Identifying the Movie Success Rate

Mittal Jain
Department of I.T
K. J. Somaiya College of Engineering
Mumbai, India
mittal.jain@somaiya.edu

Sameer Patel
Department of I.T
K.J. Somaiya College of Engineering
Mumbai, India
sameer.patel@somaiya.edu

Sarvesh Pai
Department of I.T
K.J. Somaiya College of Engineering
Mumbai, India
sarvesh.pai@somaiya.edu

***Abstract*— Predicting the success of a movie's opening has always been a difficult problem since the success of a movie does not always depend on its quality only. There are many external factors such as other competing movies; time of the year and even weather influences the success because these factors also impact the Box Office sales. It is very important in terms of Box Office ticket sales to predict the movie's opening success in order to make proper plans, cost plans and make the sales profitable. We introduce a simple solution for predicting movie success in terms of cost and financial success. We have achieved decent results, allowing planning of theatre and small studios. Machine Learning has always been used to increase sales in different domains like stock market etc. This paper focuses on the different machine learning algorithms like Adaboost, SVM, Logistic Regression, Naïve Bayes Classifier and K-Nearest Neighbors to predict the success of different movies in the Box Office Studio.**

***Keywords* — Support Vector Machine (SVM), Receiver Operating Characteristic Curve (ROC), Area Under Receiver Operating Characteristic Curve (AUC), K – Nearest Neighbor (KNN)**

I. INTRODUCTION

Hundreds and Thousands of films are released every year in India and many other countries. Since 1930s, the industry has earned more money every year than that of any other country by a huge margin. Cinema is the industry where even films of small budget earn around billion dollars. Huge production houses are the main control head with spending dollars on advertising alone. Advertising are one of the factors which contribute a lot to the entire budget of the movie. Sometimes these investments lead to heavy losses for the producers. Warner Brothers had a deep fall in their revenues last year even after advertising their movies. Now the question arises , If it was somehow possible to know beforehand the success of a movie , the production house would actually focus on these issues to increase their sales. This would help the producers increase their sales by focusing on these factors. Using these factors , they would be able to know when the market is actually down or when it is not. Knowing these factors would surely help producers understand their competitors and work on this factors. Many people have worked on this topic and tried to achieve this goal of predicting success rate of movies. Many different techniques and algorithms such as sentiment analysis and analysis on the profits have been done in the past. None of the studies have been succeeded in suggesting what actually they wanted to predict. [1].

II. TASK DESCRIPTION

The objective of our project is to predict the success rate of a movie based on attributes such as the actors involved, directors, year in which they were released, movie genre, total runtime of movie, user rating, number of votes, total revenue generated by movie, the overall meta-score, age of the users watching and recording the votes or rating, the geographical areas where movie was released, any other influences such as political movements, ongoing trends, and so on.

Our most important and difficult tasks was to get the data-set for such kind of prediction and analysis. We had to look for datasets available on the web, as we would not be able to collect historical data about past movies for our project. Following are steps we performed:

1) Searched for available datasets to support our idea and thoroughly scrutinized them, to get the most suitable dataset for our idea.

2) Shortlisted few datasets, we picked the most suitable dataset for our project.

3) Pruned the data which we required, most suitable for our prediction and analysis.

4) Collected ground truth data, and saved in the csv file format. We also binarized our attributes and used an additional success column, based on the average revenue, rating and votes received by the movie.

5) We used this data as an input to the machine learning and data mining algorithms for prediction of movie success rate.

6) We split the data into training and testing data.

7) The machine learning algorithms we have used are Logistic Regression, Linear SVM, K-Nearest Neighbor, Naïve Bayes Classifier and AdaBoost.

8) We have computed the results of our algorithms by means of confusion matrix, accuracy, recall, precision rate and ROC curve.

9) We have also used this dataset for analysis of effect of various attributes on the success rate of movie. These attributes include rating, votes, actors, directors, revenue and meta-score.

