RUTGERS, THE STATE UNIVERSITY OF NEW JERSEY

FUNDAMENTALS OF ANALYTICS AND DISCOVERY INFORMATICS - PROJECT REPORT

Prediction of the expected sports entertainment level of an international T20 cricket match

Vikhyat Dhamija
RU ID: 194003013

Table of Contents

Introduction

As we all know, sports are a huge form of entertainment for people around the world. Sports can take any name like football, cricket , baseball , basketball but the craziness they generate for their lovers is constant.

In present times, there are many sports fans who love watching their beloved sports in their comfortable recliner or sofa. But there are many other crazy sports lovers who want to see the game live because they like to see the crowd and produce the symphony of scream with them, they want to see how the events are organized, talk to other fans, visit new places, hear the crunch of collisions, watch the warmup man do his work and so much more. But back in their mind, they know that tickets are very expensive, there are other expenditure related to travel to see the live matches. Further, they are spending the most precious thing ***"Their Time".***

Consider a case where there is a company or an organization who wants to be associated with highly entertaining matches, to increase its popularity and hence increase its revenue. But it has limited marketing budget to spend, which it wants to spend wisely and judiciously.

Suppose they have a service or program in hand that based on the parameters of match like the Playing teams, Ground, Geographical information etc. tell them about the *"Expected Sports Entertainment Level"* for that match based on the historical data. This will help the viewers in making a conclusion that whether it is a good decision to spend so many resources and go to the stadium to watch the live match or just sit and relax on recliner or sofa or even do not watch the match. This will also help the companies or organizations to make judicious decision to associate with particular match.

So, the purpose of this project is to build a machine learning model to forecast or predict the expected entertainment level of the match in order to help them make a right decision.

I chose 20-20 format of the international cricket as sport whose entertainment level or value will be predicted. It is a wondering fact that Cricket is the second most popular sports in the world with the fan following of about 2.5 Billion. In Cricket, Twenty Twenty format is becoming the most popular because of its high entertainment quotient and its shorter duration.

"Twenty Twenty format is a shorter version of the cricket which is typically completed in about three hours where each innings lasting around 75–90 minutes and a 10–20-minute interval. Each innings is played over 20 overs and each team has 11 players. This is much shorter than previously existing forms of the game and is closer to the timespan of other popular team sports. It was introduced to create a fast-paced form of the game, which would be attractive to spectators at the ground and viewers on television." [1]

Motivation

This is the line from the article reported by the reputed news agency Bloomberg:

"The television advertisement rates for the ICC Cricket World Cup 2019 have jumped to their highest, beating what Star India charged for Indian Premier League. The ad rates for India games range from INR 15-16 lakh (22K USD) for 10 seconds, two people in media buying agencies said.

And for the much-awaited India-Pakistan match on June 16, a slot costs up to Rs 20 lakh (30K USD) for 10 seconds, the people said."[11]

This is the data for the latest ticket prices in England, the country which also has great fan following for the cricket matches:

"The cost of seeing every home match across all three formats (in some cases a separate pass is needed for the T20 Blast):" [12]

County	Price
Sussex	£304 (early bird £280)
Somerset	£279
Essex	£277
Northamptonshire	£265
Kent	£250
Warwickshire	£245 (early bird £220)
Middlesex	£245
Hampshire	£240
Yorkshire	£230
Worcestershire	£225
Gloucestershire	£216
Durham	£210
Surrey	£203
Glamorgan	£200 (early bird £150)
Lancashire	£195
Derbyshire	£189 (early bird £169)
Nottinghamshire	£180
Leicestershire	£149

Source: The Guardian Image 1: Ticket Prices in England for cricket matches

"Rubeena Singh, CEO, iProspect India shared her views and provided a breakdown of expenditure on advertisement in digital media and TV during this IPL season 2020. According to the breakdown provided by her, there will be an overall amount spent on the advertisement will be 3,000 – 3,300 crores."

TV – 2,300 – 2,500 crores.

Digital – 800 – 1000 crore."[13]

All above articles are just to amaze you with the fact that how much is at a stake for the viewers and the advertisers and others who spend their money for the sake of entertainment. Apart from that there may be various betting agencies staking their money on the matches. I, myself, am a great cricket lover. When I used to see the cheer leaders cheering for the sixes and fours in a T20 match, then I used to get amazed by the sheer amount of money involved in these matches that is totally dependent on this blitzkrieg. People buy so much expensive tickets to come to the ground to see this shorter format T20 Matches and the advertisers and other invest their bucks

on those matches just to associate themselves with these entertaining matches, which are not just viewed live but will be seen by cricket lovers in the form of highlights and will be repeated again and again in the news channels because of their sheer entertainment.

Literature Review

In preparing this study, a literature search was completed. Following papers have been studied which are related to the Cricket Match Predictions using Machine Learning. (Note Links have been provided in the reference section)

1. Predicting Outcome of Indian Premier League (IPL) Matches Using Machine Learning
2. Prediction of the outcome of a Twenty-20 Cricket Match
3. Sport analytics for cricket game results using machine learning: An experimental study
4. Quantitative Analysis of Forthcoming ICC Men's T20 World Cup 2020 Winner Prediction using Machine Learning.
5. Dynamic Winner Prediction in Twenty20 Cricket: Based on Relative Team Strengths
6. Cricket Match Outcome Prediction Using Machine Learning

The paper 1 was focused on developing the model for the prediction of the outcome of Twenty 20 cricket matches played in the Indian Premier League. In this study, the players performance was the main parameter chosen for making the decision regarding the win or lose of the team. In this study , based on the individual performance, total team performance was judged. In this study, the multivariate regression model was developed and was used to calculate the points of player. Player Points section of the ICC gives the player certain points based on the 6 factors like number of wickets taken, number of dot balls given, number of fours, number of sixes, number of catches, and number of stumpings. So based on these scores the multivariate regression model was developed to calculate the players points based on these 6 parameters. This was a great thing seen in practical where regression model was used to come up with the features. Then based on the individual's player points, the total weight of the team was calculated. As each team had 25 players out of which the 11 players played in a particular match, then in order to calculate the weight of the team , the players were sorted in descending order based on the frequency of the players played for their team. So top 11 most frequently played players of the team were used in the calculation of total teams' weight. The formula used was the (Summation of the scores of 11 players)/(Number of appearances of a team in a season). So ideally the weight of the team has to calculated incrementally after each match of the team but in the study the data till the final of each season was considered and used to calculate the team weight and then the same score was used for all matches. So, the scores were static for particular season. In the study, in the dataset preparation, Recursive feature Elimination method was used which recursively removes the features and builds the model using the remaining feature and then recalculates its accuracy. So, based on RFE method , the 7 most useful features were detected. In the study , the technique of 10-fold cross validation was used to calculate the accuracy of the model. The models were created using various algorithms like Naïve Bayes , Extreme Gradient Boosting , SVM and Logistic Regression, Random Forests and Multilayer Perceptron were used, and it was found out that MLP proved to be more accurate with around 70 percent accuracy and F1 score of 0.72. *This study made me understand how to approach the cricket predictions related problems , how the multivariate regression model can be used to calculate the features to be fed to the model,*

how the Recursive feature Elimination Techniques work to select the important features to be fed to the model.

The paper 2 is also an interesting paper focused on building model on the prediction outcome of Twenty 20 match. The interesting part in the study was the feature extraction and the various approaches used in creating features set for the model creation. This has been a great learning through this paper. The study used the web crawler built using python and BeautifulSoup Library. In its feature extraction process, the parameters were selected that have been used by the cricket experts. For each instance corresponding to each match, the average of various parameters for players were averaged till the Date of the match. So, the features corresponding to each player were Average Runs Continuous Average Number of 4s Continuous Average Number of 6s Continuous Average Strike Rate Continuous etc. There were multiple approaches used for the prediction problem. In approach 1 there were variations used. In variation 1, 16 * 22 = 352 features were used per instance (11 per team and as two team played a match hence 22) i.e., the match in order to predict the outcome of the match. In variation 2, it aggregated each player's features to arrive at two values i.e., the Batting Aggregate and Bowling Aggregate. So, in this case each training instance had 44 features instead of 352 features. In Variation 3, further enhancement was done over approach 2 and each player's feature i.e., the batting aggregate and bowling aggregate were converted into one value per player. So, in this variation each training instance would have 22 features. In this Variation further enhancement was done and each team's normalized player features were made to arrive at one value per team. Hence so many variations were performed in producing the feature set for the ML Model Building. In the approach 2 some innovative steps were taken to improve the accuracy of the model like the **Time Scaling** where performances of the players were scaled according to time with a weight derived from the mathematical formula = (1+ t-t min)/(1+t max – t min) where t is the time of a particular match under consideration and the and the t min is the oldest match in the dataset and t max is the latest match in the dataset , Different methodology to calculate the Batting and the Bowling Score because generally we use the aggregated statistics of the players but we do not consider the strength of opponent as it can be possible that players has good stats because it had lighter opponents. In third approach, two separate K means clustering systems were used to generate the batting and bowling score. The k-means clustering system for batting was based on the batting features of player and similar for the bowling. Various ML algorithms were used like SVM , Decision Tree , Naïve bayes etc. 10-Fold cross validation was used for the evaluation of the various ML Models. *This study made me understand various approached to feature extraction and transformation for building the good Machine learning models.*

The paper 3 is all about using machine learning to predict the Cricket Game Results. This paper focused on different strategy to come out with the best model for predicting Cricket Game Results. Here the two features set were used to predict the outcome of the match. First feature set was based on the features related to Home Team and another feature set was based on the Toss decision. The influential features had been identified using Correlation based feature selection, Information Gain, ReliefF, Wrapper etc. techniques. In this study, all experiments were performed on WEKA (The Waikato Environment for Knowledge Analysis). Various machine learning algorithms like Naive Bayes, Random Forest, K-nearest neighbor, and Model Decision Tree were then applied to the two feature sets to derive predictive models for the match result. The evaluation of the models was performed using the 10-fold cross validation in order to compare the models based on the average predictive accuracy using 10 folds cross validation. Further the recall and precision were also used to evaluate the Models. This study made me understand that

how we can apply various techniques for feature selection, how we can use different feature sets that are based on completely different approaches to prediction (like in this study we are using the home team related features and the toss decision based features as both home team and conditions also affect the result of the match and toss where the toss winning team can make decision either to bat first or ball first based on various other factors also affect the match decision) and create intelligent approach based on them and further how we can utilize the evaluation metrics like precision, recall and average predictive accuracy to compare various models.

Approach

In order to solve the problem in hand, Supervised Learning is being used. In order to solve the problem of supervised learning, the label(class) to be assigned to the instances of dataset has to be decided first. Then, based on our study, various attributes/features of instances of dataset are decided which enable us to build a good prediction model. As we have a multi class classification problem in hand, various machine learning algorithms are determined to be used to build a predictive model. And then the evaluation metrics (described in next section) are used to evaluate various models generated. So, following is the flow to solve the machine learning problem in hand.

A. Concept to be learned – its description.

As described, the concept to be learned in this project is the *expected sports entertainment Level of a cricket match.* In the dataset to be used, these classification labels will be assigned to the instances which will be the entertainment level. In a twenty 20 cricket matches, viewers are excited to view the runs scored in 4's and 6's as twenty 20 international twenty 20 cricket is all about exciting batting.

Considering this, the entertainment level will be generated by the mathematical formula:

= (Total runs scored in 4's and 6's in both innings / Total runs scored in both innings) * 100

So, whether runs scored are less or more, the excitement is more about batting blitzkrieg (4's and 6's) in the match.

This percentage was calculated. Then, the threshold is calculated based on the average value of the percentage of 4's and 6's in both innings for all the instances i.e., the matches.

