A project report on

Movie Success Prediction using Data Mining

For Data Mining and Business Intelligence(ITA5007)
of
Master of Computer Application

By

SAURABH KUMAR	18MCA0123
AVINAY MEHTA	18MCA0100
JOY PAL	18MCA0095

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School Of Information Technology and Engineering

April, 2019.

Abstract: In this project, we apply data mining technique and machine learning algorithms using **R** software to predict the success and failure of movie based on several attributes. In order of doing this, we will develop a methodology on the basis of historical data to reduce certain level of uncertainty concerned to movie's future outcome. Some of the criteria in calculating movie success included budget, actors, director, producer, story writer, movie release day, competing movie releases at the same time, music, release location and target audience. Since, movie making involves huge investment thus movie prediction plays a vital role in the movie industry. This model helps movie makers to modify the criteria of blockbusters. It also helps movie watchers to determine a blockbuster before purchasing a ticket. Each attribute has some criteria and on the basis of that weightage has been given and then prediction is made based on that. Here, we also analyse key factors for movie profitability. This project also show the power of predictive and prescriptive data analytics for information systems to aid movie business decisions. This model also helps to find out the review of the new movie.

Keywords: Data Mining, Machine Learning, Movie, R Software

Introduction: Movies is the most convenient way to entertain peoples. However only few movies get higher success and are ranked high. Many movies are produces by the movie industry in a year. A movie revenue depends on various components such as cast acting in a movie, budget for the making of the movie, film critics review, rating for the movie, release year of the movie, etc. Because of these multiple components there is no formula that helps us to provide analysis for predicting how much revenue a particular movie will be generating. However by analyzing the revenues generated by previous movies, a model can be built which can help us predict the expected revenue for a particular movie. As we know in today's world the movie is one of the biggest source of entertainment and also for business purposes. . To expend this business further we need the technology through which we can predict the success rate of the movie. If we were able to predict the movie success rate in the correct manner then it will be easy for the businessman to get higher profit from it and also if the prediction shows the success rate is low of certain movie then it helps those businessmen to improve the content of the movie so that they can get higher revenue from it. success rate of movies, models and mechanisms can be used to predict the success of a movie. It will help the business significantly.stakeholders such as actors, producers, directors etc. can use these predictions to make more informed decisions. They can make the decision before the movie release. This proposed work aims to develop a model based upon the data mining techniques that may help in predicting the success of a movie in advance thereby reducing certain level of uncertainty. The excellent way to find detailed information about almost every film ever made is through IMDb. Vast amount of data, which contains much valuable information about general trends in films. Data mining techniques enable us to uncover information which will both confirm or disprove common assumptions about movies, and also allow us to predict the success of a future film given select information about the film before its release. So here we are developing the software for data analytics through which we can predict the success rate of the movie which high accuracy. Here we are using the R-software to predicting the movie success rate into which first we have downloaded the data set from kaggle.com and after that we are generating the training and test data set. In a dataset, a training set is implemented to build up a model, while a test (or validation) set is to validate the model built. The main attributes selected for building model are critics_score, imdb_rating, imdb_num_votes, audience_score. Data points in the training set are excluded from the test (validation) set. Usually, a dataset is divided into a training set, a validation set (some people use 'test set' instead) in each iteration, or divided into a training set, a validation set and a test set in each iteration. These training and test dataset is used to build model for selected set of attributes.

On the basis of the generated model prediction have been done and result have been generated. Through the acquired result we can easily conclude that the movie is hit, superhit, blockbuster or flop. The outcome of this research is therefore twofold, it provides tools and techniques to transform the database data into a format suitable for data mining, and provides a selection of information mined from this refined data.

Literature Survey

In paper [1], predictive models for the box office performance of the movies was represented by factors derived from social media and IMDb. According to our models, we have identified the following patterns: (1) the popularity of leading actress is crucial to the success of a movie, (2) the combination of past successful genre and a sequel movie is another pattern for success, (3) a new movie in the not popular genre and an actor with low popularity could be a pattern for a Flop. It is surprising that sentiment score and view and comment counts were not identified as relevant in our experiments. Author believe it is related to how weights are assigned to each attribute. Further studies to determine different weighting methods will be beneficial. In addition, our prediction is for movies yet to be released. The preliminary result of tracking 13 of the movies shows a good prediction performance from our model. A follow-up study on the final performance of our models will be validated and presented once all of the movies are released. Future work to improve our models will include further refinement of the Neutral class and characterization of movie box office performance in terms of net profits and profit ratios.

In paper[2], author have come up with a mathematical model to find the success rating of upcoming movies based on certain factors. As per their mathematical model, it was concluded that one factor was movie genres which determined the success rating of movies. It was also determined that the movie success depends on the cast of the movies. There was a strong correlation found between actors and the genres indicating that certain actors tend to work in certain genres. The actors and genres in turn define the success rating of the movie. Our work and results can be used to predict success or failure of upcoming movies by the movie makers as well as by the audience. A limitation of our work is that it focuses on only Bollywood movies currently. In the future, [2] will expand the model to include Hollywood movies.

In paper[3], author predict movie based on online reviews is a trendy approach for public to share their sentiments and opinions and it is more useful in business intelligence. [3] author used a variety of algorithms for predicting the sales performance of bollywood and Hollywood movies using sentiment information mined from reviews and tweets. Our main work is that use of S-PLSA model. Using S-PLSA author summarize the sentiment information from online reviews and tweets, they have used the ARSA model for predicting sales performance of movies using sentiment information and past box office performance. They further classify the reviews into positive, negative and neutral after that set a simple metric PN ratio and set threshold value to predict the success of movies, i.e. Hit, Average and Flop.

In paper[4], the author shows success of a movie in the box office depends on different aspects. Yet, predicting the performance of a movie is a crucial part of decision-making among the stakeholders of the movies. Over the time several techniques have been proposed to predict the box office performance of a movie. The existing works mainly concentrated on predicting the performance of a movie by applying sentiment analysis on comments collected from several sources like YouTube, news, blogs. Without the sentiment analysis from the comments, other data like number of views, number of likes, number of dislikes, number of dislikes are totally omitted in the existing works. However, these data could play a vital role in predicting the performance of a movie in the box office. So in [4] work, author prepared a dataset which contained the data that were previously overlooked. Later on, we performed an experiment by applying different data mining techniques on the prepared dataset to identify the most suitable technique for predicting gross income of a movie. The experiment showed that Linear regression is the most suitable method for predicting the gross income of a movie. In this work, we only worked with the trailer data collected from YouTube. As for future work, we recommend performing an experiment by adding the data from different sources. Furthermore, the numerical value derived from sentiment analysis can be used along with the data introduced in this work for predicting the performance of a movie.

Dataset Description:-

The dataset requirement for our project is fulfilled through kaggle repository. From here we downloaded the dataset and used as an input. Kaggle.com is a website that provides dataset for free for its users. Thus we got dataset for free of cost. This dataset consists of 651 rows and 32 columns. The dataset we get is preprocessed so we need not to pre-process it. Our first task for this assignment is to choose which variables to include in our model. It would be easier to start with eliminating variables that will obviously not be of use for our model. Uniform Resource Locators or commonly known as URLs provide an easy way to find more information for each movie but will not provide information whether a movie is popular or not. Runtime or length of the movie in minutes is not a key ingredient for popularity for a movie. Most movies are of similar length. Runtime would probably be a good predictor of movie genre. Animation and Documentaries are generally shorter than feature films. The title of a movie is usually what a moviegoer remembers when a movie is popular but it is not what makes a movie popular. However, this not the case when it comes to actors, actresses or directors. Movie goers turn into devoted fans when a certain actor, actress or director captures their imagination and becomes a key determiner whether subsequent movies from the same person is a must see. Let us now focus our attention to choosing our response variable. Since, we are working on development of model and prediction, the requirement of our data is numerical. Thus, the attribute selected here must be an attribute containing numerical values.

Table 1:- Below is a list of the variables that measures a movie's popularity.

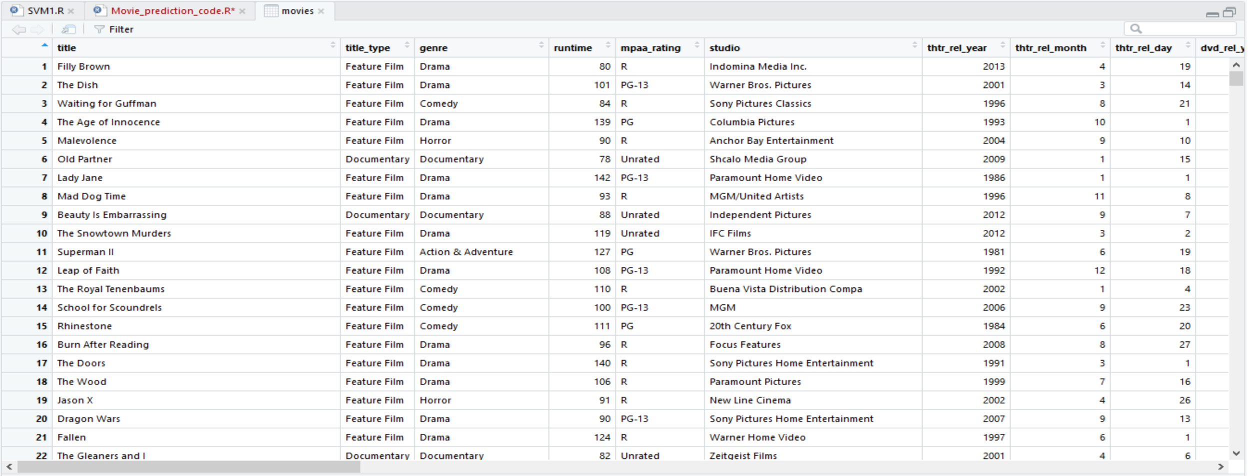
Variable	Description	Data Type
Imdb_rating	Rating on IMDB	Num
Imdb_num_votes	Number of votes on IMDB	Int
critics_rating	Critics rating on IMDB	Factor
Critics_score	Critics score on IMDB	Num
audience_rating	Audience rating on IMDB	Factor
Audience_score	Audience score on IMDB	num

Since we are doing linear regression in place of logistic regression, there is a need for us to analyse the central tendency and quintiles of these attributes. The table below show the acquired outputs:-

Table 2:- Below table show the min, max, mean, median, Q1, Q2 and Q3

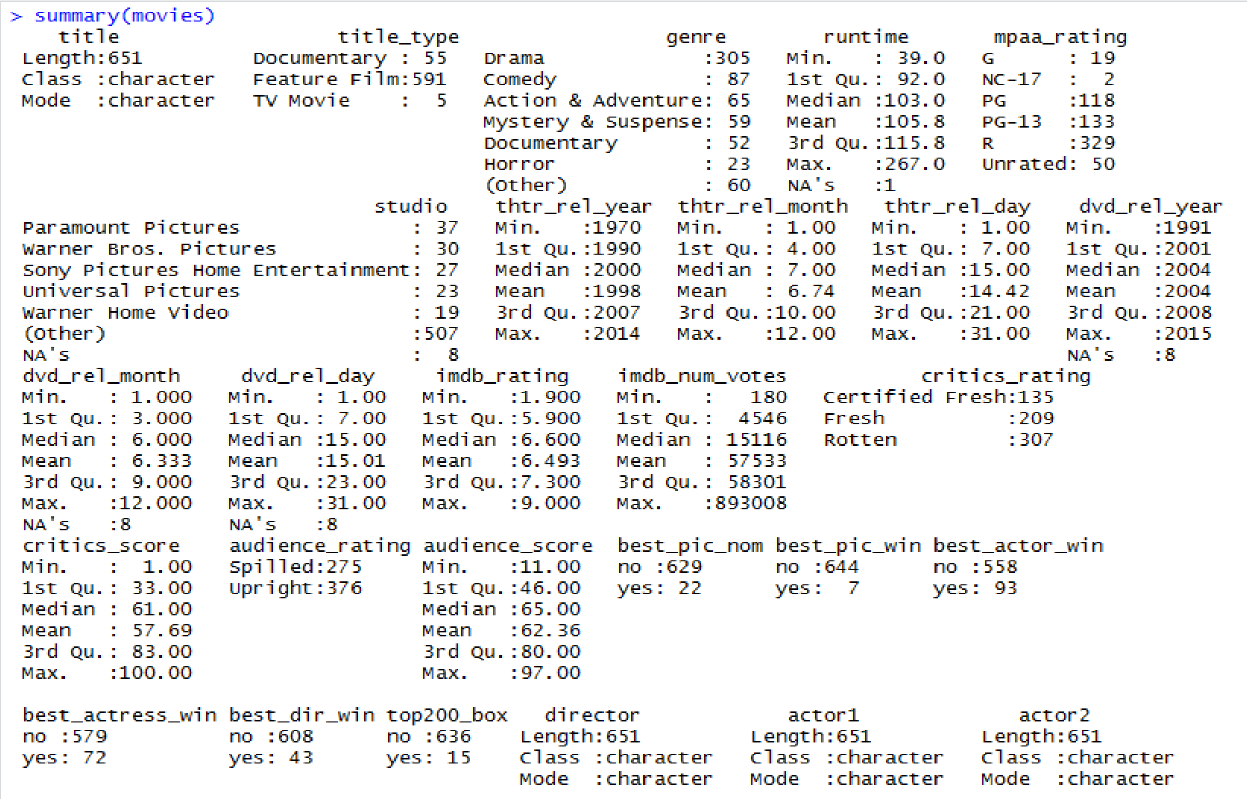
	imdb_rating	**imdb_num_votes**	**critics_score**	**audience_score**
Min	1.90	180	1.00	11.00
Max	9.00	893008	100.00	97.00
Mean	6.49	57533	57.72	62.41
Median	6.60	15116	61.00	65.00
Q1	5.90	4546	33.00	46.00
Q3	7.30	58301	83.00	80.00

