. Above we can see the most popular movie stars, pulled from a descending list of average box office gross.The names are more obscure, but these might be, from a studio perspective, the best bang for their buck.

Text Analytics: WordClouds

Word clouds can help us to measure the density* of specific words within a vector of observations. In this case, our text data is representative of the way that movies are marketed (plot keywords, synopses) and the cast relative to each observation.

Perhaps the most intuitive take away from these wordclouds are seen in the distribution by Actor. Since the data includes top rated movies, we can begin to form ideas about Lead Actors in further analysis aimed at predicting profitability or user ratings.

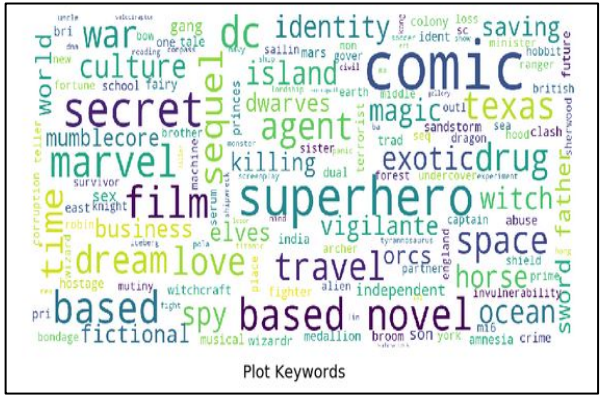


Figure 2.1

Figure 2.2

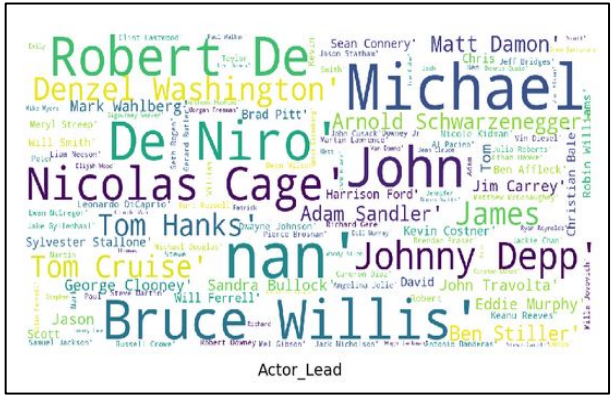


Figure 2.3

*The size of a word within a wordcloud is representative of the density of the distribution of the specific word/phrase.