That came out to be:

```
0.555021889814801
```

As a result, the labels were assigned to the instances based on the following criteria. The following function was used in order to assign the labels:

```
def label_function(x):
    if x > 0.60:
        return "H"
    elif x < 0.55:
        return "L"
    else:
        return "A"

merged_df["Elevel"] = merged_df["Elevelval"].apply(label_function)
```

Here the 55 % is considered the threshold value based on which we assign the labels or class. We have identified three classes here – High(H), Average(A), Low(L) where H > 60 % , L < 55 % , A > 55 % and A < 60 %

These labels will be the prediction values. Based on the features, these labels will be predicted hence giving users the prediction or forecast that how the cricket match will be, enabling them to decide whether they want to view the match or not.

B. Attributes

The attributes used are as follows:

S.No.	Attributes	Description
1	Team1	*Team 1* are the countries who plays the international T20 cricket. Here the current Top 10 Teams in ICC Twenty 20 Cricket ratings are considered, whose matches arise profound interest and viewership. These are : England, India, Australia, Pakistan, New Zealand, South Africa, Bangladesh, Afghanistan, West Indies, Sri Lanka
2	Team2	*Team 2* are the countries who plays the international T20 cricket. Here the current Top 10 Teams in ICC Twenty 20 Cricket ratings are considered, whose matches arise profound interest and viewership. These are : England, India, Australia, Pakistan, New Zealand, South Africa, Bangladesh, Afghanistan, West Indies, Sri Lanka
3	MatchTime	This will have three values : 1. day 2. Night 3. daynight
4	Ha1	This attribute will have value 1/0 depending on the fact that Team 1 has the home advantage or not.
5	Ha2	This attribute will have value 1/0 depending on the fact that Team 2 has the home advantage or not.
6	T1w	This attribute will have value 1/0 depending on the fact that Team 1 has won the toss or not.
7	T2w	This attribute will have value 1/0 depending on the fact that Team 2 has won the toss or not.

8	T1b	This attribute will have value 1/0 depending on the fact that Team 1 bat first or not.
9	Groundcapacity	This will have the values of the various cricket grounds seating capacities as it directly relates to the viewership of the match which may be a variable on which the performance of the players can be dependent as more viewers means more noise which may positively affect the results increasing the motivation of the players or they can increase the noise so as to create disturbance. So, it also forms an important feature to decide the performances of teams in the match.
10	AvgSR	This is the average batting strike rate of the top 5-6 players in the team line up. As the top 5-6 players in a team line up are the top batsmen of the team, they are mostly responsible for most of the total score and even the major proportion of 4's and 6's.
11	CRating	This will have the sum of the numbers assigned based on the ranks(ratings) of the Team1 and Team2 in the year when match was played as higher the rank of the two teams more entertaining the match will be.
12	Elevel	This is the class label for the Entertainment Level that has three values – High(H) , Average(A) and Low(L)

C. Data Collection (Obtain and clean Data)

Data was obtained using the web scrapers built using Python and its BeautifulSoup package. ESPN Cricinfo Website has all the statistics related to all the international Matches in various formats like One Day, Test and T20 formats. Python script was built where the different arguments like year and team code were passed to the URL, to scrawl the year and team wise match records and then the URL of each match record was extracted so as to move to access that URL in order to extract the relevant attributes of that particular match.

Sample URL of the particular match:

https://stats.espncricinfo.com/ci/engine/records/team/match_results.html?class=3;id=2013;team=1;type=year

Generic URL of the match to be used by passing various year and team codes as an arguments :

https://stats.espncricinfo.com/ci/engine/records/team/match_results.html?class=3;id='+str(year)+';team='+str(team)+';type=year

Match results

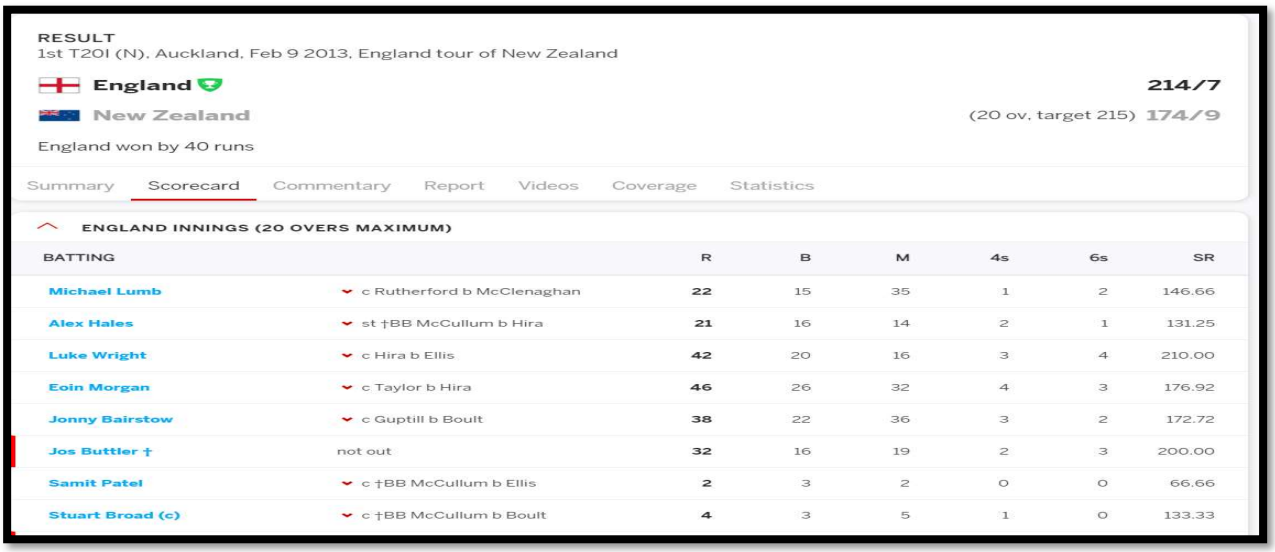
Team 1	Team 2	Winner	Margin	Ground	Match Date	Scorecard
New Zealand	England	England	40 runs	Auckland	Feb 9, 2013	T20I # 301
New Zealand	England	New Zealand	55 runs	Hamilton	Feb 12, 2013	T20I # 302
New Zealand	England	England	10 wickets	Wellington	Feb 15, 2013	T20I # 304
England	New Zealand	New Zealand	5 runs	The Oval	Jun 25, 2013	T20I # 317
England	New Zealand	no result		The Oval	Jun 27, 2013	T20I # 318
England	Australia	Australia	39 runs	Southampton	Aug 29, 2013	T20I # 328
England	Australia	England	27 runs	Chester-le-Street	Aug 31, 2013	T20I # 329

Here the ScoreCard URL is provided with each match record which was extracted, and that URL was accessed in order to extract the relevant information related to the match.

In the Score card page also, we have to scrape through the various URLs in order to access the relevant information. For example, Ground URL was accessed to access the ground information. Even each player URL was accessed in order to collect the batting strike rate of the player in that particular year in T20 format . Then their average was taken to come out with the Average Strike Rate of the two teams playing the match which directly relates to the Batting performance of the two teams playing the match and in turn the amount of blitzkrieg in the form of 6's and 4's in the short format of the match.

For example:

"https://stats.espncricinfo.com" + match.group(1)" here the match.group(1) is the portion of the URL of the scorecard of each match, extracted as described above.



RESULT
1st T20I (N), Auckland, Feb 9 2013, England tour of New Zealand

England 214/7
New Zealand (20 ov, target 215) 174/9
England won by 40 runs

Summary | Scorecard | Commentary | Report | Videos | Coverage | Statistics

ENGLAND INNINGS (20 OVERS MAXIMUM)

BATTING		R	B	M	4s	6s	SR
Michael Lumb	c Rutherford b McClenaghan	22	15	35	1	2	146.66
Alex Hales	st †BB McCullum b Hira	21	16	14	2	1	131.25
Luke Wright	c Hira b Ellis	42	20	16	3	4	210.00
Eoin Morgan	c Taylor b Hira	46	26	32	4	3	176.92
Jonny Bairstow	c Guptill b Boult	38	22	36	3	2	172.72
Jos Buttler †	not out	32	16	19	2	3	200.00
Samit Patel	c †BB McCullum b Ellis	2	3	2	0	0	66.66
Stuart Broad (c)	c †BB McCullum b Boult	4	3	5	1	0	133.33

Likewise other relevant URLs like the one for the ground related information, was extracted in order to collect the relevant attribute related to the ground where match was played.[The dataset obtained, and the python script has been attached with this report.]

Other python script was built to extract the other relevant parameters like ICC (International Cricket Council) ratings for the particular team in the particular year in the T20 format from the Wikipedia. Wikipedia has a web page related to all cricket information related to the particular year in the International Cricket.

"https://en.wikipedia.org/wiki/International_cricket_in_"+str(year)

This URL was used with the year passed as the parameter in order to scrawl to the Wikipedia webpage related to the ICC International Cricket information in that particular year. This led us to create the Ratings CSV file in order to merge with the main Dataset created as described above.

These datasets were merged using the left outer join procedure using the merge function available in the python pandas. (Note all these steps are available in the Jupyter Notebook attached with the report)

Following steps were taken for cleaning the Merged Dataset:

1. Duplicates instances were removed.
2. Ground Capacity which was Null was replaced with mean capacity of 20000 which had already been done during dataset creation using python web scraping script which was used for data collection.
3. Missing Ratings values were imputed using the average rating value of 128. It is to be noted that ICC had started giving the ICC T20 cricket ratings from 2012 onwards so in order to create the instances for the years starting from the year 2007, the ratings related to the ODI format i.e., the 50 overs format were selected while extracting data from the Wikipedia as it was the fastest format at that time.
4. As described and presented in the code snippet below, the formula of :
   (No. of Fours * 4 + No. Of Sixes * 6)/ Total Runs, was used in order to calculate the Entertainment label.

```
#Producing the Labels and summing up the ratings
merged_df["Elevelval"]=( merged_df["Fours"]*4 + merged_df["Sixes"]*6)/merged_df["Runs"]

#Here imputing the missing values with the average rating across all years
merged_df['Rating1'] = merged_df['Rating1'].fillna(128)
merged_df['Rating2'] = merged_df['Rating2'].fillna(128)

#Combined Ratings adding the ratings of both team
merged_df["CRating"]=merged_df["Rating1"] + merged_df["Rating2"]

#Dropping the ratings of team1 and team2
merged_df=merged_df.drop(["Rating1","Rating2"], axis = 1)
```

5. One hot encoding was performed on the categorical variables like the Team1 and Team2 in order to convert the categorical data to the numerical data to be used for the machine learning algorithm.
6. SMOTE technique was used to create a more balanced dataset. (Description of SMOTE is present in the Discussion section)
7. Correlation was checked between the various features in order to remove the highly correlated features.

```
merged_df=merged_df.drop(["Runs","Fours","Sixes"], axis = 1)
merged_df.corr()
```

	Ha1	Ha2	T1w	T2w	T1b	Groundcapacity	AvgSR	CRating
Ha1	1.00	-0.40	-0.00	0.00	0.05	0.08	0.08	0.23
Ha2	-0.40	1.00	0.07	-0.07	0.02	-0.09	-0.07	-0.08
T1w	-0.00	0.07	1.00	-1.00	-0.11	0.03	0.02	-0.00
T2w	0.00	-0.07	-1.00	1.00	0.11	-0.03	-0.02	0.00
T1b	0.05	0.02	-0.11	0.11	1.00	0.04	0.05	0.09
Groundcapacity	0.08	-0.09	0.03	-0.03	0.04	1.00	-0.00	0.02
AvgSR	0.08	-0.07	0.02	-0.02	0.05	-0.00	1.00	0.15
CRating	0.23	-0.08	-0.00	0.00	0.09	0.02	0.15	1.00

As it can be seen that there is no as such correlation, positive or negative, between the various features used.

