BUSINESS REPORT-CRICKET WIN PREDICTION

TABLE OF CONTENTS

Sl .No.	**Review Parameters**	**Page No.**
1.	**Introduction**	4,5
	- Brief introduction about the problem statement and the need of solving it.	
2.	**EDA and Business Implication**	5-13
	- Uni-variate / Bi-variate / Multi-variate analysis to understand relationship b/w variables. How your analysis is impacting the business?	
	- Both visual and non-visual understanding of the data.	
3.	**Data Cleaning and Pre-processing**	13-17
	- Approach used for identifying and treating missing values and outlier treatment (and why)	
	- Need for variable transformation (if any)	
	- Variables removed or added and why (if any)	
4.	**Model building**	17-24
	- Clear on why was a particular model(s) chosen.	
	- Effort to improve model performance.	
5.	**Model validation**	25,26
	- How was the model validated? Just accuracy, or anything else too?	
6.	**Final interpretation / recommendation**	27-30
	- Detailed recommendations for the management/client based on the analysis done.	

LIST OF TABLES

Table No.	Title	Page No.
2.1	Descriptive statistics for categorical variables	6
2.2	Descriptive statistics of continuous variables	7
2.3	Inference of different univariate plots for continuous variables	8
3.1	Missing values for different variables before and after imputation	13
3.2	VIF values for different variables	16
4.1	WIN PER CENT OF ENTIRE DATASET	22
4.2	WIN PER CENT OF ODI FORMAT CLUSTER	23
4.3	WIN PER CENT OF T20 FORMAT CLUSTER	23
4.4	WIN PER CENT OF TEST FORMAT CLUSTER	24
5.1	PERFORMANCE METRICS OF DIFFERENT MODELS BUILD FOR OVERALL DATA	25
5.2	PERFORMANCE METRICS OF DIFFERENT MODELS BUILD FOR DIFFERENT CLUSTERS	26
5.3	PERFORMANCE METRICS AFTER MODEL TUNING FOR DIFFERENT CLUSTERS	26
6.1	VARIABLE IMPORTANCE FOR TEST MATCH CLUSTER	27
6.2	VARIABLE IMPORTANCE FOR T20 MATCH CLUSTER	28
6.3	VARIABLE IMPORTANCE FOR ODI MATCH CLUSTER	29

LIST OF FIGURES

Figure No.	Title	Page No.
2.1	Cat plot for categorical variables	8
2.2	Count plots for the each of the independent categorical variables with Result	9
2.3	Scatter plot for each continuous variable with the categorical variable 'Result'	10
3.1	Outliers in the dataset	14
3.2	Box plot shown after Outlier Treatment	15

LIST OF ANNEXURES

Annexure No.	LIST	Page No.
Annexure 1	Distribution plot and box plots for all the continuous variables of sports data	31,32
Annexure 2	Pair plot for all continuous variables of sports dataset	33
Annexure 3	Heat map depicting correlation between continuous variables of sports dataset	34
Annexure 3a	Heat map for encoded and scaled variables	35

Annexure 4	OUTPUT FOR MODEL DECISION TREE CLASSIFIER (DTC) FOR OVERALL DATA	36
Annexure 5	OUTPUT FOR MODEL RANDOM FOREST CLASSIFIER (RFC) FOR OVERALL DATA	37
Annexure 6	OUTPUT FOR MODEL NEURAL NETWORK CLASSIFIER (NNC) FOR OVERALL DATA	38
Annexure 7	OUTPUT FOR MODEL LOGISTIC REGRESSION MODEL (LRM) FOR OVERALL DATA	39
Annexure 8	OUTPUT FOR MODEL LINEAR DISCRIMINANT ANALYSIS (LDA) FOR OVERALL DATA	40
Annexure 9	OUTPUT FOR MODEL NAÏVE BAYES WITH SMOTE (NBS) FOR OVERALL DATA	41
Annexure 10	OUTPUT FOR MODEL KNN WITH SMOTE (KNNS) FOR OVERALL DATA	42
Annexure 11	OUTPUT FOR TUNED MODEL ENSEMBLE RFC FOR OVERALL DATA	43
Annexure 12	OUTPUT FOR TUNED MODEL BAGGING FOR OVERALL DATA	44
Annexure 13	OUTPUT FOR TUNED MODEL GRADIENT BOOSTING FOR OVERALL DATA	45
Annexure 14	OUTPUT FOR TUNED MODEL ADA BOOST FOR OVERALL DATA	46
Annexure 15	OUTPUT FOR MODEL RANDOM FOREST CLASSIFIER (RFC) FOR ODI DATA	47
Annexure 16	OUTPUT FOR MODEL NEURAL NETWORK CLASSIFIER (NNC) FOR ODI DATA	48
Annexure 17	OUTPUT FOR TUNED MODEL ENSEMBLE RFC FOR ODI DATA	49
Annexure 18	OUTPUT FOR TUNED MODEL BAGGING FOR ODI DATA	50
Annexure 19	OUTPUT FOR MODEL KNN WITH SMOTE (KNNS) FOR T20 DATA	51
Annexure 20	OUTPUT FOR MODEL RANDOM FOREST CLASSIFIER (RFC) FOR T20 DATA	52
Annexure 21	OUTPUT FOR MODEL NEURAL NETWORK CLASSIFIER (NNC) FOR T20DATA	53
Annexure 22	OUTPUT FOR TUNED MODEL ENSEMBLE RFC FOR T20 DATA	54
Annexure 23	OUTPUT FOR TUNED MODEL BAGGING FOR T20 DATA	55
Annexure 24	OUTPUT FOR MODEL LOGISTIC REGRESSION MODEL (LRM) FOR TEST DATA	56
Annexure 25	OUTPUT FOR MODEL NEURAL NETWORK CLASSIFIER (NNC) FOR TEST DATA	57
Annexure 26	OUTPUT FOR TUNED MODEL GRADIENT BOOSTING FOR TEST DATA	58
Annexure 27	OUTPUT FOR TUNED MODEL ADA BOOST FOR TEST DATA	59

1. Introduction

- Brief introduction about the problem statement and the need of solving it.

1. INTRODUCTION ABOUT PROBLEM STATEMENT AND NEED OF SOLVING IT

External analytics consulting firm was hired by BCCI for data analytics of Sports data. The sports data set with the information of games played previously were provided to develop strategies for winning cricket game by Indian cricket team.

BUSINESS INSIGHT FOR CRICKET WIN PREDICTION

Cricket is liked by most of the people across the globe and has a huge finance involved in organising them. In turn, the BCCI should also earn profits by organising them. The BCCI with the winning strategies and recommendations provided by the analysed data of the past games played, can be used to train Indian team so that the Indian team can have a better win and in turn have profits for the BCCI from the tickets sold to audience, advertising companies, promotions for products, parking payments etc.

There is rapid expansion of sports data analytics every year. Technological advances in development of different machine learning models, evaluating models have gained insight to formulate strategies for sports so that chances of win can be increased or achieved. Based on the models' certain strategies can be thought of which can be followed so that chance of winning for Indian cricket team is favored. Machine learning algorithms can predict easily using the classification function for various attributes in the dataset. Certain descriptive modelling involving machine learning models of CART, Random Forest, Neural network classifier, predictive modelling involving logistic regression, LDA, KNN and Naïve Bayes with SMOTE are best suited to predict winning strategy. Overfitting, underfitting, model tuning can be done for improving the performance of models. Model evaluation for all the models based on the performance metrics to identify the best model showing highest win accuracy.

As a data analyst of external analytics consulting firm the sports dataset is to be analyzed with the primary **business insight of developing strategies for various formats of cricket by statistical analysis and building various models and tuning if required and evaluating the model which is best for that format and proposing the strategies to win for Indian team.**

Recently, the T20 format is getting major attention due to its shortest format and huge score within 20 overs.

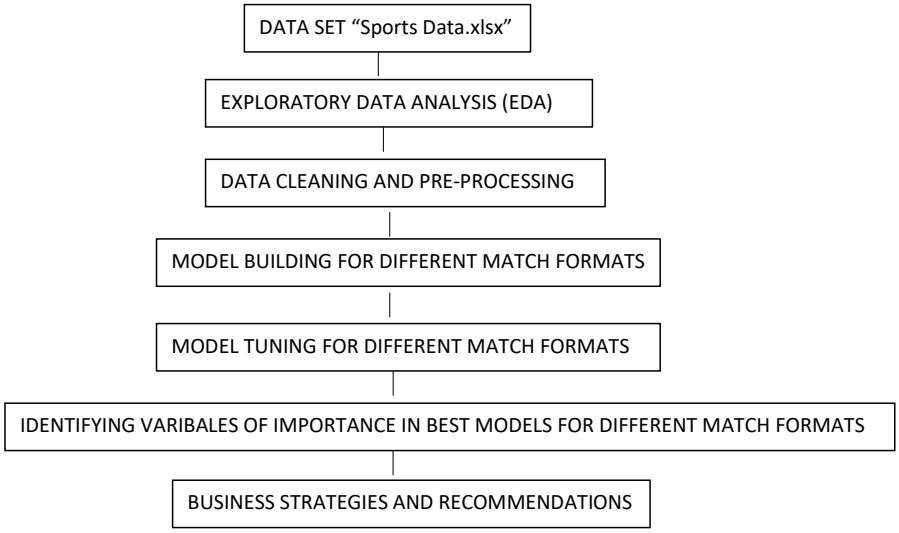
Objective:

To propose different win strategies for Indian cricket to be played in various formats *i.e.*, Test (1 strategy), ODI (2 strategies), T20 (2 strategies) for INDIAN CRICKET TEAM WIN. The strategies are to be provided for matches played in India against Sri Lanka (ODI), Australia(T20) during winter season as day and night matches and for test match to be played against England in England as day match in rainy season.

Data Set given: 'Sports Data.xlsx'

Target Variable: Result

METHODOLGY FOR CRICKET WIN PREDICTION



2. EXPLORATORY DATA ANALYSIS AND BUSINESS IMPLICATIONS

Insight: Visualising the dataset in python jupyter notebook and understanding the attributes or varibales and their distribution, relationship between the variables and their importance.
Steps followed for EDA:

- Import necessary libraries of numpy, pandas, matplotlib, seaborn in the python jupyter notebook.
- Read the dataset "Sports Data.xlsx". The dataset comprises of data dictionary utilised for sports data in sheet 1 and Sports data in sheet 2.
- Glimpse of the dataset i.e., both head and tail of the dataset.
- Shape of the dataset.
- Structure of the dataset describing all variables information.
- Data types of the dataset.
- Visualising descriptive statistics as well as categorical and numerical column variables.
- Renaming of the column variables if required.
- Value counts for the categorical variables for identifying uniques.
- Replacing or converting 20-20 into T20 and bat with batting for Match_format and First_selection variables.
- Visualising the dataset after conversions.

- Exploratory data analysis comprising of univariate, bivariate and multivariate distributions with their business implications

INFERENCES WITH OUTPUT:

DATA REPORT:

Data was collected for 2930 cricket game played in the past which were assigned different Game numbers. For each game played a win or loss result was given which will be considered as a target variable or dependent variable. About 21 columns of independent variables along with game numbers were given in the dataset.

Visual inspection of data (rows, columns, descriptive details)

- Data shape: 23 column variables and 2390 rows.
- The shape of the dataset is 23 column variables and 2390 rows.
- There are no duplicates in the dataset.
- The information or data structure depicts 10 object, 9 float and 4 integer types
- Making different list for categorical and numerical columns

The Categorical variables comprise of Game_number; Result; Match_light_type; Match_format; First_selection; Opponent; Season; Offshore; Players_scored_zero; Player_highest_wicket.

The Numerical or continuous variables comprise of Avg_team_Age ; Bowlers_in_team; Wicket_keeper_in_team; All_rounder_in_team ; Audience_number; Max_run_scored_1over; Max_wicket_taken_1over; Extra_bowls_bowled; Min_run_given_1over; Min_run_scored_1over; Max_run_given_1over; extra_bowls_opponent; player_highest_run.

The descriptive statistics of categorical variables is shown in Table 2.1

The descriptive statistics of continuous variables is shown in Table 2.2

Based on the above data,

Match_format and First_selection has 4 unique formats of which T20 and 20-20 are similar. So, 20-20 may be converted to T20 format. Similarly, First_selection has 3 unique formats, of which bat and batting are similar. So, bat converted to batting.

Players_scored_zero and player_highest_wicket look like numeric and the value counts show certain numeric written in words which has to modified or converted into numeric.

Value counts are done to visualise the changes made.

Table 2.1: Descriptive statistics for categorical variables

	count	unique	top	freq
Game_number	2930	2930	Game_2013	1
Result	2930	2	Win	2457
Match_light_type	2878	3	Day	2041
Match_format	2860	4	ODI	1865
First_selection	2871	3	Bowling	1722
Opponent	2894	9	South Africa	640
Season	2868	3	Rainy	1309
Offshore	2866	2	No	2057
Players_scored_zero	2930	5	3	1730
player_highest_wicket	2930	6	1	1084

Table 2.2: Descriptive statistics of continuous variables

	count	mean	std	min	25%	50%	75%	max
Avg_team_Age	2833	29.24285	2.26423	12	30	30	30	70
Bowlers_in_team	2848	2.913624	1.023907	1	2	3	4	5
Wicket_keeper_in_team	2930	1	0	1	1	1	1	1
All_rounder_in_team	2890	2.722491	1.092699	1	2	3	4	4
Audience_number	2849	46267.96	48599.58	7063	20363	34349	57876	1399930
Max_run_scored_1over	2902	15.19986	3.66101	11	12	14	18	25
Max_wicket_taken_1over	2930	2.713993	1.080623	1	2	3	4	4
Extra_bowls_bowled	2901	11.25267	7.780829	0	6	10	15	40
Min_run_given_1over	2930	1.95256	1.678332	0	0	2	3	6
Min_run_scored_1over	2903	2.762659	0.705759	1	2	3	3	4
Max_run_given_1over	2896	8.669199	5.003525	6	6	6	9.25	40
extra_bowls_opponent	2930	4.229693	3.626108	0	2	3	7	18
player_highest_run	2902	65.88939	20.33161	30	48	66	84	100

Business Insights:

- Unique game numbers are given for each game
- Result is showing more data for games won. Most game data is for ODI matches
- India has played many games with South Africa.
- India selected bowling as first selection.
- The mean and median showed much variation depicting outliers for few variables
- The counts are showing that missing values are present in the dataset.

