# A COMPLETE BREAKDOWN ON MOVIE DATASET AND SENTIMENT ANALYSIS ON CREDITS

Nikesh Kumar Reddy Shettipalli
Rutgers University
(nrs113@scarletmail.rutgers.edu)

Anirudh Bandi
Rutgers University
(ab1721@scarletmail.rutgers.edu)

December 17, 2017

**ABSTRACT :**

The Internet Movie Database is a well-known site that contains information on movies, shows and personalities. The propose of this system is to analyse and breakdown the entire movie dataset and make predictions which will help the film industry while choosing their next option with maximum accuracy and throughput. They can use this system to adjust budget, cast and movie genre to optimize the movie box office gross. We attempt to build a scalable model to perform this analysis. We start by preparing the data, cleaning and preprocessing. Then, we proceed to Exploratory Data Analysis and Visualization. After visualization, we use Machine Learning Algorithms to predict the output and increase the efficiency. Lastly, we use LSTM Network to analyse sentiment based on user feedback and provide sentimental analysis on movies. We try to unveil the important factors influencing the score of IMDB Movie Data. We perform an exploratory analysis of the data and observe some interesting phenomenon, which also helps us improve our prediction strategy. Our results finally show that we achieve a good prediction of score on this dataset.

## I. INTRODUCTION

Every year, hundreds of movies are produced and released. There are great movies, average movies and crappy ones among them. Therefore how do we know their qualities before we do not see the movie ourselves? Or how can we choose a great movie to enjoy and have a relaxed weekend? Rotten Tomatoes and IMDB website are just a good choice to refer at this time. Due to its popularity, IMDB website contains a great deal of information about movies and the comments from audiences. The scores which IMDB gives are highly recognized by the people, representing the quality of content as well as audience's favor to some extent. The user comments in the Rotten Tomatoes help us analyse the flow of the movie. Analyzing the sentiment of the users for a movie also plays an important role in determining the success of the movie. Therefore, in this paper, we will try to unveil the important factors influencing the score on IMDB website and propose an efficient approach to predict it. We will break down the data and analyse each and every piece of information, visualize it and use it for out prediction. The data we use in our paper comes from IMDB 5000 Movie Dataset and Rotten tomatoes on Kaggle. It contains 28 variables for 5042 movies. From our experience, the state-of-the-art methods currently employed to study this type of data are using different regression models, classification models and LSTM Neural Networks.

## II. APPROACH

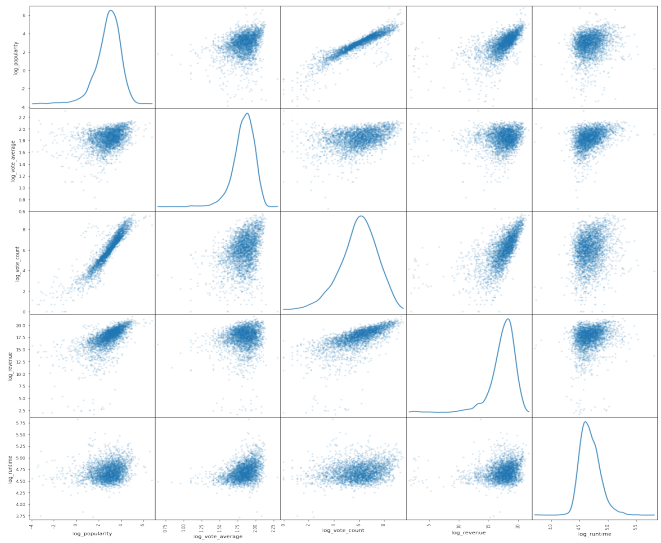
### A. Data Cleaning and Data Pre-Processing

When we look into the dataset, we conduct following things to achieve the pre-processing.

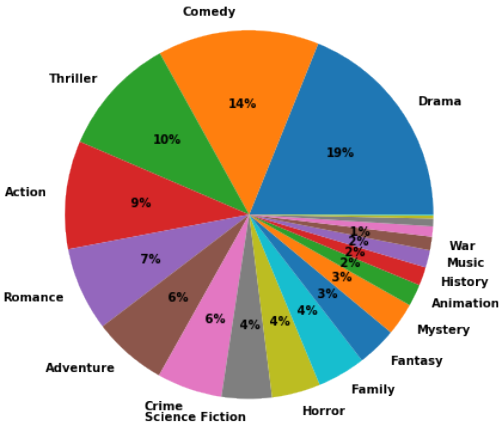
1. Deal with missing/vacant information. For example, few movies might not have a tagline. So, In this specific case we must deal with the missing values by finding the cluster of words that have similar meaning with the title and find the nearest neighbour to it.
2. Handle the data import problem. For example, If there are some mistakes in some lines when we import the .csv file into python workplace, we check it further and eventually find out that it is caused by the punctuation ”,” or “ ”. Since these punctuation ”,” appear, the python recognizes the title to be two terms instead of one, leading to the errors. Therefore, we substitute it with ” ” in the Excel software and successfully solve this problem.
3. Note some variables with significant vacant information. When we explore and investigate into the data, we find out the variable budget has large vacant information, specifically, only less than half of the data have the budget information. Compared with this, gross information is very complete and can give much more clues to predict the scores than budget in the financial level. Thus in the experiments we decide to not consider the budget information which is the most logical choice.
4. Transform Nonnumerical Information. For example, if we have categorical variable in the data like class_type is class A or class B, then we can create 2 new features by the categorical features and maintain their count or 1 if present and 0 if absent in the table. This way, we can convert a Non numerical information into Numerical information.