8. Normalization of the numerical features like AvgSR, Groundcapacity was performed for the Logistic Regression Classifier as it best works with Normalized Data otherwise the heavy dimension or independent variable would have influenced the model.

D. Machine Learning Algorithms (Use of Algorithms)

The following five machine learning algorithms were used to build model for this supervised classification problem.

1. Logistic Regression
2. Decision Trees
3. Naïve Bayes
4. Support Vector Machines
5. Random Forest (Ensemble Learning from group of Decision Trees)

OneVsRest classification, heuristic methodology was used for converting the binary classification algorithms to be used for multi class classification. With this approach, the multiclass classification problem is converted into various binary classification problems and the most confident result from these classification problems was chosen.

K-Fold Cross Validation(7-fold) was used to come out with the average metrics of the model derived from the above algorithms and then these models were compared based on the various Evaluation metrics. Cross validation is a good technique to compare the models neglecting the overfitting scenarios.

E. Evaluation

In this project, the problem of multi class classification is being dealt. So, in order to evaluate the model for classification following metrics will be calculated.

1. Accuracy: This is the measure of percentage of True Positives or correct class predictions in the total dataset. So, this parameter simply shows that for how many instances the correct output was predicted out of the total dataset instances.

2. Macro averaged precision: The precision for each class is given by :
   True Positives/(True Positives + False Positives)
   So, it conveys that for each class, the total number of correct predictions of that class out of total predictions of that class. Then average is taken for the precision of all the classes.

3. Macro averaged Recall: The recall for each class is given by :
   True Positives /(True Positives + False Negatives)
   So, it conveys that for each class the total number of correct predictions of that class out of total actual values for that class. Then average is taken for the recall of all the classes.

4. F1 Score: This is the harmonic mean of the Macro averaged Recall and Macro averaged Precision.

In case of classification problem, depending on the dataset in hand whether it is balanced having instances almost equally distributed for each class or it is unbalanced having instances of certain classes more than other classes, the accuracy or the F1 score takes the prominent role. As the dataset has been made synthetically balanced by the SMOTE technique, still F1 Score is important, providing the balance of both precision and recall. Note that in order to check whether our model is performing well, the model was compared with the baseline ZeroR model which just predict the majority class and then its evaluation metrics were compared to the other models being generated.

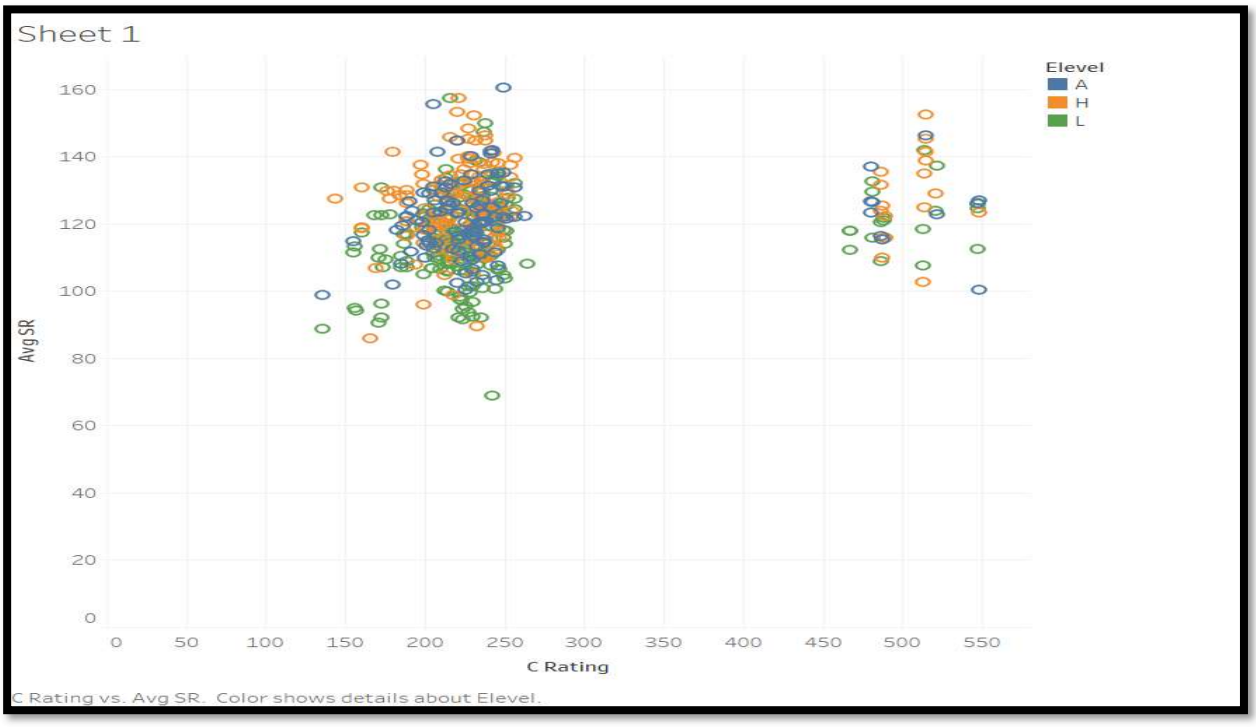
Results

1. Graphs

a. Exploratory Data Analysis using Tableau.

EDA helps us to peek into the data and help us better understand the patterns, relationships among the variables and any outliers in the data points. Most often visualization techniques are used for the exploratory data analysis. When EDA is done, it leads to insights and hence make us decide the features that can be used for sophisticated modelling through machine learning. That is why EDA through Visualization was performed and various observations were made.

A. *Scatter Plot with the axes AvgSR i.e., Average Batting Strike Rate and the Combine team Rating of the Two teams in a particular year and the corresponding Entertainment which seems to the most appropriate independent variables affecting the predicted class.*

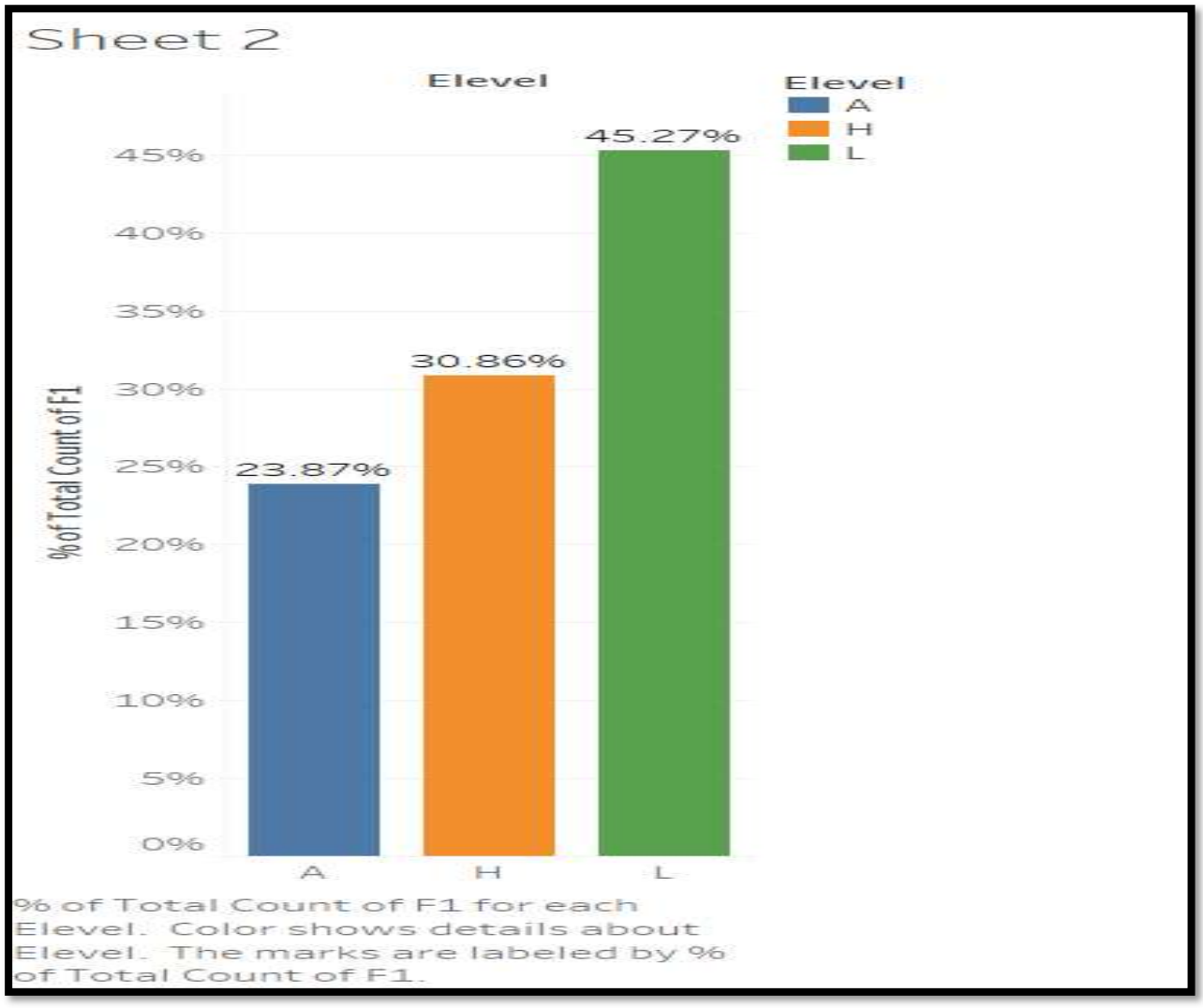


Above Scatter Plot shows that how the green points and the Blue points are clustered in the particular territory or the range of AvgSR and the CRating. So, they seem to be separable, but the High Entertainment Level (H) does not seem to have defined territory in this range.

B. *Checking the Balanced or the Unbalanced Nature of the Dataset*

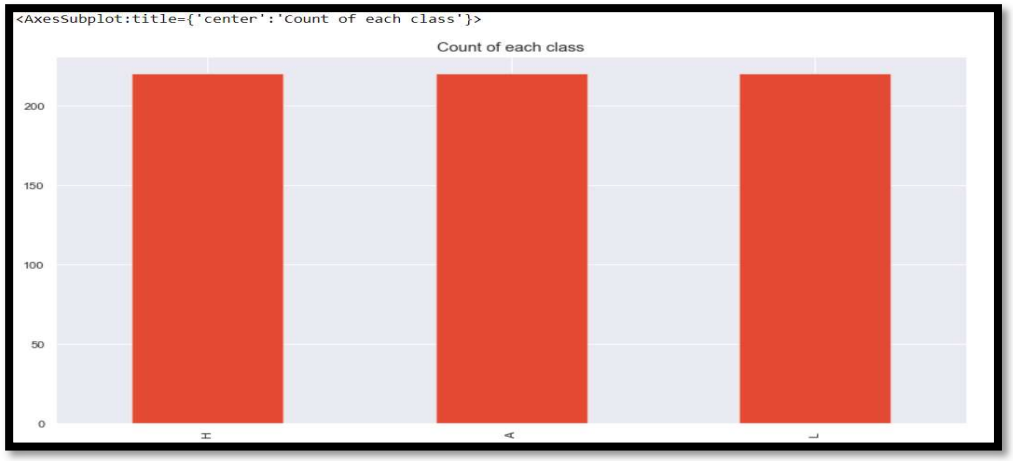
Based on the figure shown the dataset is more skewed towards the low level and this situation has been handled by the SMOTE Technique in order to make a balanced dataset in order to generate the instances using KNN technique.