Univariate analysis (distribution and spread for every continuous attribute, distribution of data in categories for categorical ones)

Approach: Distibution plots and box plots are drawn for all the continuous variables. For categorical variables cat plots are given. Box plots depict the presence or absence of outliers for that variable in the dataset.

Cat plot for categorical variables are shown in Figure 2.1

Inference:

The distribution plot and box plots for all the continuous variables are shown in Annexure 1. The inference is shown in Table 2.3.

Table 2.3: Inference of different univariate plots for continuous variables

DISTRIBUTION	VARIBALES
Normal distribution	Bowlers_in_team, Wicket_keeper_in_team, player_highest_run
Left skewed	Avg_team_Age, All_rounder_in_team, Max_wicket_taken_1over, Min_run_scored_1over
Right skewed	Audience_number, Extra_bowls_bowled, Min_run_given_1over, Max_run_scored_1over, Max_run_given_1over, extra_bowls_opponent

Inference : The cat plots for categorical variables show highest count for wins, day matches, South Africa, ODI matches, bowling for first selection, rainy season.

b) Bivariate analysis (relationship between different variables, correlations):

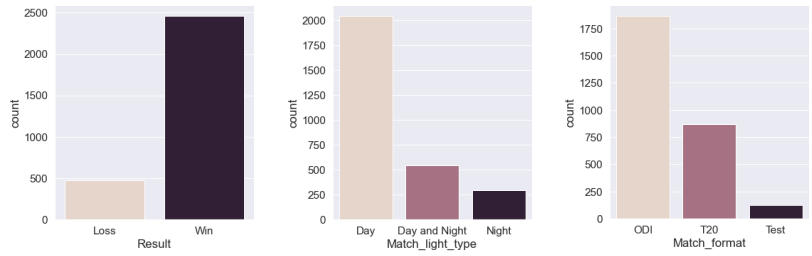
Approach: Scatter plots are drawn for each of continuous variables with the target variable *i.e* result. A scatter plot is a visual representation of the degree of correlation between any two columns. The pair plot function in seaborn makes it very easy to generate joint scatter plots for all the columns in the data. A scatter plot can also be plotted for two individual columns. Pair plots are also drawn for continuous variables which shows the density of that variable with respect to other variables. The correlation coefficients for all the continuous variables are depicted as heat maps and shown in Annexure 3.

Inference:

The scatter plot between each continuous variables and dependent variable 'result' are shown in Figure 2.2. Pairplot shows the correlation among the variables and for the present dataset is shown in Annexure 2. Heat map was generated for all the continuous variables which is shown in Annexure 3. Highest correlation is observed between Max_run_given_1over and Extra_bowls_bowled (0.62), followed by Extra_bowls_bowled and Audience_number (0.57). This is followed by extra_bowls_opponent and Extra_bowls_bowled (0.46), Max_run_given_1over and Audience_number (0.44), extra_bowls_opponent and Audience_number (0.31), Avg_team_age and Bowlers_in_team (0.24).

The scatter plots are shown in Figure 2.3 which describes the density of win or loss with respires to the variable as shown in plots.

Figure 2.1: Cat plot for categorical variables



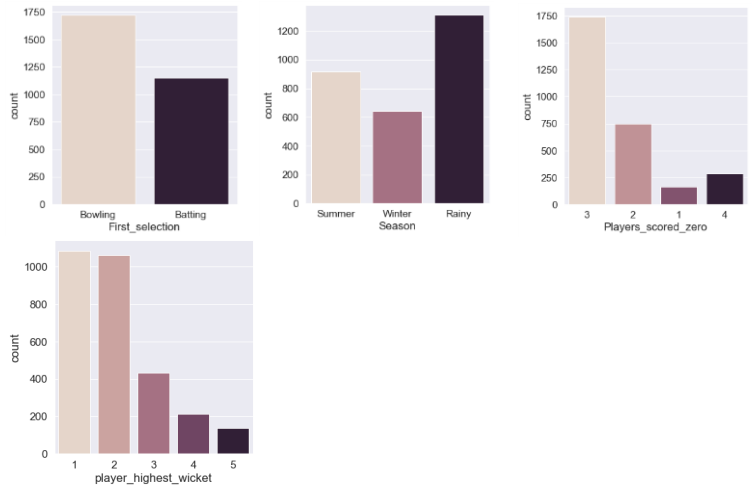
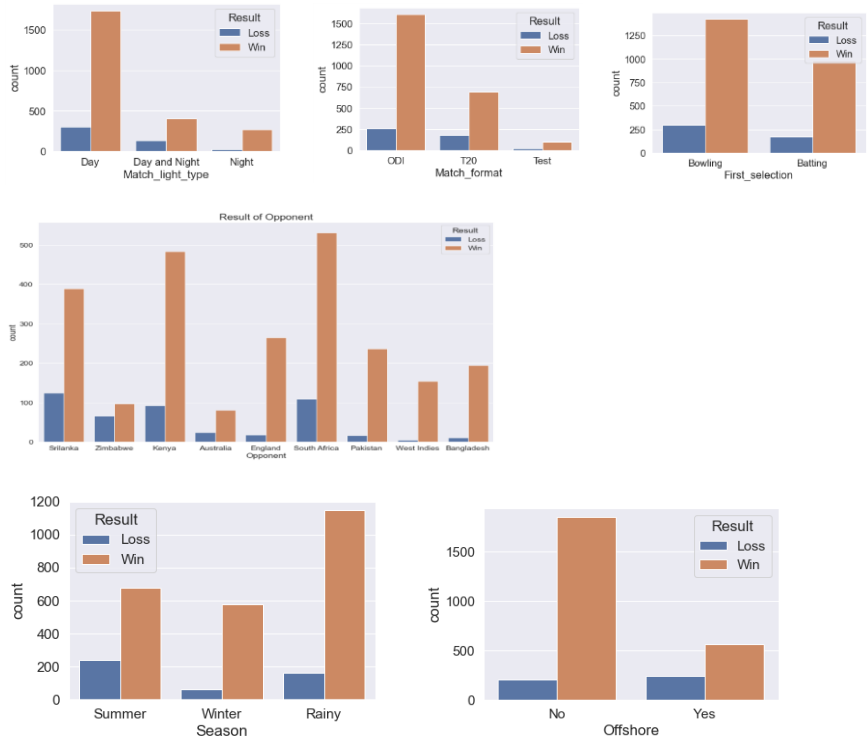


Figure 2.2: Count plots for the each of the independent categorical variables with Result



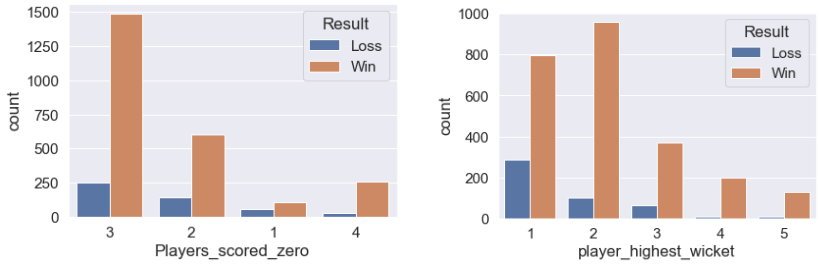
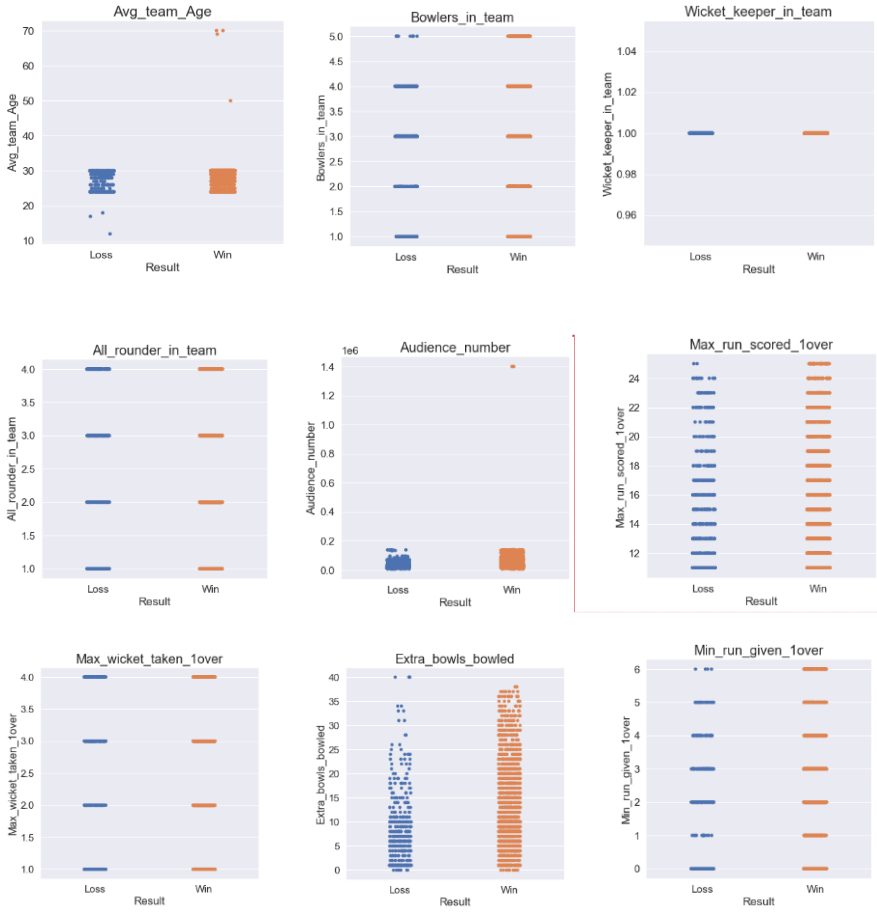
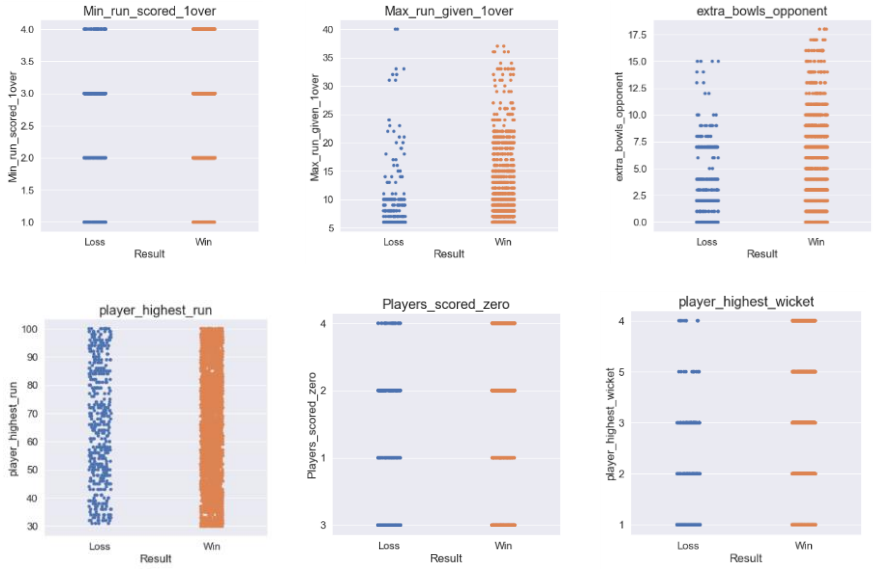


Figure 2.3: Scatter plot for each continuous variable with the categorical variable 'Result'



Commented [DHD1]:



BUSINESS IMPLICATIONS FOR EDA:
Business insights for data distribution is shown in Box 1.
Business insights for relation between variables (scatter plot, heat map and pairplot) shown in Box 3.
Business insights for relationship between categorical variables are shown in Box 3.