Figure 1:- Cleaned Dataset



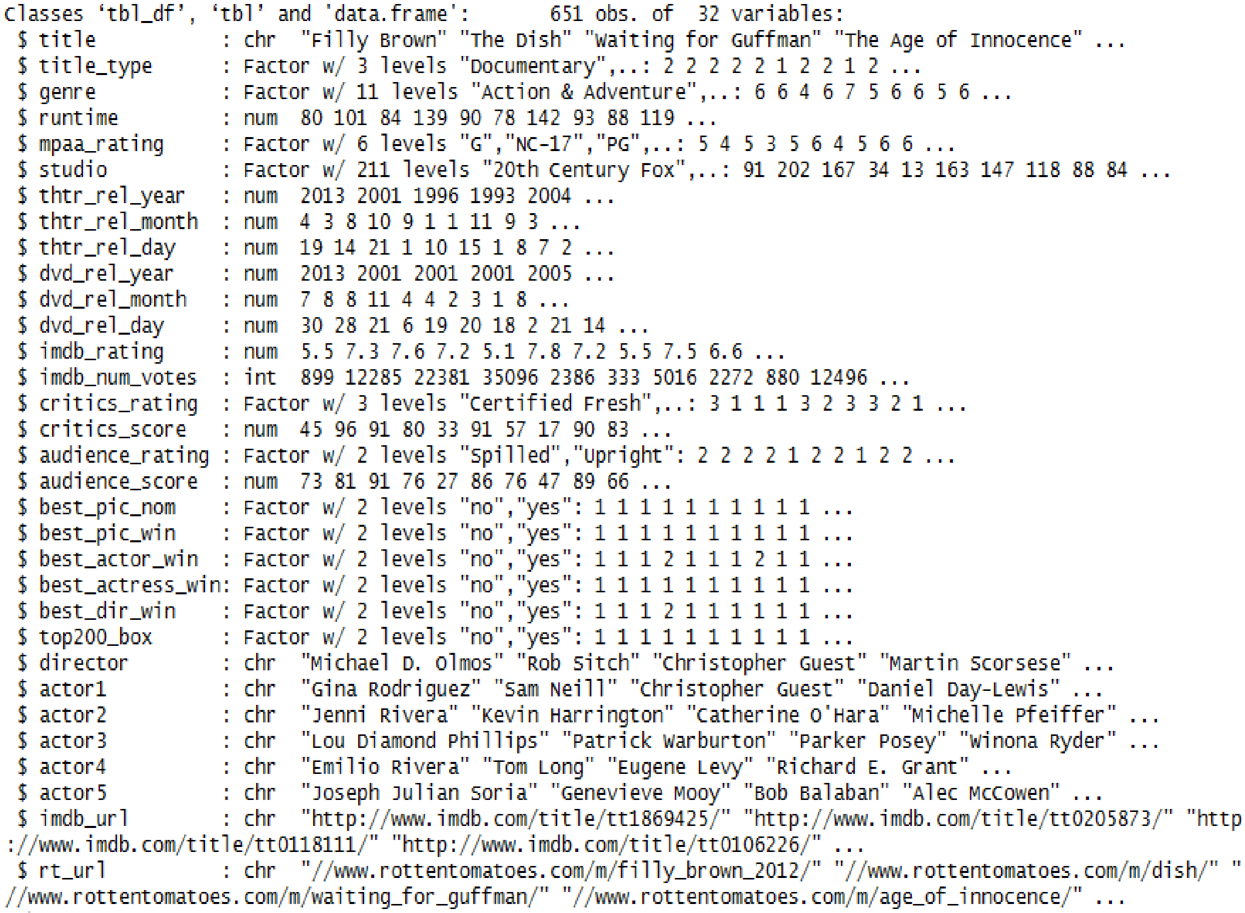
	title	title_type	genre	runtime	mpaa_rating	studio	thtr_rel_year	thtr_rel_month	thtr_rel_day	dvd_rel_y
1	Filly Brown	Feature Film	Drama	80	R	Indomina Media Inc.	2013	4	19	
2	The Dish	Feature Film	Drama	101	PG-13	Warner Bros. Pictures	2001	3	14	
3	Waiting for Guffman	Feature Film	Comedy	84	R	Sony Pictures Classics	1996	8	21	
4	The Age of Innocence	Feature Film	Drama	139	PG	Columbia Pictures	1993	10	1	
5	Malevolence	Feature Film	Horror	90	R	Anchor Bay Entertainment	2004	9	10	
6	Old Partner	Documentary	Documentary	78	Unrated	Shcalo Media Group	2009	1	15	
7	Lady Jane	Feature Film	Drama	142	PG-13	Paramount Home Video	1986	1	1	
8	Mad Dog Time	Feature Film	Drama	93	R	MGM/United Artists	1996	11	8	
9	Beauty Is Embarrassing	Documentary	Documentary	88	Unrated	Independent Pictures	2012	9	7	
10	The Snowtown Murders	Feature Film	Drama	119	Unrated	IFC Films	2012	3	2	
11	Superman II	Feature Film	Action & Adventure	127	PG	Warner Bros. Pictures	1981	6	19	
12	Leap of Faith	Feature Film	Drama	108	PG-13	Paramount Home Video	1992	12	18	
13	The Royal Tenenbaums	Feature Film	Comedy	110	R	Buena Vista Distribution Compa	2002	1	4	
14	School for Scoundrels	Feature Film	Comedy	100	PG-13	MGM	2006	9	23	
15	Rhinestone	Feature Film	Comedy	111	PG	20th Century Fox	1984	6	20	
16	Burn After Reading	Feature Film	Drama	96	R	Focus Features	2008	8	27	
17	The Doors	Feature Film	Drama	140	R	Sony Pictures Home Entertainment	1991	3	1	
18	The Wood	Feature Film	Drama	106	R	Paramount Pictures	1999	7	16	
19	Jason X	Feature Film	Horror	91	R	New Line Cinema	2002	4	26	
20	Dragon Wars	Feature Film	Drama	90	PG-13	Sony Pictures Home Entertainment	2007	9	13	
21	Fallen	Feature Film	Drama	124	R	Warner Home Video	1997	6	1	
22	The Gleaners and I	Documentary	Documentary	82	Unrated	Zeitgeist Films	2001	4	6	

Figure 2:- Shows the summary information of dataset



```
> summary(movies)
    title                      title_type                     genre          runtime        mpaa_rating
 Length:651         Documentary : 55   Drama               :305   Min.   : 39.0   G      : 19
 Class :character   Feature Film:591   Comedy              : 87   1st Qu.: 92.0   NC-17  :  2
 Mode  :character   TV Movie    :  5   Action & Adventure: 65   Median :103.0   PG     :118
                                       Mystery & Suspense: 59   Mean   :105.8   PG-13  :133
                                       Documentary         : 52   3rd Qu.:115.8   R      :329
                                       Horror              : 23   Max.   :267.0   Unrated: 50
                                       (Other)             : 60   NA's   :1
                              studio   thtr_rel_year  thtr_rel_month   thtr_rel_day    dvd_rel_year
 Paramount Pictures              : 37   Min.   :1970   Min.   : 1.00   Min.   : 1.00   Min.   :1991
 Warner Bros. Pictures           : 30   1st Qu.:1990   1st Qu.: 4.00   1st Qu.: 7.00   1st Qu.:2001
 Sony Pictures Home Entertainment: 27   Median :2000   Median : 7.00   Median :15.00   Median :2004
 Universal Pictures              : 23   Mean   :1998   Mean   : 6.74   Mean   :14.42   Mean   :2004
 Warner Home Video               : 19   3rd Qu.:2007   3rd Qu.:10.00   3rd Qu.:21.00   3rd Qu.:2008
 (Other)                         :507   Max.   :2014   Max.   :12.00   Max.   :31.00   Max.   :2015
 NA's                            :  8                                                  NA's   :8
 dvd_rel_month     dvd_rel_day     imdb_rating    imdb_num_votes          critics_rating
 Min.   : 1.000   Min.   : 1.00   Min.   :1.900   Min.   :   180   Certified Fresh:135
 1st Qu.: 3.000   1st Qu.: 7.00   1st Qu.:5.900   1st Qu.:  4546   Fresh          :209
 Median : 6.000   Median :15.00   Median :6.600   Median : 15116   Rotten         :307
 Mean   : 6.333   Mean   :15.01   Mean   :6.493   Mean   : 57533
 3rd Qu.: 9.000   3rd Qu.:23.00   3rd Qu.:7.300   3rd Qu.: 58301
 Max.   :12.000   Max.   :31.00   Max.   :9.000   Max.   :893008
 NA's   :8        NA's   :8
 critics_score    audience_rating audience_score  best_pic_nom best_pic_win best_actor_win
 Min.   :  1.00   Spilled:275     Min.   :11.00   no :629      no :644      no :558
 1st Qu.: 33.00   Upright:376     1st Qu.:46.00   yes: 22      yes:  7      yes: 93
 Median : 61.00                   Median :65.00
 Mean   : 57.69                   Mean   :62.36
 3rd Qu.: 83.00                   3rd Qu.:80.00
 Max.   :100.00                   Max.   :97.00

 best_actress_win best_dir_win top200_box   director            actor1             actor2
 no :579          no :608      no :636    Length:651         Length:651         Length:651
 yes: 72          yes: 43      yes: 15    Class :character   Class :character   Class :character
                                          Mode  :character   Mode  :character   Mode  :character
```

Figure 3:- Description of attributes of dataset

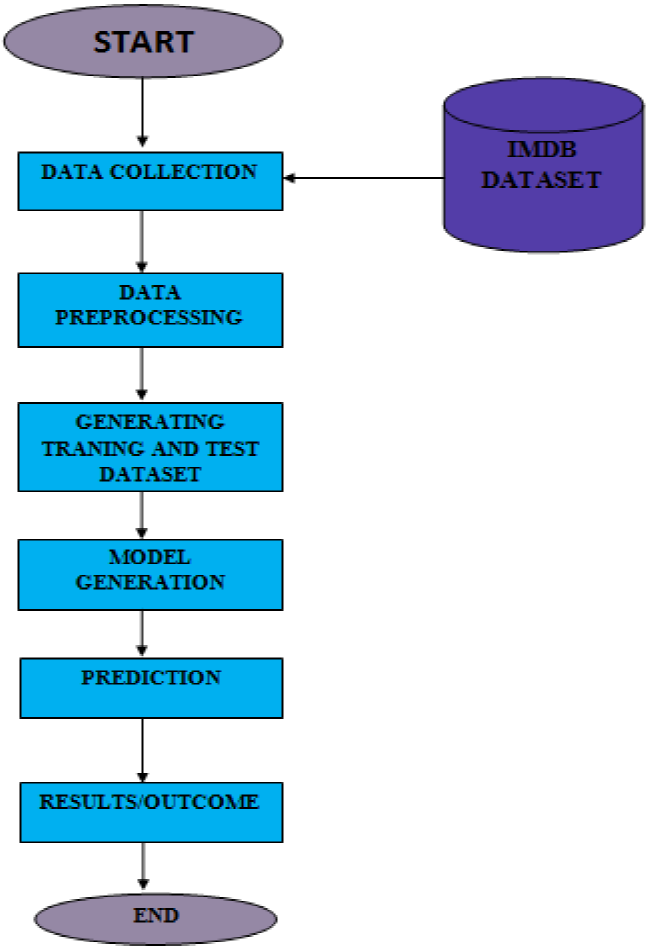


```
Classes ‘tbl_df’, ‘tbl’ and 'data.frame':	651 obs. of  32 variables:
 $ title           : chr  "Filly Brown" "The Dish" "Waiting for Guffman" "The Age of Innocence" ...
 $ title_type      : Factor w/ 3 levels "Documentary",..: 2 2 2 2 2 1 2 2 1 2 ...
 $ genre           : Factor w/ 11 levels "Action & Adventure",..: 6 6 4 6 7 5 6 6 5 6 ...
 $ runtime         : num  80 101 84 139 90 78 142 93 88 119 ...
 $ mpaa_rating     : Factor w/ 6 levels "G","NC-17","PG",..: 5 4 5 3 5 6 4 5 6 6 ...
 $ studio          : Factor w/ 211 levels "20th Century Fox",..: 91 202 167 34 13 163 147 118 88 84 ...
 $ thtr_rel_year   : num  2013 2001 1996 1993 2004 ...
 $ thtr_rel_month  : num  4 3 8 10 9 1 1 11 9 3 ...
 $ thtr_rel_day    : num  19 14 21 1 10 15 1 8 7 2 ...
 $ dvd_rel_year    : num  2013 2001 2001 2001 2005 ...
 $ dvd_rel_month   : num  7 8 8 11 4 4 2 3 1 8 ...
 $ dvd_rel_day     : num  30 28 21 6 19 20 18 2 21 14 ...
 $ imdb_rating     : num  5.5 7.3 7.6 7.2 5.1 7.8 7.2 5.5 7.5 6.6 ...
 $ imdb_num_votes  : int  899 12285 22381 35096 2386 333 5016 2272 880 12496 ...
 $ critics_rating  : Factor w/ 3 levels "Certified Fresh",..: 3 1 1 1 3 2 3 3 2 1 ...
 $ critics_score   : num  45 96 91 80 33 91 57 17 90 83 ...
 $ audience_rating : Factor w/ 2 levels "Spilled","Upright": 2 2 2 2 1 2 2 1 2 2 ...
 $ audience_score  : num  73 81 91 76 27 86 76 47 89 66 ...
 $ best_pic_nom    : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
 $ best_pic_win    : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
 $ best_actor_win  : Factor w/ 2 levels "no","yes": 1 1 1 2 1 1 1 2 1 1 ...
 $ best_actress_win: Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
 $ best_dir_win    : Factor w/ 2 levels "no","yes": 1 1 1 2 1 1 1 1 1 1 ...
 $ top200_box      : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
 $ director        : chr  "Michael D. Olmos" "Rob Sitch" "Christopher Guest" "Martin Scorsese" ...
 $ actor1          : chr  "Gina Rodriguez" "Sam Neill" "Christopher Guest" "Daniel Day-Lewis" ...
 $ actor2          : chr  "Jenni Rivera" "Kevin Harrington" "Catherine O'Hara" "Michelle Pfeiffer" ...
 $ actor3          : chr  "Lou Diamond Phillips" "Patrick Warburton" "Parker Posey" "Winona Ryder" ...
 $ actor4          : chr  "Emilio Rivera" "Tom Long" "Eugene Levy" "Richard E. Grant" ...
 $ actor5          : chr  "Joseph Julian Soria" "Genevieve Mooy" "Bob Balaban" "Alec McCowen" ...
 $ imdb_url        : chr  "http://www.imdb.com/title/tt1869425/" "http://www.imdb.com/title/tt0205873/" "http
://www.imdb.com/title/tt0118111/" "http://www.imdb.com/title/tt0106226/" ...
 $ rt_url          : chr  "//www.rottentomatoes.com/m/filly_brown_2012/" "//www.rottentomatoes.com/m/dish/" "
//www.rottentomatoes.com/m/waiting_for_guffman/" "//www.rottentomatoes.com/m/age_of_innocence/" ...
```