## B. Exploratory Data Analysis

To begin with, we plot few basic plots to see how the data is arranged. The correlation between features are plotted using the scatter matrix with their log values



We start with analysing the genres.



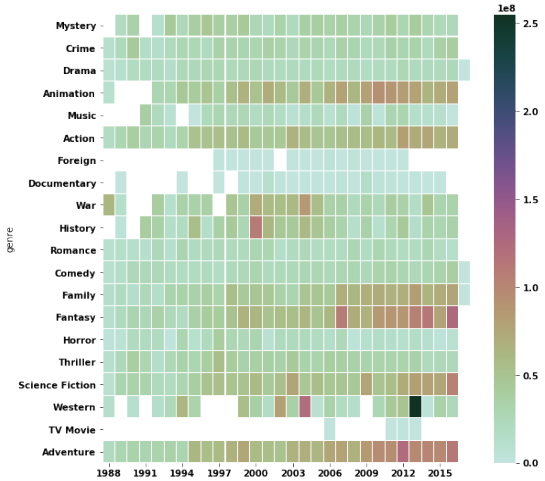

We find that drama movies are most common, followed by comedy. Afterwards, thriller and action movies are the most popular. Interestingly, half of the movies is from the top 5 genres. (51%). This suggest that the main genre of the most movies are drama, comedy, thriller, action. However, the top 5 most common genres could be seen as more general descriptions. For example, movies with the genre war might also be tagged as action movies or drama movies.
We move one step further and look deep into the genre section. We calculate the average votes, budget, and revenue for the different genres. We see which genres are the best scoring ones in each category.

| | 0 | mean_votes_average | mean_budget | mean_revenue | profit |
|---|---|---|---|---|---|
| 11 | Animation | 6.341453 | 6.646590e+07 | 2.256930e+08 | 1.592271e+08 |
| 15 | Adventure | 6.156962 | 6.632686e+07 | 2.086602e+08 | 1.423333e+08 |
| 16 | Fantasy | 6.096698 | 6.356061e+07 | 1.933542e+08 | 1.297936e+08 |
| 18 | Family | 6.029630 | 5.071951e+07 | 1.623455e+08 | 1.116260e+08 |
| 6 | Science Fiction | 6.005607 | 5.186555e+07 | 1.524565e+08 | 1.005910e+08 |

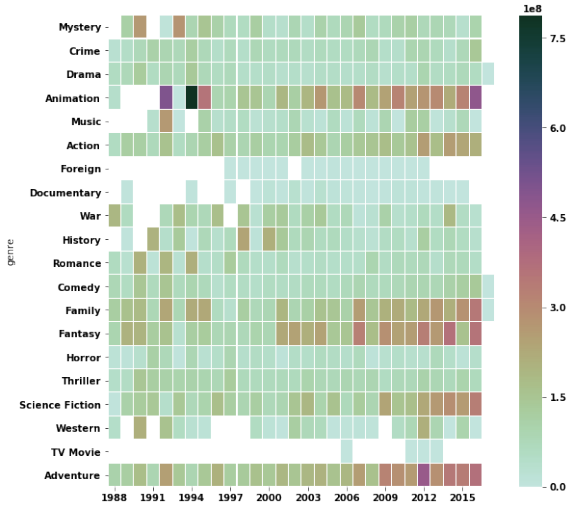
It's very interesting to see that the top 5 highest vote average consists of History, War, Drama, Music and Foreign, while none of these genres are in either one of the other three categories, which all have the same top 3: Animation, Adventure, Fantasy.
However, we would have expected a higher correlation between the budget and the quality of a movie.