BOX 1: BUSINESS INSIGHT BASED ON DISTRIBUTION OF DATA (UNIVARIATE ANALYSIS)

- Most observations are around 30.
- Most games have 3 bowlers.
- Only one wicket keeper in team.
- Most games have 3 or 4 all rounders
- Most games scored 10-15 runs in 1 over by the team
- Most games show 3 wickets taken in 1 over by team
- Most games show 10-12
- Most games show 0 runs
- Minimum 3 runs are scored in 1 over
- <10 runs are given 1 over
- Most games show 2 extra bowls by opponent
- 100 runs are scored in many games by team

BOX 2: BUSINESS INSIGHT BASED ON CORRELATION BETWEEN VARIABLES OF DATASET (BIIVARIATE AND MULTIVARIATE ANALYSIS)

- Most of the matches whether won are lost is centred around 20-30 years average age

- The percentage of winning increases from 1 to 3 bowlers and decreases slightly after3 upto 5 bowlers
- The total no. of wicket keepers are constant (only 1) in all the games (either loss or win. So, it has no significant effect on deciding the result.
- The percentage of winning increases from 1 to 2 all-rounders and decreases slightly after 2 till 4 all-rounders.
- High audience provide victory to the team. So, it is significant variable as the audience support the team during the game.
- If the no.of runs scored in an over is high, it makes the match victory. The losses at high runs/over may be decided by the other factors. Here, 24 runs/match has higher victory compared to loss
- Almost similar distribution for loss and win. May be insignificant in determining the result
- If extras are more considering India, then India may lose the match. If extra bowls are bowled by opponent, India may win the match.
- If runs are given by India (r.p.o for opponent=6 and =1), it easy for India to chase frequently. But in certain matches, r.p.o of the opponent=6 and =1, loss was also observed. Between 1< min_runs_given_1over(r.p.o opponent)<6 and at min_runs_given_1over =0, the distribution is similar for win and loss
- Almost similar distribution for loss and win. May be insignificant in determining the result
- India lost even after securing runs>40 several times which need to be considered. The matches won have highest max_runs_given1over at 36-37 and India has won in them. For lower runs below 30 the distribution is highly favourable for winning or result =1.
- The greater the extra bowls bowled by opponent, more easily India can secure the victory
- More victory as more players score higher runs is observed.
- Almost similar distribution for loss and win. May be insignificant in determining the result
- If more wickets (4-5) are taken by a bowler, India won. Rarely, India lost taking 4-5 wickets. For <4 wickets the win and loss distribution is similar.

BOX 3: BUSINESS INSIGHTS FOR RELATIONSHIP BETWEEN CATEGORICAL VARIBALES

- Day matches has highest win
- ODI matches has highest win
- First selection of bowling showed highest wins
- India won most matches played against South Africa followed by Kenya. It is having minimum loses with West Indies
- Maximum matches were won in rainy season
- Maximum matches are won when played in India
- Even 3 players were out for 0, India still managed to win the matches.

- India won the matches when bowlers took 4 and 5wickets

3. DATA CLEANING AND PRE-PROCESSING

- Approach used for identifying and treating missing values and outlier treatment (and why)
- Need for variable transformation (if any)
- Variables removed or added and why (if any)

Steps involved in data cleaning and pre-processing:

a) Missing value treatment
b) Outlier treatment
c) Variable transformation
d) Removal of unwanted variables.

a) Missing Value treatment (if applicable):

Approach: Check for the missing values in the dataset.
Missing values treatment is done **by imputing median values for missing data of continuous variables and imputing mode values for categorical data**.
After imputation checking missing values.

Inference:

There are 789 missing values in the dataset.
After imputing missing values there are no missing values in the dataset. The missing values for different variables before and after imputation are shown in Table 3.1

Table 3.1: Missing values for different variables before and after imputation

Missing values (Before imputation)	Missing values (After imputation)
Result 0	Result 0
Avg_team_Age 97	Avg_team_Age 0
Match_light_type 52	Match_light_type 0
Match_format 70	Match_format 0
Bowlers_in_team 82	Bowlers_in_team 0
All_rounder_in_team 40	All_rounder_in_team 0
First_selection 59	First_selection 0
Opponent 36	Opponent 0
Season 62	Season 0
Audience_number 81	Audience_number 0
Offshore 64	Offshore 0
Max_run_scored_1over 28	Max_run_scored_1over 0
Max_wicket_taken_1over 0	Max_wicket_taken_1over 0
Extra_bowls_bowled 29	Extra_bowls_bowled 0
Min_run_given_1over 0	Min_run_given_1over 0
Min_run_scored_1over 27	Min_run_scored_1over 0
Max_run_given_1over 34	Max_run_given_1over 0
extra_bowls_opponent 0	extra_bowls_opponent 0
player_highest_run 28	player_highest_run 0

Players_scored_zero 0	Players_scored_zero 0
player_highest_wicket 0	player_highest_wicket 0
Total missing values: 789	Total missing values: zero

b) Outlier treatment (if required):
Approach: Check for outliers in the dataset by the box plot for continuous variables before and after removing outliers. Outlier treatment is done by IQR.
Inference:
There are outliers present in 5 numerical columns, but considering that outliers for Extra_bowls_bowled, extra_bowls_opponent and Max_run_given_1over must be considered as it may happen in future matches also. The avg_team_age and audiecne_number can be removed from outliers as it does not decide the win and lose condition significantly. However, Outliers are removed from all the five variables mentioned above.
The box plot before and after outlier treatment are shown in Figure 3.1 and Figure 3.2

Figure 3.1: Outliers in the dataset

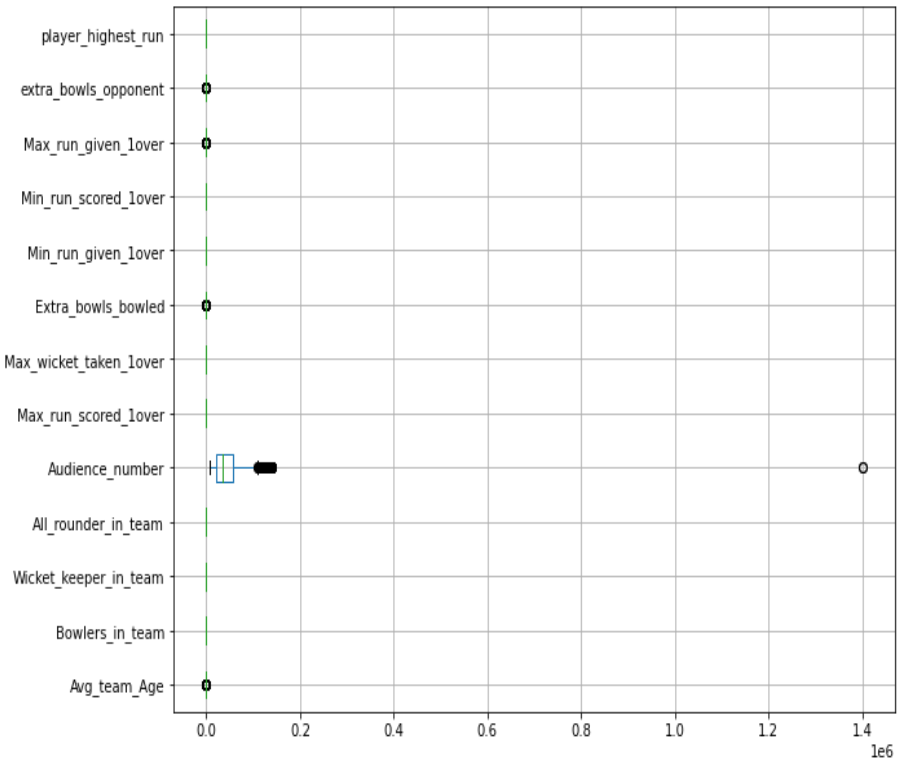
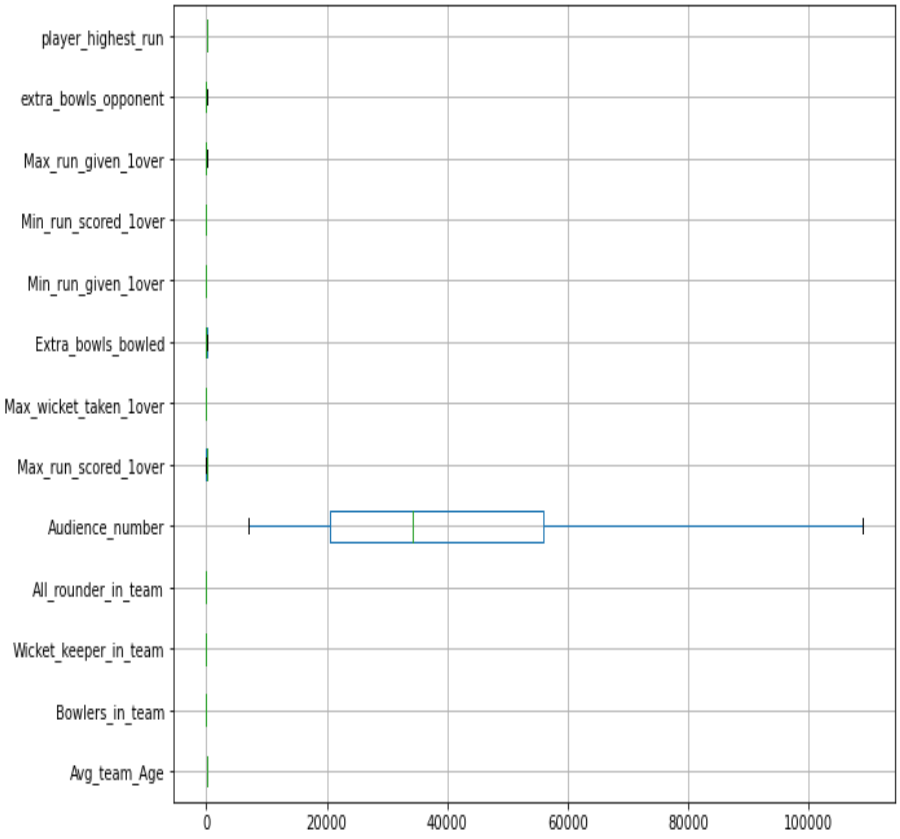


Figure 3.2 : Box plot shown after Outlier Treatment



c) Variable transformation (if applicable)

Approach:

- For categorical varibales , variable transformation is done by Label encoding.
- For continuous variables, scaling using standard scaler is followed.

Import Label encoder from sklearn.preprocessing and encode all the categorical variables.
See the value counts after label encoding as mentioned in jupyter notebook.
Transformation on the data using Standard Scaler
Scaling of the dataset is done by importing Standard Scaler from sklearn.preprocessing
Returns the z-score of every attribute.
Fit and transform.

For the encoded and scaled data visualise the correlations through heat maps in python jupyter notebook.

Inference:

The heat map generated for the scaled data is shown in Annexure-3a.
High correlation exists between
Player_highest_wicket and Audience_number (0.93)
Player_highest_wicket and Extra_bowls_bowled (0.78)
Extra_bowls_bowled and Audience_number (0.75)
Max_run_given_1over and Extra_bowls_bowled (0.57)
No new variables are required, hence not added any variable.

d) Removal of unwanted variables (if applicable)

Game_number have unique ids for each game and are dropped .

Table 3.2: VIF values for different variables

	feature	VIF
0	Result	1.184994
1	Avg_team_Age	0
2	Match_light_type	1.014881
3	Match_format	1.545418
4	Bowlers_in_team	1.028235
5	All_rounder_in_team	1.030228
6	Wicket_keeper_in_team	0
7	First_selection	1.008614
8	Opponent	1.165731
9	Season	1.010222
10	Audience_number	**7.097595**
11	Offshore	1.090127
12	Max_run_scored_1over	1.008721
13	Max_wicket_taken_1over	1.016987
14	Extra_bowls_bowled	3.084353
15	Min_run_given_1over	1.616754
16	Min_run_scored_1over	1.019664
17	Max_run_given_1over	3.036599
18	extra_bowls_opponent	2.663806
19	player_highest_run	1.010529
20	Players_scored_zero	1.029654
21	player_highest_wicket	**8.096399**

INFERENCE:

Two variables “Audience_number”, “player_highest_wicket” are showing VIF above 5.0 and can be dropped.

DATA BALNACED/UNBALANCED?
Approach:
Copy all the predictor variables into X dataframe.
Copy target into the y dataframe
See the value counts for the target.
The data proportion is shown in Box 4.

Box 4: DATA PROPORTION WITH RESPECT TO TARGET VARIBALE 'RESULT'
The data of the target variable is unbalanced as it depicts target variable=1 for 83.8% of the cases and 16.2% for target variable=0. The data for the target variable=1 can be randomly considered upto 60 percent of the entire dataset to make it balanced data. The model predicts quite good for train and test data considering 60-40 split rather than 80-20 split of the tarrget variable.

Clustering:

Any business insights using clustering (if applicable)

K-means clustering for scaled data:
Appraoch: From sklearn.cluster import KMeans
from sklearn.metrics import silhouette_samples, silhouette_score
perform k-means clustering based on match format as shown in jupyternotebook.
Inference:
Optimal clusters are identified based on point plot as shown in figure

Cluster values based on match format for ODI, T20 and Test matches

```
0    1935
1     870
2     125
Name: cluster, dtype: int64
```

```
0    1935
1     870
2     125
Name: Match_format, dtype: int64
```

Business insights:
The clustered matches will further be considered for evaluating the unique strategies and insights based on match format.

Train-test-split for the dataset:
Splitting into train and test set with 70: 30 and following random state=2 . Checking the dimensions of the training and test data.

4 MODEL BUILDING:
4a. Clear on why was a particular model(s) chosen:
Business Insight:
Various descriptive/predictive/prescriptive models enable to identify the importance of the variables which are important in designing the effective win strategy.

- Descriptive analytic models involve data mining which allows to condense big data into smaller and more useful nugget of information.
- Predictive analytic models can forecast what might happen in the future as they are all probabilistic in nature. It uses statistical and machine learning algorithms.

- Prescriptive analytic models are the next step of predictive analytics and advises on the possible outcomes and results in actions that are likely to maximise business metrics.

Approach used:

- Model building and tuning is done after EDA, data cleaning and pre-processing, VIF and dropping the variables having VIF value>5; proportions of the data set and splitting into train and test set with 70: 30 and following random state=2. Checking the dimensions of the training and test data.