III. MAJOR CHALLENGES

A. *Dataset Description*

1) Some key attributes like genre were comma separated values in a CSV file.

2) Converting the above data into binary values for the model and other data cleaning process required some serious effort.

3) Implementation of cross-validation from scratch without using external libraries to extract all relevant information required several brainstorming sessions.

B. *Visualization*

1) To get the details of visualization libraries in python a detailed investigation was required.

2) To Finalize the key attributes with which movie data needed to be analyzed took a lot of time.

3) Most of the attributes in the dataset were too related to each other. Dividing them to separate entities to infer useful information was a challenge.

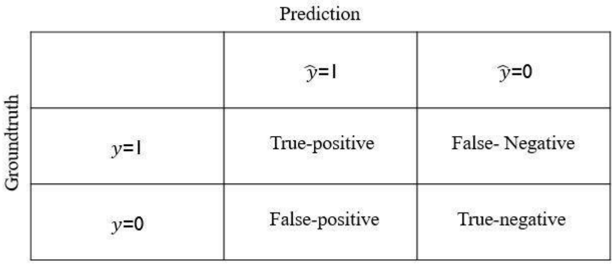
IV. EXPERIMENTS

A. *Dataset Description*

1) Rank - Rank of the movie
2) Title - Title of the movie
3) Genre - Genre of the movie
4) Description - Description of the movie
5) Director - Director of the movie
6) Actors - Actors of the movie
7) Year - Year of the movie
8) Runtime (Minutes) - Runtime of the movie
9) Rating - Rating of the movie
10) Votes - Votes of the movie
11) Revenue (Millions) - Revenue of the movie
12) Meta-score – Meta-score of the movie

B. *Evaluation Metrics*

To visualize the performance of an algorithm, typically a supervised learning confusion matrix is used. Also, known as error matrix, each column of the confusion matrix signifies an instance of a predicted class and each row signifies an instance of the actual class.



		Prediction	
		$\hat{y}$=1	$\hat{y}$=0
Groundtruth	y=1	True-positive	False- Negative
	y=0	False-positive	True-negative

The above table is an example of a confusion matrix. If the prediction and ground truth are equal then it is either True-positive or True negative based on the classification labels. If the prediction is not equal to the ground truth, then it is either False-positive or False-Negative based on the classification labels. From the table we are calculating the Accuracy, Precision and Recall.

We are also determining the ROC curve to evaluate the performance of the algorithm.

Logistic Regression

Logistic regression is a statistical method for analyzing a dataset where there are one or more independent variables that determine an outcome. The outcome can be measured with a dichotomous variable in which we have two outcomes. The dependent variable is always either binary or dichotomous, i.e. it contains data coded as either 1 or 0. The binary logistic model can be used to estimate the probability of the binary response depending on the different prediction variables. [4]

The main goal of Logistic regression is to find the accurate fitting model which can be used to describe the relationship between the different characteristics and features of the dataset.

Logistic regression equation - Here p is the probability that the characters of interest are present. The logit transformation is defined as the logged odds:

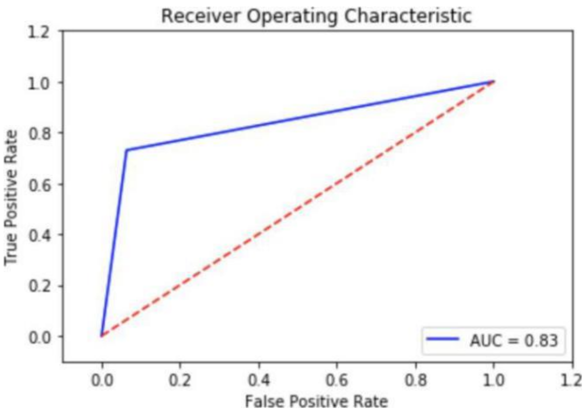
Odds=p/(1-p)
and
Logit(p)=ln(p/(1-p))

Major Results for Logistic Regression:

Confusion Matrix:

Predictions			
Ground Truth		1	0
	1	162	11
	0	10	27

Accuracy: 0.9000
Precision: 0.7105
Recall: 0.7297
ROC:



Analysis

From the above results, it can be inferred that when we consider binary values as input the Logistic regression classifier has a good accuracy of 90.0% and the ROC curve gives an AUC of 0.83.

