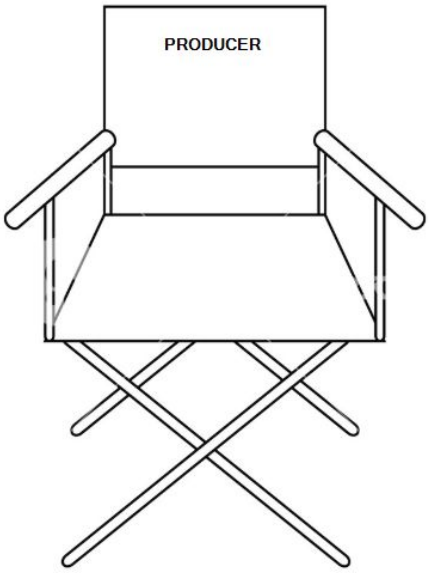


Predicting Film Box Office Revenues Pre-Release

BUSN 41204 Machine Learning
Professor Mladen Kolar, Winter 2019

Benjamin Cox
Luca Ferrara
Ian Lenny
Ramasamy Sridharan

Project Mission

"Show me the money!" Jerry Maguire, 1996

While many films are regarded as artistic masterpieces and of seminal cultural importance, the reality has long been that Hollywood is at its core a business first and artistic vehicle second. In fact, between 2000 and 2010 only 36% of films had profitable box-office returns[1]. In an increasingly global market, artistic enterprises achieve financial success only rarely, with the vast majority of projects failing to recoup expenses. However, with the advent of better data collection and analytical techniques, the ability to better anticipate the success of projects has become increasingly achievable. While revenue streams for a film are numerous, the most critical is the box office revenues, which essentially dictate any licensing and syndication fees that may follow on after theatrical showings. As such, the ability of a given studio to predict the financial success of a particular project is an essential element to success in an increasingly competitive landscape. Particularly, as studios consolidate and allocate an increasing amount of budget to a smaller number of "blockbuster" films, their ability to predict future revenues has an immense importance for their growth and even survival.

Our goal, therefore, is to use a rich dataset from The Movie Database (TMDB) to generate a predictive regression model that would add value to the core business strategy of these studios. The dataset includes a number of features about the film which all would be known *prior* to release, making application for future usage possible. Due to the industry dynamics, with increasing predictive power directly comes increased strategic power. Hence, we believe that there is a very strong business case to explore machine learning techniques in the hunt for the next Blockbuster film.

Film Industry Dynamics

The film industry, or better defined, the movie and video production industry, produce and distribute motion pictures across the globe. This is a relatively highly capital intensive, reputation and network based market that has high barriers to entry. Some of the content they produce is original, purchased from creative artists and writers via a competitive marketplace for ideas. Other content is produced using existing intellectual property, licensed or purchased directly for usage in film -- for example the recent Marvel series of films relies on a universe of characters and plotlines that are already established. Movie studios are primarily responsible for maintaining timelines of production for film and video, arranging financing, producing, publicizing, and frequently distributing the film. In 2018, this industry generated $32 Billion in revenue and $5.5 Billion in profit at a 5-year growth rate of 2.0% with 2.4% projected through 2023[2]. Estimates of global box office revenue were at $40.6 Billion in 2017[3]. This market therefore is mature, with slow growth and several headwinds in the form of competing goods, most notable of which is content streaming at home through platforms such as Netflix or Hulu. Competition within the industry is quite fierce, as consolidation between 21st Century Fox and Disney as well as Time Warner with AT&T has created a highly concentrated market, with over 80% of total revenue generated by the top six major studios, including The Walt Disney Company (34.8%), NBCUniversal Media LLC (14.90%) and AT&T (11.1%). This consolidation as well as digital distribution and content generation has improved profitability margin of the industry from 11% in 2011 to 14.6% in 2018, but has also decreased the basis of competition on operations and focused it on content. While the domestic box

---

1 "Predicting Movie Success with Data Analytics" https://storyfit.com/using-analytics-to-predict-movie-success/

2 "About this Industry", IBISWorld Industry REport 51211a, "Movie & Video Production in the US", September 2018

3 The-numbers.com via Statista as of October 25, 2019

office accounts for 26.1% of revenues for the industry, the remaining sources, distribution and television licensing, are all highly path-dependent on success in this market. The 60% of revenue generated via this licensing and global distribution therefore highly rely on domestic success, which has lead to a competitive dynamic where studios are often competing for a share of a fixed pie of consumer spend each year, resembling a zero-sum game in many respects. Indeed, the number of projects film studios are "green-lighting" each year has declined significantly, with the average budget per film increasing and the quest for global "blockbusters" becoming an industry competitive norm.[4] [Appendix A]

Global Industry

The film industry has become increasingly global, leading other forms of globalization and integration due to the ease of dissemination in digital format and the increase in consumer income globally in the past decades, a primary determinant of discretionary spending. Thus, the trend has been for more film industry revenue to be sourced overseas, which has an impact on product strategy and revenue prediction. The leading markets worldwide in 2017 by gross box office revenue were the United States at $10.24 Billion followed by China at $8.42 Billion, with countries such as Japan, the UK, South Korea, and Germany all near $1.5 Billion. Notably, however, the largest number of cinema screens worldwide are in Asia Pacific, at over 70,000 versus 42,000 in the U.S. Growth in box office revenues is largely coming from the APAC region and is a key determinant of strategy.[5] [Appendix A]