MODEL BUILDING

Steps involved in model building for different models"

MODEL 1: DECISION TREE CLASSIFIER

Approach:

- Building a decision tree classifier:
- Initialise a Decision Tree Classifier with criterion = 'gini' and random state= 2.
- Fit the model.
- from sklearn import tree
- The above code will save a .dot file in your working directory.
  WebGraphviz is Graphviz in the Browser. Copy paste the contents of the file into the link below to get the visualization http://webgraphviz.com/
- Variable Importance
- Predicting Test Data
- Regularising the Decision Tree
- Adding Tuning Parameters
- Generating New Tree
- Variable Importance
- Predicting on Training and Test dataset
- Getting the Predicted Classes
- Getting the Predicted Probabilities

MODEL 2: RANDOM FOREST CLASSIFIER (RFC)

- **Initialise random forest classifier**
- **Grid Search for finding out the optimal values for the hyper parameters like max_depth, n_estimators, min_samples_leaf, min_samples_split etc.**
- **Fit the model on train data.**
- Variable Importance
- Predicting on Training dataset
- Getting the Predicted Classes
- Getting the Predicted Probabilities

MODEL 3: NEURAL NETWORK CLASSIFIER

- Building Neural Network Model for scaled data
- Grid Search for finding out the optimal values for the hyper parameters like solver, estimator, hidden layer size, etc.
- Fit the model on the training data
- Predicting training data

- Predicting on Training dataset
- Getting the Predicted Classes
- Getting the Predicted Probabilities

MODEL 4: LOGISTIC REGRESSION

- Fit the Logistic Regression model using hyperparameters like solver, verbose, n_jobs etc.
- Predict on training and test data
- Getting the predicted classes
- Getting the Predicted Probabilities

MODEL 5: LINEAR DISCRIMINANT ANALYSIS

- Fit the LDA model on the train data set.
- Predict on training and test data
- Getting the predicted classes
- Getting the Predicted Probabilities
- Accuracy score of train and test dataset.

MODEL 6: NAÏVE BAYES WITH SMOTE

- Import module GaussianNB.
- Fit the GaussianNBmodel on the train data set.
- Predict the model score on training and test data.

MODEL 7: KNN WITH SMOTE

- Import module KNeighborsClassifier.
- Fit the KNeighborsClassifier on the train data set.
- Predict the model score for training and test data.

MODEL BUILDING FOR VARIOUS CLUSTERS OF MATCH FORMAT

MATCH FORMAT	**MODELS BUILD**
ODI MATCH	RANDOM FOREST CLASSIFIER (RFC)
	NEURAL NETWORK CLASSIFIER (NNC)
T20 MATCH	RANDOM FOREST CLASSIFIER (RFC)
	NEURAL NETWORK CLASSIFIER (NNC)
	KNN WITH SMOTE (KMMS)
TEST MATCH	LOGISTIC REGRESSION (LOG REG)
	NEURAL NETWORK CLASSIFIER (NNC)

The steps followed as mentioned above were considered and respective models were built for the cluster data of ODI, T20 and TEST instead of overall dataset.

4b. EFFORT TO IMPROVE THE MODELS- MODEL TUNING:
Model Tuning and business implication
INSIGHT:

To find out win prediction in different types of match format like test, ODI and T20 matches, and provide business insights to BCCI which help to design the winning strategies for Indian team based on the best model.
For the sports dataset, Ensemble Random Forest, Bagging, Gradient boosting and AdaBoost are tuned on models built for the overall data comprising of all match formats without making clusters.
After having 3 different types of clusters based on match format as depicted by k-means clustering, the better models were built for the three different types of clusters. Later these clustered models are further tuned to improve the performance metrics using Ensemble Random Forest and Bagging for ODI and T20 clusters, Gradient boosting and AdaBoost for test match cluster.
Ensembles are machine learning methods for combining predictions from multiple separate models.
Idea of Ensemble modelling:
For each model, the prediction for test data is performed individually and later all the predictions are summed up to generate the final evaluation (prediction).
Approach for different ensemble models:

ENSEMBLE MODEL 1: RANDOM FOREST ENSEMBLE

- Import RandomForestClassifier form sklearn.ensemble.
- Select the hyperparameters n_estimators = 100 and random_state =2
- Then fit the model for the entire train data.
- The model score for both train and test data are calculated.

ENSEMBLE MODEL 2: BAGGING

- **Import random forest classifier and Bagging classifier from sklearn.ensemble**
- **By choosing appropriate hyperparameters like base_estimator = random forest model, random_state etc, we tune the model.**
- Then fit the model for the entire train data.
- The model score for both train and test data are calculated.

ENSEMBLE MODEL 3: GRADIENT BOOSTING

Building Gradient Boosting Model:

- **Import Gradient Boosting Classifier from sklearn.ensemble**
- **By choosing appropriate hyperparameters like random_state=2 we tune the model.**
- Then fit the model for the entire train data.
- The model score for both train and test data are calculated.

ENSEMBLE MODEL 4: ADA BOOST

Modelling using Adaboost Model:

- **Import AdaBoostClassifier from sklearn.ensemble**
- **By choosing appropriate hyperparameters like n_parameters =100, random_state=2 we tune the model.**

Then fit the model for the entire train data. The model score for both train and test data are calculated.

MODEL TUNING FOR VARIOUS CLUSTERS OF MATCH FORMAT

MATCH FORMAT	MODELS BUILD
ODI MATCH	ENSEMBLE RANDOM FOREST (ERFC)
	BAGGING
T20 MATCH	ENSEMBLE RANDOM FOREST (ERFC)
	BAGGING
TEST MATCH	GRADIENT BOOSTING
	ADA BOOST

INFERENCE AND OUTPUT:

The individual models were interpreted based on their performance metrics in terms of precision, recall, F1 score, Accuracy of the train and test data. The Area Under Curve (AUC) as well as Receiver Characteristic Operator (ROC) of the train and test were plotted for the respective models.
Model Evaluation:
Done by performance metrics

- Classification metrics for the train and test data
    - **Confusion Matrix for the train and test data**
    - **Measuring AUC and ROC Curve for the train and test data.**

For every model, confusion matrix is given as:

		ACTUAL VALUES	
		POSITIVE	NEGATIVE
PREDICTED VALUES	POSITIVE	TP	FP
	NEGATIVE	FN	TN

The individual models after tuning with ensemble models were interpreted based on their performance metrics in terms of precision, recall, F1 score, Accuracy of the train and test data. The Area Under Curve (AUC) as well as Receiver Characteristic Operator (ROC) of the train and test were plotted for the respective models.

- The output for various models of DTC, RFC, NNC, LRM, LDA, NBS, and KNNS for the entire dataset are shown in Annexure 4-10.
- The output after model tuning using ERF, Bagging, Gradient Boosting and Ada boost for the entire dataset is shown in Annexure 11-14.
- The output for various models of RFC, NNC for the ODI match cluster is shown in Annexure 15,16.
- The output after model tuning using ERF, bagging for the ODI match cluster is shown in Annexure 17,18.
- The output for various models of KNNS, RFC, NNC for the T20 match cluster is shown in Annexure 19-21.

- The output after model tuning using ERF, bagging for the T20 match cluster is shown in Annexure 22,23.
- The output for various models of LRM, NNC for the Test match cluster is shown in Annexure 24,25.
- The output after model tuning using Gradient boosting and Ada boost for the ODI match cluster is shown in Annexure 26,27.

Interpretation of the model(s)
Based on the confusion matrix, we evaluate the loss and opportunity loss to predict the total loss. Later, the total loss is subtracted from 1 to obtain the total win percentage. The models with high win percentage is to be considered.

From the above table, the following calculations are done:
Accuracy: Out of all cases how much is correctly predicted given by:
(TP + TN)/(TP + TN + FP +FN)
Recall is TP rate i.e., TP/(TP + FN)
Precision is specificity i.e., TN rate which is TN/(TN + FP)
F1-SCORE measures precision and recall at the same and is
2*Recall*Precision/(Recall+Precision)
Total win = 1-Total loss. Total loss is (0.95*loss+0.05*Opp.loss), where loss is FN/(FN + TN) and opportunity loss is FP/(FP + TP).

TOTAL WIN PERCENT IS CALCULATED BY TOTAL WIN*100

The models with high win percentage is to be considered.
Table 4.1: WIN PER CENT OF ENTIRE DATASET

MODEL BUILDING		WIN PERCENT
DECISION TREE CLASSIFIER	TRAIN SET	77.42
DECISION TREE CLASSIFIER	TEST SET	75.19
RANDOM FOREST CLASSIFIER	TRAIN SET	86.38
RANDOM FOREST CLASSIFIER	TEST SET	85.34
NEURAL NETWORK CLASSIFIER	TRAIN SET	84.08
NEURAL NETWORK CLASSIFIER	TEST SET	83.49
LOGISTIC REGRESSION	TRAIN SET	76.23
LOGISTIC REGRESSION	TEST SET	74.60
LDA	TRAIN SET	76.55
LDA	TEST SET	74.93
NAÏVE BAYES WITH SMOTE	TRAIN SET	67.99

NAÏVE BAYES WITH SMOTE	TEST SET	66.47
KNN WITH SMOTE	TRAIN SET	75.63
KNN WITH SMOTE	TEST SET	74.93
MODEL TUNING		
ERF	TRAIN SET	100.00
ERF	TEST SET	97.29
BAGGING	TRAIN SET	99.99
BAGGING	TEST SET	97.08
GRADIENT BOOSTING	TRAIN SET	95.66
GRADIENT BOOSTING	TEST SET	92.19
ADABOOST	TRAIN SET	90.26
ADABOOST	TEST SET	87.46

Table 4.2: WIN PER CENT OF ODI FORMAT CLUSTER

MODEL BUILDING		WIN PERCENT
RANDOM FOREST CLASSIFIER	TRAIN SET	86.83
RANDOM FOREST CLASSIFIER	TEST SET	83.90
NEURAL NETWORK CLASSIFIER	TRAIN SET	79.76
NEURAL NETWORK CLASSIFIER	TEST SET	77.88
MODEL TUNING		
ERF	TRAIN SET	100.00
ERF	TEST SET	96.91
BAGGING	TRAIN SET	99.90
BAGGING	TEST SET	96.64

Table 4.3: WIN PER CENT OF T20 FORMAT CLUSTER

MODEL BUILDING		WIN PERCENT
RANDOM FOREST CLASSIFIER	TRAIN SET	78.77
RANDOM FOREST CLASSIFIER	TEST SET	73.08
NEURAL NETWORK CLASSIFIER	TRAIN SET	60.25

NEURAL NETWORK CLASSIFIER	**TEST SET**	**52.83**
KNN WITH SMOTE	**TRAIN SET**	**79.76**
KNN WITH SMOTE	**TEST SET**	**65.40**
MODEL TUNING		
ERF	**TRAIN SET**	**100.00**
ERF	**TEST SET**	**95.44**
BAGGING	**TRAIN SET**	**100.00**
BAGGING	**TEST SET**	**96.80**

Table 4.4: WIN PER CENT OF TEST FORMAT CLUSTER

MODEL BUILDING		**WIN PERCENT**
NEURAL NETWORK CLASSIFIER	**TRAIN SET**	**84.39**
NEURAL NETWORK CLASSIFIER	**TEST SET**	**76.62**
LOGISTIC REGRESSION	**TRAIN SET**	**84.85**
LOGISTIC REGRESSION	**TEST SET**	**87.48**
MODEL TUNING		
GRADIENT BOOSTING	**TRAIN SET**	**100.00**
GRADIENT BOOSTING	**TEST SET**	**92.35**
ADABOOST	**TRAIN SET**	**100.00**
ADABOOST	**TEST SET**	**92.35**

BEST MODEL BUILD IS:
The best model for the entire dataset is **RANDOM FOREST CLASSIFIER**
The best models for ODI is **RANDOM FOREST CLASSIFIER.**
The best models for T20 is **RANDOM FOREST CLASSIFIER**
The best models for TEST is **LOGISTIC REGRESSION**

BEST MODEL AFTER MODEL TUNING IS:
ENTIRE DATASET: ENSEMBLE RANDOM FOREST
ODI FORMAT: ENSEMBLE RANDOM FOREST
T20 FORMAT: BAGGING
TEST FORMAT: GRADIENT BOOSTING

5. MODEL VALIDATION:

INSIGHT: The models built thus are analysed and best model is to be identified based on the performance metrics. Then, based on match formats, we decide the best model for each match format to tune that model.

APPROACH:

Based on the classification metrics, the following calculations are done to obtain performance metrics:

Accuracy: Out of all cases how much is correctly predicted given by:
(TP + TN)/ (TP + TN + FP +FN)
Recall is TP rate i.e., TP/ (TP + FN)
Precision is specificity i.e., TN rate which is TN/ (TN + FP)
F1-SCORE measures precision and recall at the same and is
(2*Recall*Precision)/ (Recall + Precision)

TABLE 5.1 PERFORMANCE METRICS OF DIFFERENT MODELS BUILD FOR OVERALL DATA

		PERFORMANCE METRICS					
MODEL BUILDING	DATA	PRECISION	ACCURACY	RECALL	F1 SCORE	AUC	Win %
DTC	TRAIN SET	0.77	0.77	0.77	0.77	0.86	77.82
DTC	TEST SET	0.88	0.72	0.76	0.82	0.732	75.21
RFC	TRAIN SET	0.84	0.84	0.87	0.85	0.924	86.38
RFC	TEST SET	0.88	0.8	0.8	0.88	0.776	85.34
NNC	TRAIN SET	0.73	0.77	0.85	0.79	0.863	84.08
NNC	TEST SET	0.91	0.80	0.85	0.88	0.789	83.5
LOG REG	TRAIN SET	0.77	0.77	0.76	0.76	0.829	73.6
LOG REG	TEST SET	0.91	0.73	0.75	0.82	0.752	74.59
LDA	TRAIN SET	0.77	0.76	0.76	0.76	0.828	76.56
LDA	TEST SET	0.91	0.74	0.75	0.83	0.754	74.63
NBS	TRAIN SET	0.74	0.72	0.68	0.71	0.719	67.93
NBS	TEST SET	0.9	0.66	0.67	0.77	0.657	65.73
KNNS	TRAIN SET	0.99	0.9	0.8	0.89	0.899	84.08
KNNS	TEST SET	0.95	0.74	0.72	0.82	0.764	83.5

INTERPREATION:

RANDOM FOREST CLASSIFIER IS THE BEST MODEL BUILD ON ENTIRE DATASET.