The predictions are quite high, and this algorithm is very stable when we consider the dataset with more than one independent variable.

Support Vector Machines

Support Vector Machine (SVM) is a supervised machine learning algorithm which can be used for both classification and regression challenges However, it is mostly used in classification problems. SVM is like a sharp knife – it works on smaller datasets, but on them, it can be much stronger and powerful in building models.

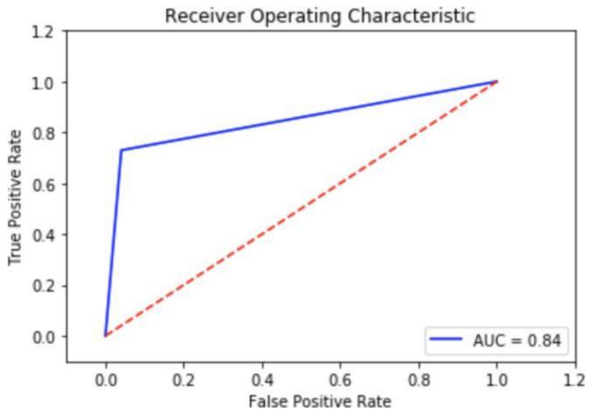
In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a coordinate.

Major Results for SVM:

Confusion Matrix:

Predictions			
Ground Truth		1	0
	1	166	7
	0	10	27

Accuracy: 0.9190
Precision: 0.7941
Recall: 0.7297
ROC:



Analysis:

The results of SVM algorithm, the accuracy of 91%, and the AUC is good, even though we get good accuracy, it seems not to be the best suited algorithm for our prediction.

K-Nearest Neighbor Algorithm

KNN is a non-parametric method used for classification and regression. This method is used widely in classification problems in the industry. In this method, the majority vote of its neighbors classifies an object and it is then being assigned to the class most common among its k nearest neighbors where k is a positive integer, typically small.

K-Nearest is also a lazy algorithm because it doesn’t use the training data points to do any generalization. In simple words, there is no explicit training phase or it is very minimal. This means the training phase is fast and lack of generalization means that KNN keeps all the training data. All the training data is needed during the testing phase [3].

KNN Algorithm:

In the classification of K-nearest neighbor algorithm , the algorithm burns down to form a major count between the k most identical instances or outputs to a given particular scene or observation. Similarity can be explained as the distance metric between the two data points. A popular option is the Euclidean distance given by

$$d(x,x') = \sqrt{(x_1-x'_1)^2+(x_2-x'_2)^2+\ldots+(x_n-x'_n)^2}$$

But other techniques can be more appropriate such as the Manhattan Distance, and Hamming Distance. Basically, given a positive integer Q, an unseen observation a, and a similarity metric d, KNN classifier undergoes the following steps:

1) It goes through the entire dataset computing d between a and each training observation. We’ll call the Q points in the training data that are closest to a the set A. Note that Q is usually odd to prevent tie situations.

2) It then calculates the conditional probability for each structure/class i.e., the fraction of points in AA with that given class label.

$$P(y=j|X=x) = \frac{1}{K}\sum_{i\in \mathcal{A}} I(y^{(i)}=j)$$

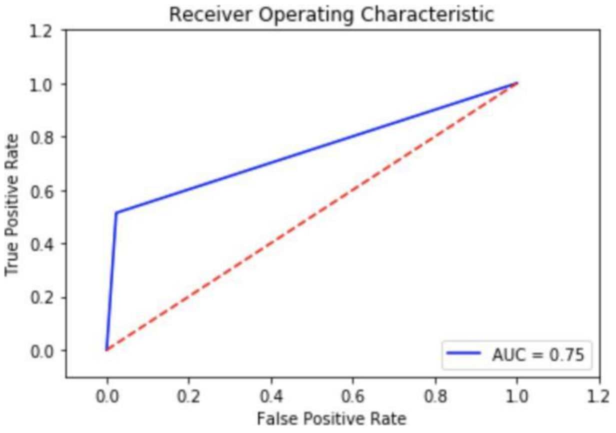
Lastly, the input a gets assigned to the class with the largest probability.