In our project, we go even further to get more-in-depth results. Therefore, we first extend the dataframe. with the year of release per movie. We find average votes, average runtime, and average budget per release year and per genre.
Lets look at how the genre is distributed over other sections

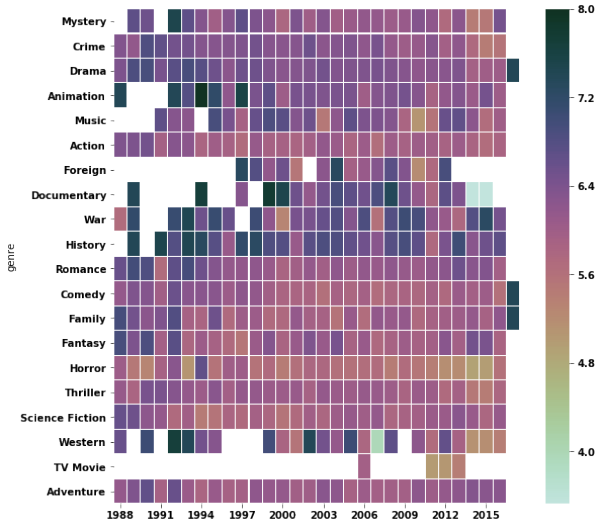
Revenue vs Genre




Profit vs Genre

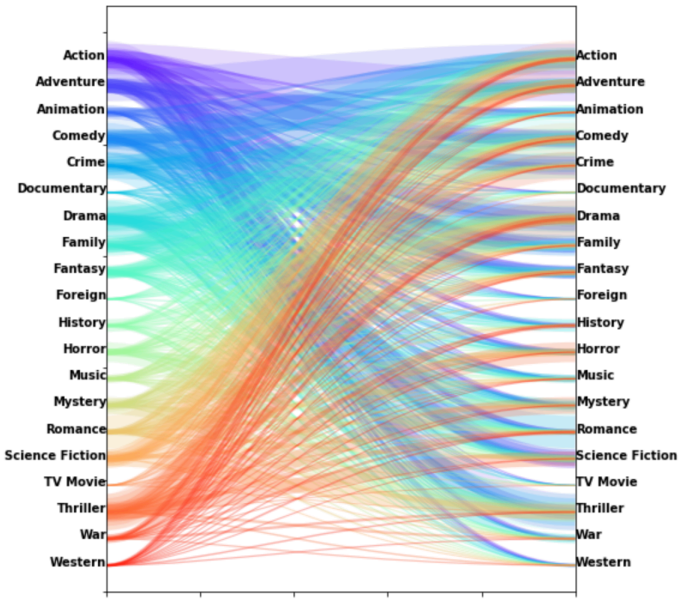


Average Vote vs Genre




## C. Advanced EDA

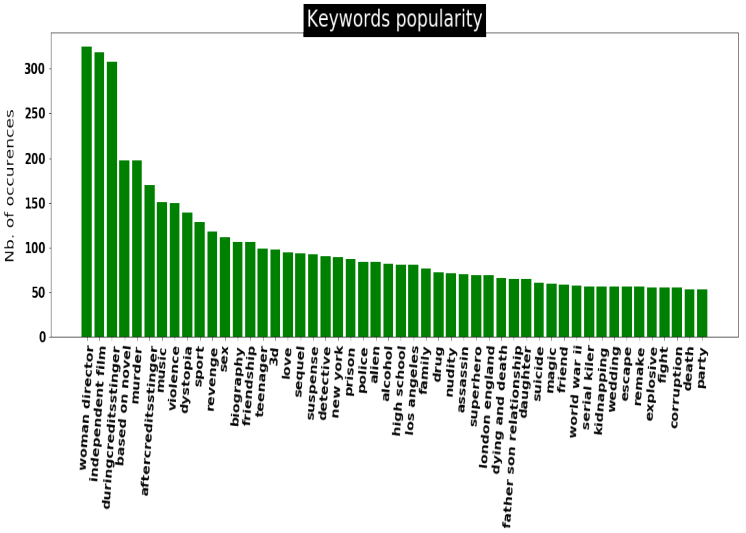
We did not stop with that! Lets see how the genres are interrelated.



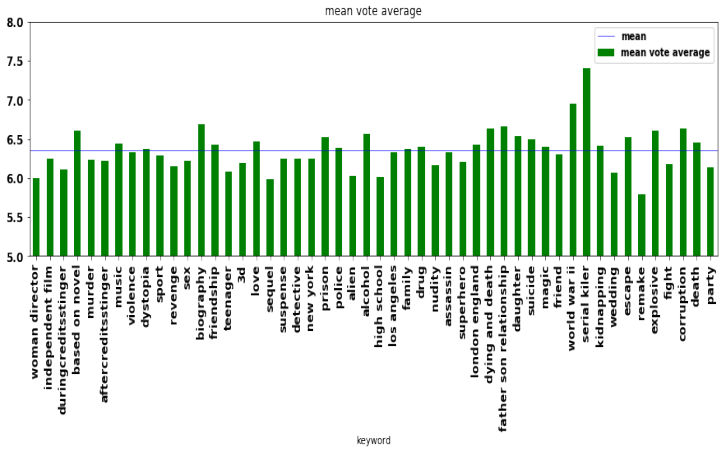

A detailed table view/charts are clearly given in the code-document for all the other features as well.

A small description of how we analyzed the influence of keywords on movie ratings or revenue.