VARIABLE IMPORTANCE OF 0.1 CAN BE CONSIDERED FOR DEFINING STRATEGIES TO WIN

TABLE 5.2 PERFORMANCE METRICS OF DIFFERENT MODELS BUILD FOR DIFFERENT CLUSTERS

			PERFORMANCE METRICS					
	MODEL BUILDING	DATA	PRECISION	ACCURACY	RECALL	F1 SCORE	AUC	Win %
ODI-MATCH	RFC	TRAIN SET	0.85	0.86	0.87	0.86	0.941	86.84
ODI-MATCH	RFC	TEST SET	0.92	0.81	0.86	0.89	0.722	83.89
ODI-MATCH	NNC	TRAIN SET	0.83	0.82	0.8	0.81	0.888	79.76
ODI-MATCH	NNC	TEST SET	0.94	0.77	0.79	0.86	0.737	77.89

	MODEL BUILDING	DATA	PRECISION	ACCURACY	RECALL	F1 SCORE	AUC	WIN %
T20 MATCH	RFC	TRAIN SET	0.8	0.79	0.79	0.79	0.877	83.44
T20 MATCH	RFC	TEST SET	0.86	0..70	0.74	0.79	0.738	76.92
T20 MATCH	NNC	TRAIN SET	0.89	0.76	0.59	0.71	0.854	80.8
T20 MATCH	NNC	TEST SET	0.9	0.57	0.51	0.65	0.716	76.58
T20 MATCH	KNNS	TRAIN SET	0.99	0.89	0.79	0.88	0.891	85.41
T20 MATCH	KNNS	TEST SET	0.91	0.67	0.66	0.66	0.703	82.64

	MODEL BUILDING	DATA	PRECISION	ACCURACY	RECALL	F1 SCORE	AUC	WIN %
TEST MATCH	LOG REG	TRAIN SET	0.85	0.85	0.85	0.85	0.958	84.85
TEST MATCH	LOG REG	TEST SET	0.91	0.84	0.89	0.9	0.857	87.48
TEST MATCH	NNC	TRAIN SET	0.78	0.8	0.85	0.81	0.893	84.4
TEST MATCH	NNC	TEST SET	0.9	0.75	0.77	0.83	0.825	76.7

INTERPRETATION: Based on win% and AUC values,
BEST MODEL BUILD IS:
The best models for ODI is **RANDOM FOREST CLASSIFIER.**
The best models for T20 is **RANDOM FOREST CLASSIFIER**
The best models for TEST is **LOGISTIC REGRESSION**

After Model Tuning:
ODI Format: Ensemble Random Forest
T20 Format: Bagging
Test Format: Gradient Boosting

TABLE 5.3 PERFORMANCE METRICS AFTER MODEL TUNING FOR DIFFERENT CLUSTERS

			PERFORMANCE METRICS					
	MODEL BUILDING	DATA	PRECISION	ACCURACY	RECALL	F1 SCORE	AUC	Win %
ODI MATCH	BAGGING	TRAIN SET	1	1	1	1	0.998	99.90
ODI MATCH	BAGGING	TEST SET	0.96	0.95	0.98	0.97	0.824	96.64
ODI MATCH	E RF	TRAIN SET	1	1	1	1	1	100.00
ODI MATCH	E RF	TEST SET	0.96	0.95	0.98	0.97	0.838	96.91

T20 MATCH	BAGGING	TRAIN SET	1	1	1	1	1	100.00
T20 MATCH	BAGGING	TEST SET	0.94	0.95	0.98	0.96	0.868	96.80
T20 MATCH	E RF	TRAIN SET	1	1	1	1	1	100.00
T20 MATCH	E RF	TEST SET	0.95	0.93	0.96	0.95	0.883	95.44

TEST MATCH	GB	TRAIN SET	1	1	1	1	1	100.00
TEST MATCH	GB	TEST SET	0.89	0.86	0.94	0.92	0.749	92.35
TEST MATCH	AB	TRAIN SET	1	1	1	1	1	100.00
TEST MATCH	AB	TEST SET	0.89	0.86	0.94	0.92	0.749	92.35

6. Final interpretation / recommendation
- Detailed recommendations for the management/client based on the analysis done.

Interpretation of the most optimum model and its implication on the business
Business Implications of The Optimum Model:
Insight:
The problem statement is to provide business insights based on the models built and tuned till far. BCCI need to be provided with different strategies against the following formats.
1. 1 Test match with England in England. All the match are day matches. In England, it will be rainy season at the time to match.
2. 2 T20 match with Australia in India. All the match are Day and Night matches. In India, it will be winter season at the time to match.
3. 2 ODI match with SriLanka in India. All the match are Day and Night matches. In India, it will be winter season at the time to match.
Approach:
The best built and tuned models after ensemble are considered for drawing effective strategies based on variable importance in the models which are selected based on win%.
BUSINESS INTERPRETATION:
BUSINESS STRATEGY AND RECOMMENDATION FOR TEST MATCH CLUSTER
Table 6.1: VARIABLE IMPORTANCE FOR TEST MATCH CLUSTER

odds_ratio	variable
1.30E+07	Opponent
9.82E+00	First_selection
3.66E+00	Players_scored_zero
3.24E+00	Min_run_given_1over
1.56E+00	extra_bowls_opponent
1.50E+00	Max_run_scored_1over
1.45E+00	Avg_team_Age
1.32E+00	Extra_bowls_bowled
1.26E+00	Offshore
1.04E+00	Result
1.00E+00	Match_light_type
1.00E+00	All_rounder_in_team
9.69E-01	player_highest_run
7.62E-01	Min_run_scored_1over
7.05E-01	Wicket_keeper_in_team
6.63E-01	Max_wicket_taken_1over
4.22E-01	Max_run_given_1over
2.32E-01	Season
2.19E-01	Bowlers_in_team

Based on the above table, for test match cluster Opponent, First_selection, players_scored_zero, Min_run_given_1over are important for making win prediction and defining winning strategies.

BUSINESS RECOMMENDATION FOR WINNING TEST MATCH

- India has to opt to bowl first on the offshore match at England if it wins toss. Inning extras must be reduced during the bowling half by the Indian team. Bowlers with little economy rate need to selected as they provide less runs per over to the opponent team.
- If India bats first, based on rainy climate, as there is a chance of DLS, maintaining good runs per over (r. p. o) is necessary. Also, after rains, the second batting becomes tougher. As England pitches are bouncy, selecting more (4 allrounders) and minimising duck-outs i.e., by sending opening batsmen who can stand for a long time at least 20 overs. Maintaining partnerships (at least 2-3) of 100- 120 runs and average of 4-5 runs per over is highly appreciated in the match.
- Practice matches are also preferred as it helps to acclimatise to the offshore climatic conditions at England since no test matches were played with England in rainy season based on the dataset.

BUSINESS STRATEGY AND RECOMMENDATION FOR T20 MATCH CLUSTER

Table 6.2: VARIABLE IMPORTANCE FOR T20 MATCH CLUSTER

Variable	Importance
Players_scored_zero	0.220497
Extra_bowls_bowled	0.122009
Opponent	0.108232
Min_run_scored_1over	0.090526
extra_bowls_opponent	0.074193
All_rounder_in_team	0.066362
Bowlers_in_team	0.065253
Max_wicket_taken_1over	0.064699
Season	0.060594
player_highest_run	0.036164
Offshore	0.029334
Max_run_scored_1over	0.028892
Max_run_given_1over	0.021726
First_selection	0.008177
Match_light_type	0.003341
Min_run_given_1over	0
Wicket_keeper_in_team	0
Match_format	0
Avg_team_Age	0

Based on the above table, for T20 match cluster players_scored_zero, extra bowls bowled, opponent are important for making win prediction and defining winning strategies.

BUSINESS RECOMMENDATION FOR WINNING T20 MATCH

RECOMMENDATION-1

The players must score minimum of 20 runs and no duck-out is mandatory for winning. So, the captain must deploy scoring batsmen as openers, select 3-4 allrounders in team and one all-rounder in the top 4 batting line-up. The runs per over by the team must be maintained around 8-9 at least if batting first.
The match at times due to fog in winter, the ball becomes slippery and can't slide on ground. So, it is advisable to bat first as Australia is a tough team.

RECOMMENDATION-2

Consider the bowling strategy, it is necessary to minimise the inning extras and pressurise the opponent team by taking key wickets, especially in P1 and P2. The mid overs are to be controlled by spinners and allrounders. More allrounders are generally preferred as in T20 matches, scores fly high.

BUSINESS STRATEGY AND RECOMMENDATION FOR ODI MATCH CLUSTER

Table 6.3: VARIABLE IMPORTANCE FOR ODI MATCH CLUSTER

Variable	Importance
extra_bowls_opponent	0.177686
Extra_bowls_bowled	0.141615
Max_wicket_taken_1over	0.127468
All_rounder_in_team	0.111439
Min_run_scored_1over	0.109006
Players_scored_zero	0.056168
Season	0.052319
Bowlers_in_team	0.04747
Min_run_given_1over	0.041813
player_highest_run	0.04043
Max_run_given_1over	0.034556
Max_run_scored_1over	0.022341
Offshore	0.017851
Opponent	0.014673
First_selection	0.003822
Match_light_type	0.001343
Wicket_keeper_in_team	0
Match_format	0
Avg_team_Age	0

Based on the above table, for ODI match cluster extra bowls opponent, extra_bowls_bowled, max_wickets_taken_1 over, All_rounder in team, Min_run_scored_1over are important for making win prediction and defining winning strategies.

BUSINESS RECOMMENDATION FOR WINNING ODI MATCH

RECOMMENDATION 1

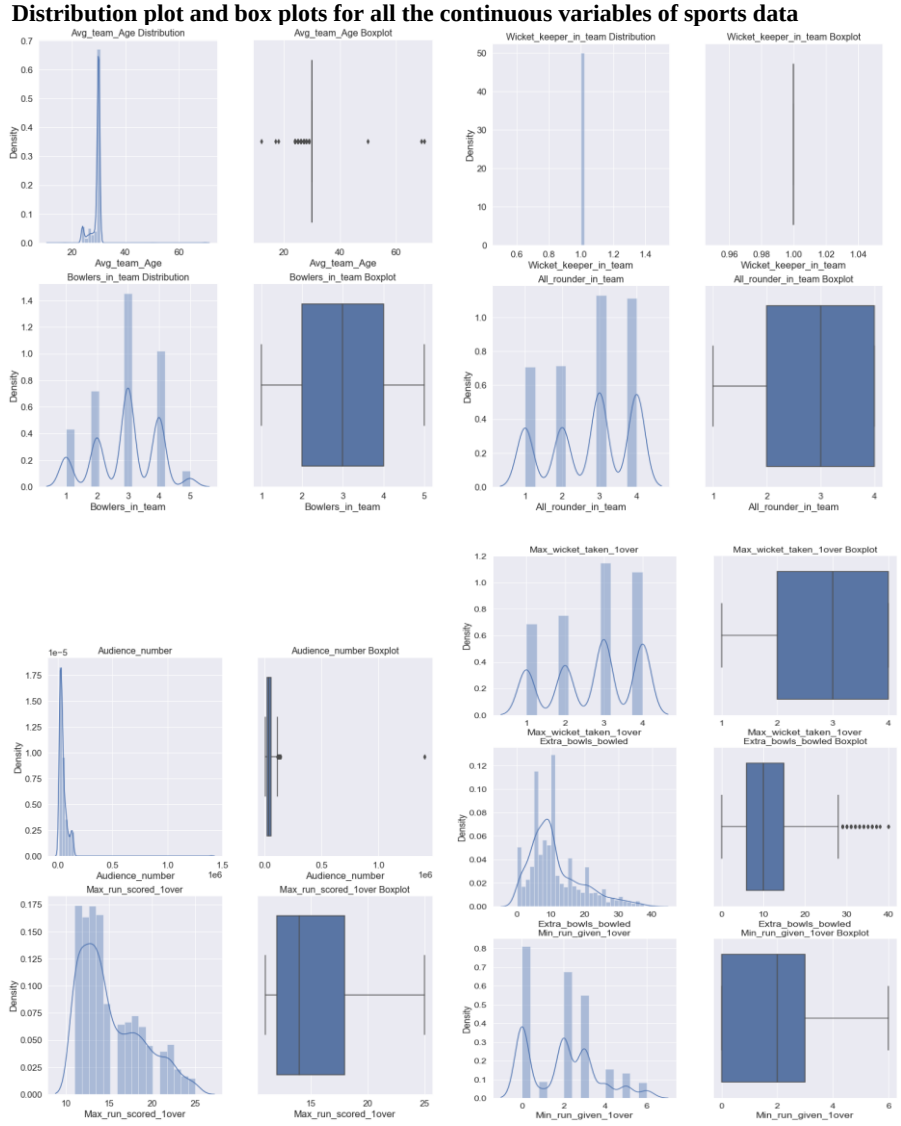
- Provide less bowling extras during the bowling innings. Fast bowlers during the P1 and P3 with good swing are to be selected in the team.
- Minimising the opponent runs per over by taking more wickets by selecting excellent spinners with less economy rate.