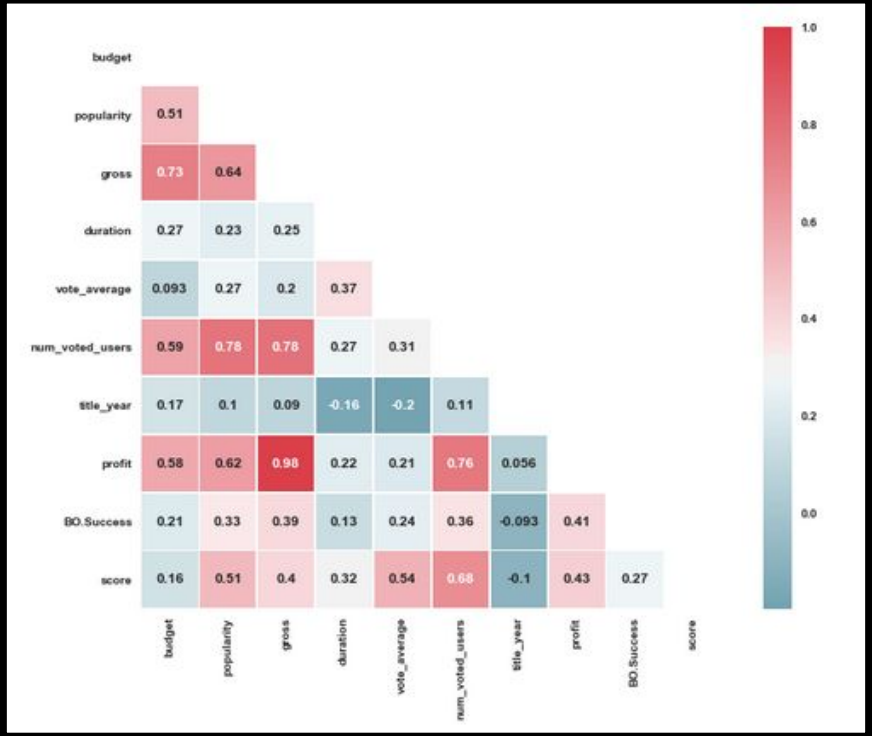


Figure 2.4

Analytics: Correlation

Next, we examine non-causal relationships present between numerical features in our dataset. Collinearity is to be expected in some regard, due to feature engineering, but we'll deal with that as we move towards building models.

- Gross box office/Profit/And our dummy var for box office success are all highly correlated with popularity (0.5 or above).
- Duration/Length of a movie has little impact on performance indicators.
- Title Year doesn't have a relationship with whether or not a movie makes money, but that could be due to the fact it's sequential.
- Conversely, average ratings have declined over the years, as new movies aren't as highly regarded as older movies.

Box Office Performance Logistic: Prediction Preparations

We use binarized representations of categorical data (text) in order to predict box office performance in cases where we are dealing with incomplete information. What we mean by incomplete information is that we wouldn't generally have access to user ratings and popularity metrics before a movie is released. Using only the data that would be available to the general public at the time of release we focus mainly on: Release Year, Top billed actors (1-3), director, genres, and language.

We use the median of gross - budget as our delineator of our dependent variable, where if a profit falls above the median, then it is classified as 1, else 0. We experimented using the mean and '0' as thresholds, but class imbalance was severe and we weren't making models that were better than randomly guessing 1 every time.The median of the profit var == $2,547,569. So we will be predicting whether or not a movie makes at least $2.5mm more than its budget.

For context, the dummying of these variables resulted in a sparse matrix consisting of 8000 columns against 5000 observations. After splitting this up into a train and test set, we fit our trained model against our test set. *This has to be done with more powerful statistical software due to the size of the dataset.

Box Office Performance Logistic: Prediction Results

Accuracy can be seen below. While 72.61% isn't the greatest sign of a model's ability to generalize, it is an improvement on randomly guessing whether or not a movie will be successful or not. For comparison's sake, if we opted to use tmdb data that we wouldn't have at the time of a movie release - for instance, if we were trying to predict future final box office performance after a few weeks being in theatres and garnering ratings, we could improve the accuracy of our projections by 8%:

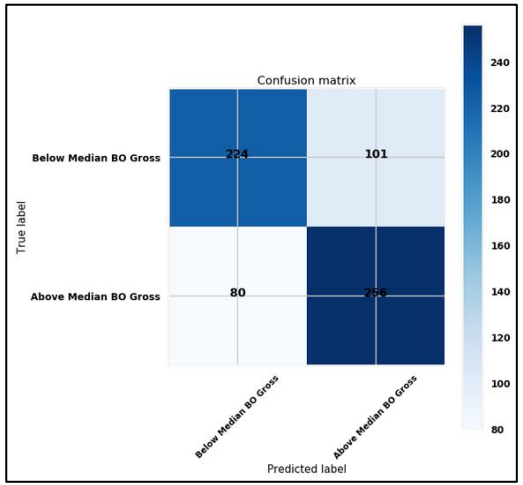


Figure 3.1

Data attainable before release

0.7261724659606656

	precision	recall	f1-score	support
0	0.74	0.69	0.71	325
1	0.72	0.76	0.74	336
avg / total	0.73	0.73	0.73	661

Figure 3.2

Data attainable only after release

0.8063540090771558

	precision	recall	f1-score	support
0	0.79	0.82	0.81	325
1	0.82	0.79	0.81	336
avg / total	0.81	0.81	0.81	661

Figure 3.3

Feature Importance

Below we can see by feature results of which categorical representations mean the most to a model when predicting profitability. To summarize, if a movie has any combination of the features shown below, it is very likely that the result of the logistic function (1/1+exp(-utility)) will be close to one, meaning a prediction of class == profitable.