Major Results for KNN:

Confusion Matrix:

Predictions			
Ground Truth		1	0
	1	169	4
	0	18	19

Accuracy: 0.8952
Precision: 0.8260
Recall: 0.5135
ROC:



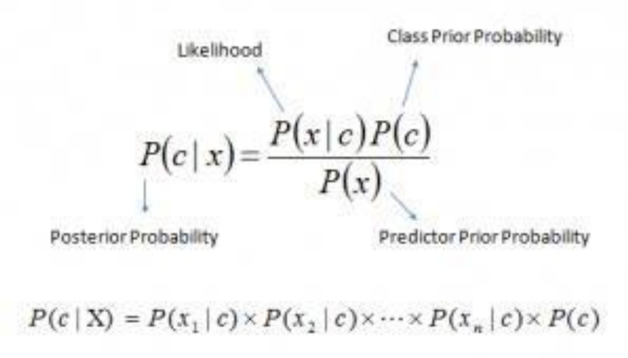
Analysis

KNN algorithm is one of the simplest classification algorithm. Even with such simplicity, it can give highly competitive results. The only difference KNN classifier has from regression is the methodology, which uses the averages of nearest neighbors rather than voting from nearest neighbors.

Based on the above results, it can be inferred that the K-Nearest Neighbor classifier at k = 5 has a good accuracy of 89.5% and the ROC curve gives an AUC of 0.75.

Naïve Bayes Algorithm

Naïve Bayes Algorithm is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In other words, a Naive Bayes classifier estimates that the presence of a feature or variable in a class is not related to the presence of any other feature. Naive Bayes model is more easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.



$$P(c \mid x) = \frac{P(x \mid c)P(c)}{P(x)}$$

$$P(c \mid X) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

Above,

- P(c|x) is the posterior probability of class (c, target) given predictor (x, attributes).
- P(c) is the prior probability of class.
- P(x|c) is the likelihood which is the probability of predictor given class.
- P(x) is the prior probability of predictor.

The working of Naïve Bayes algorithm can be explained in 3 steps

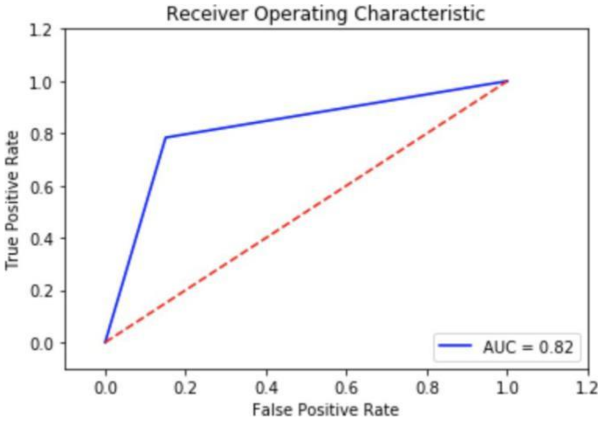
1) Converting the dataset into frequency model.
2) Create a likelihood table by finding the probabilities.
3) Use Naive Bayesian equation to calculate the posterior probability for each class. The class with the highest posterior probability is the outcome of prediction.

Major Results for Naïve Bayes:

Confusion Matrix:

		Predictions	
Ground Truth		1	0
	1	147	26
	0	8	29

Accuracy: 0.8380
Precision: 0.5272
Recall: 0.7837
ROC:



Analysis

The accuracy for Naive Bayes Classifier is 83.8%. ROC is good but in comparison to other models which are under the scope of our study, NBC seems to an underperforming model for our dataset. AUC is 0.82 which is a decent figure as well.