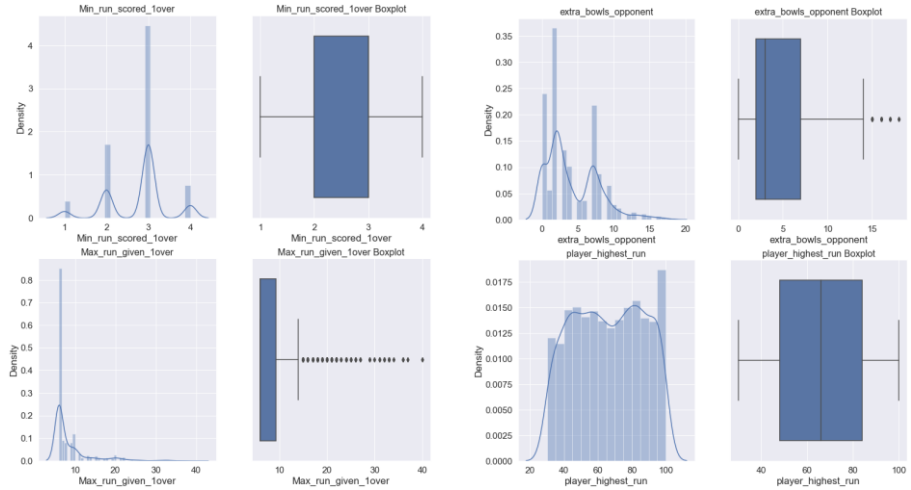
RECOMMENDATION 2

- Selecting 2-3 all-rounders in the team and opt to bat first, if India wins toss due to hard pitch, foggy conditions in the second innings.
- Maximum runs must be scored per over if India bats first.

ANNEXURE-1

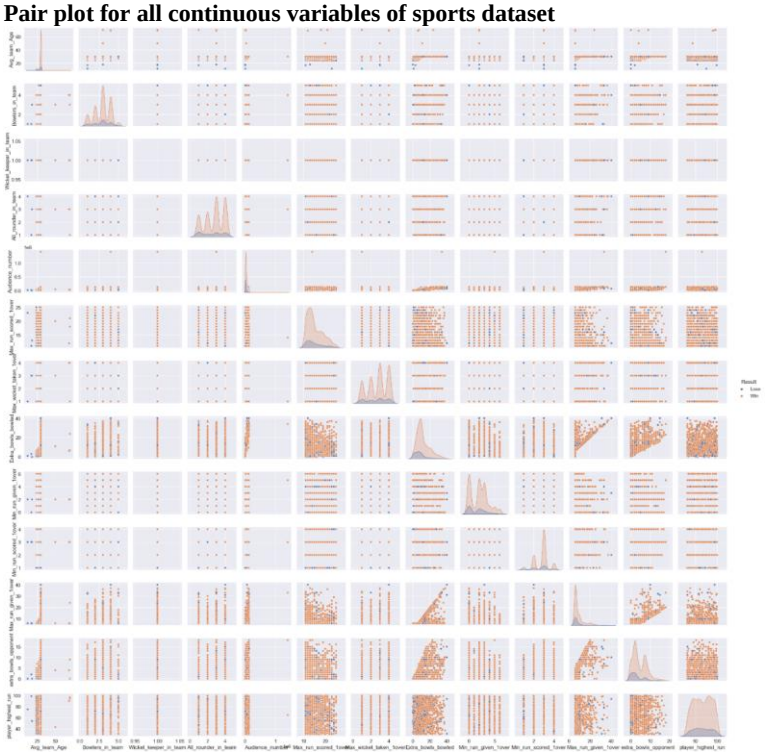
Distribution plot and box plots for all the continuous variables of sports data




Min_run_scored_1over
Min_run_scored_1over Boxplot
extra_bowls_opponent
extra_bowls_opponent Boxplot
Max_run_given_1over
Max_run_given_1over Boxplot
player_highest_run
player_highest_run Boxplot
Density

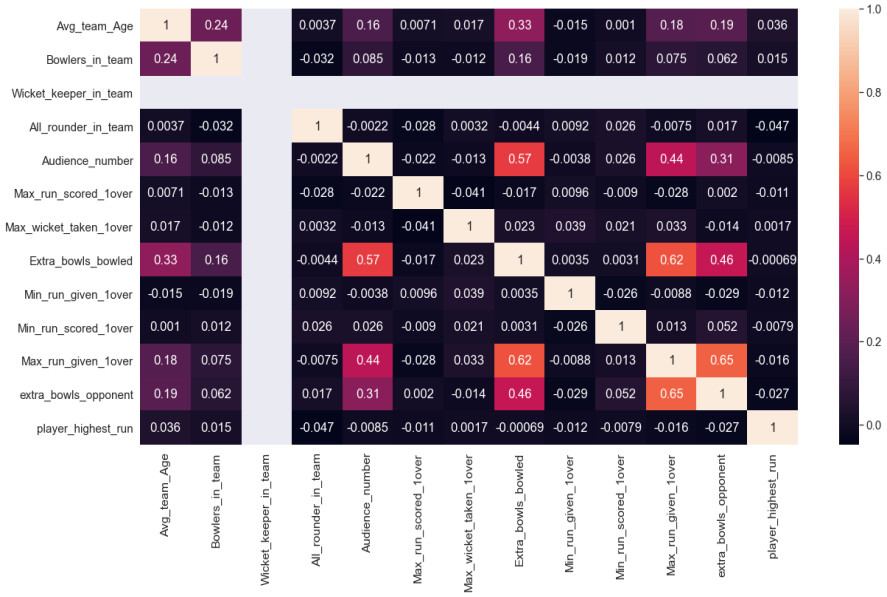
ANNEXURE-2

Pair plot for all continuous variables of sports dataset

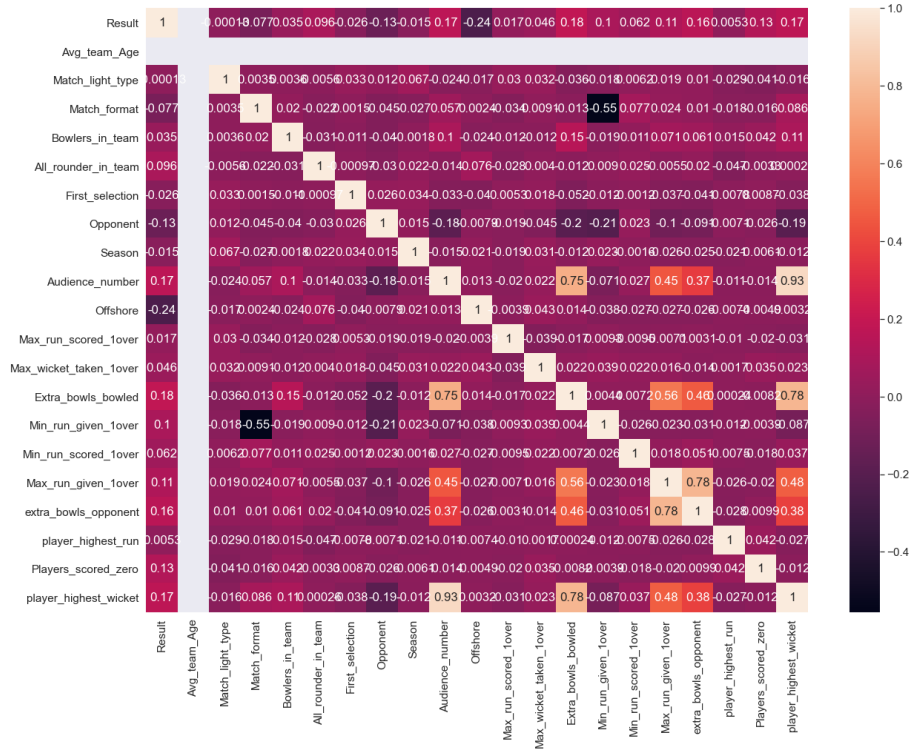


ANNEXURE-3

Heat map depicting correlation between continuous variables of sports dataset



Heat map for encoded and scaled variables



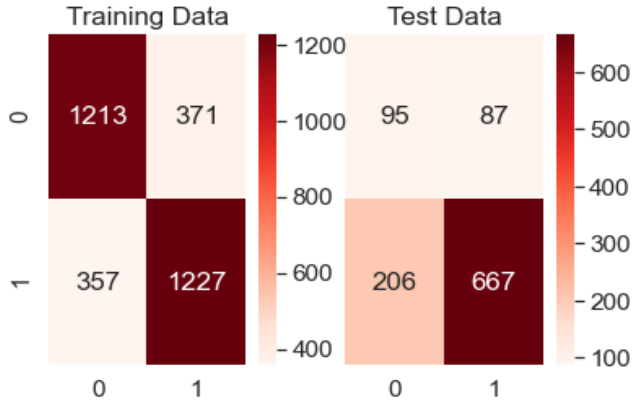
OUTPUT FOR MODEL DECISION TREE CLASSIFIER (DTC) FOR OVERALL DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

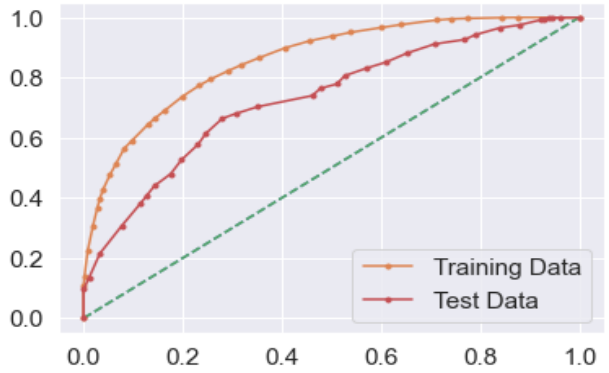
TRAIN SET	precision	recall	f1-score	support
0	0.77	0.77	0.77	1584
1	0.77	0.77	0.77	1584
accuracy			0.77	3168
macro avg	0.77	0.77	0.77	3168
weighted avg	0.77	0.77	0.77	3168

TEST SET	precision	recall	f1-score	support
0	0.32	0.52	0.39	182
1	0.88	0.76	0.82	873
accuracy			0.72	1055
macro avg	0.6	0.64	0.61	1055
weighted avg	0.79	0.72	0.75	1055

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 0.860

AUC TEST DATA: 0.732

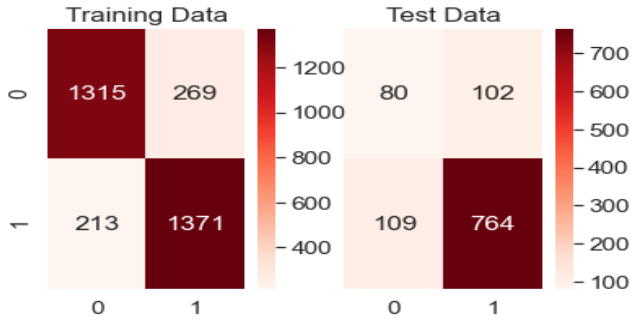
OUTPUT FOR MODEL RANDOM FOREST CLASSIFIER (RFC) FOR OVERALL DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

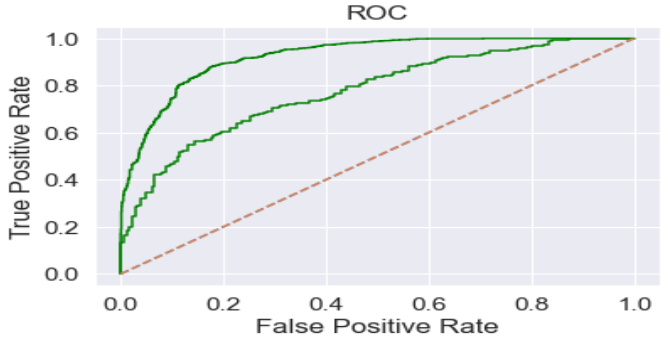
TRAIN SET	precision	recall	f1-score	support
0	0.86	0.83	0.85	1584
1	0.84	0.87	0.85	1584
accuracy			0.85	3168
macro avg	0.85	0.85	0.85	3168
weighted avg	0.85	0.85	0.85	3168

TEST SET	precision	recall	f1-score	support
0	0.42	0.44	0.43	182
1	0.88	0.88	0.88	873
accuracy			0.8	1055
macro avg	0.65	0.66	0.65	1055
weighted avg	0.8	0.8	0.8	1055

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 0.924

AUC TEST DATA: 0.776

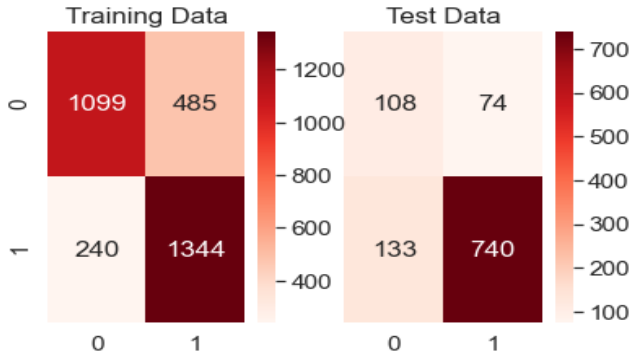
OUTPUT FOR MODEL NEURAL NETWORK CLASSIFIER (NNC) FOR OVERALL DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