Film Trends

Films, as a product, are largely classified into genres that are defined by the plot, cinematography, actors, and production level. Action and Adventure films, such as *Guardians of the Galaxy*, constitute 60% of box office revenues, with Drama, Comedy, and Thriller/Suspense each garnering ~10% each[6]. The industry trend, as many movie-goers and analysts alike have noted, is for large, blockbuster action films typically released during the summer to dominate the box-offices. Likewise, well-known stars with high fees are contracted to bring in audiences. Franchises, accordingly, have become hugely profitable assets for these studios. For example, the Marvel Cinematic Universe generated $870 Million in US Box Office revenue and Star Wars generated $564 Million, both being properties of The Walt Disney Company. Harry Potter, AT&T's largest property, also generated an incredible $626 Million in box office revenue [Appendix A]. Indeed, each of the major Studios remaining in the industry have at minimum one such Franchise product that significantly out produces the broader portfolio in box office revenue, and therefore downstream revenue.

Along demographic lines, notable differentiation and segmentation can be done to further target film success for smaller portfolio films, but the increasingly most strategically important productions are those within the action and adventure category.

Current State of Analytics

There is strong evidence that major studios today invest a great deal in forecasting revenues and incorporate recent academic works into their processes, with a rich body of academic work providing the foundation for the most advanced methods. Creating a model to predict revenue of films, therefore, ought

[4] "Industry Performance", IBISWorld Industry Report 51211a, "Movie & Video Production in the US", September 2018
[5] Statista Box Office In the U.S. Report, IHS Screen Digest: 2017
[6] "Products & Markets", IBISWorld Industry Report 51211a, "Movie & Video Production in the US", September 2018

to be evaluated on its ability to impact this decision set for studios, and there is already a great deal of research that has gone into this. Historical modeling using traditional multilinear regression techniques have found that factors such as budget, size of studio, sequel (binary), holiday season, and award nominations all were statistically significant dependent variables.[7] Other analyses have focused on the cast of films, seeking to identify the impact of stars on outcomes and finding significance there but also with regards to the parental rating assigned.[8] Yet more modelling techniques such as Bayesian methods have confirmed that utilizing information about the market and the film itself are critical elements to creating a formal model with predictive power. [9] Research along these lines has given studios a usable guideline for predicting revenues, but recent changes in consumer tastes and the speed with which information disseminates in social networks has meant that an increasingly important factor is sentiment analysis on the web.[10] In general, research across methods has found the following to be key factors[11]:

Data Type		
Textual	Categorical	Numerical
Social Sentiment	Character Types	Ratings
Dialogue Style	Movie Distribution	Awards
Plot Complexity	Release Date / Seasonality	Budget
Reviews	Genre	
	Sequel/Franchise Status	
	Star Power	

While no other studies were found on the same data used, the preponderance of literature in this area incorporates the factors above for classification purposes rather than regression, oftentimes assigning a "success" or "failure" category based on a logical expression relating to revenue vs. budget, or binning different levels of financial success into an ordinal category; such as $1-$10 Million being Category 1, $11-$20 Million being Category 2, and so on.[12] Similar studies which sought to predict the ctegorical level of profitability of success, using a similar dataset, generated results of 88.8% accuracy using a Neural Network and 84.2% using SVM.[13] Conclusions from these studies report results here are mixed, often times feature engineering and the incorporation of several datasets is required to generate usable accuracy in prediction.

Expanding our horizon of search further, we looked to a survey research paper which compared and contrasted various learning algorithms ability to predict revenue from data very similar to our own and encompassing the "factors" aforementioned. Using 10-fold CV across Logistic Regression, Boosting (AdaBoost), Random Forest, Naive Bayes, Stochastic Gradient Descent (SGD), Support Vector Machines (SVM), and Neural Networks via Multi-Layer Preceptrion (MLP) results indicated that numerous models

---

[7] NA Pangarker & E.M. Smit "The determinants of box office performance in the film industry revisited", Stellenbosch Universtiy, 2013

[8] Ravid, S Abraham, 1999. "Information, Blockbusters, and Stars: A Study of the Film Industry," The Journal of Business, University of Chicago Press, vol. 72(4), pages 463-492, October

[9] Neelamegham, R. & Chinatagunta, P. 1999. 'A Bayesian model to forecast new product performance in domestic and international markets', Marketing Science, 18(2): 115-136.

[10] Hayden, Erik ."The Art of Predicting Box Office Gold", *Pacific Standard.* https://psmag.com/economics/the-art-of-predicting-box-office-gold-7183

[11] "Predicting Movie Success with Data Analytics" https://storyfit.com/using-analytics-to-predict-movie-success/

[12] Quader, Nahid & Gani, Md & Chaki, Dipankar. (2018). Performance evaluation of seven machine learning classification techniques for movie box office success prediction. 1-6. 10.1109/EICT.2017.8275242.

[13] Rhee, Travis & Zulkernine, Farhana. (2016). Predicting Movie Box Office Profitability: A Neural Network Approach. 665-670. 10.1109/ICMLA.2016.0117.

were feasible, with MLP demonstrating the best classification performance. Notably, “genre” and “part of a series” variables were excluded, making direct comparison more difficult.[14]

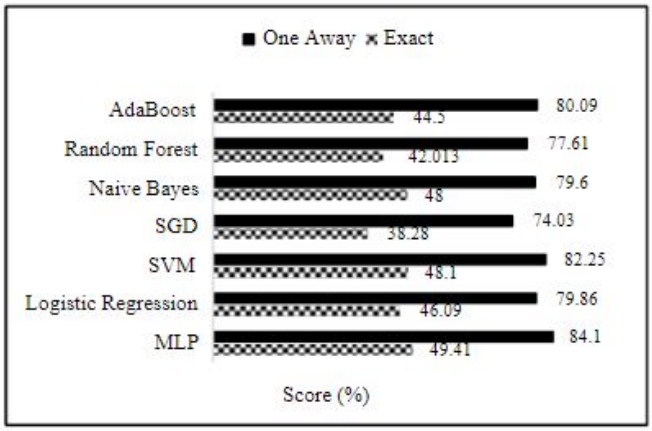


Figure 2. Performance comparison of different machine learning methods for pre-released features.