Proposed Methodology:-

The proposed methodology deals with different stages of the project which consists of data collection, data preprocessing, generating training and testing dataset, model generation, prediction and outcomes. These all methods prevents us from getting any irrelavent data which further keeps our outcomes more relevant and accurate for the prediction. Here we collected dataset from IMDB which consist of 32 attributes and 651 tuples. Further steps are explained below:-

Figure 4:- Flow Diagram of implementation



Data Collection:

The crude IMDb dataset is organized so that the greater part of its properties and information is sorted out and put away independently in compacted plain content documents. For example, the majority of the approximately 651 movies picture appraisals from the database are put away in the compacted content document evaluations. Rundown which incorporates literary informationabout the information just as a table of film rank, the number of votes and film titles. In this way, some kind of cleaning, mix and preprocessing is probably going to be required so as to utilize the information with the end goal of information mining through

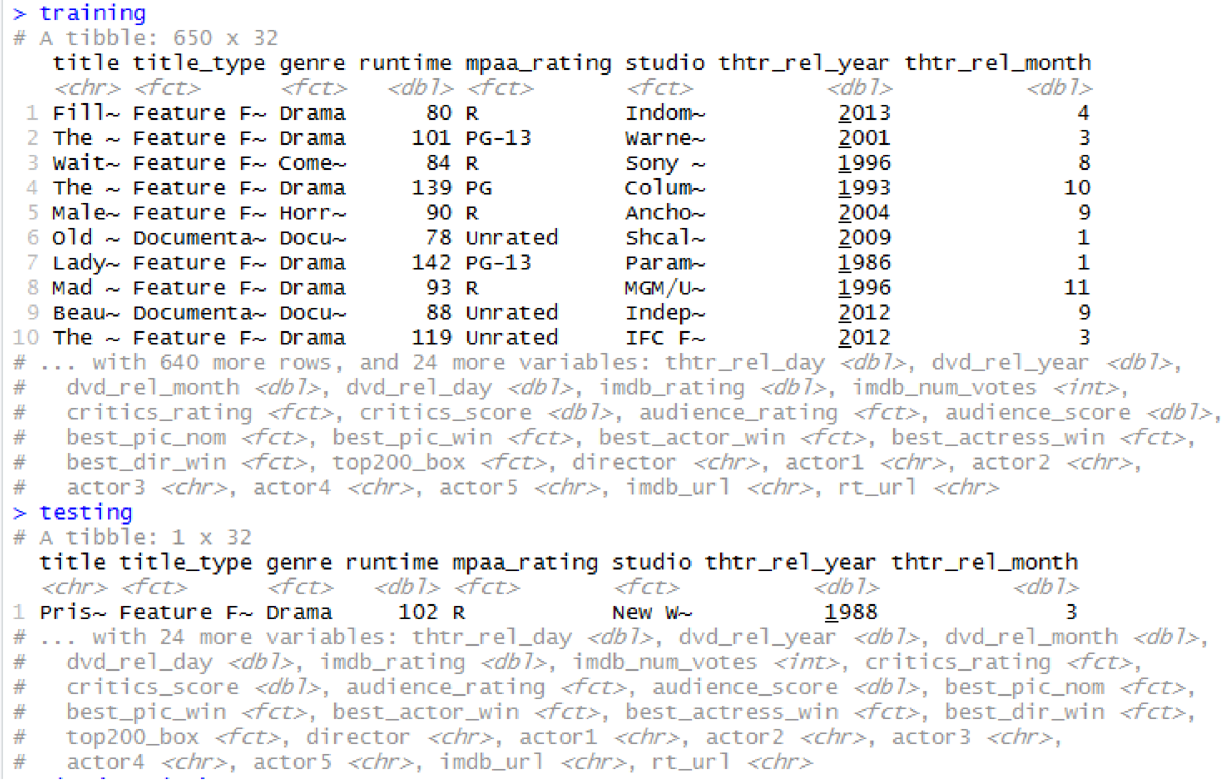
supervised machine learning strategies. The information was gathered utilizing IMDB which contains the IMDB motion picture dataset of in excess of 651 films in the dataset.

Data Preprocessing: In this stage dataset is prepared for applying data mining technique. Before applying data mining technique, pre-processing methods like cleaning, variable transformation and data partitioning and other techniques for attribute selection must be applied.

After pre-processing we have attributes or variables for each movie. Each test file will contain best attributes and rebalanced. As the data is taken in the raw format from IMDb it is first required to be pre-processed. To overcome missing value scenario central tendency method is used both mean and median and later the duplicate items are removed. Pre-processing is the crucial phase for the project as it mainly focuses on the working of the algorithm. As the data is now pre-processed next comes data integration and transformation in which the alpha numerical data need to be converted to the numerical data as it is required for regression model. The correlation between the features is identified using the greedy backwards method.

Generating Training and Test Dataset: Training dataset is a set of attributes used to fit the parameters of the model. The model like naive Bayes classifier is trained on the training dataset using supervised learning method like gradient descent or stochastic gradient descent. In practice, the training dataset often consist of pairs of an input vector and the corresponding output vector (or scalar), which is commonly denoted as the *target*. The current model is run with the training dataset and produces a result, which is then compared with the *target*, for each input vector in the training dataset. Based on the result of the comparison and the specific learning algorithm being used, the parameters of the model are adjusted. The model fitting can include both variable selection and parameter estimation. Finally, the test dataset is a dataset used to provide an unbiased evaluation of a *final* model fit on the training dataset.When the data in the test dataset has never been used in training like cross-validation, the test dataset is also called a holdout dataset.

Figure 5:- Below snapshot shows the developed training and testing dataset



```
> training
# A tibble: 650 x 32
   title title_type genre runtime mpaa_rating studio thtr_rel_year thtr_rel_month
   <chr> <fct>      <fct>   <dbl> <fct>       <fct>          <dbl>          <dbl>
 1 Fill~ Feature F~ Drama      80 R           Indom~          2013              4
 2 The ~ Feature F~ Drama     101 PG-13       Warne~          2001              3
 3 Wait~ Feature F~ Come~      84 R           Sony ~          1996              8
 4 The ~ Feature F~ Drama     139 PG          Colum~          1993             10
 5 Male~ Feature F~ Horr~      90 R           Ancho~          2004              9
 6 Old ~ Documenta~ Docu~      78 Unrated     Shcal~          2009              1
 7 Lady~ Feature F~ Drama     142 PG-13       Param~          1986              1
 8 Mad ~ Feature F~ Drama      93 R           MGM/U~          1996             11
 9 Beau~ Documenta~ Docu~      88 Unrated     Indep~          2012              9
10 The ~ Feature F~ Drama     119 Unrated     IFC F~          2012              3
# ... with 640 more rows, and 24 more variables: thtr_rel_day <dbl>, dvd_rel_year <dbl>,
#   dvd_rel_month <dbl>, dvd_rel_day <dbl>, imdb_rating <dbl>, imdb_num_votes <int>,
#   critics_rating <fct>, critics_score <dbl>, audience_rating <fct>, audience_score <dbl>,
#   best_pic_nom <fct>, best_pic_win <fct>, best_actor_win <fct>, best_actress_win <fct>,
#   best_dir_win <fct>, top200_box <fct>, director <chr>, actor1 <chr>, actor2 <chr>,
#   actor3 <chr>, actor4 <chr>, actor5 <chr>, imdb_url <chr>, rt_url <chr>
> testing
# A tibble: 1 x 32
  title title_type genre runtime mpaa_rating studio thtr_rel_year thtr_rel_month
  <chr> <fct>      <fct>   <dbl> <fct>       <fct>          <dbl>          <dbl>
1 Pris~ Feature F~ Drama     102 R           New W~          1988              3
# ... with 24 more variables: thtr_rel_day <dbl>, dvd_rel_year <dbl>, dvd_rel_month <dbl>,
#   dvd_rel_day <dbl>, imdb_rating <dbl>, imdb_num_votes <int>, critics_rating <fct>,
#   critics_score <dbl>, audience_rating <fct>, audience_score <dbl>, best_pic_nom <fct>,
#   best_pic_win <fct>, best_actor_win <fct>, best_actress_win <fct>, best_dir_win <fct>,
#   top200_box <fct>, director <chr>, actor1 <chr>, actor2 <chr>, actor3 <chr>,
#   actor4 <chr>, actor5 <chr>, imdb_url <chr>, rt_url <chr>
```

After generating, we got training dataset with 650 rows and 32 attributes and, testing dataset with 1 row and 32 attributes.

Data Analysis: In data analysis, all selected attributes are analyzed on the basis of different factor that help us to gather most accurate outcome for further stages. Selected features for analysis are as follows: imdb_rating, imdb_num_votes, critics_rating, critics_score, audience_rating and audience_score. On the basis of these attributes, we are generating various visualized graphs for analyzing the best possible attribute among these for further predictions.

An actor, actress, or director who has won an oscar award is a great motivation to analyze movie success. Movie who has won an oscar has the same weight as an actor or a director in making a movie popular. So, in our analysis we are generating scatter plot to show diference between number of oscar won by particular actor, actress and director.

Figure 6:- Distribution of critics_score and imdb_rating on scatter plot

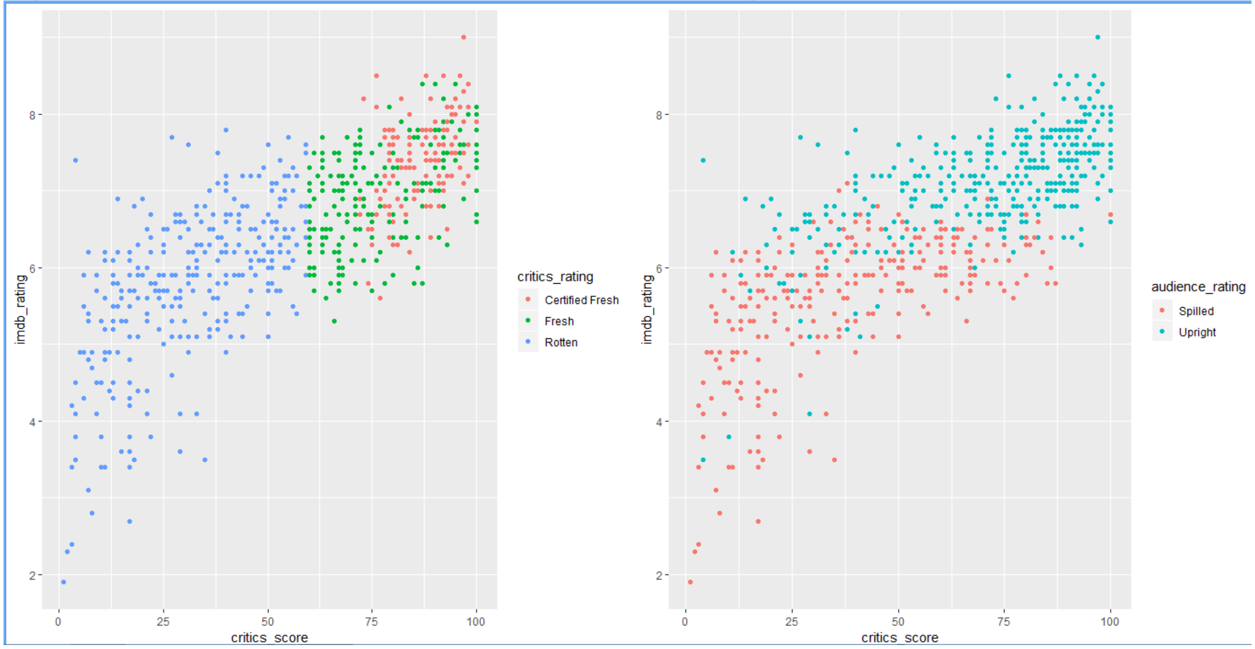


Figure 7:-Showing imdb_rating, imdb_num_votes, critics_score and audience_score data on histogram