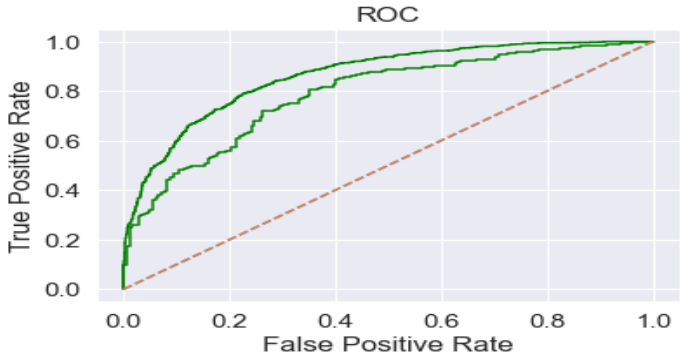
TRAIN SET	precision	recall	f1-score	support
0	0.82	0.69	0.75	1584
1	0.73	0.85	0.79	1584
accuracy			0.77	3168
macro avg	0.78	0.77	0.77	3168
weighted avg	0.78	0.77	0.77	3168

TEST SET	precision	recall	f1-score	support
0	0.45	0.59	0.51	182
1	0.91	0.85	0.88	873
accuracy			0.8	1055
macro avg	0.68	0.72	0.69	1055
weighted avg	0.83	0.8	0.81	1055

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 0.863

AUC TEST DATA: 0.789

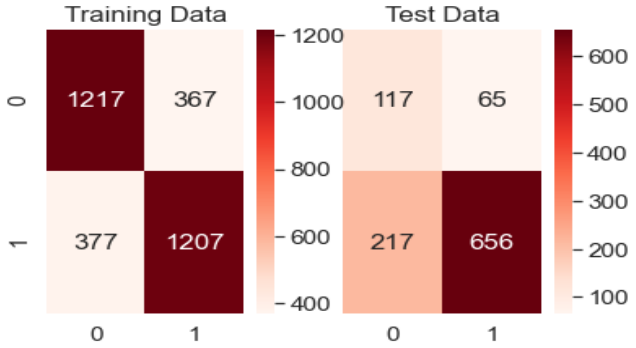
OUTPUT FOR LOGISTIC REGRESSION MODEL (LRM) FOR OVERALL DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

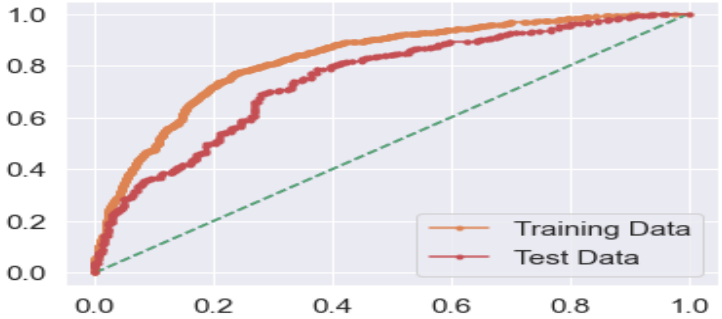
TRAIN SET	precision	recall	f1-score	support
0	0.76	0.77	0.77	1584
1	0.77	0.76	0.76	1584
accuracy			0.77	3168
macro avg	0.77	0.77	0.77	3168
weighted avg	0.77	0.77	0.77	3168

TEST SET	precision	recall	f1-score	support
0	0.35	0.64	0.45	182
1	0.91	0.75	0.82	873
accuracy			0.73	1055
macro avg	0.63	0.7	0.64	1055
weighted avg	0.81	0.73	0.76	1055

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 0.829

AUC TEST DATA: 0.752

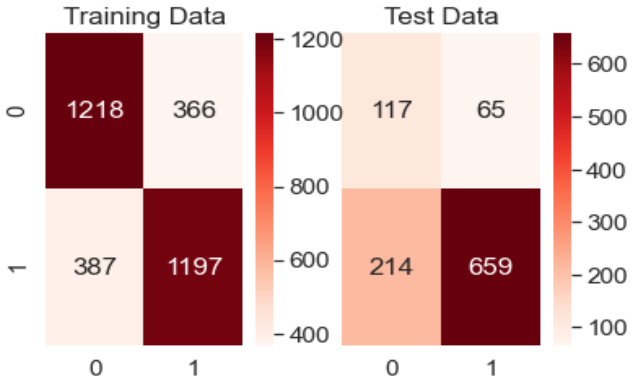
OUTPUT FOR MODEL LINEAR DISCRIMINANT ANALYSIS (LDA) FOR OVERALL DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

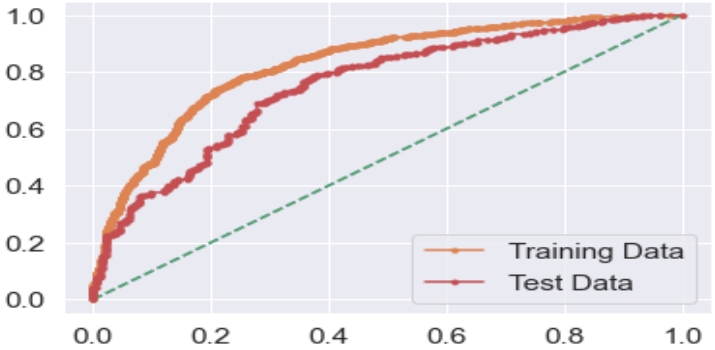
TRAIN SET	precision	recall	f1-score	support
0	0.76	0.77	0.76	1584
1	0.77	0.76	0.76	1584
accuracy			0.76	3168
macro avg	0.76	0.76	0.76	3168
weighted avg	0.76	0.76	0.76	3168

TEST SET	precision	recall	f1-score	support
0	0.35	0.64	0.46	182
1	0.91	0.75	0.83	873
accuracy			0.74	1055
macro avg	0.63	0.7	0.64	1055
weighted avg	0.81	0.74	0.76	1055

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 0.828

AUC TEST DATA: 0.754

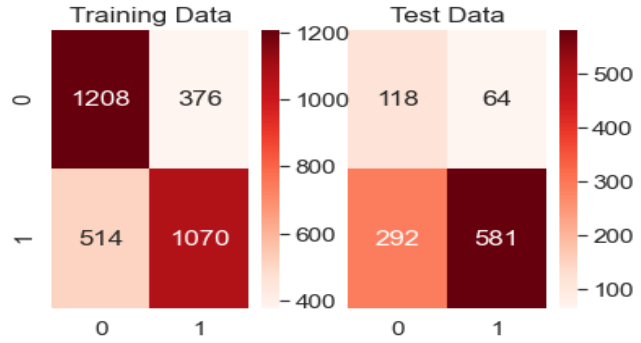
OUTPUT FOR MODEL NAÏVE BAYES WITH SMOTE (NBS) FOR OVERALL DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

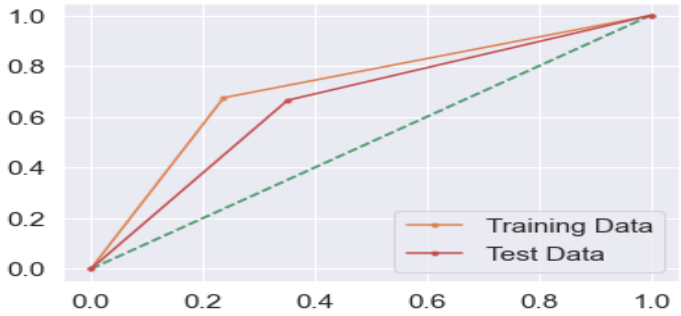
TRAIN SET	precision	recall	f1-score	support
0	0.7	0.76	0.73	1584
1	0.74	0.68	0.71	1584
accuracy			0.72	3168
macro avg	0.72	0.72	0.72	3168
weighted avg	0.72	0.72	0.72	3168

TEST SET	precision	recall	f1-score	support
0	0.29	0.65	0.4	182
1	0.9	0.67	0.77	873
accuracy			0.66	1055
macro avg	0.59	0.66	0.58	1055
weighted avg	0.8	0.66	0.7	1055

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 0.719

AUC TEST DATA: 0.657

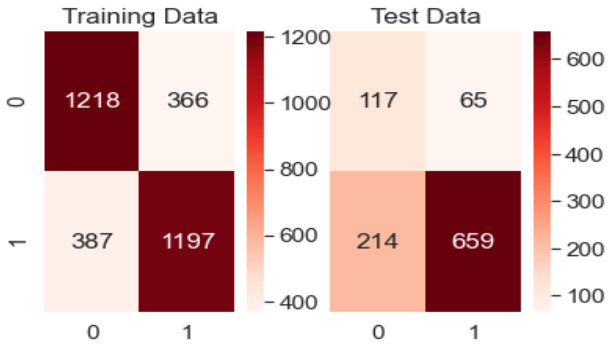
OUTPUT FOR MODEL KNN WITH SMOTE (KNNS) FOR OVERALL DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

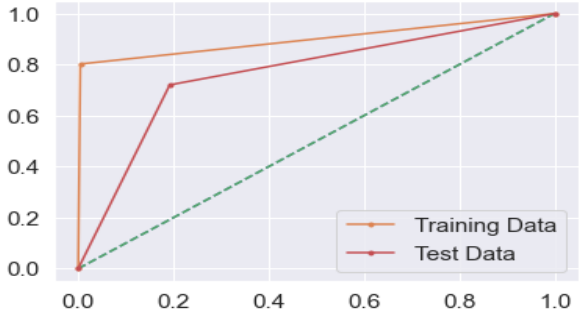
TRAIN SET	precision	recall	f1-score	support
0	0.83	1	0.91	1584
1	0.99	0.8	0.89	1584
accuracy			0.9	3168
macro avg	0.91	0.9	0.9	3168
weighted avg	0.91	0.9	0.9	3168

TEST SET	precision	recall	f1-score	support
0	0.38	0.81	0.51	182
1	0.95	0.72	0.82	873
accuracy			0.74	1055
macro avg	0.66	0.76	0.67	1055
weighted avg	0.85	0.74	0.77	1055

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 0.899

AUC TEST DATA: 0.764

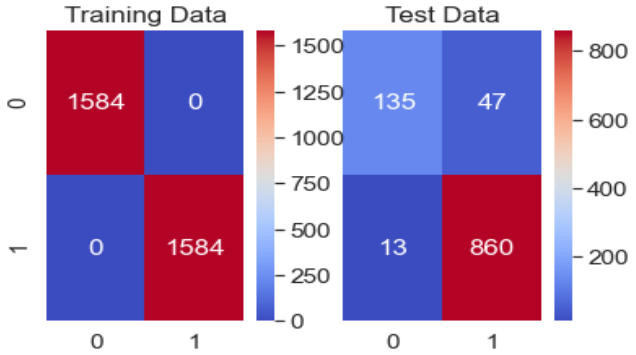
OUTPUT FOR TUNED MODEL ENSEMBLE RFC FOR OVERALL DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

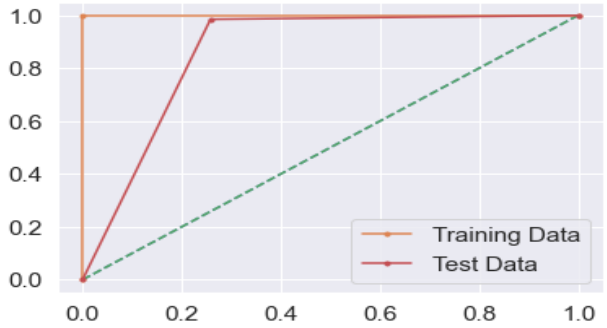
TRAIN SET	precision	recall	f1-score	support
0	1	1	1	1584
1	1	1	1	1584
accuracy			1	3168
macro avg	1	1	1	3168
weighted avg	1	1	1	3168

TEST SET	precision	recall	f1-score	support
0	0.91	0.74	0.82	182
1	0.95	0.99	0.97	873
accuracy			0.94	1055
macro avg	0.93	0.86	0.89	1055
weighted avg	0.94	0.94	0.94	1055

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 1.000

AUC TEST DATA: 0.863

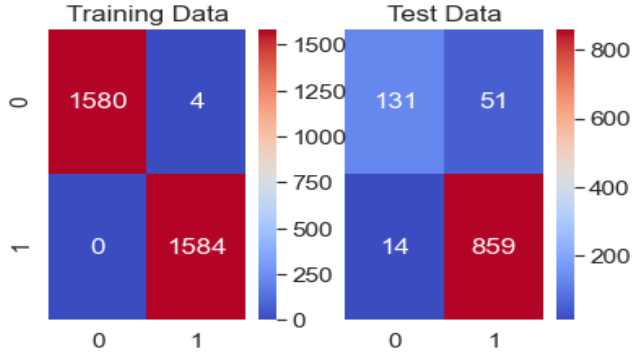
OUTPUT FOR TUNED MODEL BAGGING FOR OVERALL DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

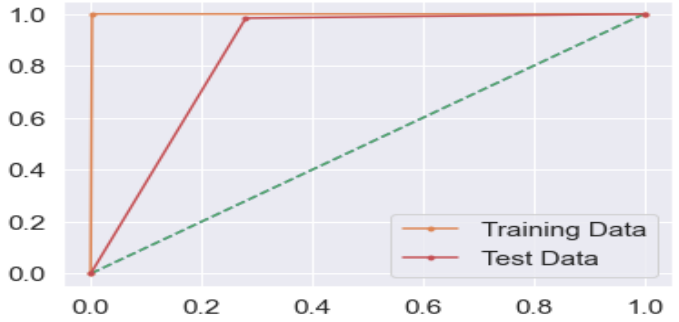
TRAIN SET	precision	recall	f1-score	support
0	1	1	1	1584
1	1	1	1	1584
accuracy			1	3168
macro avg	1	1	1	3168
weighted avg	1	1	1	3168

TEST SET	precision	recall	f1-score	support
0	0.9	0.72	0.8	182
1	0.94	0.98	0.96	873
accuracy			0.94	1055
macro avg	0.92	0.85	0.88	1055
weighted avg	0.94	0.94	0.94	1055