The authors conclude that depending on the dataset provided, any of the above classifiers are potentially good learners for the problem at hand but given our hope is to seek the greatest regression results by reducing RMSE, we will look to apply these findings as we see appropriate for our dataset and problem.

Other Notable Analysis

Indeed, increasingly analysts at film studios find that without human instinct involved in the decision process, purely historical modelling is an inadequate solution. Increasingly, however,leading indicators that arise from early information released about films is being used to better inform studios as they begin projects as to their eventual estimated revenue. Companies that invest a significant outlay in a data collection infrastructure therefore can utilize platforms such as Twitter, Facebook, and Instagram as well as message boards at popular sites to generate a more targeted estimation of future revenue. These all contribute to creating more real-time analysis of what is referred to as “hype”, “buzz”, or social sentiment. Unfortunately, because this information comes later in the stage, pre-release but after initial reviews and pre-screenings, it has limited value to a decision maker when evaluating from the standpoint of greenlighting a project at the initial phase unless one were to estimate the social media state itself.[15]

Natural Language processing techniques are also being employed to use n-gram, bag of words, sentiment analysis to evaluate scripts for their potential. This would be analysis at the very beginning of the film production pipeline and would be a complementary analysis to the one proposed. Findings here have shown promise in predicting a particular studio’s portfolio ROI and show business promise. As our data does not include these scripts and deals with a different stage of the decision-making process, we will not incorporate the findings.

Perhaps most interesting and cutting-edge is the introduction of a new technique by Disney Research called “Factorized Variational Autoencoders (FVAE), which take facial landmarks of the audience and

[14] Quader, Nahid & Gani, Md & Chaki, Dipankar. (2018). Performance evaluation of seven machine learning classification techniques for movie box office success prediction. 1-6. 10.1109/EICT.2017.8275242.

[15] “Box-Office Opening Prediction of Movies based on Hype. Analysis through Data Mining.” Ajay Siva Santosh Reddy. St. Francis Institute of. Technology.

encodes activities such as smiles, laughter, or tears during the viewing of a film to decode the information into interpretable response results. This in turn can be used to help identify successful content within movies or characters within a series which generate the most response from the audience and should be of focus in future films.[16] While interesting, this is a more in-depth product focus than we will assess as its connection to revenue is less obvious.

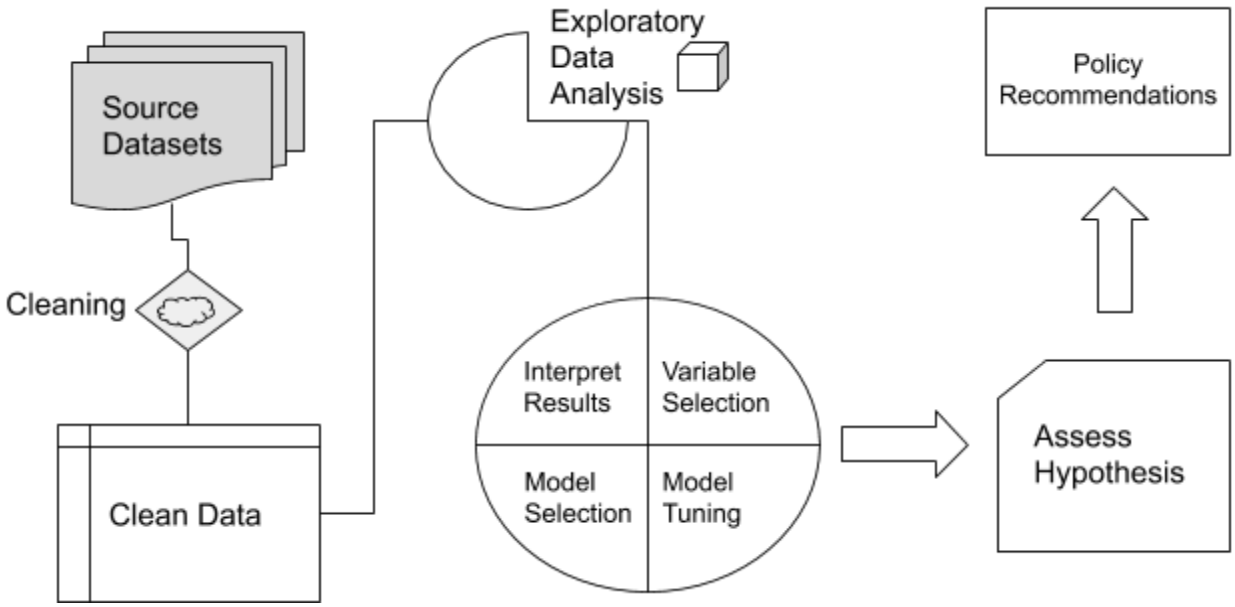
Our analysis will seek to build on this previous work and development by applying several learning algorithms to the data, which we will seek to process in a way which enables all of the “factors” mentioned above to be incorporated. We will retain the question as a regression problem rather than classification to better align with the Kaggle competition and to diversify the study from current literature.