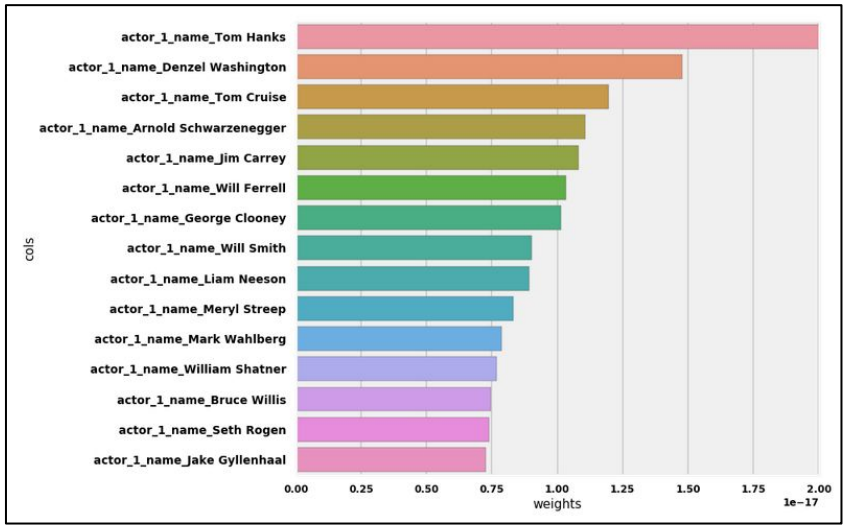


Figure 4.1 Lead Actors

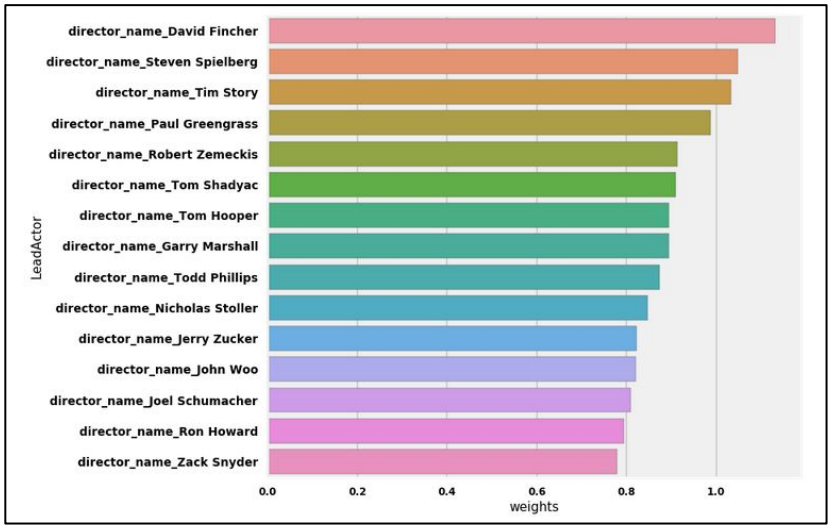


Figure 4.2 Directors

Feature Importance

Below we can see by feature results of which categorical representations mean the most to a model when predicting profitability. To summarize, if a movie has any combination of the features shown below, it is very likely that the result of the logistic function (1/1+exp(-utility)) will be close to one, meaning a prediction of class == profitable.