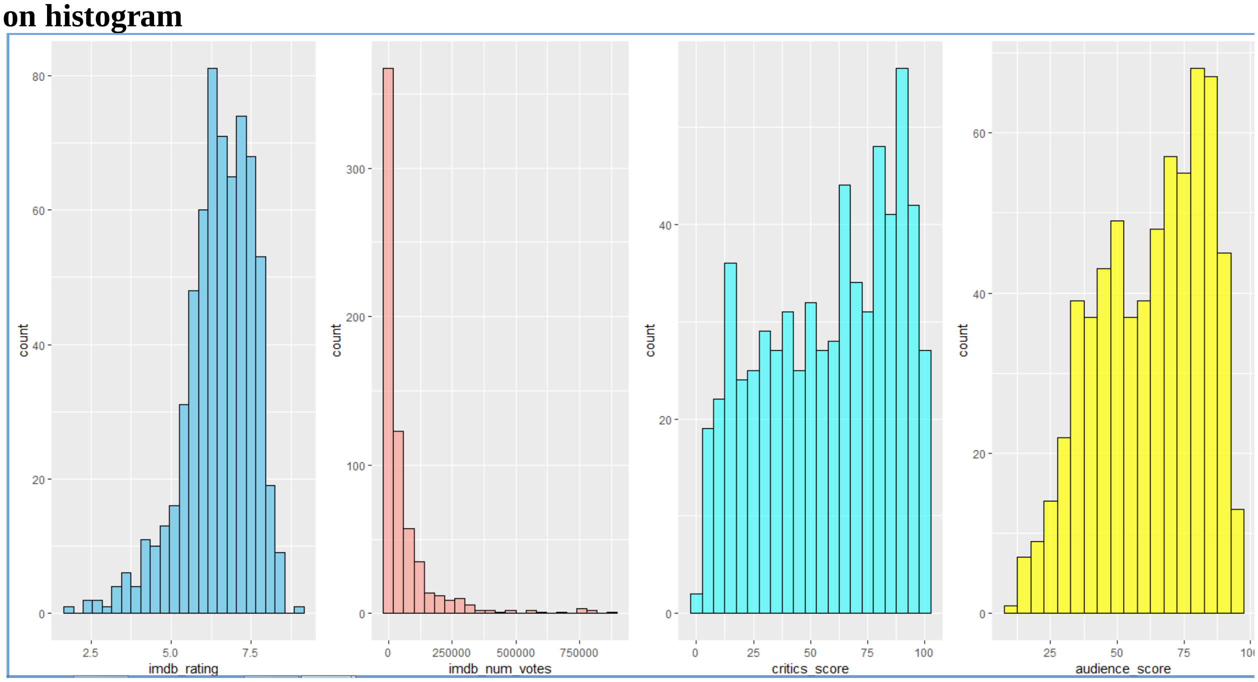
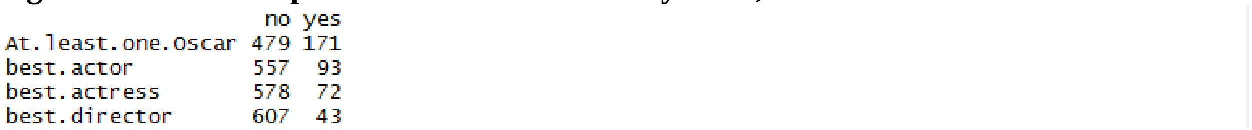
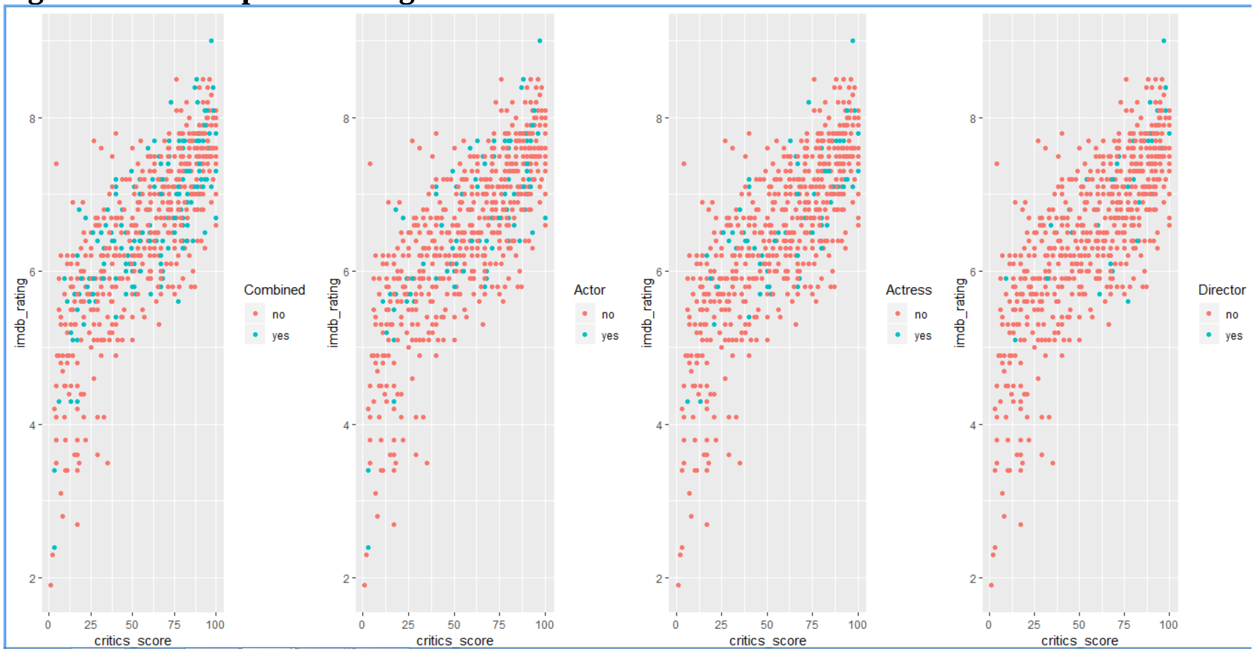


Figure 6:- Tabular representation of oscar win by actor, actress and director



```
                   no yes
At.least.one.Oscar 479 171
best.actor         557  93
best.actress       578  72
best.director      607  43
```

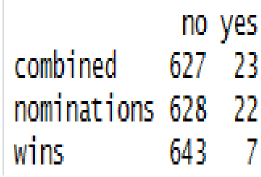
Figure7:- Scatter plot showing the above data for oscar win



From the plots above we can see the distribution of having an oscar winner as a cast in a movie. Oscar winners appear to be randomly distributed across the range of both the imdb_rating and critics_score and follows the linear trend in the data. It seems having an oscar winner in the movie is not a factor for movie success prediction.

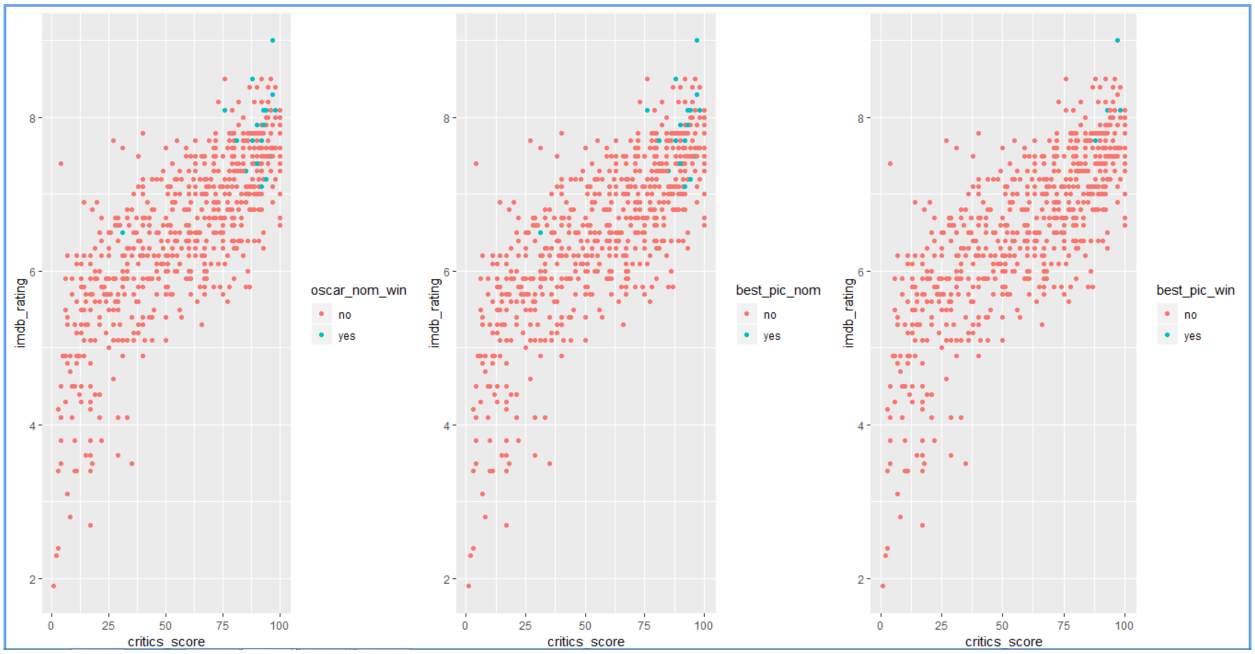
We will also be comparing whether getting an oscar nomination and winning an Oscar for best picture are the same in predicting movie popularity.

Figure 8:- Tabular representation of movie nomination and winning for oscar



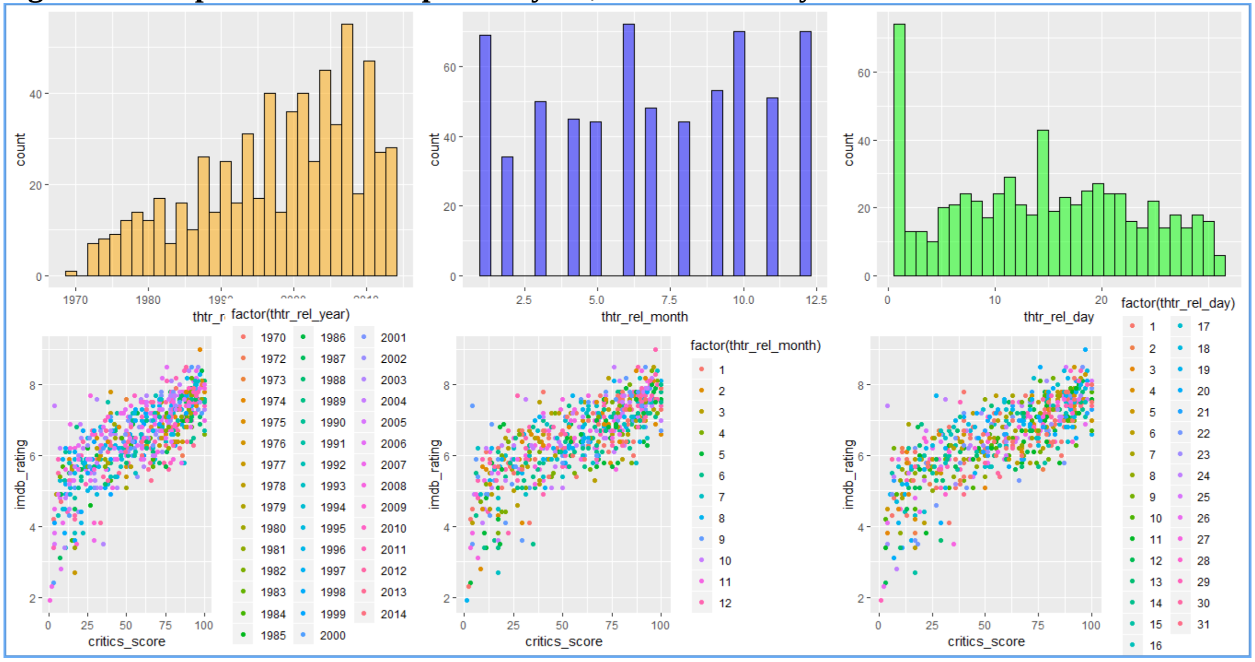
```
            no yes
combined    627  23
nominations 628  22
wins        643   7
```

Figure 9:- Scatter plot showing the above data for oscar nomination and winning



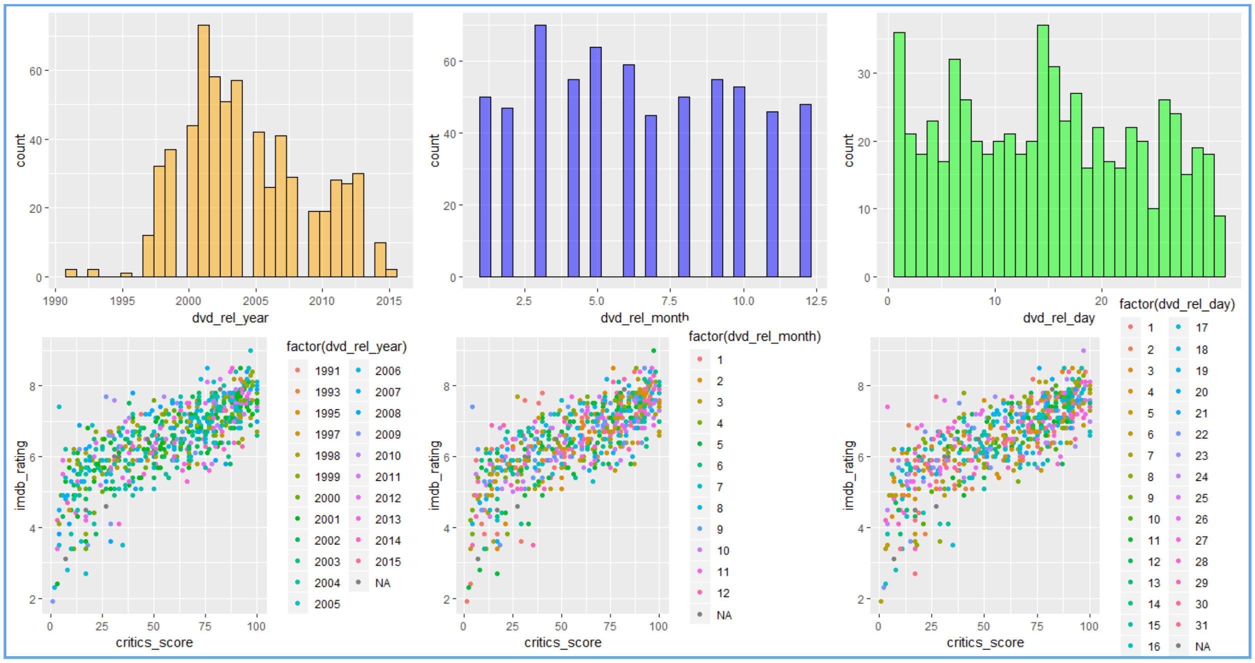
We can see that an oscar nomination for best picture has almost the same effect as winning an Oscar in terms of movie popularity. The points for the nominated films and those that won are all clustered in the extreme high end of both the imdb_rating and critics_scorescale except for one point that is in the middle.