Initial analysis:

Our dataset is relatively small, and is sourced from The Movie Database (TMDB) and was found via a Kaggle competition. The dataset contains 7398 films with features including cast, crew, plot keywords, budget, and posters as images, release dates, languages, production companies, and countries.
There are numerical, categorical, text, and image data types within the dataset which will be used to predict worldwide revenue against a Kaggle test file of 4398 movies.
Also, we will use the same evaluation metric as required by the Kaggle competition, the RMSLE. This will be the ratio of logs of predicted revenue to log of the actual revenue. Given this data label, our models will be regressors that get us close to the leaderboard of Kaggle on the evaluation metric.

Approach:

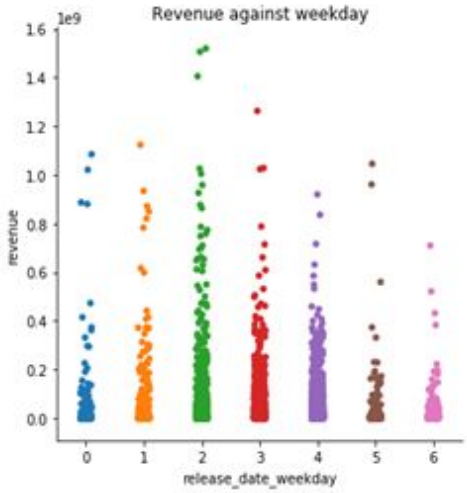
We followed the standard model development approach – transforming the given data into tabular form, preprocess & clean the data into usable form, EDA to gather insights on the data and validate preprocessing, feature selection and engineering, model building, validation and final test evaluation.

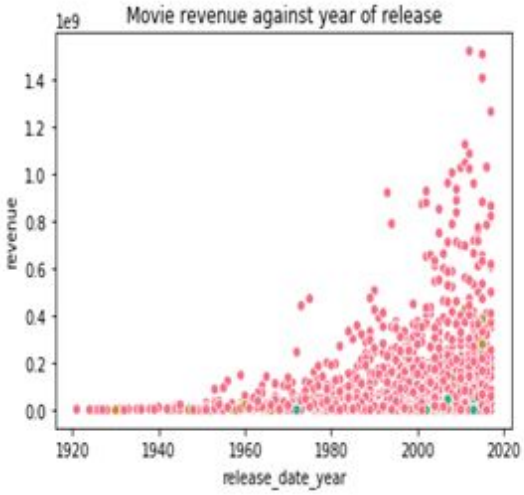


Preprocessing and EDA

[16] Pringle, Ramona. “Watching you, watching it: Disney turns to AI to track filmgoers' true feelings about its films” https://www.cbc.ca/news/technology/disney-ai-real-time-tracking-fvae-1.4233063

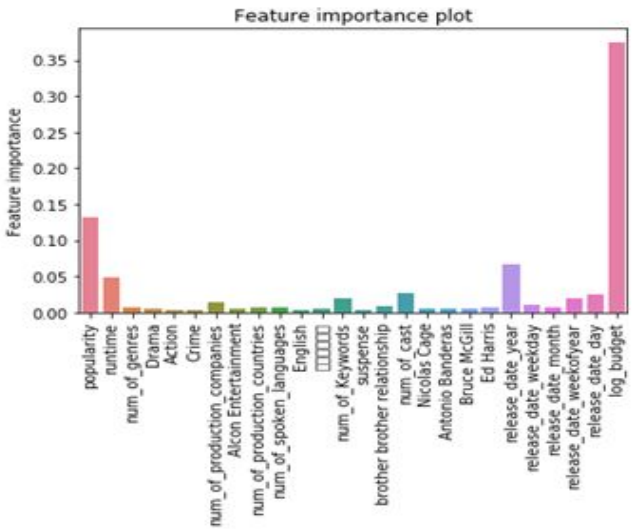
Since the train and test data were extracted from TMDB API, the data were put out as JSON strings. The important predictive factors such as director, actor details were held inside these *json* strings. We engaged in data transformation techniques to extract these predictor variables into usable form for inputting our model. Our intuition here was to enable our models to pick up any signals about how a director or an actor can be a determinant of a movie success. And whether the combination of keywords "saving + world" is magical? The transformation was extended to the release dates of the movie to extract the weekday so our models can answer whether there is any advantage in using F-S-S (Friday-Saturday-Sunday) for new releases or the budget Tuesdays. Additionally, data cleaning was done to fix wrong release dates and impute missing values with mean or default dates wherever appropriately.





Feature Selection

Our transformed data had about 500+ columns, meaning the p in n*p was very large. To reduce the overfitting, we wanted to do choose a feature selection technique that gives us the list of important predictors. Considering the large number of columns, the correlation heat map will be a bit complex. We, therefore, went with the feature importance to get us quickly find the list of important features. Also, we followed a minor iterative approach by limiting the features with importance values of 1X, 2X times greater than the mean importance of all features. We had finally settled in with about 26 features with high enough predictive power.



Model Development & Testing

The Kaggle competition for this data uses Root Mean Squared Log Error (RMSLE) as the measurement of predictive ability. RMSE or MSE or more common, but the RMSLE is often used when there is a massive range of predictions. Since movie budgets can make virtually nothing to over a billion dollars, this is a scenario when the Log RMSE is the optimal predictor. The current leader of the kaggle competition has a **RMSLE=1.67004**. Our goal is to get as close as possible to this. Note: Although our prediction function (in the code) is labeled RMSE, it is because we are predicting on log of revenue.