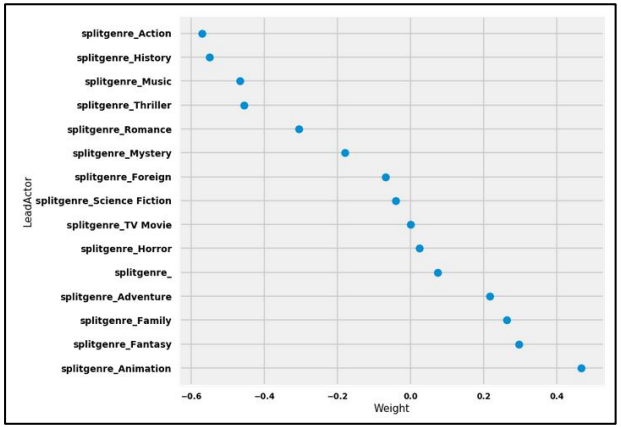


Figure 4.3 Genre

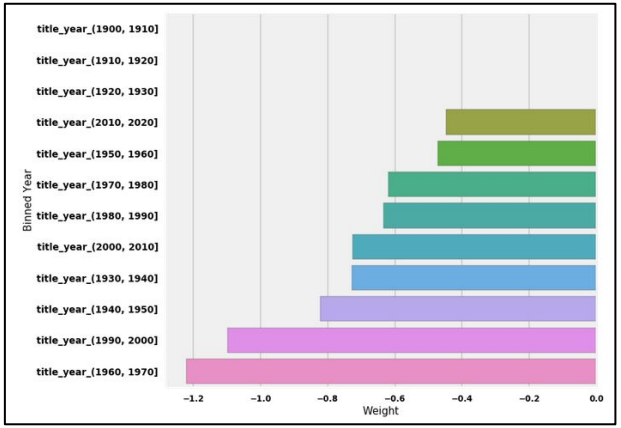


Figure 4.4 Release Year

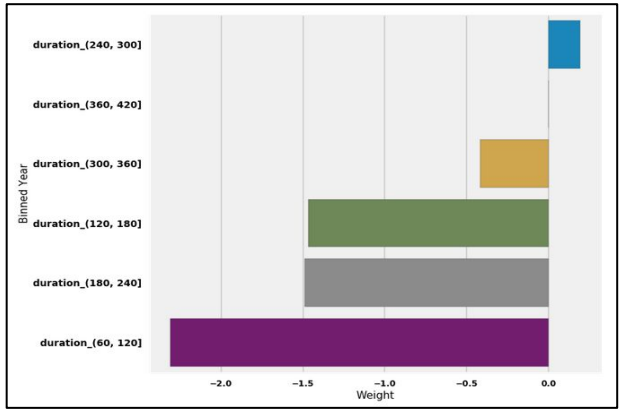


Figure 4.5 Duration

Manual Predictions

Budget is essentially moving most of the needle on our predictive capabilities before logging it. Even with logging it, the resulting value of the dotproduct between the weight and the value is responsible for most of the resulting value of the utility function.
In order to truly test the predictive power of our model, we should experiment on data that is outside of the range of our domain (also known as interpolation). This example will center around predicting binarized box office profitability for Marvel's Infinity War based on the features we included in our fitted model.

weights	cols	Value	Coeff * Weights
0.603180021	budget	19.33697148	11.66367486
-1.468117775	duration_(120, 180]	1	-1.468117775
-0.447125567	title_year_(2010, 2020]	1	-0.447125567
-0.571222416	splitgenre_Action	1	-0.571222416
-0.568149229	language_English	1	-0.568149229
-0.050034007	actor_1_name_Robert Downey Jr.	1	-0.050034007
0.244261159	actor_2_name_Chris Hemsworth	1	0.244261159
-0.155162019	actor_3_name_Mark Ruffalo	1	-0.155162019
0.034716634	director_name_Anthony Russo	1	0.034716634

Summation of Weights * Coeff	8.682841638
Probability	0.99983056
Predicted Class	1
Actual Class	1

Figure 4.6

Alternative Box Office Performance: Linear Regression Planning

Logistic regression is great for measuring categorical responses, but when dealing with financial data, a linear model with a continuous output variable is often preferred. Here, we will model the numeric features in our dataset against total gross. Because of our lack of data, we will use features such as average vote, number of votes, and popularity, which wouldn't be available to us at the time of movie release, but might be during the middle to end of the theatrical run- This would be useful to predict the kind of 'legs' a movie will have, and where it ultimately will end up as far as total box office gross.