Before SMOTE:



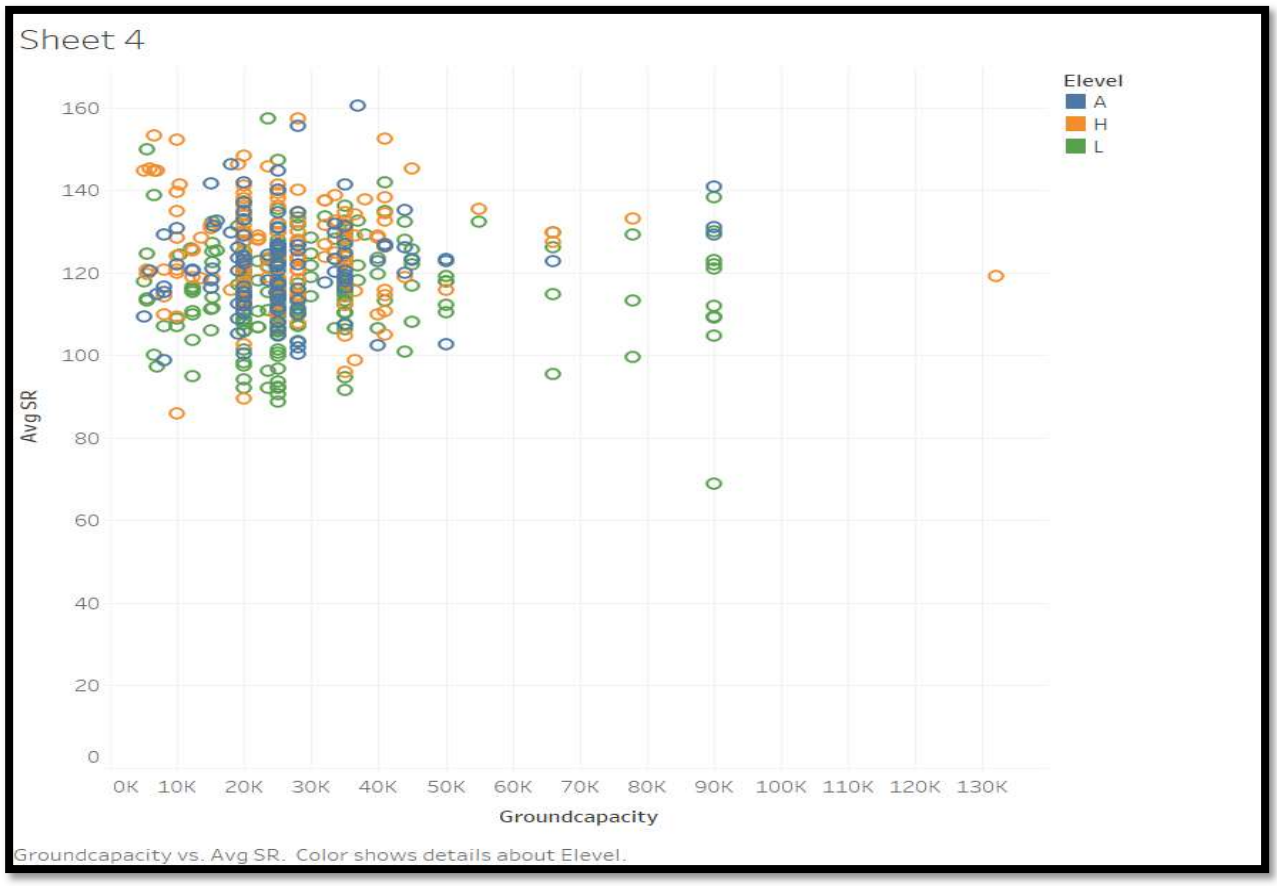
After SMOTE:

This plot was plotted using the Matplotlib Library of Python.



So, this clearly shows that how the SMOTE Technique works to create a balanced Dataset.

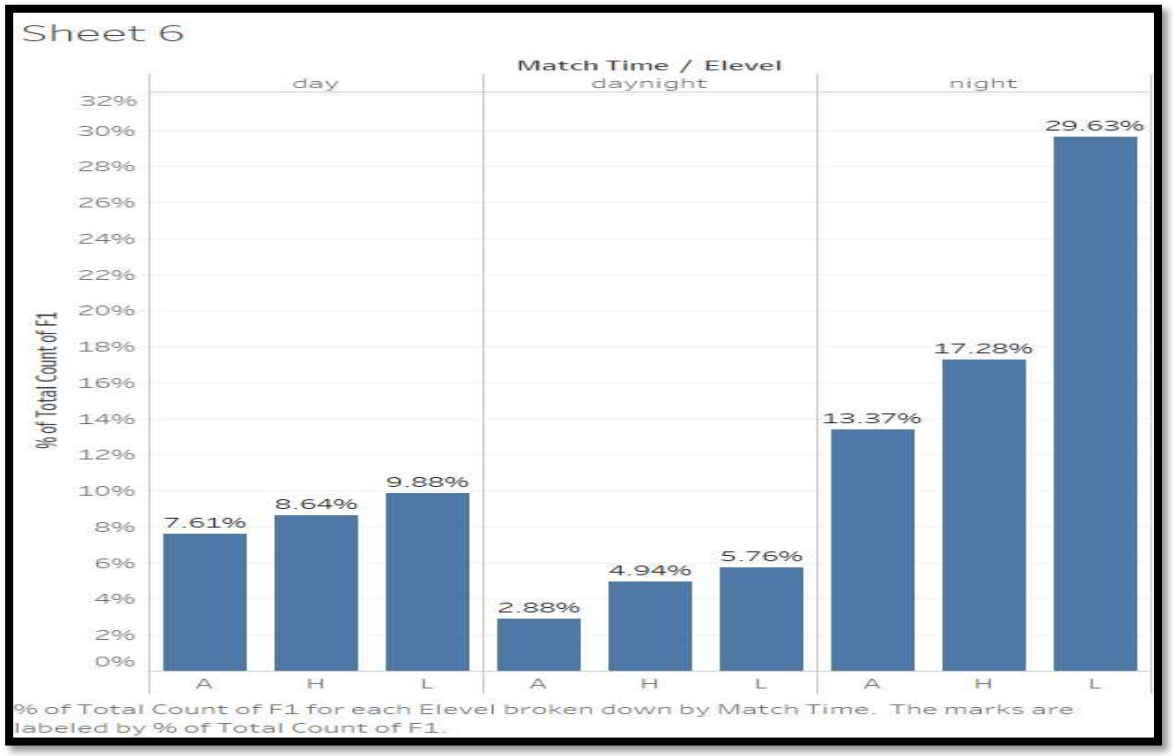
C. *Scatter Plot with the axes AvgSR i.e., Average Batting Strike Rate and the Ground Capacity and the corresponding Entertainment which seems to the most appropriate independent variables affecting the predicted class.*



Here also interesting observations are observed:

a. As the ground capacity (which is quite proportional to the ground size) like from 50k to 90k, we can see that the green rings i.e., the L matches are more and there are very few high H and A matches.
b. Blue rings are also clustered like 25K to 30K (as 20k is our imputed value) with specific range of Average Batting Strike Rate i.e., 100 to 130 so that is a good observation from the machine learning point of view that we can cut the slice.
c. It seems that models like SVM (nonlinear) which takes the data to higher dimensions and cut it, will work good on it and that is what actually happened with this dataset.

D. *This is the Bar chart corresponding to the MatchTime Vs Elevel*



Interesting Observations :

a. In case of night the match is more skewed towards the lower Entertainment Level, which is an interesting observation that under lights, players are expected to not perform that well.

E. *Below is simply the tabular representation showing the Home Advantage for the teams and also points to some interesting observations.*

Ha1	Team1	Elevel A	Elevel H	Elevel L
0	Afghanistan	3	3	3
	Australia	4	10	18
	Bangladesh	7	6	8
	England	5	9	14
	India	7	4	12
	New Zealand	3	2	17
	Pakistan	5	1	16
	South Africa	1	1	4
	Sri Lanka	3	3	7
	West Indies	2	1	1
1	Afghanistan			3
	Australia	16	8	23
	Bangladesh	2	7	7
	England	7	11	16
	India	9	14	17
	New Zealand	12	28	10
	Pakistan		4	4
	South Africa	16	21	11
	Sri Lanka	4	7	19
	West Indies	10	10	10

Ha2	Team2	Elevel A	Elevel H	Elevel L
0	Afghanistan	2	1	
	Australia	9	15	10
	Bangladesh	7	6	10
	England	14	22	9
	India	13	17	20
	New Zealand	8	9	20
	Pakistan	22	17	49
	South Africa	7	10	28
	Sri Lanka	8	15	35
	West Indies	14	27	21
1	Bangladesh			1
	New Zealand		1	1
	Pakistan	4	1	3
	South Africa	3		2
	Sri Lanka	1	2	8
	West Indies	4	7	3

Interesting Observations :

There can be various observations which can be seen about the countries' performance:

a. In case of New Zealand, it can be seen that, in case of 0 that means away from Home matches, the match had L or Lower entertainment level.
b. South Africa has also shown the same Trend as of New Zealand.
c. Sri Lanka has also shown the same trends as above.

This shows that home advantage can be a predicting factor for various countries.

Other graphs are being shown with the performance results.

2. Samples of the output representation

A. Decision Tree Classifier

Numpy representation

```
------Predicted and the actual output for Decision Tree Classifier-----
['H' 'L' 'L' 'L' 'H' 'L' 'L' 'L' 'L' 'H' 'H' 'L' 'H' 'H' 'H' 'H' 'L' 'L'
 'L' 'H' 'L' 'L' 'H' 'A' 'L' 'L' 'L' 'H' 'H' 'L' 'H' 'A' 'A' 'L' 'L' 'H'
 'H' 'A' 'H' 'H' 'H' 'H' 'H' 'L' 'A' 'L' 'H' 'L' 'L' 'L' 'L' 'L' 'L' 'L'
 'L' 'L' 'H' 'H' 'L' 'A' 'A' 'H' 'H' 'A' 'L' 'A' 'L' 'H' 'H' 'L' 'H' 'H'
 'H' 'L' 'L' 'H' 'H' 'L' 'H' 'L' 'H' 'L' 'A' 'H' 'A' 'H' 'L' 'L' 'H' 'L'
 'H' 'L' 'H' 'L' 'L' 'L' 'A' 'L' 'A' 'L' 'H' 'H' 'H' 'A' 'H' 'A' 'L' 'A'
 'L' 'L' 'L' 'H' 'L' 'L' 'L' 'A' 'H' 'L' 'L' 'L' 'L' 'H' 'H' 'L' 'L' 'H'
 'L' 'L' 'H' 'L' 'L' 'L']
['L' 'A' 'L' 'L' 'H' 'L' 'A' 'H' 'A' 'H' 'L' 'A' 'H' 'H' 'L' 'H' 'H' 'L'
 'H' 'H' 'A' 'A' 'L' 'A' 'H' 'H' 'H' 'H' 'H' 'L' 'H' 'A' 'H' 'H' 'A' 'H'
 'H' 'A' 'H' 'H' 'H' 'H' 'H' 'H' 'A' 'L' 'H' 'L' 'H' 'A' 'A' 'A' 'H' 'L'
 'L' 'H' 'H' 'L' 'H' 'A' 'A' 'L' 'H' 'A' 'A' 'A' 'L' 'L' 'H' 'A' 'H' 'H'
 'H' 'A' 'L' 'H' 'H' 'L' 'A' 'A' 'L' 'L' 'A' 'H' 'A' 'H' 'L' 'A' 'A' 'A'
 'A' 'H' 'L' 'A' 'H' 'A' 'H' 'H' 'A' 'H' 'A' 'L' 'L' 'L' 'H' 'A' 'A' 'A'
 'L' 'H' 'A' 'H' 'A' 'A' 'A' 'H' 'H' 'L' 'L' 'L' 'L' 'H' 'L' 'L' 'L' 'L'
 'L' 'L' 'H' 'H' 'H' 'L']
```