LASSO Regression

Given the regression nature of this problem, especially given we had approximately 500 variables once fully transformed, we decided to do a baseline LASSO Regression model. We chose LASSO because it functions as a traditional regression with a built in feature selection mechanic. In a traditional linear/logistic regression, too many variables ruins the ability for the model to predict effectively & determine true variable importance, so we chose to use a LASSO.

In the code we built in a cross validation function to iterate through the LASSO with various alpha & lamda parameters to identify which is our best model after 8 iterations. Our best performer yields a **RMSLE of 2.612,** not bad, but we can do better.

Random Forest

Next we wanted to attempt a random forest regressor model. Random Forests are one of the most flexible and powerful ensemble models that are easy to deploy in a variety of problems, and with SK Learn they are also quite easy to tune and cross validate.

With this random forest we attempted multiple numbers of estimators but 100 ended up being the optimal amount due to computational constraints while maintaining performance. Additionally, we gave the random forest a variety of tree depths to try during cross validation. Although random forests are not too prone to overfit, we did not want to risk having too many trees given the noise in the data set[17]. Our best Random Forest was 100 Trees with a depth of 13, resulting in a **RMSLE=2.4360.** If this validation held in the out of sample we would be in 269th place in the kaggle competition currently.

[17] https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm#remarks

XGBoost + Grid Search:

Finally, we wanted to try one of the more advanced and popular models that has been winning data science competitions on kaggle at extremely high rates. XGBoost (extreme gradient boosting) is very sophisticated engineering of a gradient boosted decision tree model. Given it's extremely efficient computational design it allows us to do the absolute highest intensity cross validation methods and hyperparameter tuning in order to find the optimal performing model.

This is our champion model with a **RMSLE of 2.1597.** This RMSLE would put our team in 221st place. The parameters in the xgboost were Eta which represents all learning rates between 0-1, depth which refers to the maximum tree depth. The learning function in gradient descent essentially acts as the size of the correction/overcorrection in prediction problems. With too high learning rates you may over shoot the optimal amount, and with too low learning it may create computational limitations. We also chose a wide range of tree depths 1-75 to really explore the optimal depth for this high dimensional problem. While XGBoost & random forests aren't too prone to overfit, we did initiative an early stopping function with 10 rounds to prevent any possibility of true overfit. This is then confirmed via testing & validation sets. Now let's look at what mattered in the model.

Variable Importance:

We output the top 20 variables weightings in our model, in order to find which weightings were most important to the model. The top 5 are: Budget, num_of_genres, release_date_year, num_of_cast, num_of_keyword. The intuition is that budget definitely makes an impact on earnings potential in films. Additionally, we can see that the multi-genre, multi-production,large cast films do much better in the box office. Largely, many of these variable interpretations are quite intuitive. Our team remains confident that this project could be significantly more effective if we were allowed to predict by genre, by budget, and other variables. When genres, production companies, and country have such high predictive power it implies that we may want to look deeper within those specific clusters to get more lift.

Conclusions

We find that a variety of learning algorithms can generate usable predictability power using widely available, "out of the box" techniques and would be usable within the scope of a business decision making process. In particular, we find that using the data that is publically available, given the appropriate data pre-processing, is of value. This gives further optimism for efforts of business to collect additional data, perhaps private, that would supplement and potentially improve on these predictions. Via Kaggle kernels and industry discussions, there are already many ancillary datasets that are being integrated into the contest to augment the capabilities of these models to learn and predict.

With the accuracy levels achieved by our XGBoost Model, we believe that a given studio using such a model would be able to identify a reasonable range of possible revenues and make a risk-based investment decision on that project. In so doing, they could better balance a portfolio of projects based on their relative risk appetite. Doing so could prove vital in securing the long term viability of their business model and strategic advantage.

Next Steps

There are numerous further areas of study which our analysis suggests would be fruitful. The main avenues for improvement of this project are the addition of outside datasets and the utilization of different types of data.

Additional Data

There are numerous dataset, some now available on Kaggle in fact, which add to the number of observations included in the training data by nearly 3,000 observations, which would provide a material uplift in predictive power. Given more time, we would seek to incorporate that new data and process it in a similar way to gain the advantages there. Moreover, there are alternative sources of data that would be available pre-launch, such as consumer spending data, real-estate prices of movie theater data, or even time series data on what genres were popular in a given preceding year that could add predictive power.

New Data Types

Within the Dataset provided, there was image data related to the movie posters of the films. Using Convolutional Neural Networks or other machine vision techniques, we might be able to categorize or understand those posters and incorporate that information into the predictive model as well. Moreover, there might be additional data insights if the trailers for these films were analyzed as well. It is likely that the features embedded in these trailers have predictive power on revenue.

Feature Engineering

There are numerous feature engineering opportunities available as well. For instance, incorporating the ratio of a given budget to the average of a genre's budget could have had outsize impact given the importance of budget in prediction. Also, because Genre was highly predictive, it is possible that accounting for Genre size could be helpful as well. From a

language perspective as well, we believe that generating a sentiment score based on the keywords of a given movie description would also be a possible improvement.