Adaboost Algorithm

AdaBoost stands for Adaptive Boosting which a machine is learning meta-algorithm formulated by Yoav Freund and Robert Schapire. This algorithm is used in conjunction with other types of learning algorithms to improve performance [7].

A family of weak learner algorithms are used together to form strong-learners. To find a weak rule, we apply base learning (ML) algorithms with a different distribution. Each time a base learning algorithm is applied, it generates a new weak prediction rule. This is an iterative process. After much iteration, the boosting algorithm combines these weak rules into a single strong prediction rule.

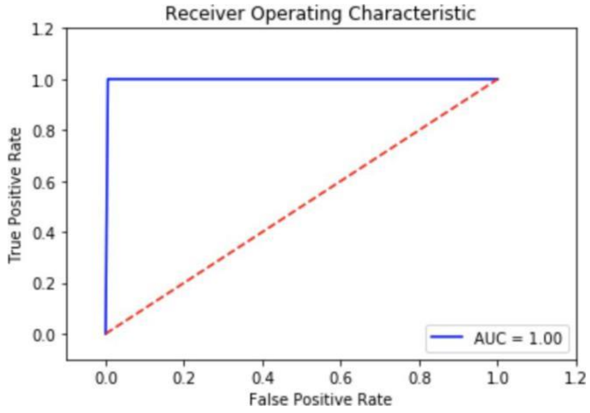
The disadvantage of Adaboost is that it can be susceptible to noisy data and outliers while being prone to overfitting.

Major Results for Adaboost:

Confusion Matrix:

Predictions			
Ground Truth		1	0
	1	172	1
	0	0	37

Accuracy: 0.9952
Precision: 0.9736
Recall:
1.0 ROC:



Analysis

The Adaboost algorithm is used in conjunction with Decision tree algorithm with maximum depth being equal to 2.

Decision trees are used with Adaboost as they are non-linear while being fast to train. They are also fast to classify and thus can be used in large numbers.

From the above results, it can be inferred that the Adaboost has a very high accuracy of 99.5%. The ROC curve gives an AUC of 1 which is a perfect score indicating a perfect test.

Visualization

We have also performed visualizations on the data.