Model parameters (LN Gross):

Source	Value	Standard error	t	Pr > |t|	Lower bound (95%)	Upper bound (95%)
Intercept	26.115	9.556	2.733	**0.006**	7.376	44.853
LN Budget	0.799	0.019	41.315	**< 0.0001**	0.761	0.837
duration	-0.003	0.002	-2.031	**0.042**	-0.006	0.000
num_voted_users	0.000	0.000	11.909	**< 0.0001**	0.000	0.000
title_year	-0.011	0.005	-2.339	**0.019**	-0.020	-0.002
popularity	0.004	0.001	4.006	**< 0.0001**	0.002	0.006

Equation of the model (LN Gross):

LN Gross = 26.1148359898925+0.799104412674256*LN Budget-3.07020642119641E-03*duration+3.43264423455301E-04*num_voted_users-1.11134253868209E-02*title_year+4.30718692494366E-03*popularity

Figure 5.1

Alternative Box Office Performance: Linear Regression Results

In order to determine the predictive ability of our model, we have to use our fitted model on our test set. After we predict natural log of box office gross, we can compare it against the actual natural log of box office gross. We then compute the error between actuals and predictions. We'd normally determine a model's ability through the result of its loss function, in this case RMSE, but that's not necessary here.

Towards the higher and lower ends of the box office spectrum, we struggled to predict actual performance.

movie_title	LN Budget	duration	num_voted_users	title_year	popularity	Predicted Gross	LN Gross	Error	Pred Gross	Actual Gross	Error
E.T. the Extra-Terrestrial	16	115	3269	1982	56.105798	18.02	20.49	2.47	66,837,342	792,910,554	726,073,212
Star Wars	16	121	6624	1977	126.393695	19.55	20.47	0.92	308,270,431	775,398,007	467,127,576
Ghost	17	127	1339	1990	41.967005	17.76	20.04	2.28	51,634,219	505,000,000	453,365,781

Figure 5.2

Alternative Box Office Performance: Linear Regression Results

Our model, however, was relatively decent at predicting values within the range of 50mm-100mm in actual box office gross:

movie_title	LN Budget	duration	num_voted_users	title_year	popularity	Predicted Gross	LN Gross	Error	Abs Error	Pred Gross	Actual Gross	Error
Predator 2	17	108	730	1990	27.934978	17.92	17.86	(0.06)	0.06	60,585,803	57,120,318	(3,465,485)
Apocalypse Now	17	153	2055	1979	49.973462	18.37	18.31	(0.06)	0.06	94,983,592	89,460,381	(5,523,211)
2001: A Space Odyssey	16	149	2998	1968	86.201184	18.11	18.05	(0.06)	0.06	72,971,187	68,700,000	(4,271,187)
The Abyss	18	139	808	1989	24.961625	18.40	18.32	(0.09)	0.09	98,284,116	90,000,098	(8,284,018)
The Last of the Mohicans	18	112	732	1992	37.776566	18.03	18.14	0.10	0.10	67,988,225	75,505,856	7,517,631
Arachnophobia	17	103	433	1990	15.895661	17.68	17.79	0.11	0.11	47,876,583	53,208,180	5,331,597
The Terminator	16	108	4128	1984	74.234793	17.99	18.18	0.18	0.18	65,291,196	78,371,200	13,080,004
Trading Places	18	116	738	1983	34.022209	18.12	18.32	0.20	0.20	74,058,918	90,400,000	16,341,082
Superman III	17	125	490	1983	22.164202	17.92	18.14	0.22	0.22	60,879,454	75,850,624	14,971,170
Groundhog Day	16	101	2301	1993	52.744331	17.86	18.08	0.22	0.22	56,807,054	70,906,973	14,099,919
Tango & Cash	18	104	458	1989	22.787667	18.19	17.97	(0.22)	0.22	79,289,031	63,408,614	(15,880,417)
The Untouchables	17	119	1384	1987	38.272889	17.92	18.15	0.23	0.23	60,568,466	76,270,454	15,701,988
Jaws: The Revenge	17	89	224	1987	12.777008	17.44	17.76	0.33	0.33	37,384,184	51,881,013	14,496,829