Clustering

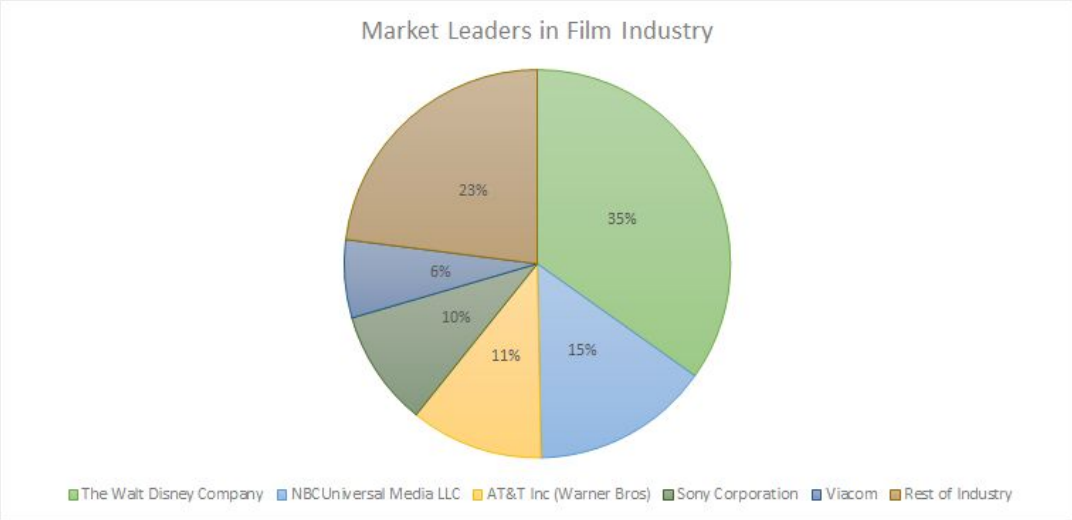
It is possible that by assessing the similarity of a given profile to another successful movies, that an estimation of the revenue potential could be reached on this basis. For instance, if we reduce the dimensionality significantly to only the most predictive variables based on our previous analysis, for instance using RF variable importance, then we might be able to create insights there. Understanding what films would be most similar to a proposed project, and averaging their revenue, could give the equivalent of a “comparables” analysis common in financial transactions that is superior in accuracy or more objective.

Ensemble Methods

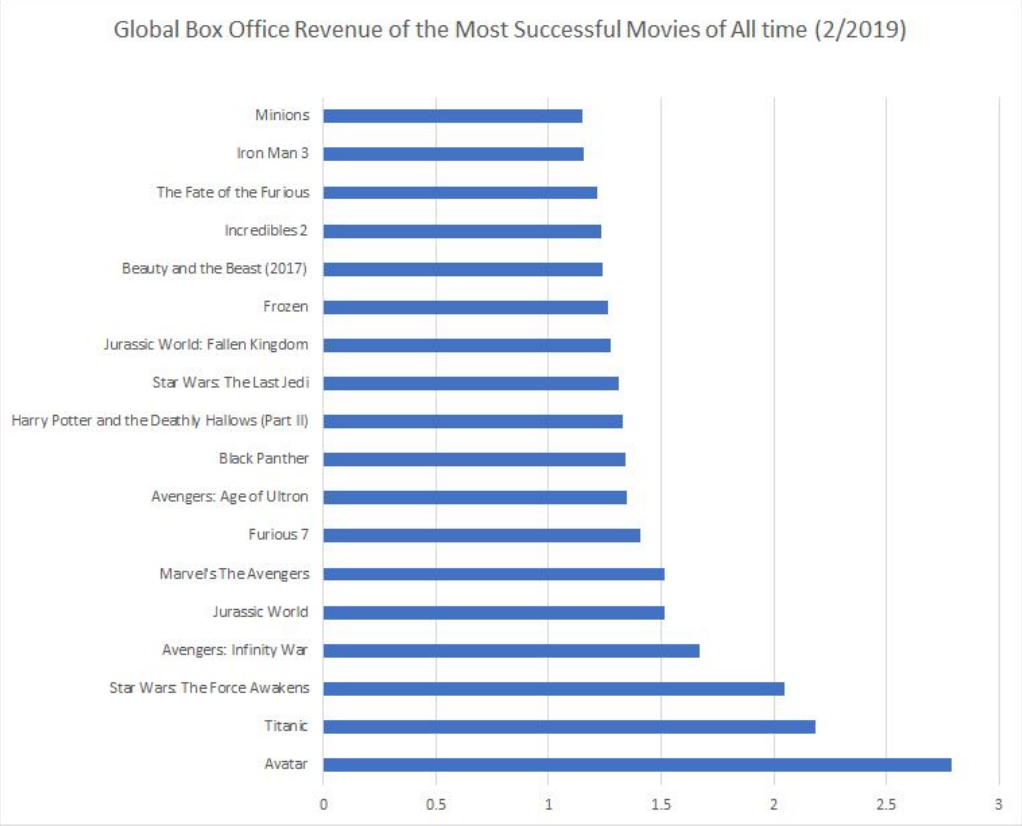
Although Random Forest and XGBoost are inherently ensemble methods, a potential avenue to try this would be via Stacking.This would combine multiple regression models via a meta-regressor, where the base level models are trained on the complete training set and the meta-model then is trained on the outputs of the base level models as features. We are unaware if this technique is being used but find it worth exploring.