By using credits.scv data, lets see what are the most popular used keywords.




We just can't rely and say that these are the best keywords. So, by using the mean vote average, lets see what keywords the people have voted for.




(similarly, for mean budget, mean revenue, profit have been shown in the code document)

Cast Analysis: The cast and the people involved in the movie are very important for a movie's success. After a bit of restructuring of the data frames, we come up with the following observation.

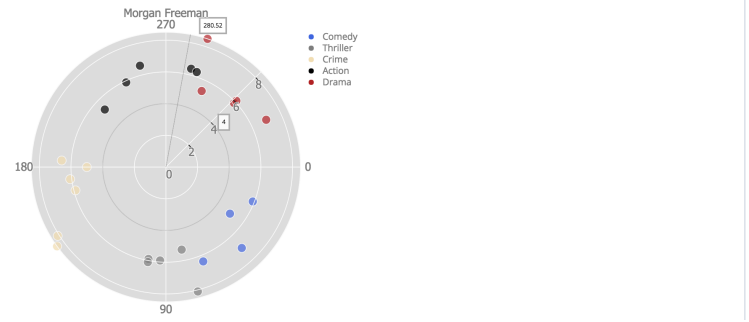
| | actor | vote_average | title_year | gross | budget | favored_genre |
|---|---|---|---|---|---|---|
| 2549 | Ian McKellen | 7.120000 | 2005.400000 | 6.826655e+08 | 1.435000e+08 | Adventure |
| 1931 | Emily Watson | 6.990000 | 2007.800000 | 5.639998e+07 | 2.180000e+07 | Drama |
| 1943 | Emma Watson | 6.930000 | 2007.700000 | 5.875647e+08 | 1.103000e+08 | Adventure |
| 3581 | Keira Knightley | 6.870000 | 2008.600000 | 3.146037e+08 | 7.795002e+07 | Drama |
| 749 | Brad Pitt | 6.842857 | 2004.714286 | 2.281057e+08 | 7.457143e+07 | Thriller |

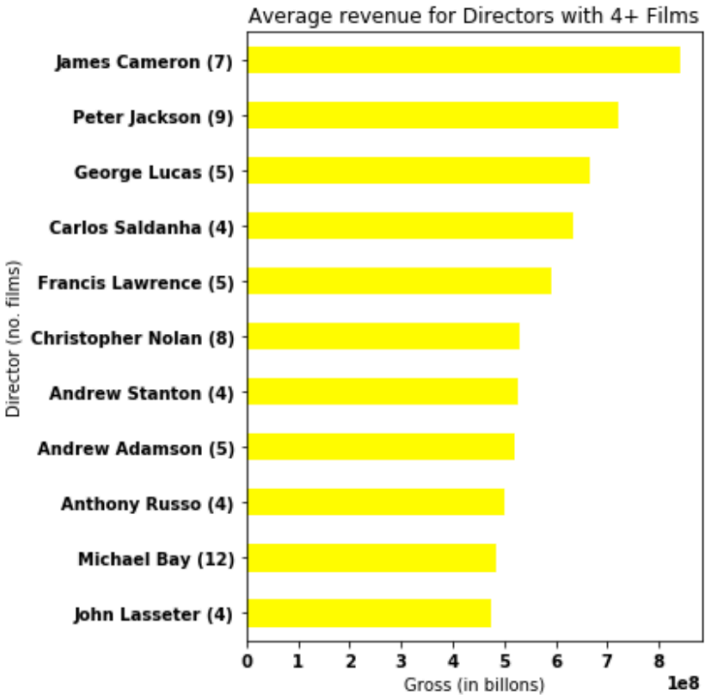
Sir Ian McKellen has had quite a career. He came out on top on all three of our attributes. (vote_average, budget, gross). He plays in the movies with the highest budget, but returns this with the highest average revenues. It makes sense that these enormous budgets lead to good movies.

This is reflected by him having the highest average score on IMDB.