Figure 5.3

Our results suggest that the variance in the response variable is not fully accounted for by the variance in our input variables. It would be of interest to binarize all categorical variables (actors/directors/etc..) and see if that reflected a better approximation of a linear function. Also, because most of the box office actuals the model was built on are between 0-100mm US dollars, we have a hard time modeling outliers, which are represented by movies that gross far above the central tendency of the data.

Recommendation Engine: Ascertaining Similarity to Drive Decision-Making

For the purpose of this dataset, a recommendation engine would be useful to determine similarity to past movies via a distance measurement of the noted variables. Being able to understand how past movies with similar traits were marketed, and whether or not they are successful, could lead to the difference between a mega-movie and a flop. Again, this is a game of incomplete information. User ratings and average votes aren't available before a movie is released. So as much information that can be derived to understand the nature of movie releases is helpful in decision making.

What features will we focus on to adequately describe our product?

- Movie Overview/Synopsis
- Plot Keywords
- Cast/Crew
- Popularity
- A combination of the above (merging all text based columns into a single column)

Recommendation Engine: Ascertaining Similarity to Drive Decision-Making

To implement a solution to film recommendations based on similarity, we need to choose a distance-based cost function with the intention of utilizing a nearest neighbor approach aimed at minimizing the distance for K. Because we're dealing with text, we default to cosine similarity.

1. Find the column(s) you'd like to use. We experiment with a number of different columns as isolates and concatenates.
2. Transform the text column into a tf-idf matrix. You can either use the default binarized vector creation, or the countvectorizer which includes token counts.
3. Compute the similarity of the matrix against itself.

Recommendation Engine: Ascertaining Similarity to Drive Decision-Making

Let's assume that we are the studio responsible for releasing Saving Private Ryan. We know the synopsis, the keywords that define the plot, and the cast. We want to see which movies, historically, are most similar to ours - Not only that, we want to see which made the most profit at the box office. If we find movies that are similar, we can devise a marketing plan and film rollout similar to those that proved to be successful. The most relevant films seem to be those that are closely related in all categorical measures (Synopsis + Keywords + Cast).

Schindler's List is the most related movie to Saving Private Ryan (Also the most popular of filtered results) and had the highest profit (Gross minus budget) at the box office. It would be beneficial for us to better understand the audience that saw/liked that movie and what mediums/channels to target advertising to them when creating a plan. It could also help us to roughly forecast some of the final domestic + foreign box office receipts.

```
Querying films similar to: Saving Private Ryan
Criteria for Matching:Cosine Similarity of Plot Overview w/ Box Office Profit
1 :Apocalypse Now | $57,960,381.00
2 :2 Guns | $70,940,411.00
3 :Brothers | $17,318,349.00
4 :Public Enemies | $134,104,620.00
5 :Evil Dead | $80,542,952.00
Querying films similar to: Saving Private Ryan
Criteria for Matching:Cosine Similarity of Plot Keywords w/ Box Office Profit
1 :The Imitation Game | $219,555,708.00
2 :Enemy at the Gates | $28,976,270.00
3 :Spy Game | $51,049,560.00
4 :Joyeux Noël | ($4,290,845.00)
5 :The Best Years of Our Lives | $21,550,000.00
Querying films similar to: Saving Private Ryan
Criteria for Matching:Cosine Similarity Cast w/ Box Office Profit
1 :Good Will Hunting | $215,933,435.00
2 :Furious 7 | $1,316,249,360.00
3 :Ocean's Eleven | $365,717,150.00
4 :The Iron Giant | ($46,840,695.00)
5 :Apollo 13 | $303,237,933.00
Querying films similar to: Saving Private Ryan
Criteria for Matching:Cosine Similarity of All Categorical Variables w/ Box Office Profit
1 :Schindler's List | $299,365,567.00
2 :Fury | $143,817,906.00
3 :The Deer Hunter | $35,000,000.00
4 :Enemy at the Gates | $28,976,270.00
5 :War Horse | $111,584,879.00
```