Dataframe Representation:

	Predicted Output	Actual Output
0	H	L
1	L	A
2	L	L
3	L	L
4	H	H
...	...	...
127	L	L
128	H	H
129	L	H
130	L	H
131	L	L

132 rows × 2 columns

B. Naïve Bayes Classifier

```
------Predicted and the actual output for Naive Bayes Classifier-----
['H' 'A' 'L' 'A' 'A' 'H' 'A' 'H' 'L' 'H' 'H' 'L' 'H' 'A' 'L' 'H' 'L' 'L'
 'A' 'A' 'H' 'H' 'L' 'A' 'H' 'H' 'L' 'H' 'A' 'H' 'L' 'A' 'H' 'A' 'L' 'H'
 'A' 'A' 'A' 'H' 'L' 'H' 'L' 'A' 'A' 'L' 'L' 'L' 'A' 'H' 'H' 'A' 'H' 'A'
 'L' 'A' 'L' 'L' 'H' 'A' 'A' 'L' 'L' 'A' 'L' 'A' 'L' 'L' 'H' 'H' 'H' 'H'
 'A' 'H' 'L' 'H' 'H' 'L' 'H' 'H' 'L' 'L' 'A' 'H' 'H' 'A' 'L' 'L' 'A' 'L'
 'L' 'L' 'H' 'A' 'H' 'L' 'H' 'H' 'L' 'H' 'H' 'L' 'L' 'H' 'H' 'A' 'H' 'L'
 'H' 'L' 'A' 'H' 'H' 'H' 'A' 'L' 'H' 'H' 'H' 'A' 'A' 'H' 'H' 'A' 'H' 'H'
 'L' 'L' 'H' 'H' 'A' 'H']
['L' 'A' 'L' 'L' 'H' 'L' 'A' 'H' 'A' 'H' 'L' 'A' 'H' 'H' 'L' 'H' 'H' 'L'
 'H' 'H' 'A' 'A' 'L' 'A' 'H' 'H' 'H' 'H' 'H' 'L' 'H' 'A' 'H' 'H' 'A' 'H'
 'H' 'A' 'H' 'H' 'H' 'H' 'H' 'H' 'A' 'L' 'H' 'L' 'H' 'A' 'A' 'A' 'H' 'L'
 'L' 'H' 'H' 'L' 'H' 'A' 'A' 'L' 'H' 'A' 'A' 'A' 'L' 'L' 'H' 'A' 'H' 'H'
 'H' 'A' 'L' 'H' 'H' 'L' 'A' 'A' 'L' 'L' 'A' 'H' 'A' 'H' 'L' 'A' 'A' 'A'
 'A' 'H' 'L' 'A' 'H' 'A' 'H' 'H' 'A' 'H' 'A' 'L' 'L' 'L' 'H' 'A' 'A' 'A'
 'L' 'H' 'A' 'H' 'A' 'A' 'A' 'H' 'H' 'L' 'L' 'L' 'L' 'H' 'L' 'L' 'L' 'L'
 'L' 'L' 'H' 'H' 'H' 'L']
```

	Predicted Output	Actual Output
0	H	L
1	A	A
2	L	L
3	A	L
4	A	H
...	...	...
127	L	L
128	H	H
129	H	H
130	A	H
131	H	L

132 rows × 2 columns

C. SVM Classifier

```
------Predicted and the actual output for SVM Classifier-----
['H' 'L' 'L' 'L' 'H' 'L' 'A' 'H' 'A' 'H' 'H' 'A' 'H' 'A' 'L' 'H' 'L' 'A'
 'H' 'A' 'L' 'A' 'L' 'A' 'H' 'H' 'L' 'H' 'H' 'H' 'L' 'A' 'H' 'H' 'A' 'H'
 'H' 'A' 'H' 'H' 'H' 'H' 'L' 'L' 'A' 'L' 'A' 'L' 'L' 'L' 'A' 'A' 'H' 'H'
 'L' 'L' 'L' 'L' 'H' 'A' 'A' 'L' 'H' 'L' 'L' 'A' 'L' 'L' 'L' 'A' 'H' 'H'
 'A' 'A' 'L' 'H' 'H' 'L' 'L' 'L' 'H' 'L' 'A' 'H' 'L' 'H' 'L' 'L' 'L' 'A'
 'L' 'L' 'H' 'L' 'L' 'A' 'H' 'L' 'L' 'H' 'H' 'A' 'L' 'H' 'H' 'A' 'L' 'A'
 'L' 'L' 'A' 'H' 'L' 'H' 'A' 'L' 'H' 'L' 'H' 'L' 'A' 'H' 'L' 'L' 'H' 'H'
 'L' 'L' 'H' 'L' 'H' 'H']
['L' 'A' 'L' 'L' 'H' 'L' 'A' 'H' 'A' 'H' 'L' 'A' 'H' 'H' 'L' 'H' 'H' 'L'
 'H' 'H' 'A' 'A' 'L' 'A' 'H' 'H' 'H' 'H' 'H' 'L' 'H' 'A' 'H' 'H' 'A' 'H'
 'H' 'A' 'H' 'H' 'H' 'H' 'H' 'H' 'A' 'L' 'H' 'L' 'H' 'A' 'A' 'A' 'H' 'L'
 'L' 'H' 'H' 'L' 'H' 'A' 'A' 'L' 'H' 'A' 'A' 'A' 'L' 'L' 'H' 'A' 'H' 'H'
 'H' 'A' 'L' 'H' 'H' 'L' 'A' 'A' 'L' 'L' 'A' 'H' 'A' 'H' 'L' 'A' 'A' 'A'
 'A' 'H' 'L' 'A' 'H' 'A' 'H' 'H' 'A' 'H' 'A' 'L' 'L' 'L' 'H' 'A' 'A' 'A'
 'L' 'H' 'A' 'H' 'A' 'A' 'A' 'H' 'H' 'L' 'L' 'L' 'L' 'H' 'L' 'L' 'L' 'L'
 'L' 'L' 'H' 'H' 'H' 'L']
```

	Predicted Output	Actual Output
0	H	L
1	L	A
2	L	L
3	L	L
4	H	H
...	...	...
127	L	L
128	H	H
129	L	H
130	H	H
131	H	L

132 rows × 2 columns

D. Logistic regression

```
------Predicted and the actual output for Logistic Regression Classifier-----
['H' 'A' 'H' 'H' 'A' 'L' 'H' 'A' 'L' 'H' 'A' 'L' 'L' 'A' 'A' 'H' 'A' 'A'
 'L' 'L' 'H' 'A' 'H' 'L' 'L' 'A' 'L' 'L' 'L' 'L' 'A' 'H' 'A' 'H' 'H' 'A'
 'L' 'A' 'A' 'A' 'L' 'L' 'A' 'L' 'A' 'H' 'L' 'L' 'L' 'H' 'A' 'A' 'L' 'H'
 'L' 'A' 'L' 'A' 'H' 'L' 'A' 'A' 'L' 'H' 'L' 'L' 'A' 'H' 'L' 'L' 'A' 'H'
 'A' 'H' 'A' 'A' 'H' 'H' 'L' 'L' 'L' 'H' 'A' 'H' 'L' 'H' 'L' 'A' 'H' 'H'
 'H' 'H' 'L' 'A' 'L' 'H' 'A' 'A' 'L' 'A' 'H' 'A' 'L' 'H' 'A' 'H' 'A' 'L'
 'A' 'L' 'H' 'H' 'L' 'A' 'A' 'A' 'H' 'L' 'A' 'A' 'H' 'L' 'L' 'A' 'H' 'L'
 'H' 'A' 'H' 'L' 'H' 'L']
['A' 'L' 'A' 'H' 'L' 'L' 'A' 'A' 'L' 'A' 'H' 'A' 'L' 'H' 'L' 'H' 'L' 'A'
 'H' 'A' 'H' 'A' 'A' 'L' 'H' 'H' 'L' 'L' 'L' 'L' 'L' 'H' 'A' 'A' 'L' 'H'
 'L' 'A' 'H' 'A' 'H' 'L' 'L' 'A' 'A' 'A' 'H' 'L' 'A' 'H' 'A' 'L' 'L' 'A'
 'L' 'A' 'L' 'A' 'A' 'H' 'A' 'A' 'L' 'A' 'L' 'L' 'A' 'L' 'A' 'L' 'L' 'H'
 'L' 'L' 'A' 'L' 'A' 'H' 'L' 'L' 'L' 'H' 'L' 'L' 'H' 'H' 'A' 'L' 'A' 'L'
 'H' 'A' 'L' 'L' 'H' 'A' 'A' 'H' 'L' 'L' 'H' 'A' 'H' 'H' 'A' 'H' 'H' 'L'
 'L' 'H' 'H' 'H' 'L' 'L' 'L' 'A' 'A' 'L' 'A' 'H' 'L' 'L' 'L' 'A' 'H' 'L'
 'L' 'A' 'L' 'L' 'H' 'L']
```

	Predicted Output	Actual Output
0	H	A
1	L	L
2	L	A
3	L	H
4	H	L
...	...	...
127	L	A
128	H	L
129	L	L
130	H	H
131	H	L

132 rows × 2 columns

3. Evaluation results

Statistical Results

A. Decision Tree
B. Naïve Bayes
C. SVM
D. Logistic Regression
E. Baseline Zero R Model
F. Random Forest

A. Results of the ***Decision Tree using SMOTE based balanced dataset with 80% and 20% split of Data****.*

```
--------------------Decision Tree--------------------------------
[[13  4 23]
 [ 3 32 19]
 [ 1 13 24]]
              precision    recall  f1-score   support

           A       0.76      0.33      0.46        40
           H       0.65      0.59      0.62        54
           L       0.36      0.63      0.46        38

    accuracy                           0.52       132
   macro avg       0.59      0.52      0.51       132
weighted avg       0.60      0.52      0.53       132

<AxesSubplot:>
```


```
For the Decision Tree classifier :
The average of the Macro averaged F1 score using the cross validation is: 0.5144991328841474
The average of the Macro averaged Precision score using the cross validation is: 0.563819424269317
The average of the Macro averaged Recall score using the cross validation is: 0.5275537634408602
The average of the Accuracy using the cross validation is: 0.5291313389857623
```

B. *Results of the* ***Naïve Bayes Classifier using SMOTE based balanced dataset with 80% and 20% split of Data****.*

```
--------------------Naive Bayes on SMOTE Balanced Dataset-------------------------------
0.5
[[17 13 10]
 [14 29 11]
 [ 5 13 20]]
              precision    recall  f1-score   support

           A       0.47      0.42      0.45        40
           H       0.53      0.54      0.53        54
           L       0.49      0.53      0.51        38

    accuracy                           0.50       132
   macro avg       0.50      0.50      0.50       132
weighted avg       0.50      0.50      0.50       132


<AxesSubplot:>
```


```
For the Naive Bayes classifier :
The average of the Macro averaged F1 score using the cross validation is: 0.4005765731906107
The average of the Macro averaged Precision score using the cross validation is: 0.4033713216846847
The average of the Macro averaged Recall score using the cross validation is: 0.4096582181259601
The average of the Accuracy using the cross validation is: 0.40947048472244446
```