Figure 9:- Barplot and Scatter plot for year, month and day basis movie release data



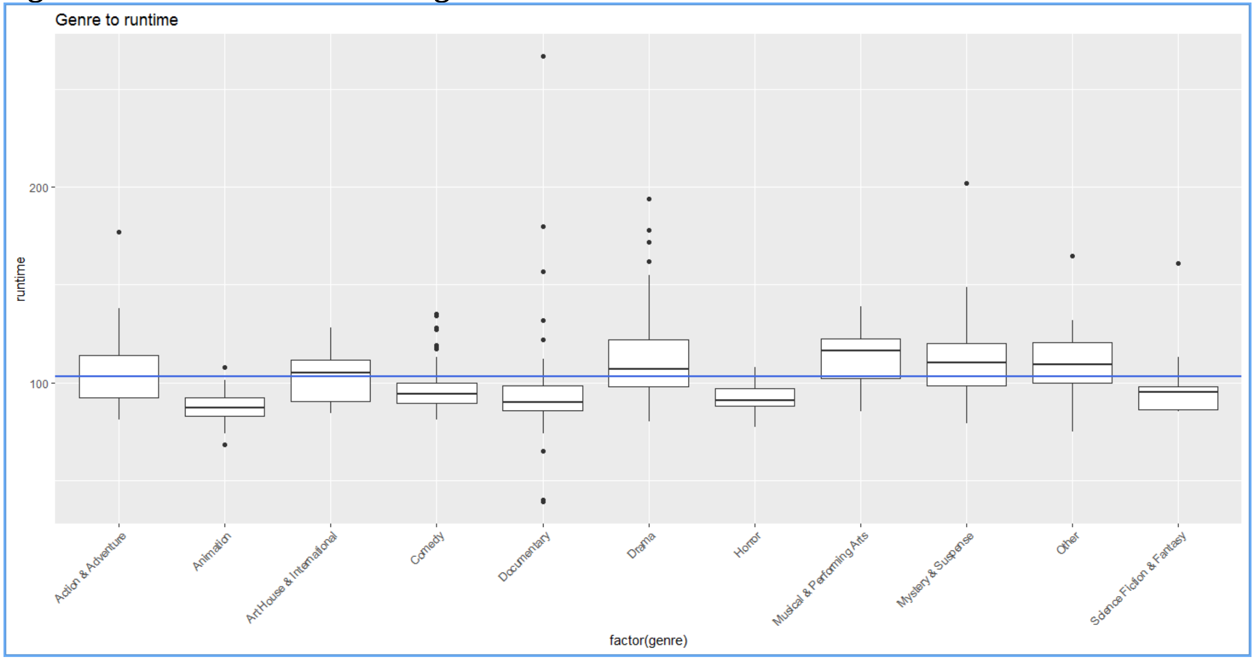
The histograms above show that there are particular years, months, and days where more movies are released in theaters for the first time. We do not observe any clustering of these points in the scatterplot along the imdb_rating and critics_score scale.

Figure 10:- Barplot and Scatter plot for year, month and day basis movie dvd release data



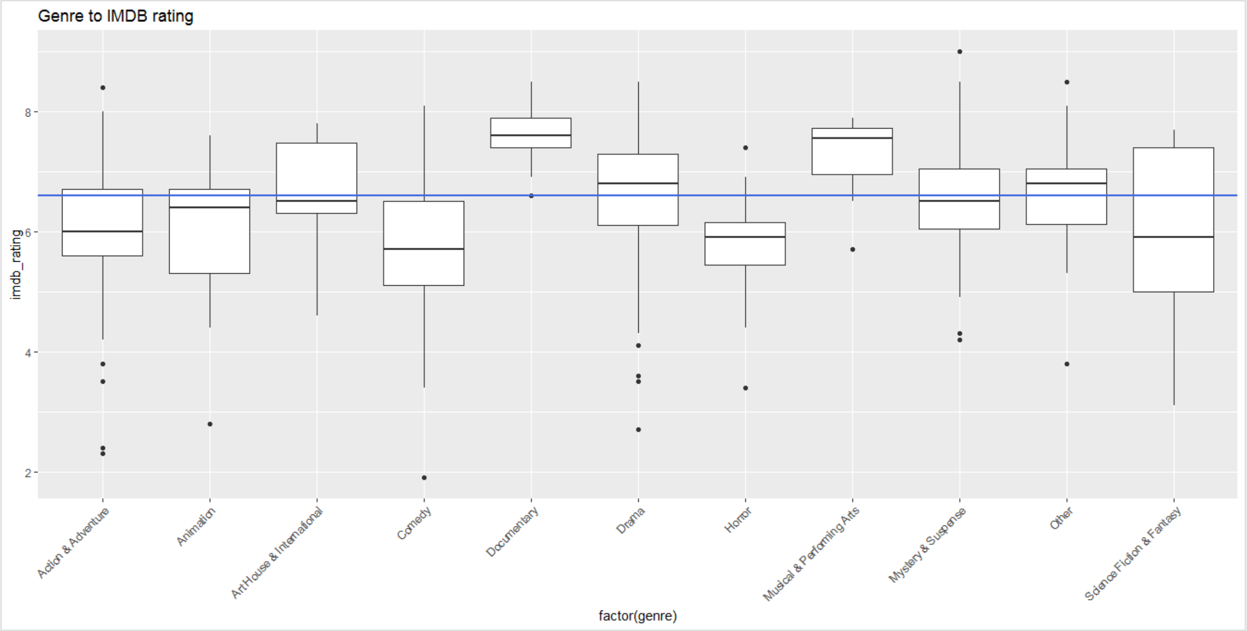
The histograms above show that there are particular years, months, and days where more dvds are released for the first time. We do not observe any clustering of these points in the scatterplot along the imdb_rating and critics_score scale.

Figure 11:- Outlier detection of genre and runtime



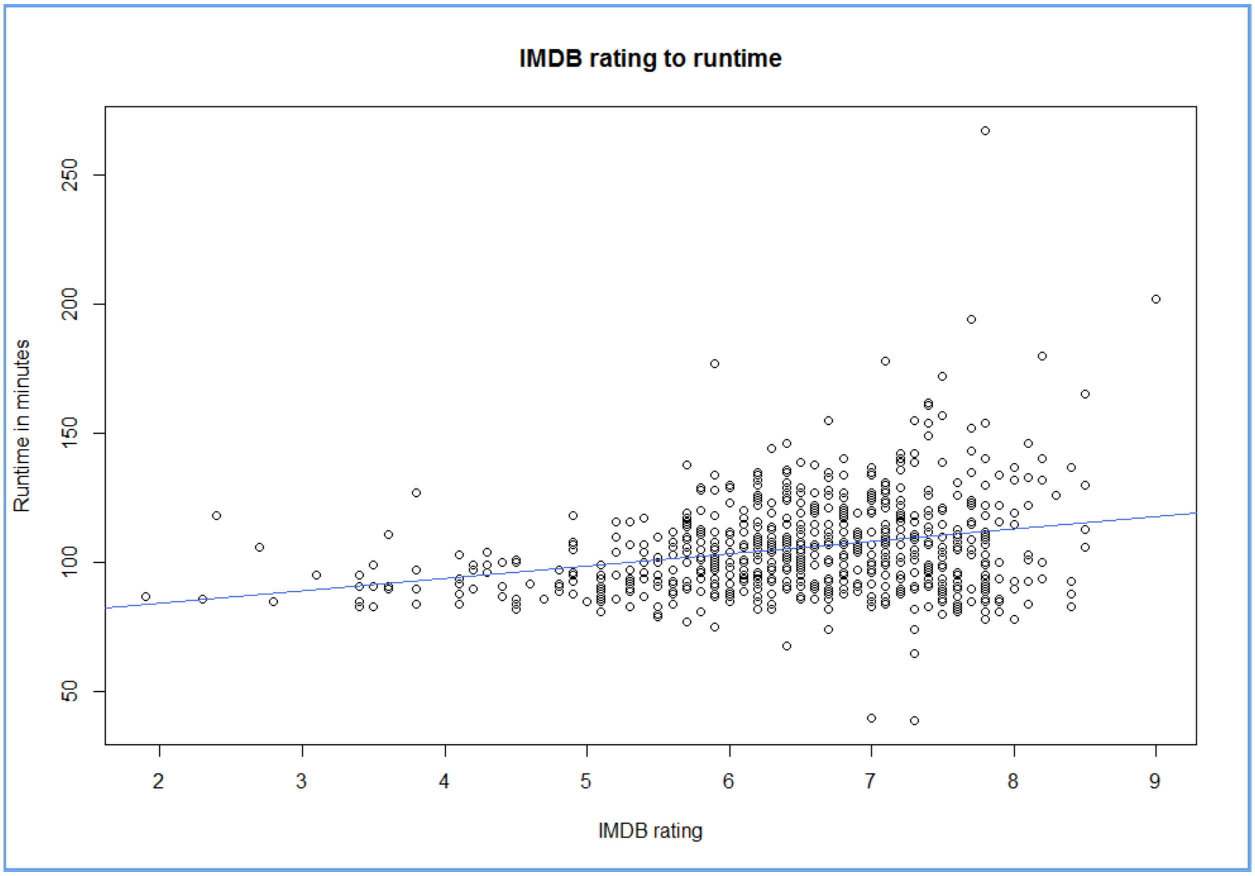
We can see all genres are oscillating around median runtime value of 103 minutes. There are lot of outliers in data, mostly documentary genre.

Figure 12:- Outlier detection of genre and imdb_rating



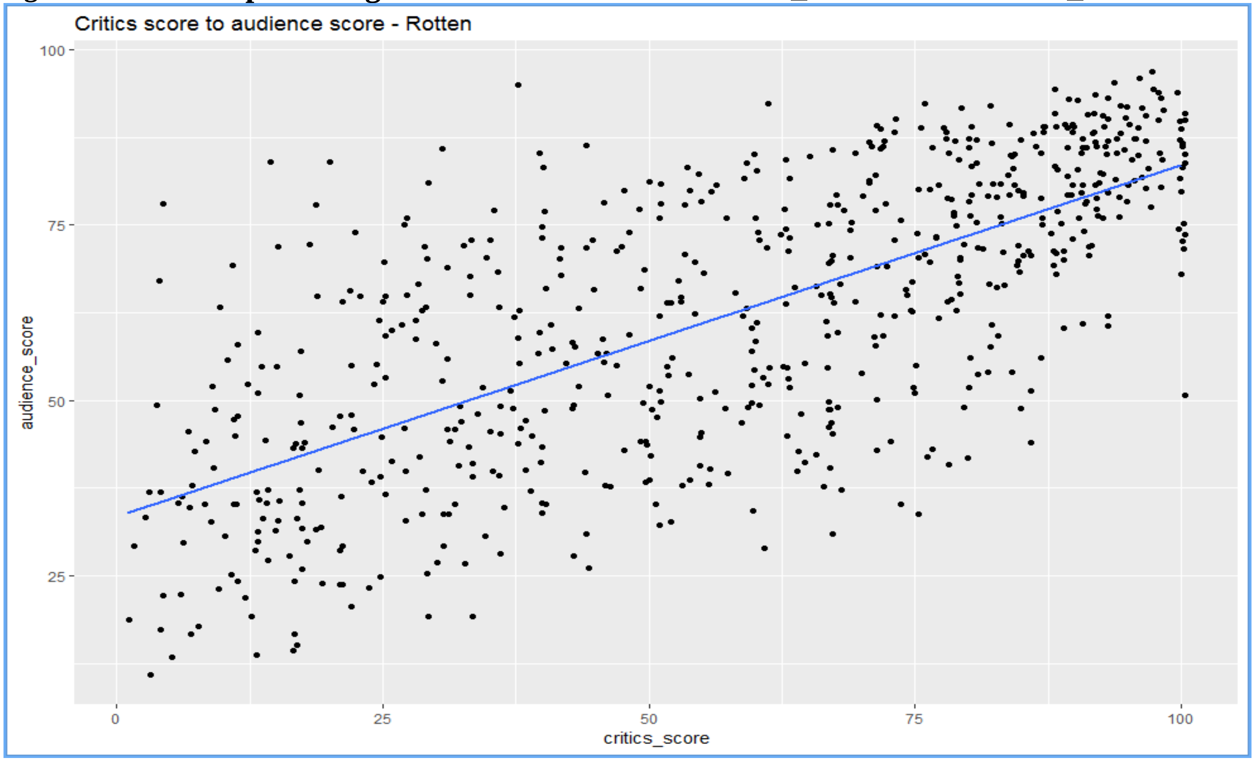
We can see that science-fiction movie has biggest variance between 1Q a 3Q, median of sci-fi genre is lower then median of all genres. Documentary performs best as Musical and performing arts movies. There are outliers too for lot of genres. From median of genre we can see that people tend to give higher rating on IMDB.