Appendix

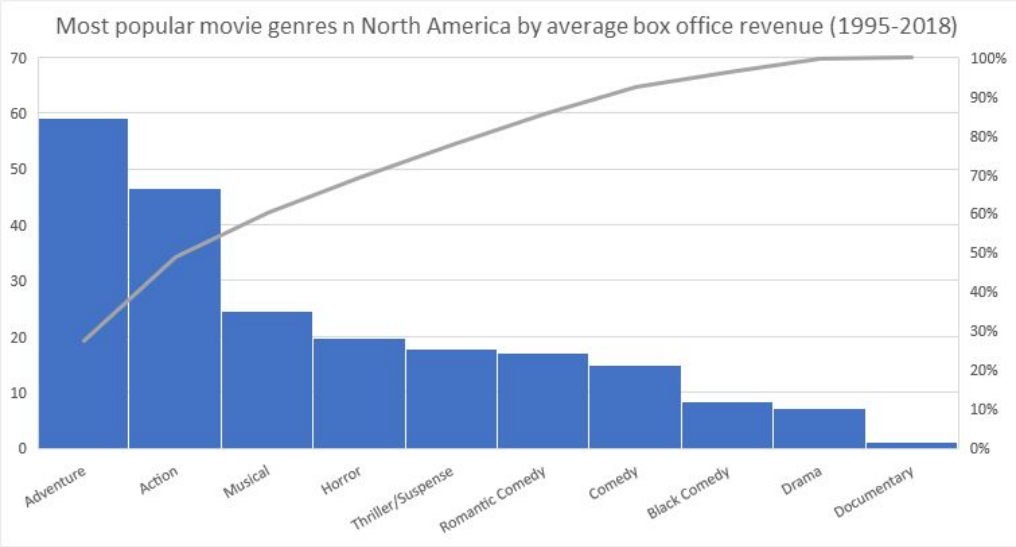
A.



Source: IBISWorld Industry Reeport "Movie & Video Production in the US", September 2018



Source: the-numbers.com via Statista 2019



Source: the-numbers.com via Statista 2019

IHS Screen Digest 2017 via Statista

```
In [41]: ### Train Lasso Regression ###
import sklearn as sk
#need to grid search to find best alpha/lamda
a_list=[-8,-7,-6,-5,-4,-3,-2,-1]
for i in range(len(a_list)):
    alpha=10**a_list[i]
    lasso=sk.linear_model.Lasso(alpha=alpha,fit_intercept=True,normalize=True)
    lasso.fit(xtrain,ytrain)
    yhat=lasso.predict(xcv)
    RMSE=np.sqrt(np.mean((yhat-ycv)**2))
    print(i,RMSE)

Users/bencox/anaconda3/lib/python3.6/site-packages/sklearn/linear_model/coordinate_descent.py:491: Converge
ceWarning: Objective did not converge. You might want to increase the number of iterations. Fitting data wi
h very small alpha may cause precision problems.
 ConvergenceWarning)

 2.7592292895265804

Users/bencox/anaconda3/lib/python3.6/site-packages/sklearn/linear_model/coordinate_descent.py:491: Converge
ceWarning: Objective did not converge. You might want to increase the number of iterations. Fitting data wi
h very small alpha may cause precision problems.
 ConvergenceWarning)

 2.7591380883279464

Users/bencox/anaconda3/lib/python3.6/site-packages/sklearn/linear_model/coordinate_descent.py:491: Converge
ceWarning: Objective did not converge. You might want to increase the number of iterations. Fitting data wi
h very small alpha may cause precision problems.
 ConvergenceWarning)

 2.7581475152326274
 2.752227409140702
 2.712811699133855
 2.6121582881970684
 2.848690752179863
 3.22044461035713
```

LASSO Code

```
In [42]: ##### Now lets do some data visualization
### Random Forest ###
from sklearn.ensemble import RandomForestRegressor
depth=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,20,30]
for i in range(len(depth)):
    RF=RandomForestRegressor(n_estimators=100,max_depth=depth[i],n_jobs=-1)
    RF.fit(xtrain,ytrain)
    yhat=RF.predict(xcv)
    RMSE=np.sqrt(np.mean((yhat-ycv)**2))
    print(i,RMSE)

 2.748232316294994
 2.623679793892502
 2.5573131608935866
 2.536937593006695
 2.5075107361875837
 2.495133561990009
 2.4847798912645866
 2.4909292425276215
 2.4625677516988067
 2.47763868033144
0 2.4572815428850947
1 2.454233267006848
2 2.436076615604799
3 2.4483692561206154
4 2.477608969334646
5 2.4665245862961394
6 2.463233722487598
```

Random Forest Regressor Code

```
In [10]:
##Try XGBOOST
##XGBOOST+GRIDSEARCH###
import xgboost as xgb
from xgboost.sklearn import XGBRegressor
from sklearn import cross_validation, metrics   #Additional scklearn functions
from sklearn.grid_search import GridSearchCV   #Perforing grid search
import matplotlib.pylab as plt
%matplotlib inline
from matplotlib.pylab import rcParams
rcParams['figure.figsize'] = 12, 4
from sklearn.metrics import mean_squared_log_error


xgtrain = xgb.DMatrix(xtrain, label=ytrain)
xgcv = xgb.DMatrix(xcv, label=ycv)


depth=[1,2,3,4,5,6,7,8,9,10,20,30,50,75]
eta=[0.01,0.05,0.1,0.15,0.2,0.3,.4,.5,.75,.9]
mse=10000
for i in range(len(depth)):
    for j in range(len(eta)):
        param={'max_depth':depth[i],'eta':eta[j],'silent':1}
        evallist  = [(xgcv,'eval'), (xgtrain,'train')]
        bst=xgb.train(param,xgtrain,100,evallist,early_stopping_rounds=10)
        yhat=bst.predict(xgcv)
        RMSE=np.sqrt(np.mean((yhat-ycv)**2))


        print(i,j,RMSE)
        if RMSE<mse:
            mse=RMSE
            best=[i,j,RMSE]
            best_model = bst
```