C. *Results of the* ***SVM Classifier using SMOTE based balanced dataset with 80% and 20% split of Data***.

```
--------------------SVM on SMOTE Balanced Dataset-------------------------------
Test Set Accuracy : 62.121212121212125 %


[[23  2 15]
 [ 4 35 15]
 [ 3 11 24]]
              precision    recall  f1-score   support

           A       0.77      0.57      0.66        40
           H       0.73      0.65      0.69        54
           L       0.44      0.63      0.52        38

    accuracy                           0.62       132
   macro avg       0.65      0.62      0.62       132
weighted avg       0.66      0.62      0.63       132


<AxesSubplot:>
```


```
For the SVM based classifier :
The average of the Macro averaged F1 score using the cross validation is: 0.5836740468011709
The average of the Macro averaged Precision score using the cross validation is: 0.5897392862236653
The average of the Macro averaged Recall score using the cross validation is: 0.5900537634408601
The average of the Accuracy using the cross validation is: 0.5898736202207647
```

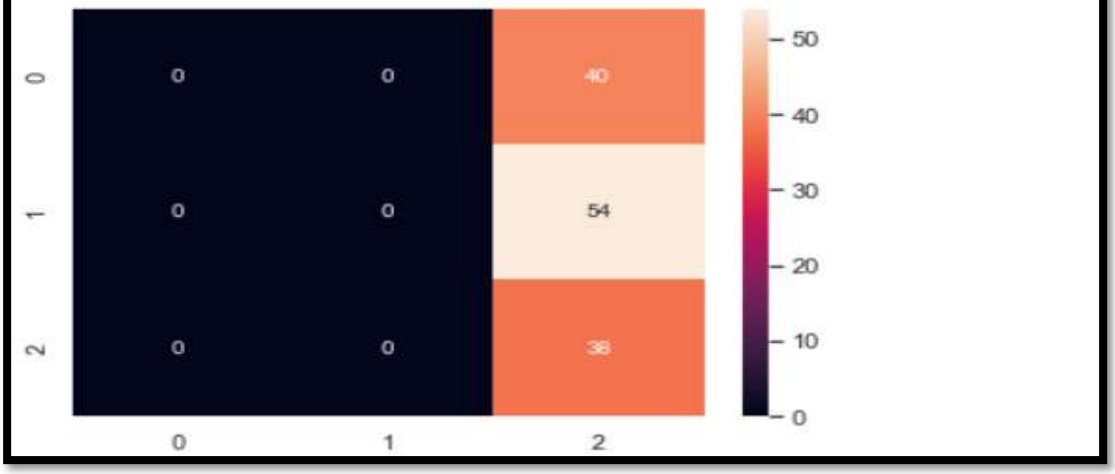
D. *Results of the* ***Logistic Regression using SMOTE based balanced dataset with 80% and 20% split of Data****.*

```
--------------------Logistic Regression on SMOTE Balanced Dataset---------------
Test Set Accuracy : 52.27272727272727 %


[[21 15  6]
 [ 8 17  9]
 [17  8 31]]
              precision    recall  f1-score   support

           A       0.46      0.50      0.48        42
           H       0.42      0.50      0.46        34
           L       0.67      0.55      0.61        56

    accuracy                           0.52       132
   macro avg       0.52      0.52      0.51       132
weighted avg       0.54      0.52      0.53       132


<AxesSubplot:>
```


```
For the Logistic Rgeression based classifier :
The average of the Macro averaged F1 score using the cross validation is: 0.47100740691750004
The average of the Macro averaged Precision score using the cross validation is: 0.48078801883530253
The average of the Macro averaged Recall score using the cross validation is: 0.4780145929339477
The average of the Accuracy using the cross validation is: 0.47774756039033756
```

E. *Results of the* ***Base Line ZeroR Model using SMOTE based balanced dataset with 80% and 20% split of Data****.*

```
--------------------ZeroR on SMOTE Balanced Dataset----------
[[ 0  0 40]
 [ 0  0 54]
 [ 0  0 38]]
              precision    recall  f1-score   support

           A       0.00      0.00      0.00        40
           H       0.00      0.00      0.00        54
           L       0.29      1.00      0.45        38

    accuracy                           0.29       132
   macro avg       0.10      0.33      0.15       132
weighted avg       0.08      0.29      0.13       132

<AxesSubplot:>
```



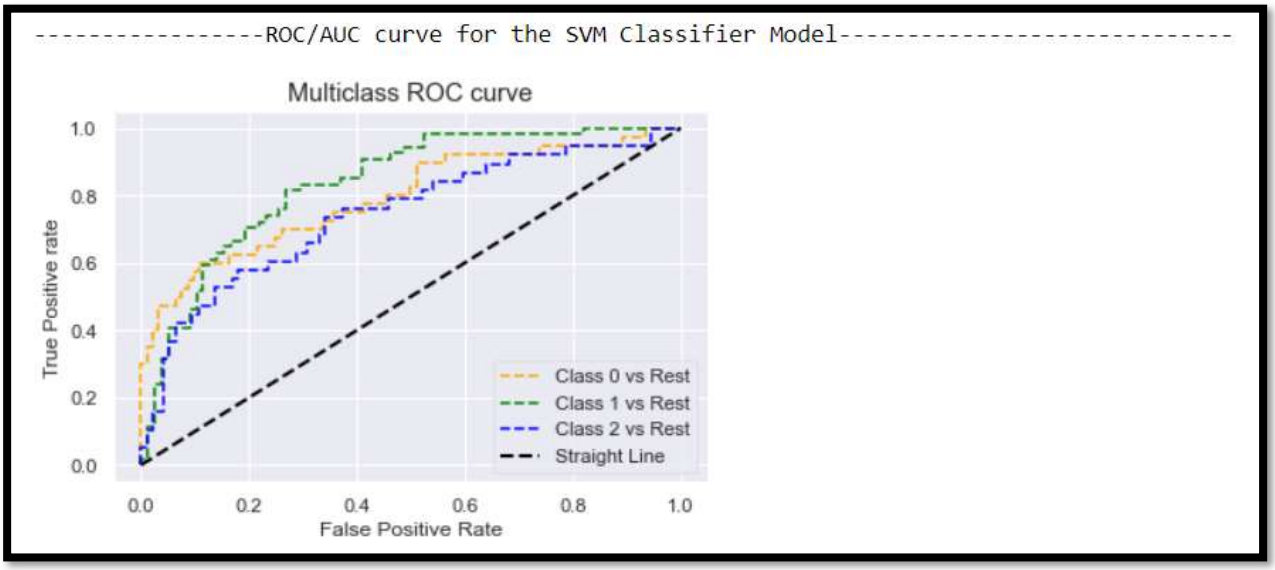
F. *Results of the* ***Random Forest using SMOTE based balanced dataset with 80% and 20% split of Data****.*

```
For the Random Forest based Ensemble classifier :
The average of the Macro averaged F1 score using the cross validation is: 0.6266006042556326
The average of the Macro averaged Precision score using the cross validation is: 0.6478044333386317
The average of the Macro averaged Recall score using the cross validation is: 0.6274001536098311
The average of the Accuracy using the cross validation is: 0.6367941129419293
```

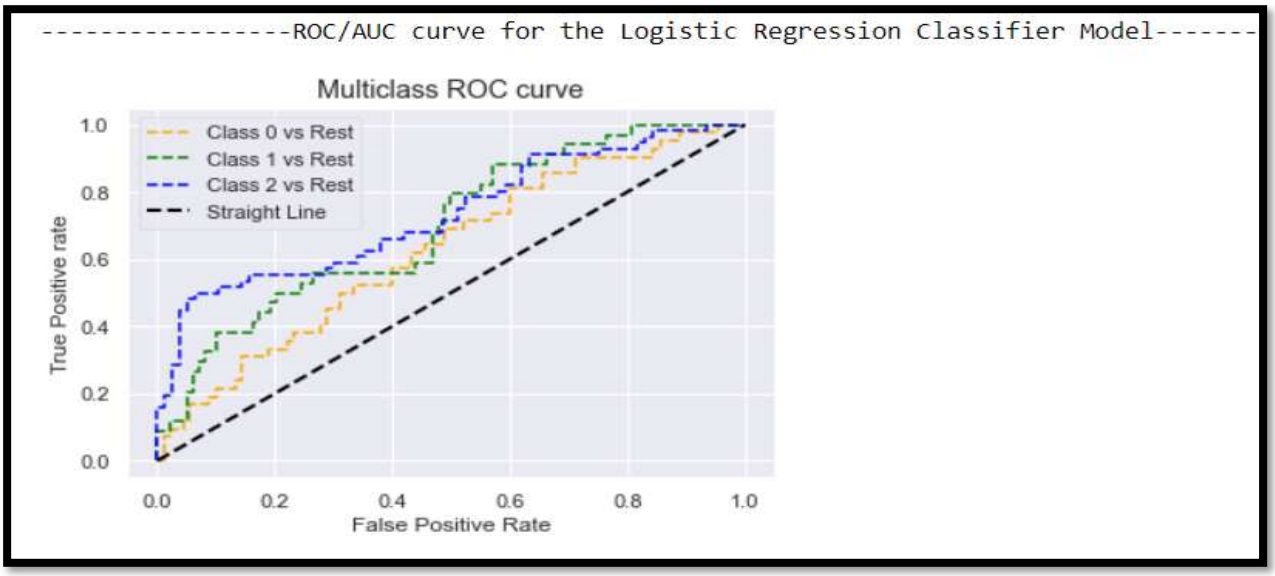
```
------------------Random Forest Based Ensemble Algorithm on SMOTE Balanced Dataset--
Test Set Accuracy : 61.36363636363637 %


[[20  3 17]
 [ 2 38 14]
 [ 4 11 23]]
              precision    recall  f1-score   support

           A       0.77      0.50      0.61        40
           H       0.73      0.70      0.72        54
           L       0.43      0.61      0.50        38

    accuracy                           0.61       132
   macro avg       0.64      0.60      0.61       132
weighted avg       0.65      0.61      0.62       132


<AxesSubplot:>
```