Figure 13:- Scatter plot along with line chart to show imdb_rating and movie runtime in minutes



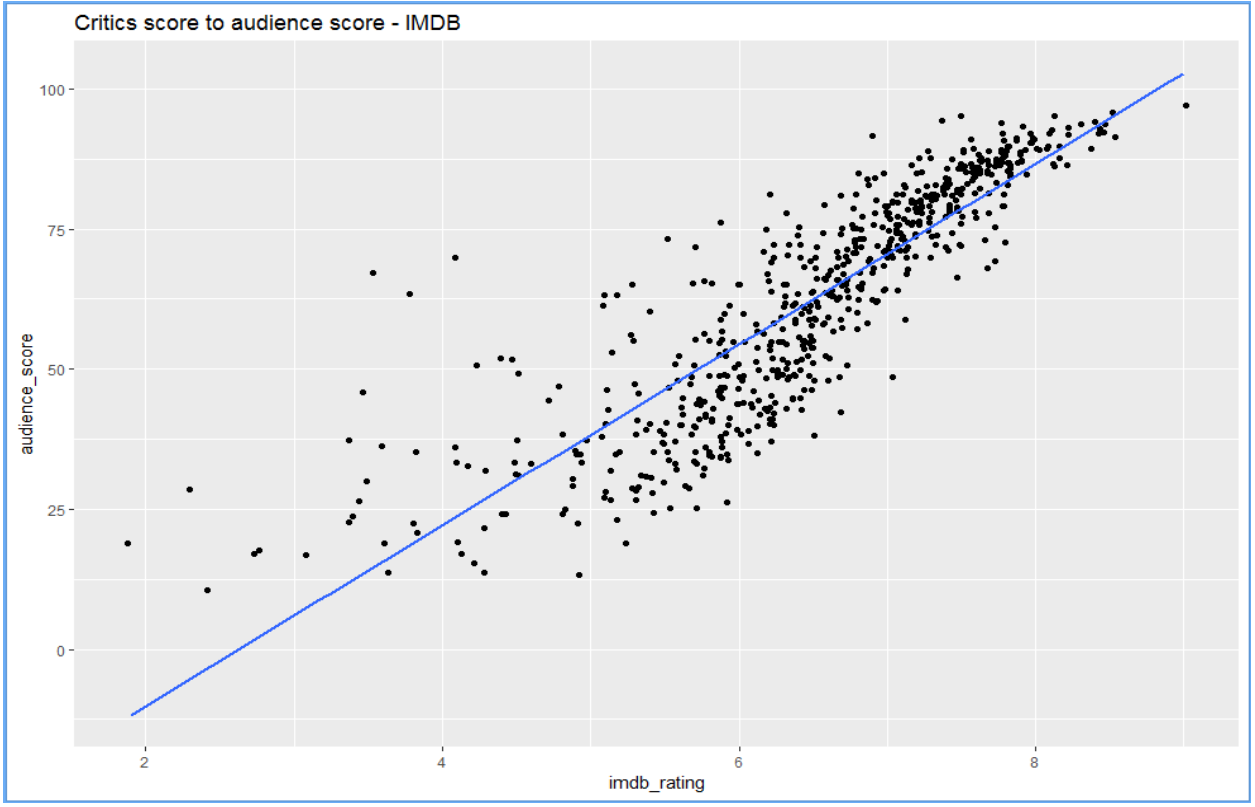
From imdb_rating and runtime of movie we can see that we can expect better IMDB rating if runtime is longer.

Figure 14:- Scatter plot along with line chart to show critics_score and audience_score



We can see there is weak correlation between critics score and audience score.

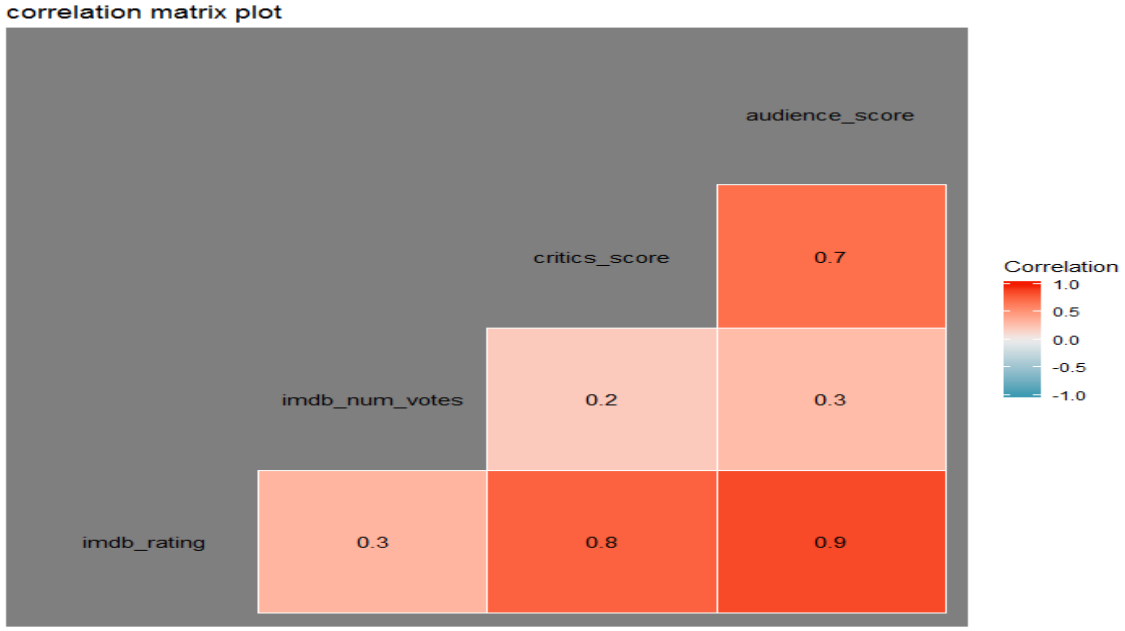
Figure 15:- Scatter plot along with line chart to show critics_score and audience_score of IMDB



We can see there is a strong correlation between critics score and audience score of IMDB dataset. Thus, our analysis of dataset is complete. Here, we got a strong correlation between critics_score and audience_score.

Model Generation: In simple terms, modeling is a simplified, mathematically-formalized way to approximate reality and optionally to make predictions from this approximation. The statistical model is the mathematical equation that is used. Representing a quantity by an average and a standard deviation is a very simple form of statistical modeling. In this project we are using correlation along with correlation matrix plot to generate our model. We also using linear regression for model generation.

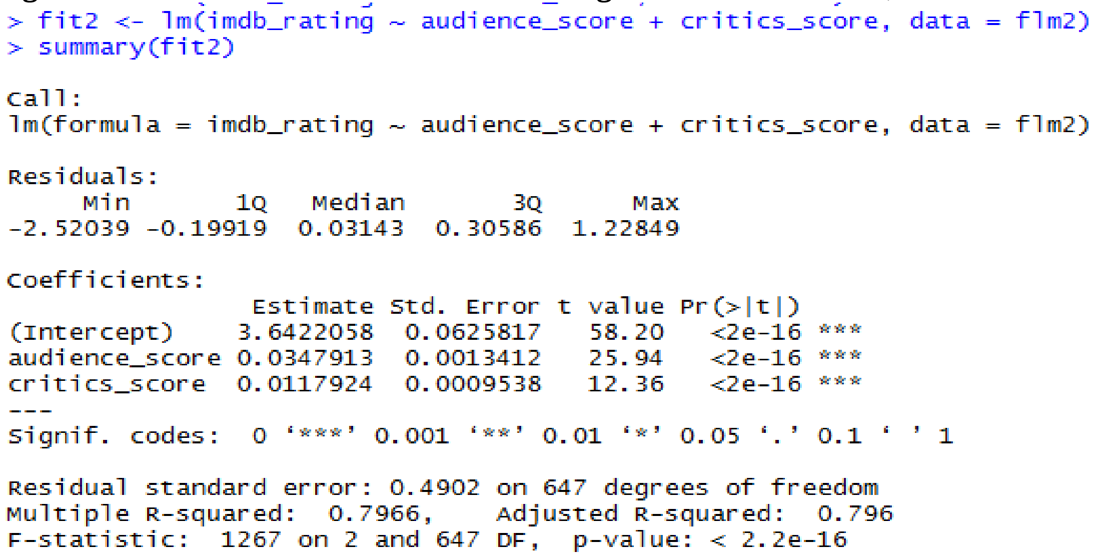
Figure 16:- Correlation matrix plot of selected attributes.



In this figure we are analizing that which attributes plays the most important role in the movie success prediction. Therefore in this graph we analysed that the audience score and critics score are strongly correlated to each other also imdb rating and audience score are strongly correlated to each other. Therefore these four attributes plays the major role in our movie success pridction.]

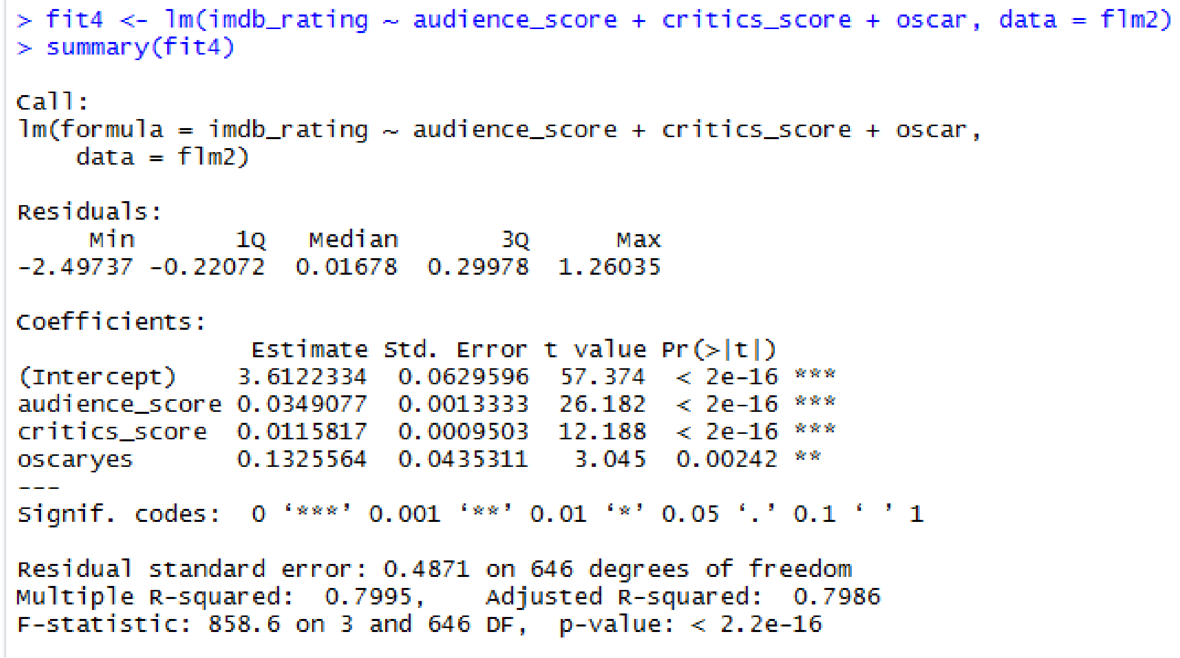
We choose critics_score as our second predictor variable. The model's R-squared increased together with the adjusted R-squared and the p-values indicate that they are both significant predictors. Figure 17 and Figure 18 shown below shows that critics score as our one of the predictor.

Figure 17:- Correlation between imdb_rating and audience_score, critics_score



```
> fit2 <- lm(imdb_rating ~ audience_score + critics_score, data = flm2)
> summary(fit2)

Call:
lm(formula = imdb_rating ~ audience_score + critics_score, data = flm2)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.52039 -0.19919  0.03143  0.30586  1.22849 

Coefficients:
                Estimate Std. Error t value Pr(>|t|)    
(Intercept)    3.6422058  0.0625817   58.20   <2e-16 ***
audience_score 0.0347913  0.0013412   25.94   <2e-16 ***
critics_score  0.0117924  0.0009538   12.36   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4902 on 647 degrees of freedom
Multiple R-squared:  0.7966,	Adjusted R-squared:  0.796 
F-statistic:  1267 on 2 and 647 DF,  p-value: < 2.2e-16
```

Figure 18:- Correlation between imdb_rating and audience_score, critics_score, oscar

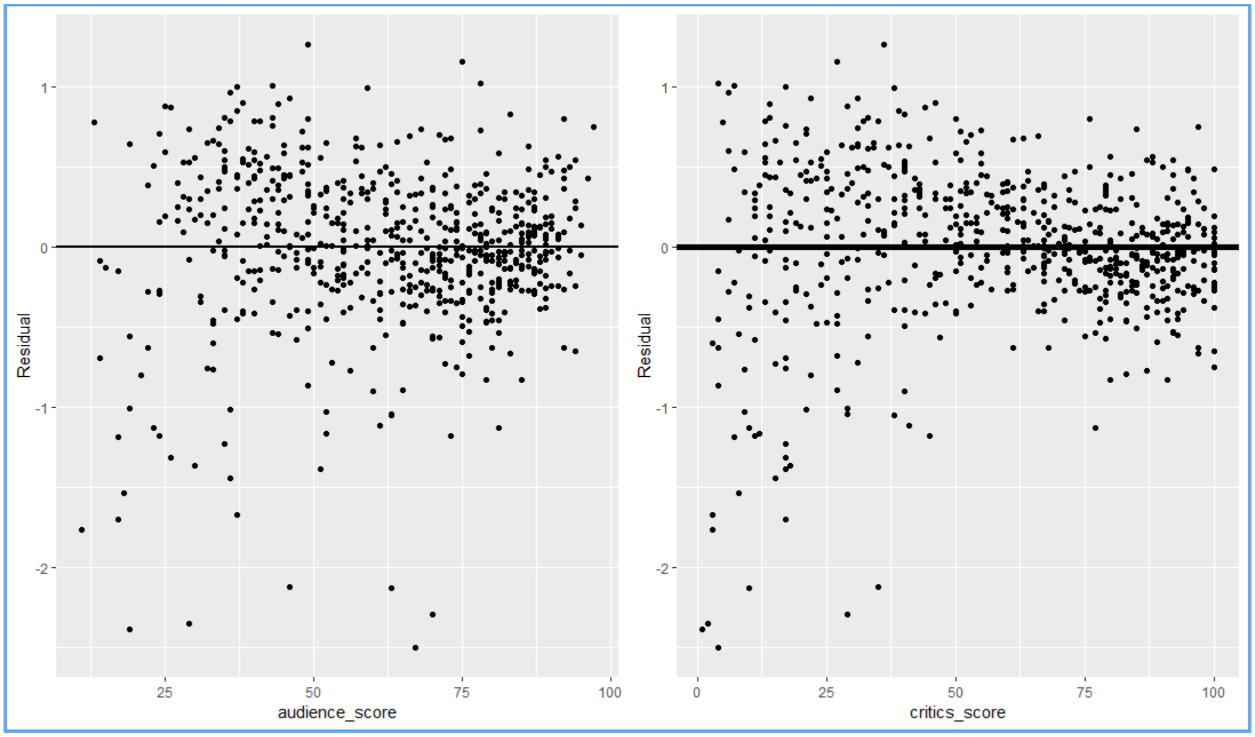


```
> fit4 <- lm(imdb_rating ~ audience_score + critics_score + oscar, data = flm2)
> summary(fit4)

Call:
lm(formula = imdb_rating ~ audience_score + critics_score + oscar, 
    data = flm2)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.49737 -0.22072  0.01678  0.29978  1.26035 

Coefficients:
                Estimate Std. Error t value Pr(>|t|)    
(Intercept)    3.6122334  0.0629596  57.374  < 2e-16 ***
audience_score 0.0349077  0.0013333  26.182  < 2e-16 ***
critics_score  0.0115817  0.0009503  12.188  < 2e-16 ***
oscaryes       0.1325564  0.0435311   3.045  0.00242 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4871 on 646 degrees of freedom
Multiple R-squared:  0.7995,	Adjusted R-squared:  0.7986 
F-statistic: 858.6 on 3 and 646 DF,  p-value: < 2.2e-16
```

A model with audience_score and critics_scoreas a predictor accounts for **79.66 %** of the variation in imdb_rating. Adding the categorical variable improves this to only **79.95 %**. However, I am interested in quantifying the difference between imdb_rating of movies with a director, actor or actress who has won an Oscar award compared to one without an oscar award. These values are correlated, which means I do not need to add more of them as this will not be helpful. I will use only critics score, as IMDB score and audience_score are strongly correlated (both are audience score).So critics score is going to by my explanatory variable.