Recommendation Engine In Action

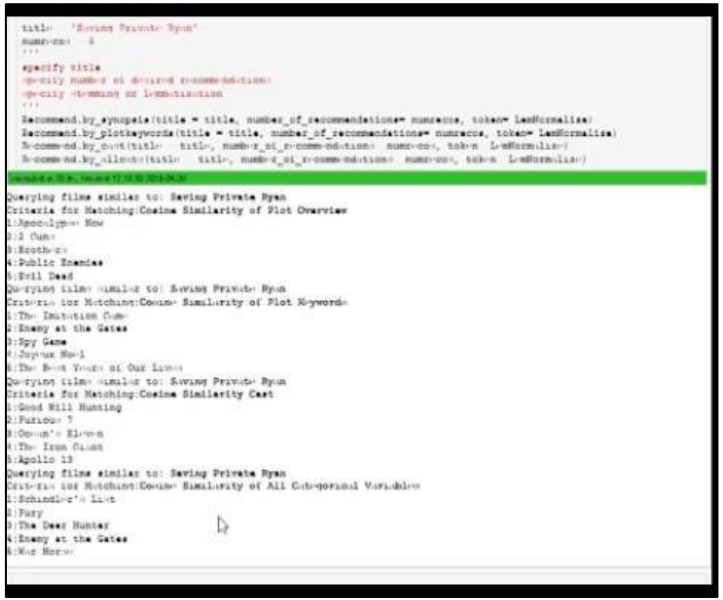


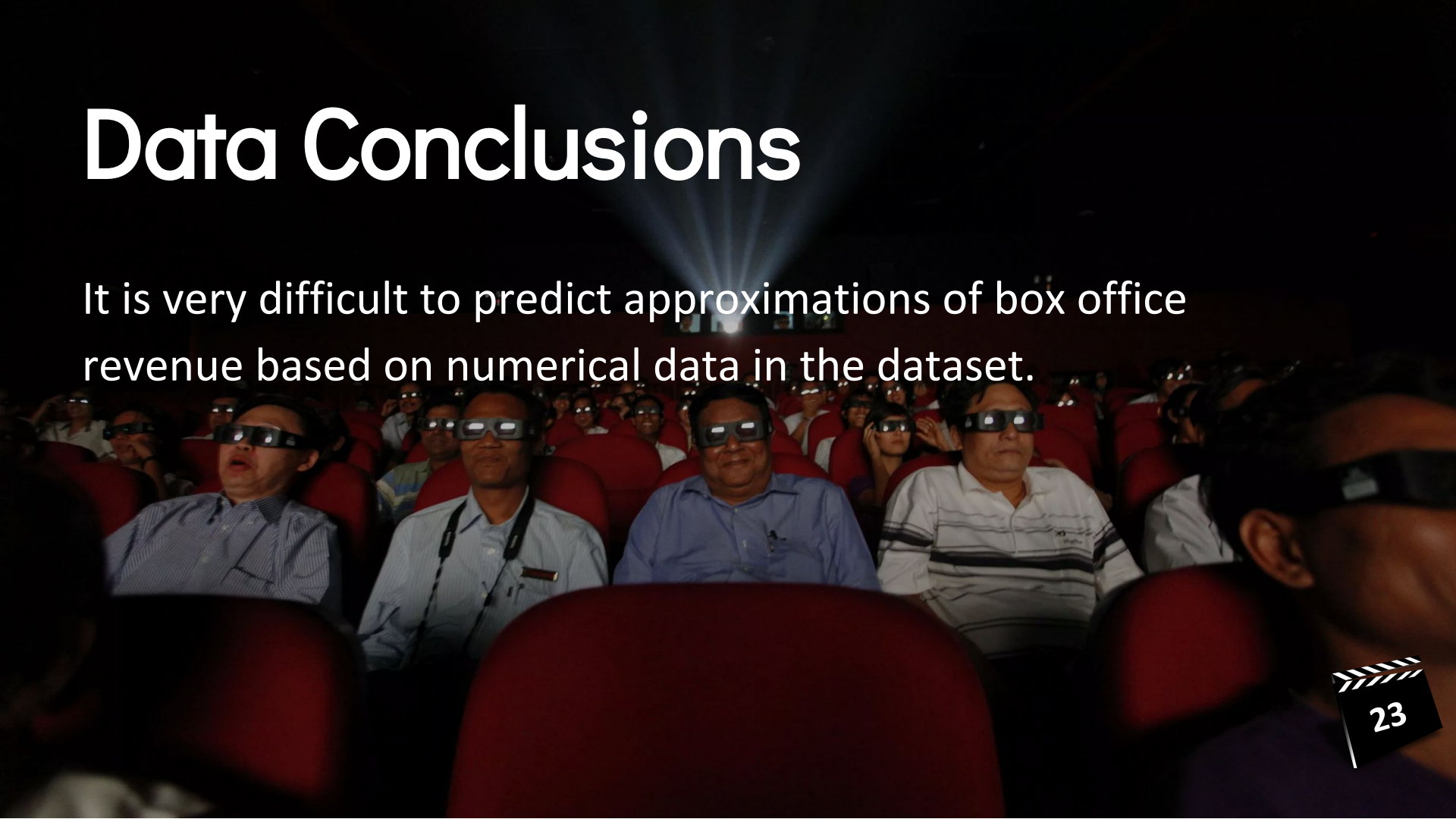
Figure 6.1

All code available per the below link. When running the code, a user must specify a movie title, number of recommendations, and the text cleaning (LemNormalize for Lemmatization, or StemNormalize for stemming). All functions live within a class and can be called by typing *Recommend.*. There are 4 functions within the class: by_synopsis, by_plotkeywords, by_cast, by_allcats.

Link to source code (.py file)

Data Conclusions

It is very difficult to predict approximations of box office revenue based on numerical data in the dataset.



Data Conclusions

A logistic model centered around profitability worked well at predicting whether or not a movie will gross more than 2mm above its box office budget or not.

Data Conclusions

The recommendation engine helps us to understand a prospective movie against past movies and get a better idea of which movies that had similar keywords, genres, actors, directors, or plots performed well at the box office. This is useful in the planning stages of production to ultimately maximize ROI.