XGBoost Code

```
        bst=xgb.train(param,xgtrain,100,evallist,early_stopping_rounds=10)
        yhat=bst.predict(xgcv)
        RMSE=np.sqrt(np.mean((yhat-ycv)**2))


        print(i,j,RMSE)
        if RMSE<mse:
            mse=RMSE
            best=[i,j,RMSE]
            best_model = bst
```

```
89]    eval-rmse:2.17408       train-rmse:1.81327
90]    eval-rmse:2.16753       train-rmse:1.81175
91]    eval-rmse:2.1638        train-rmse:1.80945
92]    eval-rmse:2.16605       train-rmse:1.80836
93]    eval-rmse:2.16522       train-rmse:1.80584
94]    eval-rmse:2.16332       train-rmse:1.79993
95]    eval-rmse:2.16395       train-rmse:1.79862
96]    eval-rmse:2.16361       train-rmse:1.79614
97]    eval-rmse:2.16186       train-rmse:1.7933
98]    eval-rmse:2.16127       train-rmse:1.79117
       eval-rmse:2.16078       train-rmse:1.78757
 7 2.15970 1796547653
       eval-rmse:2.16078       train-rmse:4.6276
ultiple eval metrics have been passed: 'train-rmse' will be used for early stopping.

ill train until train-rmse hasn't improved in 10 rounds.
1]     eval-rmse:2.61146       train-rmse:2.59785
2]     eval-rmse:2.35729       train-rmse:2.36612
3]     eval-rmse:2.32649       train-rmse:2.30536
4]     eval-rmse:2.33127       train-rmse:2.27222
5]     eval-rmse:2.31841       train-rmse:2.24768
```

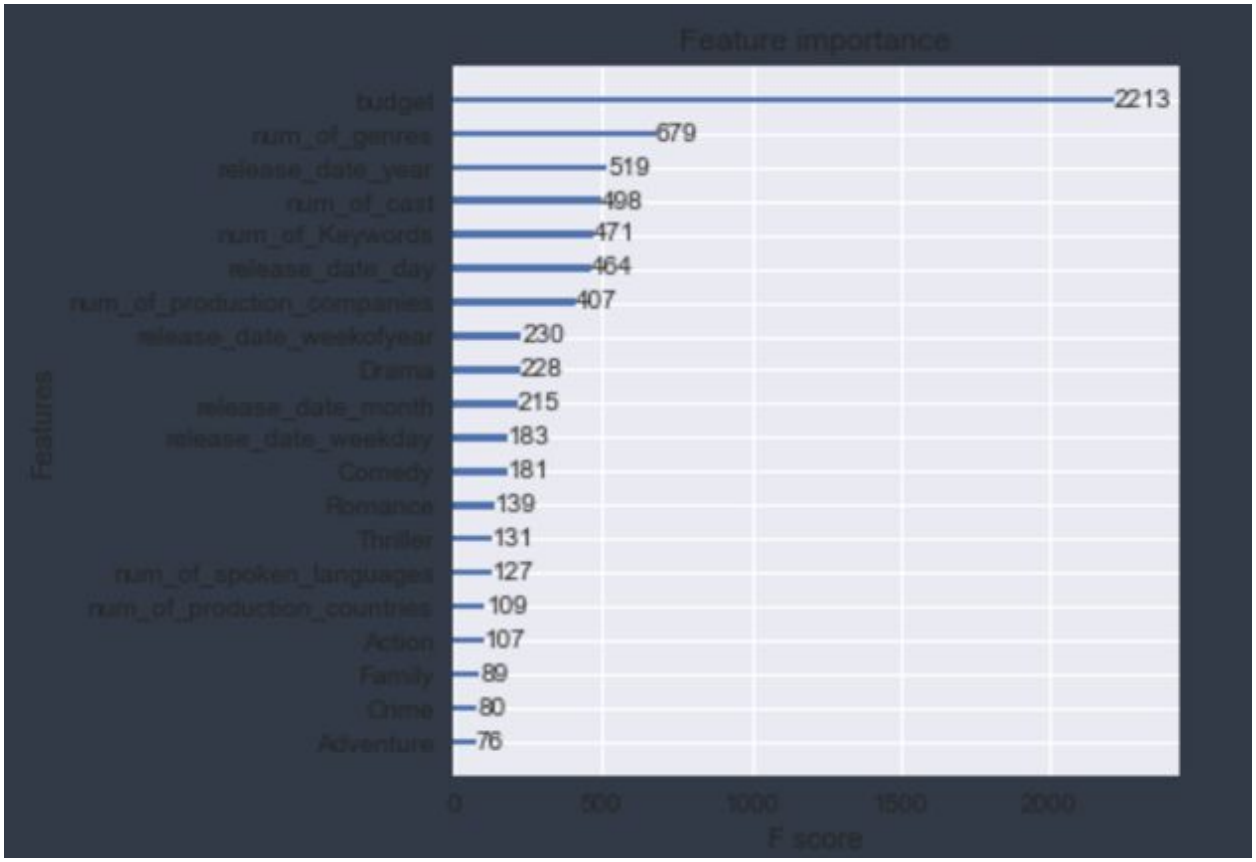
```
In [12]:
best_model

 <xgboost.core.Booster at 0x10983d438>
```

```
In [13]:
yhat=best_model.predict(xgcv)
RMSE=np.sqrt(np.mean((yhat-ycv)**2))
print(RMSE)

.1597071796547653
```

XGBoost Code+Output



Champion Model Variable Importance