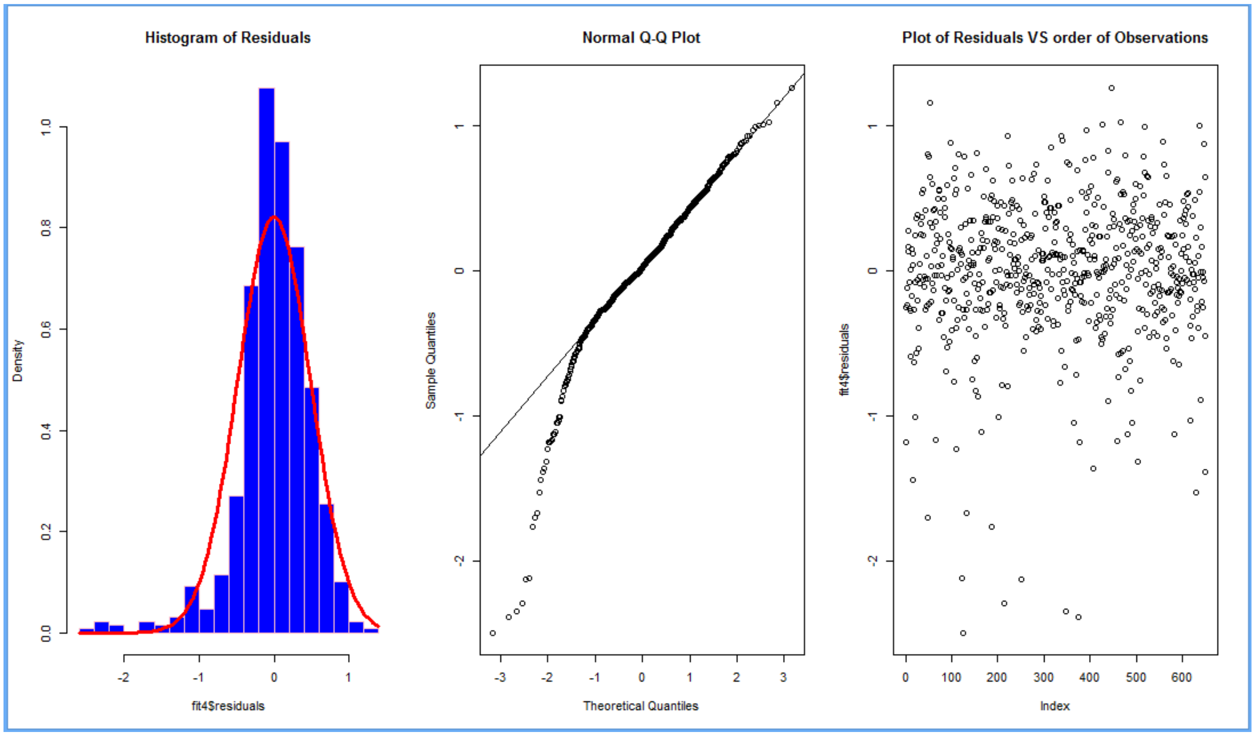
We will now proceed to perform model diagnostics to see if our model meets the requirements for the linear conditions to be valid.

Figure 19:- linear relationships between each predictor and response



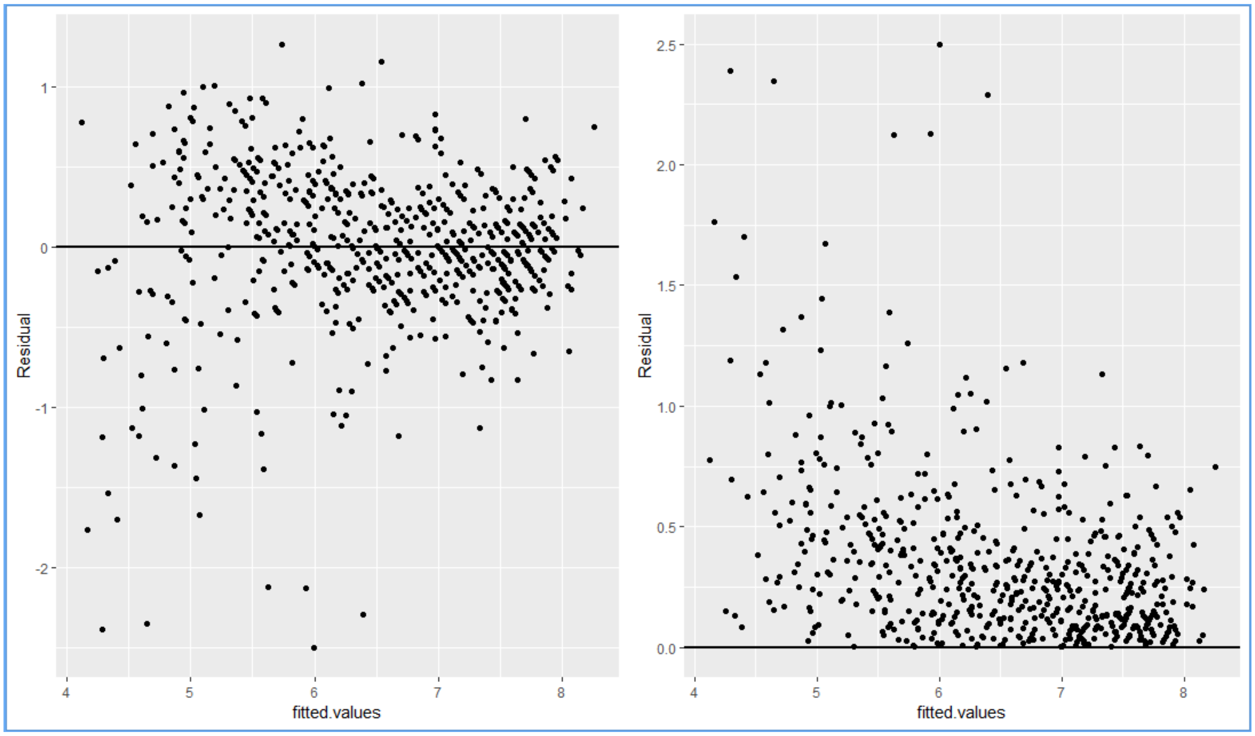
Based on the residual plots above, we can see a linear trend between our residuals and our predictor variables. This condition is met by our model.

Figure 20:- Nearly normal residuals with mean 0 and independent residuals



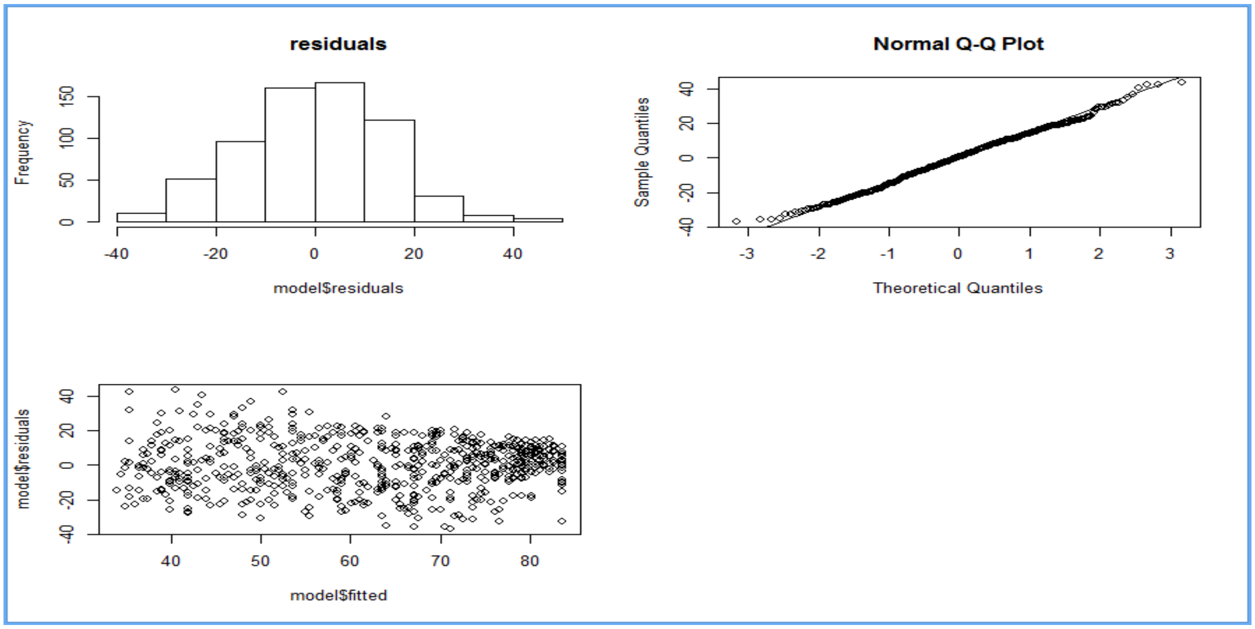
We can see a strong left skew in the histogram of our residuals (left) and the normal probability plots (center) above. Our model fails to meet this requirement for linearity. This means that our model is not very reliable when audience score or critics score is low. The plot of the residuals and the order of observations (right) above reveal a random pattern. The observations appear to be independent of each other. This condition is met.

Figure 21:- constant variability of residuals



The residuals plot versus the fiited values show heteroscedasticity. There is more variance in the observations in the lower end compared to the upper end forming a fan-shaped pattern. This condition is also not met. This means that our model has more variability when predicting low imdb_rating. The predictions are unreliable for imdb_ratings that are low.

Figure 22:-Normal residuals with mean 0 and independent residuals

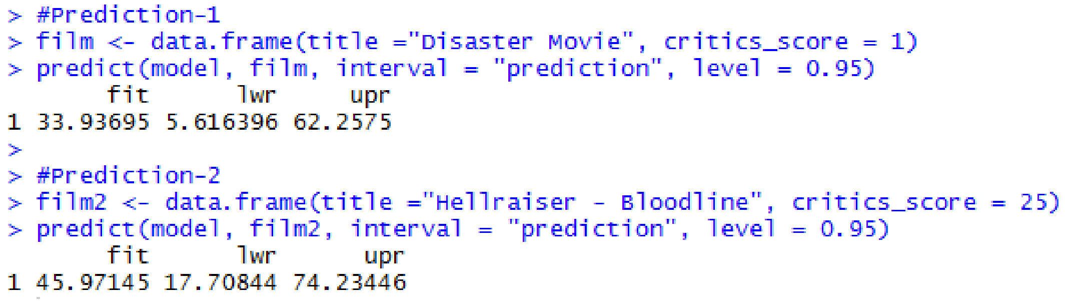


We can see that normaly distributed around zero. Few outliers do not infere with data, therefore we do not care for them. Also there is linear relationships between x and y, we have got nearly normal residuals and we see from constant variability of residuals the independence of residuals. From summary table is clear what we did know - there is strong relationship between critics score and audience score. And we can see that critic score is very significant preedictor for audience score.

Prediction : Prediction in data mining is to identify data points purely on the description of another related data value. It is not necessarily related to future events but the used variables are unknown. By using prediction we can derive the relationship between a thing we know and a thing we want to predict. The prediction in data mining is known as Numeric Prediction. Generally regression analysis is used for prediction.

Here in our project we are using predefined function predict() where we use the critics score as our input to predict the audience score. Here, we give movie name as one of input along with critics score and then it will predict the audience score. If the audience score is nearer to old data then and it fit in the range of lower limit and upper limit of predicted value, then we can say that our movie is successful or hit.

Figure 23:- Prediction of Movie audience score



```
> #Prediction-1
> film <- data.frame(title ="Disaster Movie", critics_score = 1)
> predict(model, film, interval = "prediction", level = 0.95)
       fit      lwr     upr
1 33.93695 5.616396 62.2575
>
> #Prediction-2
> film2 <- data.frame(title ="Hellraiser - Bloodline", critics_score = 25)
> predict(model, film2, interval = "prediction", level = 0.95)
       fit      lwr      upr
1 45.97145 17.70844 74.23446
```

Here, we can see that we have got prediction1 of 33.94 audience score with 95% confidence level that our score will be between 5.62 and 62.26. Well, actual audience score of this movie is 19. We can see that predicted audience score is greater than actual audience score. So, we can say that movie “Disaster Movie” is more successful than its actual performance.

Similarly, we can see that we have got prediction2 of 45.97 audience score with 95% confidence level that our score will be between 17.71 and 74.23. Well, actual audience score of this movie is 47. We can see that predicted audience score is less than actual audience score. So, we can say that movie “Hellraiser -Bloodline” is not a successful movie.

Conclusion: In this project we are just trying to determine if there is any association between different attributes present in our dataset. Here, our main aim is to find association between numeric type attributes that is used s a scoring systems and how we can use this association for prediction. As a result we found that critics score is strongly positive relationship between critics score and audience score. And we can also conclude that critics score are best predictor of audience scores. Thus, we can predicted our movies success on the basis of critics score. In future, we can add many attributes as our predictors and build model for that attributes to perform prediction. Here, we can assume that if we have movie gross score and movie net profit along with movie manufacturing cost, then we can build a more strong model for movie success prediction. In future, we can apply other machine learning algorithms for movie success prediction.