A. Visualization based on Ratings.

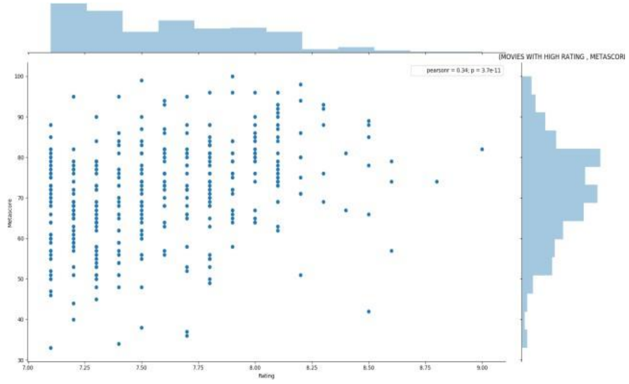


Fig 1. Movie with high rating & metascore

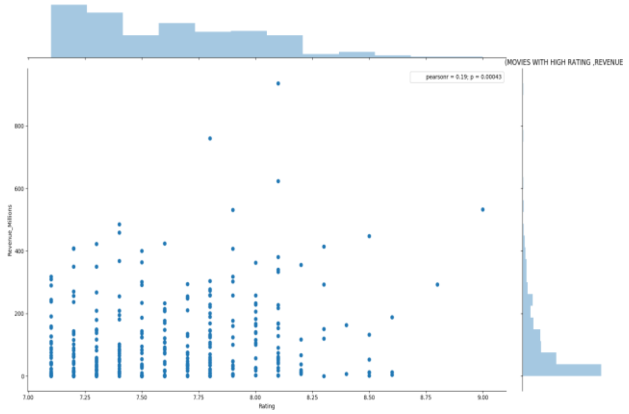


Fig 2. Movies with high rating & revenue

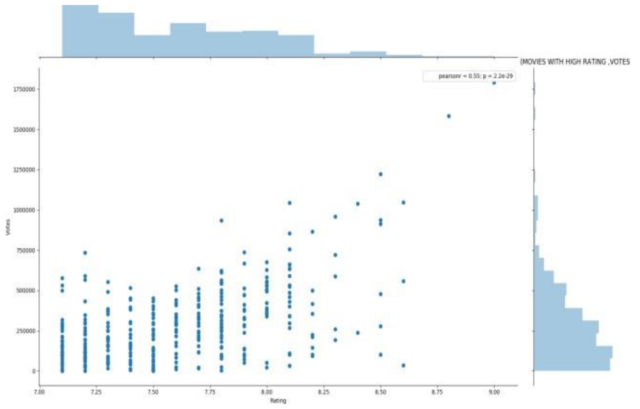


Fig 3. Movies with High rating & Votes

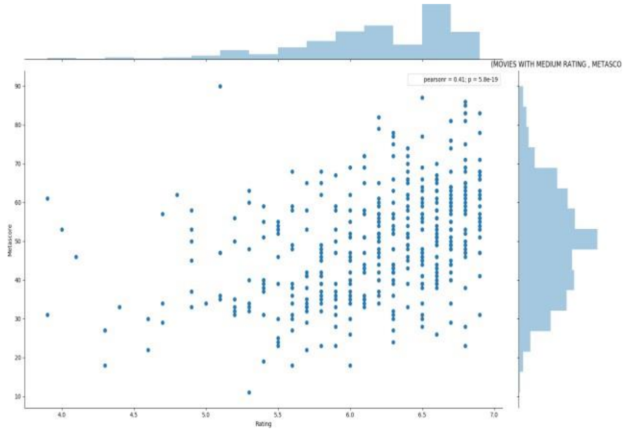


Fig 4. Movies with Medium rating & metascore

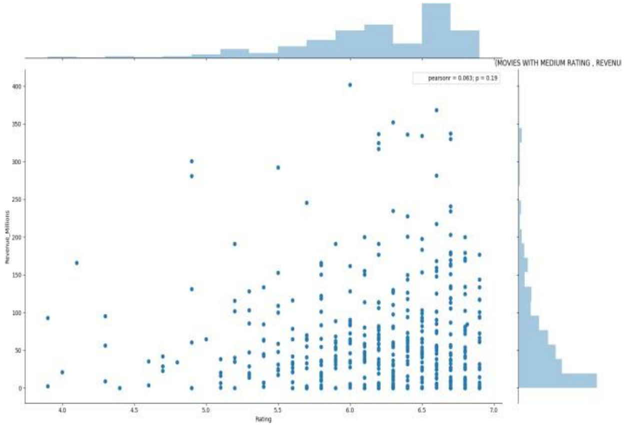


Fig 5. Movies with Medium rating & revenue

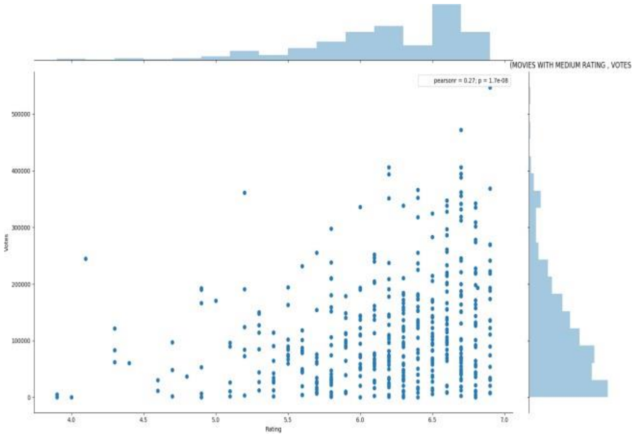
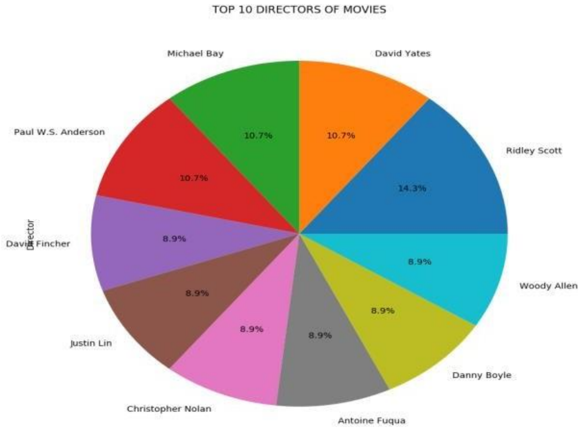
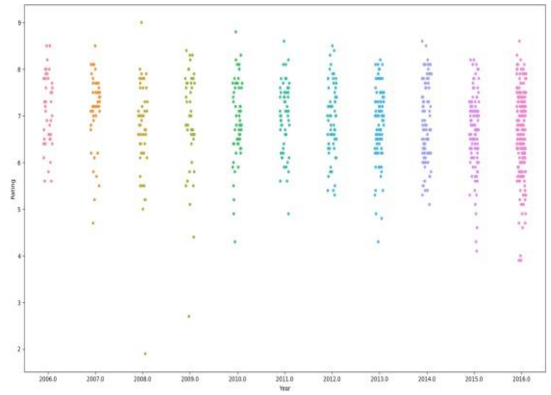


Fig 6. Movies with medium rating & votes

B. *Top Ten Directors*



C. *Rating V/S Year*



V. Conclusion AND FUTURE WORKS

Algorithms	Accuracy	Precision	Recall	AUC
Logistic Regression	0.9000	0.7105	0.7297	0.8300
Support Vector Machines	0.9190	0.7941	0.7297	0.8400
K-Nearest Neighbor Algorithm	0.8309	0.5272	0.7837	0.8200
Naïve Bayes Algorithm	0.5333	0.2706	0.9729	0.7100
Adaboost Algorithm	0.9952	0.9736	1.0000	1.0000

After building the models we found out that the success percentage for all models were nearly the same however the Adaboost with Decision Tree model had the highest accuracy in our case for predicting the movies success. We can notice that a huge and larger training data becomes the key to improve the performance