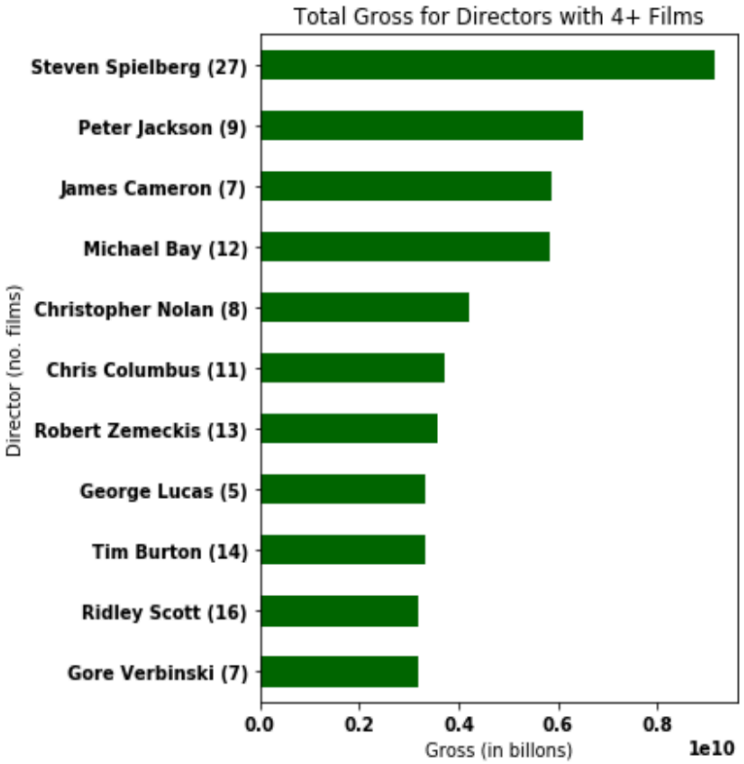
We can analyse each and every actors preferences and the type of movies they tend to incline towards. Let's see what Morgan Freeman preferences and style of movies are

| | Drama,π | Comedy,▼ | Action,▼ | Drama,▼ | Thriller, t | Comedy, t | Crime,▼ | Thriller,▼ | Crime, t | Action, t | K |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 316.8 | 5 | 6.4 | 5.9 | 100.8 | 36 | 5 | 5.9 | 180 | 284.40000000000003 | |
| 2 | 295.2 | 7 | 5.9 | 5.3 | 79.2 | 46.800000000000004 | 6.6 | 5.3 | 183.6 | 244.79999999999998 | |
| 3 | 334.8 | 6.4 | 5.3 | 7 | 100.8 | 68.39999999999999 | 6.1 | 6.1 | 172.79999999999998 | 223.20000000000002 | |
| 4 | 316.8 | 5.9 | 6.6 | 6.1 | 75.60000000000001 | 21.599999999999998 | 8.1 | 8.1 | 147.6 | 255.6 | |
| 5 | 288 | | 6.3 | 8.5 | 93.60000000000001 | | 8.5 | 5.9 | 144 | 288 | |