Model Comparisons using ROC-AUC Curve

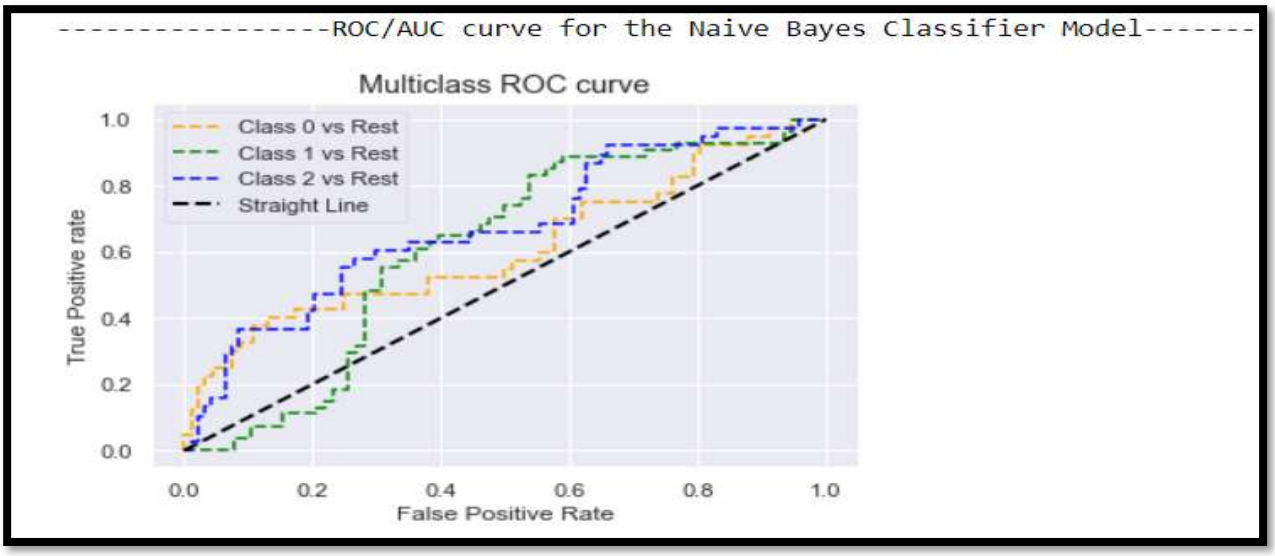
A. SVM Classifier



B. Logistic Regression Classifier

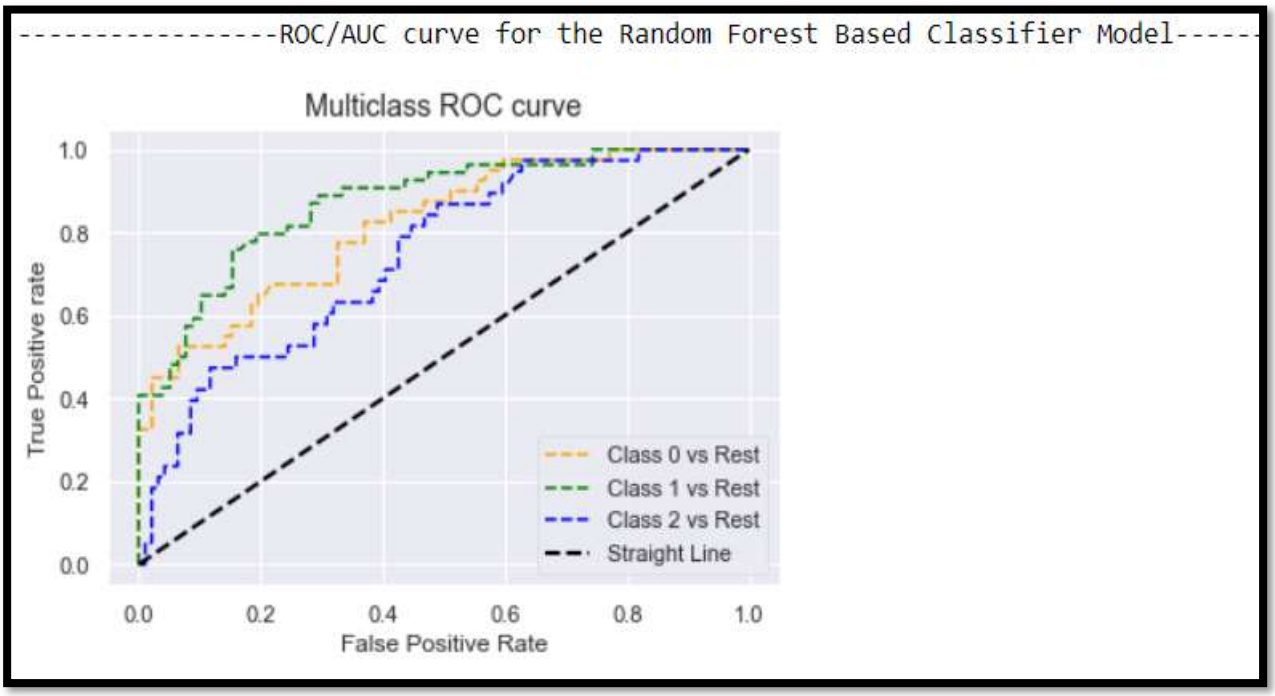


C. Naïve Bayes Classifier



*Note Class 0 , 1 , 2 = A, H, L respectively

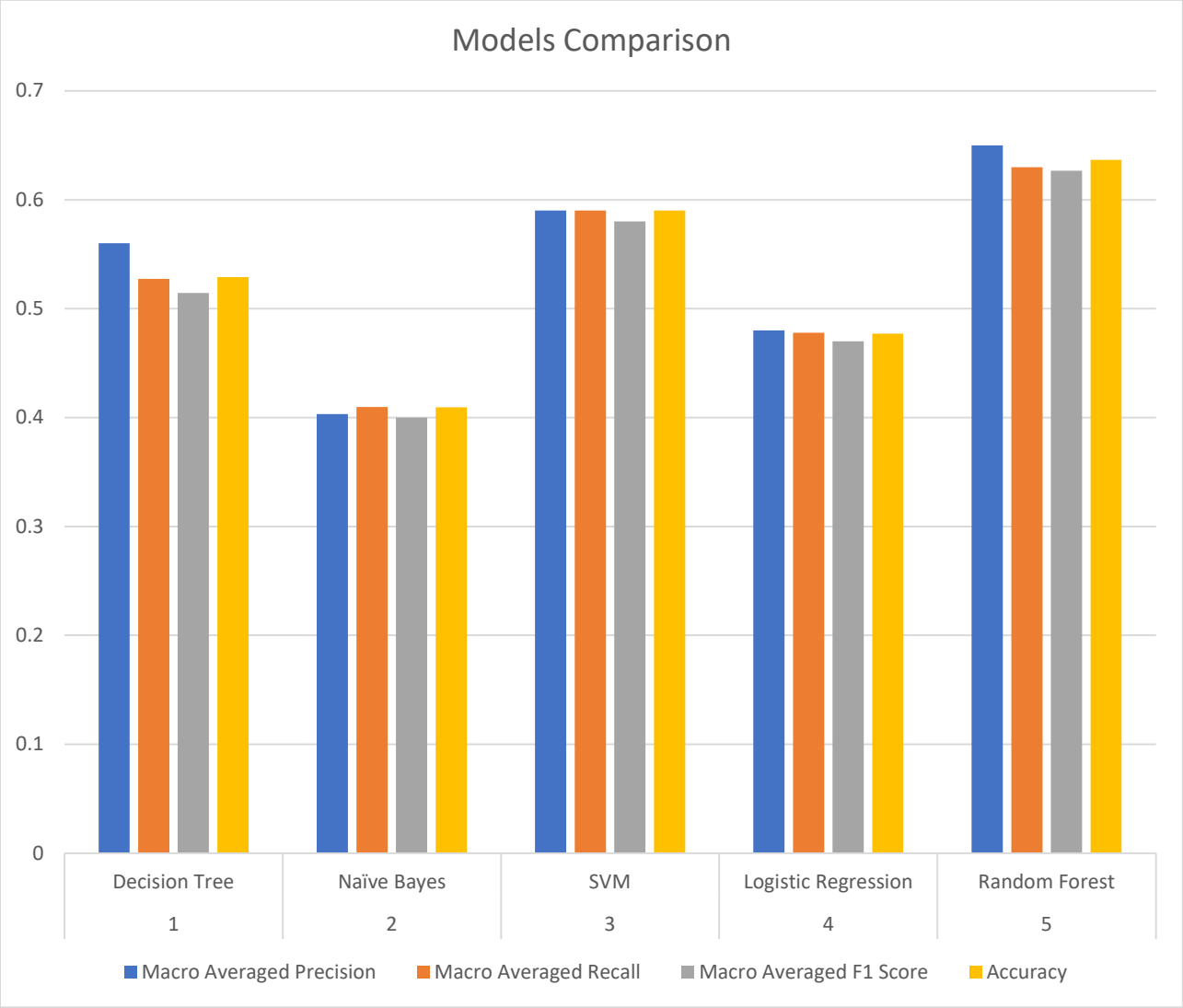
D. Random Forest Based Ensemble Classifier



Performance Results

- It is important to note that greater the Area Under Curve, better is the performance of the model at distinguishing between the positive and negative classes. So, above clearly shows that our SVM Model and Random Classifier Models are performing much better than the other models developed.
- Even based on the statistical parameters like Precision, Recall, Accuracy and F1 Score , SVM classifier and Random Classifier Model seem to work better in comparison to others.

Graphical Comparison of Mean Evaluation Metrics calculated using 7-fold Cross Validation.



Discussion

It can be seen that the performance of the machine learning models (non-ensemble) created through various algorithms have not performed that well i.e., their accuracy, precision, recall and F1 score is around 50-60 % but the SVM Classifier has worked better as it has shown the result of over 60 % and the mean of evaluation metrics calculated using cross validation are also almost 60 %. Random Classifier has worked best with all the macro averaged evaluation metrics above 60 % and the mean of evaluation metrics calculated using cross validation are 62 - 65 %.

Baseline Comparison

In order to check the performance of our models, the comparison was made to the baseline model i.e., the ZeroR on a balanced Dataset and it shows that our models performed better than the case when we pick the majority class and predict it as an output for every instance. It is very important to note that when we are working on an interesting problem in hand like the present one where we are collecting the data attributes ourselves and then building models, baseline results tell us that whether we are adding value building prediction models using the complex algorithms.

Models Comparison

The probable reasons that SVM performed better in our case can be that in SVM the best marginal distance is calculated between the line and the support vectors which highly reduce the errors. In Logistic Regression, decision boundaries are chosen near the optimal point. Other reason may be that SVM use the geometrical properties or the spatial representation of data as the criteria for classification and it seems to work better as indicated during the exploratory analysis. Further, as per the studies available on the web, it is shown that sometimes the decision tree performance suffers in case the features are one hot encoded. So, this may be one of the issues with the Decision Tree Algorithm not working that well.

Further, Naïve Bayes performance has been quite abysmal. The probable reason for that may be that the Naïve Bayes consider the features as independent . But SVM still consider interactions to a certain degree especially when we are using the non-linear kernel like Gaussian(rbf) being used in our case. One thing more which have been noticed is that Naïve Bayes is a probabilistic method and the SVM as conveyed earlier use the spatial representation. It can not generally commented that which model is suitable as a ML practitioner and usually experiments are performed with various models. Based on this dataset, may be the hyperplane separation technique of separating the data points between the classes used in SVM is more suitable for the dataset in hand.

It is interesting to see that Random Forest based ensemble learning is performing much better than the individual Decision Tree Classifier and even all other classifiers used. It is producing the cross validated accuracy, F1 score, precision and recall around 65 % which is quite commendable based on the small dataset being used. The reason may be the fact in the decision tree algorithm, at each node, the best choice is made for splitting based on the purity of class but the problem with this approach is that it does not always lead to the global optimum. It means that it is still unsure that there may be other best route of choices. Here the ensemble learning like Random Forest Classifier works well as they create various decision trees and choose the results through majority voting. This gives us the direction to test the data with more ensemble-based learning methods in future work.

Note that for every algorithm used, cross validation has been used. The mean of the performance parameters calculated using cross validation have been computed to compare their performances. Cross validation results lead to better comparison neglecting the effect of overfitting of models on certain combination of training and test split.

ROC-AUC Comparison

Further, as the models are also compared using ROC-AUC Curve , the ROC-AUC curves were plotted for the various Machine Learning Algorithms. ROC – AUC Curves are the plots of the True Positive Rate Vs False Positive Rate. TPR is also called as the recall or the sensitivity and the FPR is 1- specificity where the specificity is the TNR which is the True Negative Rate. So, in general sense, they may be called as the plot between the hit rate vs the false alarm rate. In this curve, the threshold is varied for making the decision between the positive and the negative class and then it is seen that how our model is performing, that how is the Hit Rate and False Alarm Rate. Generally, the models in case of binary classification should be above the line passing through origin with slope 1. Otherwise, the model has no skill as the hit rate is less than the false alarm rate. As the multi class classification problem is being dealt in present case, One Vs Rest approach was followed, where the particular class is a positive class and the other two are negative for it. So, confusion matrix can be imagined and based on that ROC-AUC curve can be plotted for that particular class. That is why TPR Vs FPR i.e., the ROC curve was plotted for each class. Further, it is known that greater is the Area Under Curve , better is the performance of the model at distinguishing between the positive and negative classes. This clearly shows that our SVM Model and Ensemble based Random Classifiers are performing much better than the other models. And it can be clearly seen that SVM classifier has shown the best ROC-AUC curve with largest Area under Curve in comparison to others. This clearly shows that for various thresholds, our TPR is always more than the FPR. This makes us decide the threshold based on the problem in hand i.e., the more TPR and no problem with high FPR or we want to optimize them both. Further, the Logistic Regression did fare in ROC-AUC curve, but the Naïve Bayes showed the poor performance.