of the model. We also need to look into additional features such as locations, age of voters and viewers, current trends, newspaper analysis, plot analysis and social network data analysis. This could be done and the information obtained could be added to the training data. We can also use Google trends result to improve the result.

References

[1] Darin Im, Minh Thao, Dang Nguyen, Predicting Movie Success in the U.S. market, Dept.Elect.Eng, Stanford Univ., California, December,2011

[2] Jiawei Han, Micheline Kamber, Jian Pei, Data Mining Concepts and Techniques, 3rd ed.MA:Elsevier, 2011, pp. 83-117

[3] Richard O. Duda, Peter E. Hart, David G. Stork, Pattern Classifification, 2nd. NewYork: Wiley, 1973

[4] Cohen, J., Cohen P., West, S.G., & Aiken, L.S. (2003). Applied multiple regression correlation analysis for the behavioral sciences. (2nd ed.) Hillsdale, NJ: Lawrence Erlbaum Associates

[5] Christopher M. Bishop (2006), Pattern Recognition and Machine Learn-ing, Springer, p. 205.

[6] The International Movie Database (IMDb): https://www.kaggle.com/PromptCloudHQ/imdb-data

[7] Freund, Y.: An adaptive version of the boost by majority algorithm. Machine Learning 43(3), 293–318 (2001)

[8] Haiyi Zhang, Di Li Jodrey School of Computer Science Acadia University, Canada, Naïve Bayes Text Classifier (2007)

[1]