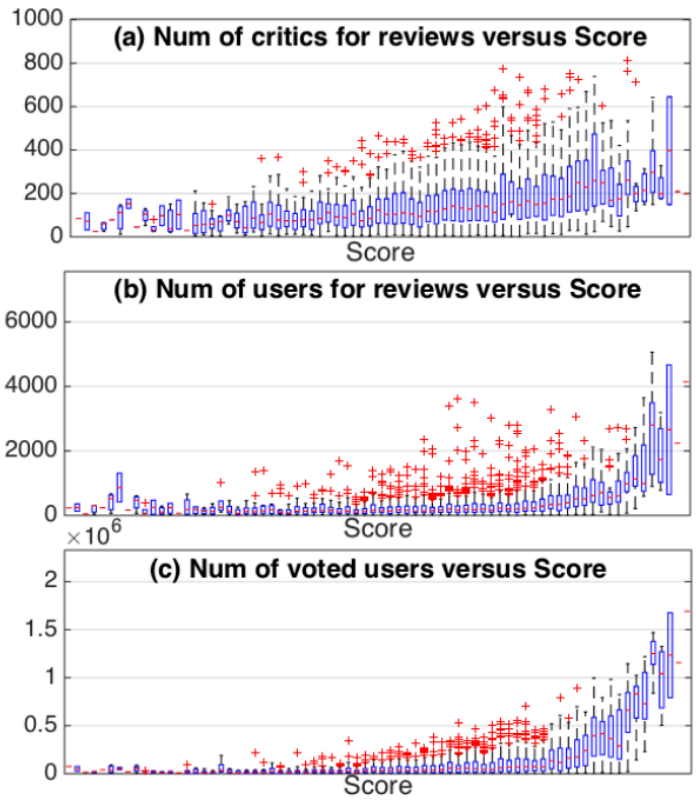


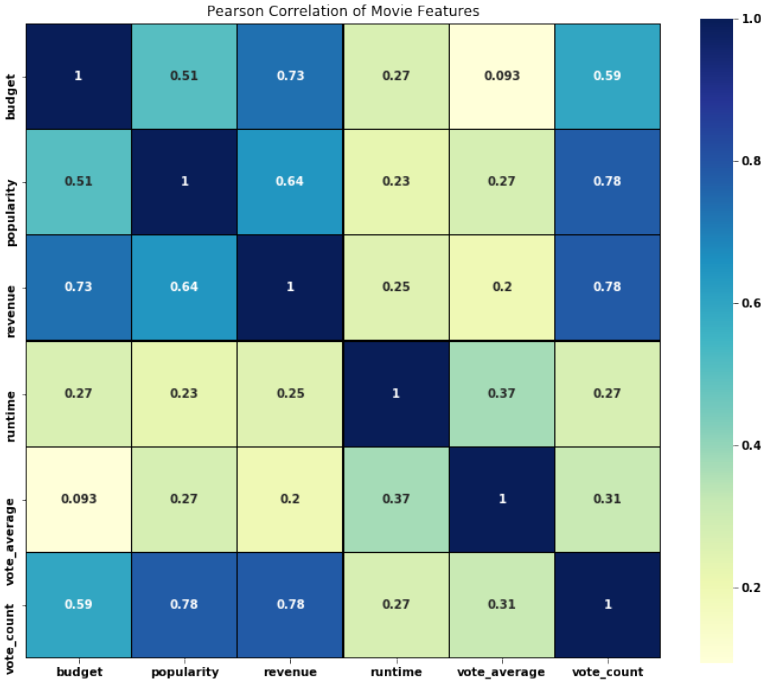

Director's idea, taking are very crucial as well. In a similar and in-depth analysis fashion, we have analysed all the directors as well. We started the actual analysis by computing the average per movie and total gross of the directors. We only took into account the directs for which we have at least 4 movies as observations, to exclude extreme outliers. Not surprisingly, the top rated directors are probably directors you have heard about.




Let's take a look at critics opinions and scores

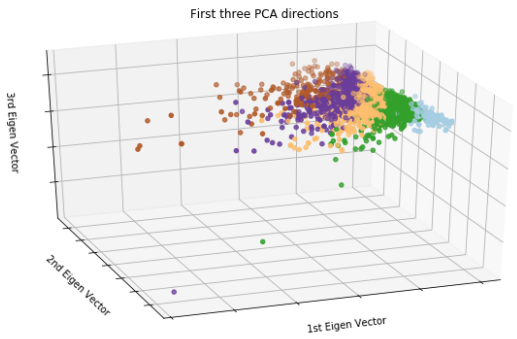


On an over all look, lets see how all the features are co-rellated.




## III. FEATURE SELECTION

There are many ways for feature selection process and its identification. Feature selection addresses the issue of most important attribute from the dataset. It eliminates those attributes that do not add information to the analysis process. It finds out the weight of mot contributed attribute and the attribute that contribute the least. There are many approaches for feature selection we will use Information Gain for feature section process. The information gain depends on the value of entropy. Information gain will be high if the entropy of an attribute is maximum. Entropy calculates on the basis of instance division into subsets. It calculates dependency in dataset. If it is completely dependent then entropy will be zero. We can use Principal Component Analysis which is based on eigen vectors.



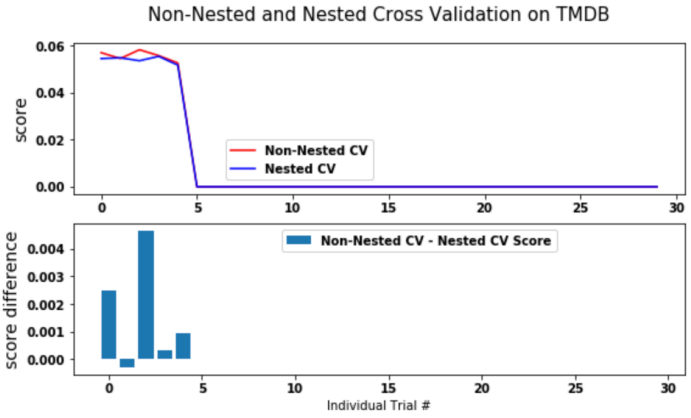

## IV. PREDICTION AND MACHINE LEARNING

(for all cases, we have split the dataset into training and testing set and calculated accuracy based on the data and output)

Many classifiers expect categorical values as the target vector, so we convert our training scores with a little help of scikit's labelencoder function.
We applied Logistic Regression, SVM, KNearest Neighbors, Gaussian Naïve Based, Perceptron, Linear SVC, Stocastic Gradient Descent, Decision Tree, Random Forest.

```
('logistic regression:', 6.65)
('Support Vector Machine:', 92.14)
('KNN:', 38.25)
('Gaussian Naive Bayes:', 5.78)
('Perceptron:', 1.21)
('linear SVC:', 0.65)
('Stochastic Gradient Descent:', 2.42)
('Decision Tree:', 100.0)
('Random forest:', 100.0)
```

SVM, Decision Tree and Random forest have given a very good accuracy.
Overfitting is always a problem in any Machine Learning problem. So, we have used Non-nested and nested cross validation. We were actually very interested in knowing the difference between the scores of both of them.