Other

The various other factors that generally affects the models like feature scaling, class imbalance was also taken care. As the Logistic regression is highly affected by the scaling issues so the standard scaler was used to standardize the results, to negate the influence of particular dimension or independent variable. The class imbalance was taken care by using the SMOTE Technique (Synthetic Minority Oversampling Technique). In this technique, the number of minority classes is balanced. In this one minority instance is selected and then the K nearest neighbors in the feature space are selected where one neighbor is chosen at random. Then a synthetic instance is created by the convex combination of the chosen minority class instance and the selected neighbor. This technique provided me with the balanced dataset as well as the increased number of instances for problem in hand.

Further, the major target for our machine learning model are the advertisers and agencies and people investing in the matches. Therefore, it is felt that the precision of the Class "H" is very important to enable them to put their best bet on the match and retrieve the better return. And that seems to better be provided by the Random Forest Classifier based on the current dataset.

As the machine learning or Data Science problem all depends on the data as a food for producing better results, it seems that in order to bring better results more features and instances are required and hence the dataset needs to be augmented.

Conclusion and future work

In conclusion, it can be seen that evaluation metrics for the model generated through various algorithms used are higher than the baseline Zero R. But apart from SVM and Random Forest, which are showing the accuracy, F1 Score, and other evaluation metrics above 60 %, others' metrics are hovering in the range of 50 to 60 % while Naïve Bayes performing badly. These evaluation metrics have also been derived based on the cross validation to negate the effect of overfitting. The probable reasons that have already been discussed are that the ensemble methods like Random Forest generates various decision trees and then take majority voting hence, tends to be more accurate. Actually, Random Forests produce unpruned and diverse trees which increase the resolution of the feature space. Further, Random Forests get off the overfitting problem based on its randomness and voting mechanisms. SVM also has performed better than the others and the reason that seems to be causing this is the use of the geometrical representation by SVM to slice the data points into various classes.

Some future strategies to increase the performance of machine learning models:

1. Increase the feature space.

In current problem, the relied features are those that directly affects the batting like the ratings, average batting strike rate, ground size and so on. As the entertainment level which is judged by the percentage of fours and sixes out of the total runs, it can also be affected by the capabilities of the bowlers. In that case, it is felt that in future work, the bowlers' data will also be scraped and will be used in the future enhancements for the project. As mentioned in the first reviewed paper, it will be tried to calculate the team weights using the individual ratings in the particular year and these team weights will supplement the features' space.

2. More algorithms will be used.

As we have multi class classification problem in hand, in that case, other algorithms will also be tried on this problem. Further, as more features are collected, the results of other algorithms like gradient boosted decision trees, Multilayer Perceptron etc can also be seen on this problem apart from the ones currently being used.

3. Scaling methodology

As learnt in the second reviewed paper, the parameters were time scaled so as to bring them on par with respect to time. Similarly, it is felt that the features like strike rates are increasing as the more players are tending to incline towards faster batting. Hence, the older average strike rates of the team are not at par with the current average strike rates. Therefore, it is felt that such scaling can improve the performance of the models.

References

1. Predicting Outcome of Indian Premier League (IPL) Matches Using Machine Learning 1809.09813.pdf (arxiv.org)
2. Prediction of the outcome of a Twenty-20 Cricket Match Microsoft Word - Final_Report.docx (wisc.edu)
3. Sport analytics for cricket game results using machine learning: An experimental study ACI-j.aci.2019.11.006_aq 1..1 (emerald.com)
4. Quantitative Analysis of Forthcoming ICC Men's T20 World Cup 2020 Winner Prediction using Machine Learning. Quantitative Analysis of Forthcoming ICC Men's T20 World Cup 2020 Winner Prediction using Machine Learning (ijcaonline.org)
5. Dynamic Winner Prediction in Twenty20 Cricket: Based on Relative Team Strengths MLSA17_paper_5.pdf (kuleuven.be)
6. Cricket Match Outcome Prediction Using Machine Learning 480_9.CRICKET_MATCH_OUTCOME_PREDICTION_USING.pdf (ijasret.com)
7. Performance measure on multiclass classification (169) Performance measure on multiclass classification [accuracy, f1 score, precision, recall] - YouTube
8. Analytics Vidhya Website for various information https://www.analyticsvidhya.com/
9. Machine Learning Mastery website for various information https://machinelearningmastery.com/
10. Towards Data Science Website https://towardsdatascience.com/
11. https://www.bloombergquint.com/business/why-advertising-rates-have-hit-a-record-this-cricket-world-cup
12. https://www.theguardian.com/sport/2020/mar/11/cricket-ticket-prices-fans-good-value-money
13. https://adscholars.com/blog/advertisement-spend-ipl-2020/

Time Log for Vikhyat Dhamija

Date	Time	Duration	Work
04/10/2021	18:00 – 00:00 EST	5	Studied the methodology and Libraries like Beautiful Soup Libraries of Python to be used for Web Scraping.
04/12/2021	11:00 – 16:00 EST	5	Worked on building script for scraping through the ESPN Cricinfo Website for getting the relevant attributes and storing them in the CSV file.
04/13/2021	08:00 – 13:30 EST	5.5	Worked on building script for scraping through the ESPN Cricinfo Website for getting the relevant attributes and storing them in the CSV file.
04/14/2021	09:00 – 14:00 EST	5	Worked on building script for scraping through the ESPN Cricinfo Website for getting the relevant attributes and storing them in the CSV file.
04/15/2021	19:30 – 00:00 EST	4.5	Worked on building script for scarping the Wikipedia Website for getting the relevant attributes and storing them in the CSV file.
04/16/2021	20:00 – 01:00 EST	5	Studied the methodology to perform various Data Cleaning tasks like Duplicate Removals, Filling the Null Values, Performing the joins using the Merge operations and various other necessary Data cleaning and Feature Engineering operations to be performed using the Python Pandas.
04/17/2021	19:30 – 23:30 EST	4	Performed the Data Cleaning Tasks on the Dataset so as to come out with the combined and cleaned Data set to be used. Note that the calculations were performed to generate the Class Labels
04/18/2021	10:00 – 14:30 EST	4.5	Applied various Machine Learning Algorithms on the Dataset to check their performances
04/19/2021	09:00-13:00 EST 20:30- 23:30 EST	7	Based on the suggestions from Dr. Bill, studied about the SMOTE and applied SMOTE to generate more Balanced Dataset and increase the number of instances. Applied various Machine Learning Algorithms again on the Dataset to check their performances. Tried to drop some features and performed some hit and trial to check the performances.
04/20/2021	21:30 – 23:00 EST	2.5	Build the function for ZeroR algorithm and then check the performance of ZeroR model for our dataset in order to check that whether we are gaining from the machine Learning Algorithms.

04/20/2021	08:00 – 12:00 EST 20:00 – 22:00 EST	6	Research ways to enhance the metrics and understood that one of the best ways is to introduce the best attributes or independent variables that can lead to improved prediction. Put the thought process to have such feature that can be extracted also through Web Scraping and came out with the Average Batting Strike rates of the Teams Playing. Modified the python script in order to extract the above-mentioned field from the website.
04/21/2021	11:00 – 14:00 EST	3	Reworked the Data Cleaning, merging tasks in order to produce the final Dataset again.
04/22/2021	10:00 – 14:00 EST	4	Based on the suggestion from the professor and for checking the relevancy of the attributes and to gain further insights about our data points, the exploratory analysis was performed using Data Visualization using Tableau. Feature correlation check was also performed through Pandas.
04/23/2021	21:00 – 00:00 EST	3	Applied various Machine Learning Algorithms again on the Dataset to check their performances. Here I got my performance improved for various models like SVM and Random Forest based classifier with parameters reaching 60-65 %.
04/23/2021	09:00 – 12:00 EST 13:00 – 17:00 EST	7	Studied the methodology to generate the ROC-AUC curves using Python Pandas. Created ROC-AUC curves for various Machine Learning Algorithms for comparing various Models.
04/24/2021	12:00 – 16:00 EST 21:30 - 23:30 EST	6	Report Writing
04/25/2021	12:00 – 16:00 EST 21:30 - 23:30 EST	6	Report Writing
04/27/2021	19:00 – 00:00 EST	5	Presentation Preparation
05/02/2021	20:00 - 21:00 EST	1	Report Writing
05/03/2021	22:00 - 23:00 EST	1	Report Writing
05/04/2021	16:00 - 17:30 EST	1.5	Report Writing

EST – Eastern Standard Time

Sample from the Dataset

Team1	Team2	MatchTime	Ha1	Ha2	T1w	T2w	T1b	Groundcapacity	AvgSR	CRating	Elevel
West Indies	England	daynight	1	0	0	1	1	12400	125.3082353	487	H
West Indies	England	daynight	1	0	1	0	0	8000	115.3088889	487	A
West Indies	England	daynight	1	0	1	0	1	8000	109.9153333	487	H
England	Pakistan	day	1	0	0	1	0	5500	124.6469231	547	L
New Zealanc	England	day	1	0	0	1	1	18000	146.2533333	515	A
New Zealanc	England	day	1	0	0	1	1	33500	138.7875	515	H
New Zealanc	England	day	1	0	1	0	1	6000	145.1994118	515	H
New Zealanc	England	night	1	0	1	0	0	10500	141.3675	515	H
New Zealanc	England	day	1	0	0	1	1	41000	152.3853333	515	H
India	Australia	night	1	0	0	1	1	25000	123.9527778	521	L
India	Australia	night	1	0	0	1	1	40000	129.0390909	521	H
Australia	Sri Lanka	day	1	0	0	1	1	50000	115.9666667	488	H
Australia	Sri Lanka	night	1	0	0	1	0	37000	121.7271429	488	L
Australia	Sri Lanka	night	1	0	1	0	0	90000	121.0761538	488	L
Australia	Pakistan	day	1	0	1	0	0	44002	126.2111111	547	A
Australia	Pakistan	night	1	0	0	1	0	12000	125.8815385	547	L
Australia	Pakistan	daynight	1	0	1	0	0	20000	112.5541667	547	L
South Africa	Pakistan	night	1	0	0	1	1	25000	127.0915789	548	A
South Africa	Pakistan	day	1	0	0	1	1	28000	123.3471429	548	H
South Africa	Pakistan	night	1	0	1	0	0	20000	100.4281818	548	A
South Africa	Sri Lanka	night	1	0	1	0	0	25000	115.9594737	489	L
South Africa	Sri Lanka	night	1	0	0	1	1	20000	122.37125	489	H
South Africa	Sri Lanka	day	1	0	0	1	1	28000	115.856	489	H
India	South Africa	night	1	0	1	0	0	20000	137.2258333	522	L
India	South Africa	night	1	0	1	0	1	40000	122.9392857	522	A
India	West Indies	day	0	1	1	0	0	20000	108.9247368	486	L
India	West Indies	day	0	1	1	0	1	20000	120.6253846	486	L
West Indies	India	day	1	0	0	1	1	15000	116.3407692	486	A