Studio Recommendations:

- Use the recommendation engine to develop a marketing plan for new films based on previous, similar films success.
- Remember that budget is the key driver of revenue, but there are several other elements (directors, actors) that you can use to impact your success.
- Preliminary projections are possible, but much more accurate results come from directly post-release.

CREDITS/RESOURCES

Specialty Costumes by	FILM ILLUSIONS, INC.
Creatures Created by	AFX STUDIO
Creature Design by	NEVILLE PAGE
Makeup Department Head	DAVID LEROY ANDERSON
Assistant Makeup Department Head	DEBORAH PATINO RUTHERFORD
Makeup Artists	KAREN IVERSON • VERA STEIMBERG
	DON RUTHERFORD • JEANNE VAN PHUE
Makeup Effects Artists	DAVE SNYDER • BARNEY BURMAN
	JAMIE KELMAN • BRIAN SIPE
	SCOTT WHEELER

EXTERNAL SOURCES

help.imdb.com/article/imdb/track-movies-tv/faq-for-imdb-ratings/G67Y87TFYYP6TWAV#

https://www.kaggle.com/fabiendaniel/film-recommendation-engine/comments

https://www.techemergence.com/use-cases-recommendation-systems/

https://sites.temple.edu/tudsc/2017/03/30/measuring-similarity-between-texts-in-python/

LINKS TO PROJECT CHECKPOINTS

Checkpoint 1

Checkpoint 2