Project R Code:

```
#Packages loading
setwd("F://Rsoftware/New")
getwd()
library(ggplot2)
library(dplyr)
library(caret)
library(statsr)
library(gridExtra)
library(GGally)
library(ggthemes)
library(knitr)

#Loading Data
load("movies.Rdata")
str(movies)
summary(movies)
#Generating Training and Testing Dataset
set.seed(5329)
inTrain <- createDataPartition(y=movies$imdb_rating, p=0.994, list=FALSE)
inTrain
training <- movies[inTrain,]
testing <- movies[-inTrain,]

#Dimention of training and Testing Dataset
dim(training)
dim(testing)

#normal distribution of imdb_num_votes
quantile(training$imdb_num_votes, c(0, 0.25, 0.5, 0.75, 0.9, 1))

#distribution of critics_score and imdb_rating
d1 <- ggplot(data = training, aes(y = imdb_rating, x = critics_score, colour = critics_rating)) +
geom_point()
d2 <- ggplot(data = training, aes(y = imdb_rating, x = critics_score, colour = audience_rating)) +
geom_point()
grid.arrange(d1, d2, nrow = 1, ncol = 2)

#Finding minimum,maximum,mean and median of imdb_rating, imdb_num_votes, critics_score,
audience_score
minv <- training %>% select(imdb_rating, imdb_num_votes, critics_score, audience_score)
%>% sapply(min) %>% sapply(round,2)
maxv <- training %>% select(imdb_rating, imdb_num_votes, critics_score, audience_score)
%>% sapply(max) %>% sapply(round,2)
meanv <- training %>% select(imdb_rating, imdb_num_votes, critics_score, audience_score)
%>% sapply(mean) %>% sapply(round,2)
medianv <- training %>% select(imdb_rating, imdb_num_votes, critics_score, audience_score)
%>% sapply(median) %>% sapply(round,2)
df <- rbind(minv, maxv, meanv, medianv)
rownames(df) <- c("min", "max", "mean", "median")
```

```r
kable(df)

#Representation of imdb_rating,imdb_num_votes,critics_score and audience_score on histogram
p1 <- ggplot(data = training, aes(x = imdb_rating)) + geom_histogram(colour = "black", fill = "skyblue", binwidth = .3)
p2 <- ggplot(data = training, aes(x = imdb_num_votes)) + geom_histogram(colour = "black", fill = "salmon", binwidth = 40000, alpha = 0.5)
p3 <- ggplot(data = training, aes(x = critics_score)) + geom_histogram(colour = "black", fill = "cyan", binwidth = 5, alpha = 0.5)
p4 <- ggplot(data = training, aes(x = audience_score)) + geom_histogram(colour = "black", fill = "yellow", binwidth = 5, alpha = 0.7)
grid.arrange(p1, p2, p3, p4, nrow = 1, ncol = 4)

#Data representation on class basis of best_actor_win, best_actress_win, best_director_win in the form of table
actr <- table(training$best_actor_win)
acts <- table(training$best_actress_win)
dir <- table(training$best_dir_win)
flm2 <- training %>% mutate(oscar = ifelse(best_actor_win == "yes" | best_actress_win == "yes" | best_dir_win == "yes", "yes", "no"))
osc <- flm2 %>% select(oscar) %>% group_by(oscar) %>% table() %>% rbind(actr, acts, dir)
rownames(osc) <- c("At.least.one.Oscar", "best.actor", "best.actress", "best.director")
osc

#Representation of above creates table data on the basis of imdb reting and critics score
oscar_in_cast <- flm2 %>% filter(oscar == "yes") %>% arrange(imdb_rating) %>% select(title) %>% data.frame() %>% head(6)
x1 <- ggplot(data = flm2, aes(y = imdb_rating, x = critics_score, colour = oscar)) + geom_point() + scale_colour_discrete(name="Combined") + scale_fill_hue(name="Combined")
x2 <- ggplot(data = flm2, aes(y = imdb_rating, x = critics_score, colour = best_actor_win)) + geom_point() + scale_colour_discrete(name="Actor") + scale_fill_hue(name="Actor")
x3 <- ggplot(data = flm2, aes(y = imdb_rating, x = critics_score, colour = best_actress_win)) + geom_point() + scale_colour_discrete(name="Actress") + scale_fill_hue(name="Actress")
x4 <- ggplot(data = flm2, aes(y = imdb_rating, x = critics_score, colour = best_dir_win)) + geom_point() + scale_colour_discrete(name="Director") + scale_fill_hue(name="Director")
grid.arrange(x1, x2, x3, x4, nrow = 1, ncol = 4)

#Data representation on class basis of best_pic_win, bes_pic_nom and oscar nomination win in tabular form
nom <- table(flm2$best_pic_nom)
win <- table(flm2$best_pic_win)
flm2 <- flm2 %>% mutate(oscar_nom_win = ifelse(best_pic_nom == "yes" | best_pic_win == "yes", "yes", "no"))
nom_win <- table(flm2$oscar_nom_win)
comb_nom_win <- rbind(nom_win, nom, win)
rownames(comb_nom_win) <- c("combined", "nominations", "wins")
comb_nom_win

#Representation of above creates table data on the basis of imdb reting and critics score
```

```
w1 <- ggplot(data = flm2, aes(y = imdb_rating, x = critics_score, colour = oscar_nom_win)) +
geom_point()
w2 <- ggplot(data = flm2, aes(y = imdb_rating, x = critics_score, colour = best_pic_nom)) +
geom_point()
w3 <- ggplot(data = flm2, aes(y = imdb_rating, x = critics_score, colour = best_pic_win)) +
geom_point()
grid.arrange(w1, w2, w3, nrow = 1, ncol = 3)
outlier_best_pic_nom <- flm2 %>% filter(best_pic_nom == "yes") %>% arrange(imdb_rating)
%>% data.frame() %>% head(1)

#Histogram along with Scatterplot showing theater release day, month and year data
g1 <- ggplot(data = flm2, aes(x = thtr_rel_year)) + geom_histogram(colour = "black", fill =
"orange", alpha = 0.5)
g2 <- ggplot(data = flm2, aes(x = thtr_rel_month)) + geom_histogram(colour = "black", fill =
"blue", alpha = 0.5)
g3 <- ggplot(data = flm2, aes(x = thtr_rel_day)) + geom_histogram(colour = "black", fill =
"green", alpha = 0.5)
g4 <- ggplot(data = flm2, aes(y = imdb_rating, x = critics_score, colour = factor(thtr_rel_year)))
+ geom_point()
g5 <- ggplot(data = flm2, aes(y = imdb_rating, x = critics_score, colour =
factor(thtr_rel_month))) + geom_point()
g6 <- ggplot(data = movies, aes(y = imdb_rating, x = critics_score, colour =
factor(thtr_rel_day))) + geom_point()
grid.arrange(g1, g2, g3, g4, g5, g6, nrow = 2, ncol = 3)

#Histogram along with Scatterplot showing dvd release day, month and year data
g7 <- ggplot(data = flm2, aes(x = dvd_rel_year)) + geom_histogram(colour = "black", fill =
"orange", alpha = 0.5)
g8 <- ggplot(data = flm2, aes(x = dvd_rel_month)) + geom_histogram(colour = "black", fill =
"blue", alpha = 0.5)
g9 <- ggplot(data = flm2, aes(x = dvd_rel_day)) + geom_histogram(colour = "black", fill =
"green", alpha = 0.5)
g10 <-ggplot(data = flm2, aes(y = imdb_rating, x = critics_score, colour = factor(dvd_rel_year)))
+ geom_point()
g11 <- ggplot(data = movies, aes(y = imdb_rating, x = critics_score, colour =
factor(dvd_rel_month))) + geom_point()
g12 <- ggplot(data = flm2, aes(y = imdb_rating, x = critics_score, colour = factor(dvd_rel_day)))
+ geom_point()
grid.arrange(g7, g8, g9, g10, g11, g12 ,nrow = 2, ncol = 3)

#Building Correlation plot
num_var <- flm2 %>% select(imdb_rating,imdb_num_votes, critics_score,audience_score)
ggcorr(num_var, name = "Correlation", label = TRUE, alpha = TRUE, palette = "PuOr") +
ggtitle("correlation matrix plot") + theme_dark()

#Finding best fitted attribute for model development and prediction
fit1 <- lm(imdb_rating ~ audience_score, data = flm2)
fit2 <- lm(imdb_rating ~ audience_score + critics_score, data = flm2)
summary(fit2)
fit3 <- lm(imdb_rating ~ audience_score + critics_score + imdb_num_votes, data = flm2)
```

```
fit4 <- lm(imdb_rating ~ audience_score + critics_score + oscar, data = flm2)
summary(fit4)

#Ploting residual data
t1 <-  ggplot(data = flm2, aes(x = audience_score, y = resid(fit4))) + geom_hline(yintercept = 0,
size = 1)  + xlab("audience_score") + ylab("Residual") + geom_point()
t2 <-  ggplot(data = flm2, aes(x = critics_score, y = resid(fit4))) + geom_hline(yintercept = 0,
size = 2)  + xlab("critics_score") + ylab("Residual") + geom_point()
grid.arrange(t1, t2, nrow = 1, ncol = 2)

#Ploting residual data for analsys
par(mfrow = c(1,3))
hist(fit4$residuals, breaks = 25, main = "Histogram of Residuals", col = "blue", border = "pink",
prob = TRUE)
curve(dnorm(x, mean = mean(fit4$residuals), sd = sd(fit4$residuals)), col="red", add=T, lwd =
3)
qqnorm(fit4$residuals)
qqline(fit4$residuals)
plot(fit4$residuals, main = "Plot of Residuals VS order of Observations")

#Ploting graph on fit4 data
t3 <- ggplot(data.frame(x = fit4$fitted.values, y = resid(fit4)), aes(x=x, y=y)) +
geom_hline(yintercept = 0, size = 1)  + xlab("fitted.values") + ylab("Residual") + geom_point()
t4 <- ggplot(data.frame(x =fit4$fitted.values, y = abs(resid(fit4))), aes(x=x, y=y)) +
geom_hline(yintercept = 0, size = 1)  + xlab("fitted.values") + ylab("Residual") + geom_point()
grid.arrange(t3, t4, nrow = 1, ncol = 2)

#Box plot model for Genere wise movie runtime in minutes
p_genrerun <- ggplot(movies, aes(x=factor(genre), y=runtime)) +
  geom_boxplot() + theme(axis.text.x = element_text(angle = 45, hjust = 1))
p_genrerun + ggtitle("Genre to runtime") + geom_hline(yintercept =median(movies$runtime,
na.rm = TRUE), col = "royalblue",lwd = 1)

#Box plot model for Genere wise IMDB_rating
p_genreimdb <- ggplot(movies, aes(x=factor(genre), y=imdb_rating)) +
  geom_boxplot() + theme(axis.text.x = element_text(angle = 45, hjust = 1))
p_genreimdb + ggtitle("Genre to IMDB rating") + geom_hline(yintercept
=median(movies$imdb_rating, na.rm = TRUE), col = "royalblue",lwd = 1)

#Scatter plot of imdb_rating and runtime in minutes
plot(movies$imdb_rating,movies$runtime, main="IMDB rating to runtime", xlab = "IMDB
rating", ylab="Runtime in minutes")
abline(lm(movies$runtime~movies$imdb_rating),col = "royalblue",lwd = 1)

#Scatter plot of audience score and critics score
ggplot(data = movies, aes(x = critics_score, y = audience_score)) +
  geom_jitter() +  geom_smooth(method = "lm", se = FALSE) + ggtitle("Critics score to
audience score - Rotten")

#Scatter plot of imdb_rating and audience score
```

```
ggplot(data = movies, aes(x = imdb_rating, y = audience_score)) +
  geom_jitter() +  geom_smooth(method = "lm", se = FALSE) + ggtitle("Critics score to
audience score - IMDB")

#Scatter plot of imdb_rating and critics score
ggplot(data = movies, aes(x = critics_score, y = imdb_rating)) +
  geom_jitter() +  geom_smooth(method = "lm", se = FALSE) + ggtitle("IMDB vs. Rotten")

#regression analysis of audience_score with imdb_rating and critics_score
regres <- lm(movies$audience_score ~ movies$critics_score + movies$imdb_rating)
summary(regres)

#regression analysis of critics_score and imdb_rating
reg <- lm( movies$critics_score ~ movies$imdb_rating)
summary(reg)

#Model development for final prediction
model <- lm(audience_score ~ critics_score, data=movies) #funny thing, if you will not use
"data=" you will not be able to get predict() to work properly
par(mfrow=c(2,2)) #combine plots to 2x2 table
hist(model$residuals, main="residuals")
qqnorm(model$residuals)
qqline(model$residuals)
plot(model$residuals ~ model$fitted)
summary(model)

#Prediction-1
film <- data.frame(title ="Disaster Movie", critics_score = 1)
predict(model, film, interval = "prediction", level = 0.95)

#Prediction-2
film2 <- data.frame(title ="Hellraiser - Bloodline", critics_score = 25)
predict(model, film2, interval = "prediction", level = 0.95)
```

References:

[1] Krushikanth R. Apala ; Merin Jose ; Supreme Motnam ; C.-C. Chan ; Kathy J. Liszka ; Federico de Gregorio” Prediction of Movies Box Office Performance Using Social Media”, 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining(ASONAM 2013), PP 1209- 1214.

[2] Javaria Ahmad ; Prakash Duraisamy ; Amr Yousef ; Bill Buckles” Movie Success Prediction Using Data Mining”, 2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT), PP 1- 4.

[3]Susmita S. Magdum ; J. V. Megha” *Mining Online Reviews and Tweets for Predicting Sales Performance and Success of Movies*”, 2017 International Conference on Intelligent Computing and Control Systems (ICICCS)PP 334-339.

[4] Md Shamsur Rahim ; A Z M Ehtesham Chowdhury ; Md. Asiful Islam ; Mir Riyanul Islam” Mining Trailers Data from YouTube for Predicting Gross Income of Movies”,2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC),PP 551-554.