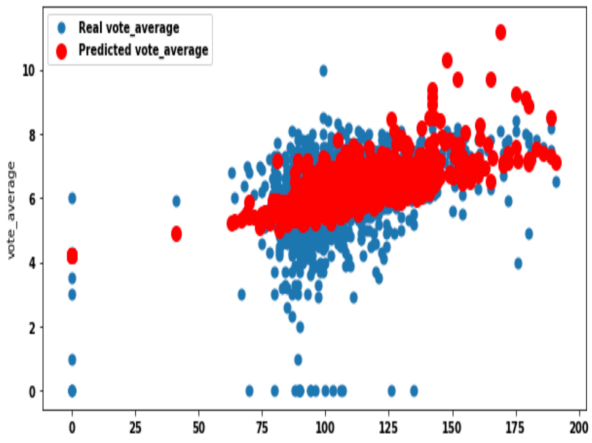

After some fine tuning, better cross validation, and optimizing hyper parameters, we again wanted to see if there would be any difference to the score of the Machine Learning algorithms used before.

```
('logistic regression:', 43.51)
('Support Vector Machine:', 93.47)
('KNN:', 63.3)
('Gaussian Naive Bayes:', 29.96)
('Perceptron:', 34.12)
('linear SVC:', 41.27)
('Stochastic Gradient Descent:', 19.3)
('Decision Tree:', 100.0)
('Random forest:', 100.0)
```

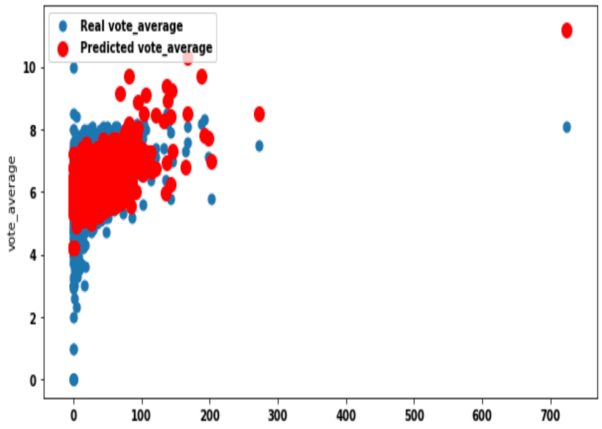
As we can see, out optimization has worked really well. The score of almost all the Algorithms have increased multi fold fashion.

We want to compare a few regression techniques to help us in making predictions. We'll use linear regression and random forest. We start by recreating our numerical data frame. (The feature we selected here is vote_average)

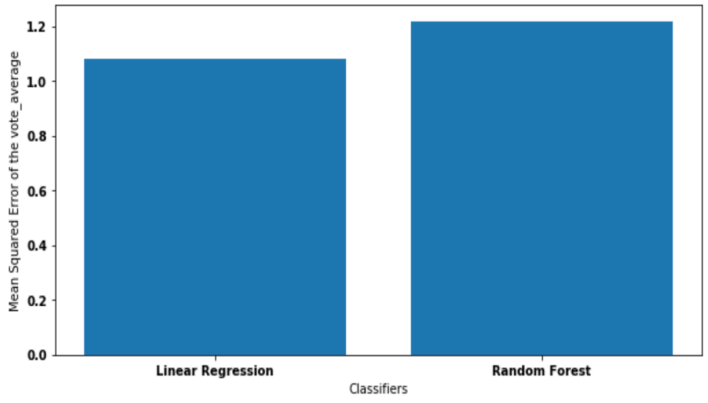
Let's see what happens if we use linear regression




Now, lets see what happens when we use random forest




Random Forest best predicts the data




## V. LSTM FOR SENTIMENT ANALYSIS

Let's see our last stage of Advanced Prediction of sentiment using LSTM Neural network

The dataset is comprised of tab-separated files with phrases from the Rotten Tomatoes dataset. The train/test split has been preserved for the purposes of benchmarking, but the sentences have been shuffled from their original order. Each Sentence has been parsed into many phrases by the Stanford parser. Each phrase has a PhraseId. Each sentence has a SentenceId. Phrases that are repeated (such as short/common words) are only included once in the data.

- train.tsv contains the phrases and their associated sentiment labels. We have additionally provided a SentenceId so that you can track which phrases belong to a single sentence.
- test.tsv contains just phrases. You must assign a sentiment label to each phrase.

The sentiment labels are:

0 - negative
1 - somewhat negative
2 - neutral
3 - somewhat positive
4 - positive

We used a nltk tokenizer, stop-words list and a stemmer for text pre-processing.
We then build a dictionary consisting of training and test phrases and created a tokenId sequence for each phrase that was used for training an LSTM network.
The LSTM network can be tuned and trained on more epochs to achieve better sentiment prediction score.

| | PhraseId | SentenceId | Phrase | Sentimen |
|---|---|---|---|---|
| 0 | 156061 | 8545 | An intermittently pleasing but mostly routine ... | 3 |
| 1 | 156062 | 8545 | An intermittently pleasing but mostly routine ... | 3 |
| 2 | 156063 | 8545 | An | 3 |
| 3 | 156064 | 8545 | intermittently pleasing but mostly routine effort | 3 |
| 4 | 156065 | 8545 | intermittently pleasing but mostly routine | 3 |
| 5 | 156066 | 8545 | intermittently pleasing but | 3 |
| 6 | 156067 | 8545 | intermittently pleasing | 3 |
| 7 | 156068 | 8545 | intermittently | 2 |
| 8 | 156069 | 8545 | pleasing | 3 |
| 9 | 156070 | 8545 | but | 2 |
| 10 | 156071 | 8545 | mostly routine | 2 |

## VI. CONCLUSION

The results obtained from each of the classifier is shown in the code document. These result shows the average percentage of time we are able to correctly predict the instances. We are able to achieve highest accuracy with Random Forest, Decision Tree and Support Vector Machine 94.34 % , 94.15% and 90.15%respectively. We also implemented some other classifier, neural network classifier that is multilayer perceptron produces accuracy

pf 79.07%. We used LSTM to perform sentiment analysis on movie reviews and predict the sentiment accurately.

## VII. FUTURE WORK

The proposed work aims to predict movies popularity. We have used machine learning approach for our experimentation. Machine learning have powerful classification algorithms for classification. Our research aims to improve previous researches. Performing data mining on IMDB is a hard task because of so many attributes related to a movie and all in different dimensions with lots of noisy data and missing fields. After performing classification, we have found out that our best results are achieved through SVM, Random Forest at around 92 %. The attributes that contributed the most to information are metascore and number of votes for each movie. We would like to take it further by taking these results and predicting what would have happened if a particular actor has acted in another movie instead of the current crew. Combining this data with Oscars data, we also have a side project of predicting Oscars for 2018.

## VIII. REFERENCES :


[1] A Survey of Collaborative Filtering Techniques; Su et al; https://www.hindawi.com/journals/aai/2009/421425/

[2] Google News Personalization: Scalable Online Collaborative Filtering; Das et al; https://www2007.org/papers/paper570.pdf

[3] Intro to Recommender Systems: Collaborative Filtering; http://blog.ethanrosenthal.com/2015/11/02/intro-to-collaborative -filtering/

[4] Collaborative Filtering Recommender Systems; Stanford Student project; http://cs229.stanford.edu/proj2014/Rahul %20Makhijani,%20Saleh%20Samaneh,%20Megh%20Mehta,%20 Collaborative%20Filtering%20Recommender%20Systems.pdf

[5] MMDS course slides; Jeffrey Ullman; http://infolab.stanford.edu/ ullman/mmds/ch9.pdf https://www.kaggle.com/willacy/director-and-actor-s-total-gross-and-imdb-score

[6] Recommending items to more than a billion people; Kabiljo et al; https://code.facebook.com/posts/861999383875667/ recommending-items-to-more-than-a-billion-people/

[7] Imdbpy: Python script for retreiving data from imdb. http://imdbpy.sourceforge.net/ support.html#documentation/.

[8] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing. Mixed membership stochastic blockmodels. J. Mach. Learn. Res., 9:1981–2014, June 2008.

[9] A. Kimmig, S. H. Bach, M. Broecheler, B. Huang, and L. Getoor. A short introduction to probabilistic soft logic.

[10] R. N. Lichtenwalter, J. T. Lussier, and N. V. Chawla. New perspectives and methods in link prediction. In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '10, pages 243–252, New York, NY, USA, 2010. ACM. https://www.kaggle.com/alaaawad/imdb-dataset-exploration

[11] A. K. Menon and C. Elkan. Link prediction via matrix factorization. In Machine Learning and Knowledge Discovery in Databases, pages 437–452. Springer, 2011.

[12] K. T. Miller, T. L. Griffiths, and M. I. Jordan. Nonparametric latent feature models for link prediction.

[13] G. Rossetti, M. Berlingerio, and F. Giannotti. Scalable link prediction on multidimensional networks. In Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops, ICDMW '11, pages 979–986, Washington, DC, USA, 2011. IEEE Computer Society.https://www.kaggle.com/diegoinacio/imdb-genre-based-analysis

[14] https://en.wikipedia.org/wiki/Film , Accessed on August 1st, 2015

[15] https://en.wikipedia.org/wiki/Internet_Movie_Database , Accessed on August 1st, 2015 [16] Darin Im and Minh Thao Nguyen : "PREDICTING BOXOFFICE SUCCESS OF MOVIES IN THE U.S. MARKET ", CS 229, Fall 2011

[17] Jeffrey S. Simonoff and liana R.Sparrow : "Predicting Movie Grosses : Winners and Losers, Blockbusters and Sleepers" . Chance, vol. 13(3), pp. 15–24, 2000.

[18] Ramesh Sharda , Dursun Delen :"Predicting box-office success of motion pictures with neural networks", Expert Systems with Applications 30 (2006) 243–254

[19] Nithin VR, Pranav M, Sarath Babu PB, Lijiya "A Predicting movie success based on IMDB data" International journal of data mining and techniques , Volume 03, june 2014, pages 365-368