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 0.999

AUC TEST DATA: 0.852

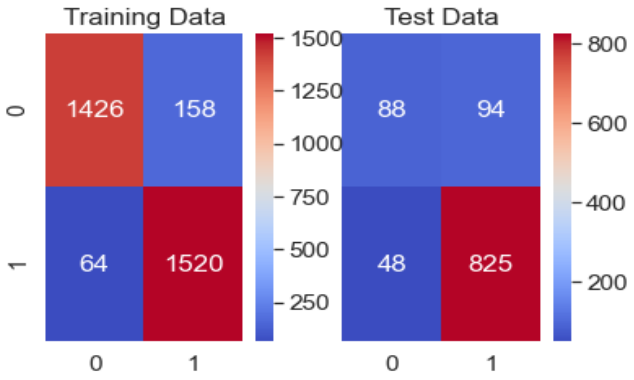
OUTPUT FOR TUNED MODEL GRADIENT BOOSTING FOR OVERALL DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

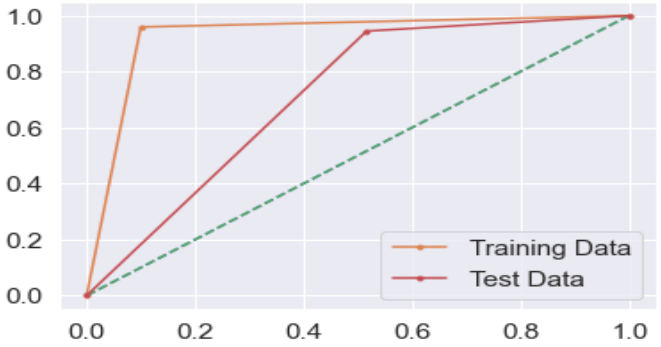
TRAIN SET	precision	recall	f1-score	support
0	0.96	0.9	0.93	1584
1	0.91	0.96	0.93	1584
accuracy			0.93	3168
macro avg	0.93	0.93	0.93	3168
weighted avg	0.93	0.93	0.93	3168

TEST SET	precision	recall	f1-score	support
0	0.65	0.48	0.55	182
1	0.9	0.95	0.92	873
accuracy			0.87	1055
macro avg	0.77	0.71	0.74	1055
weighted avg	0.85	0.87	0.86	1055

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 0.930

AUC TEST DATA: 0.714

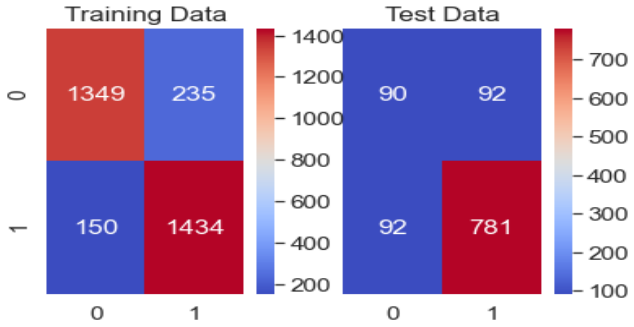
OUTPUT FOR TUNED MODEL ADABOOST FOR OVERALL DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

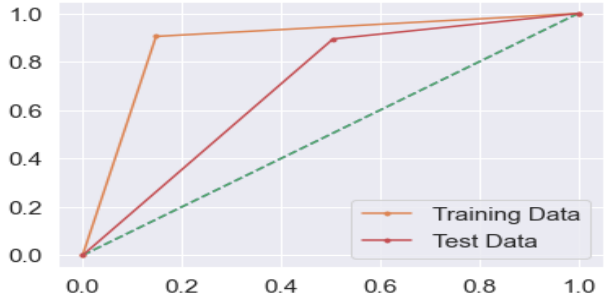
TRAIN SET	precision	recall	f1-score	support
0	0.9	0.85	0.88	1584
1	0.86	0.91	0.88	1584
accuracy			0.88	3168
macro avg	0.88	0.88	0.88	3168
weighted avg	0.88	0.88	0.88	3168

TEST SET	precision	recall	f1-score	support
0	0.49	0.49	0.49	182
1	0.89	0.89	0.89	873
accuracy			0.83	1055
macro avg	0.69	0.69	0.69	1055
weighted avg	0.83	0.83	0.83	1055

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 0.878

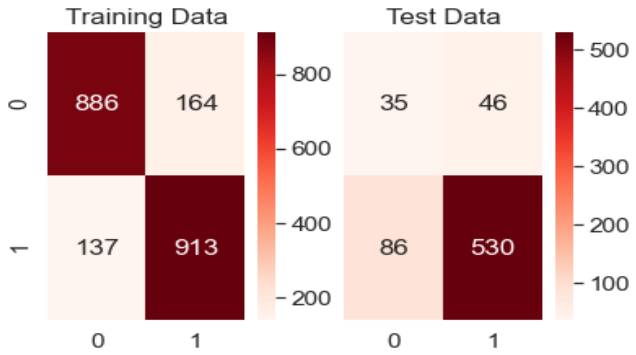
AUC TEST DATA: 0.695

OUTPUT FOR MODEL RANDOM FOREST CLASSIFIER (RFC) FOR ODI DATA

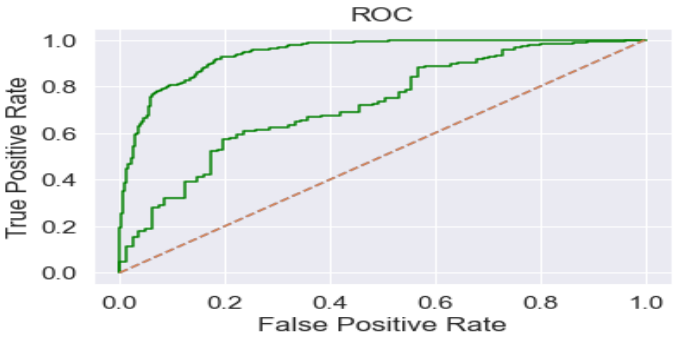
CLASSIFICATION METRICS OF TRAIN AND TEST SET

TRAIN SET	precision	recall	f1-score	support		TEST SET	precision	recall	f1-score	support
0	0.87	0.84	0.85	1050		0	0.29	0.43	0.35	81
1	0.85	0.87	0.86	1050		1	0.92	0.86	0.89	616
accuracy			0.86	2100		accuracy			0.81	697
macro avg	0.86	0.86	0.86	2100		macro avg	0.6	0.65	0.62	697
weighted avg	0.86	0.86	0.86	2100		weighted avg	0.85	0.81	0.83	697

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 0.941

AUC TEST DATA: 0.722

OUTPUT FOR MODEL NEURAL NETWORK CLASSIFIER (NNC) FOR ODI DATA

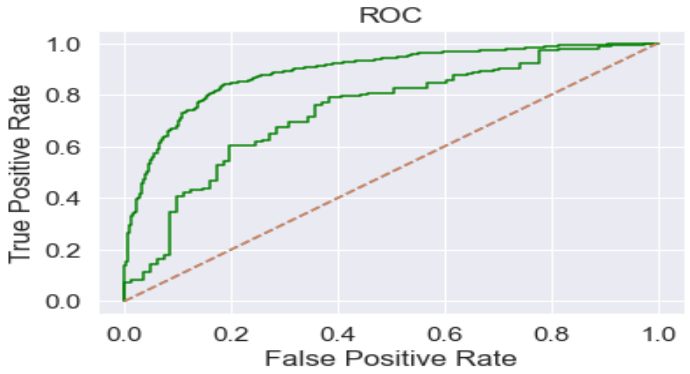
CLASSIFICATION METRICS OF TRAIN AND TEST SET

TRAIN SET	precision	recall	f1-score	support
0	0.8	0.84	0.82	1050
1	0.83	0.8	0.81	1050
accuracy			0.82	2100
macro avg	0.82	0.82	0.82	2100
weighted avg	0.82	0.82	0.82	2100

TEST SET	precision	recall	f1-score	support
0	0.28	0.62	0.38	81
1	0.94	0.79	0.86	616
accuracy			0.77	697
macro avg	0.61	0.7	0.62	697
weighted avg	0.86	0.77	0.8	697

CONFUSION MATRIX OF TRAIN AND TEST SET

ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 0.888

AUC TEST DATA: 0.737

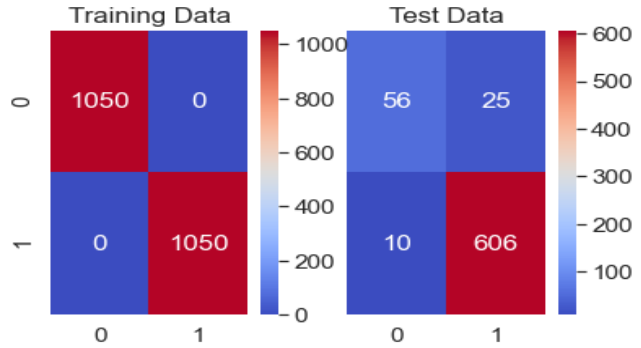
OUTPUT FOR TUNED MODEL ENSEMBLE RFC FOR ODI DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

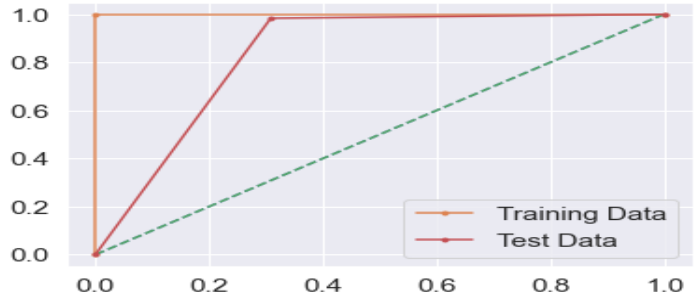
TRAIN SET	precision	recall	f1-score	support
0	1	1	1	1050
1	1	1	1	1050
accuracy			1	2100
macro avg	1	1	1	2100
weighted avg	1	1	1	2100

TEST SET	precision	recall	f1-score	support
0	0.85	0.69	0.76	81
1	0.96	0.98	0.97	616
accuracy			0.95	697
macro avg	0.9	0.84	0.87	697
weighted avg	0.95	0.95	0.95	697

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 1.000

AUC TEST DATA: 0.838

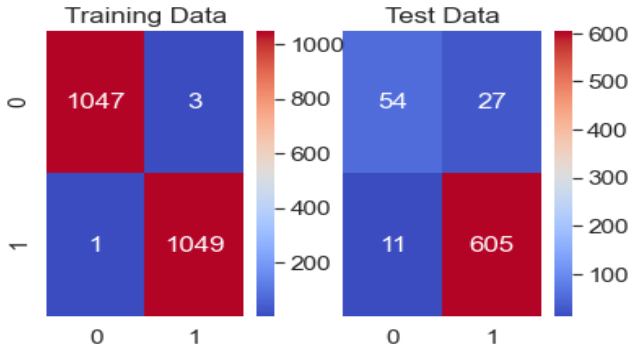
OUTPUT FOR TUNED MODEL BAGGING FOR ODI DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

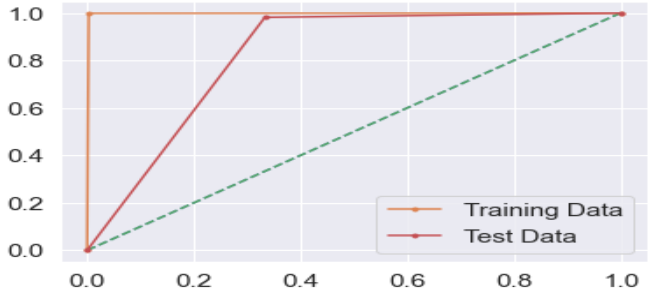
TRAIN SET	precision	recall	f1-score	support
0	1	1	1	1050
1	1	1	1	1050
accuracy			1	2100
macro avg	1	1	1	2100
weighted avg	1	1	1	2100

TEST SET	precision	recall	f1-score	support
0	0.83	0.67	0.74	81
1	0.96	0.98	0.97	616
accuracy			0.95	697
macro avg	0.89	0.82	0.85	697
weighted avg	0.94	0.95	0.94	697

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 0.998

AUC TEST DATA: 0.824

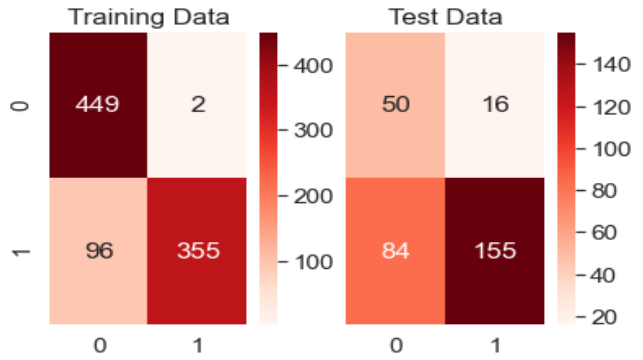
OUTPUT FOR MODEL KNN WITH SMOTE (KNNS) FOR T20 DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

TRAIN SET	precision	recall	f1-score	support
0	0.82	1	0.9	451
1	0.99	0.79	0.88	451
accuracy			0.89	902
macro avg	0.91	0.89	0.89	902
weighted avg	0.91	0.89	0.89	902

TEST SET	precision	recall	f1-score	support
0	0.37	0.76	0.5	66
1	0.91	0.65	0.76	239
accuracy			0.67	305
macro avg	0.64	0.7	0.63	305
weighted avg	0.79	0.67	0.7	305

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET

AUC TRAIN DATA: 0.891

AUC TEST DATA: 0.703

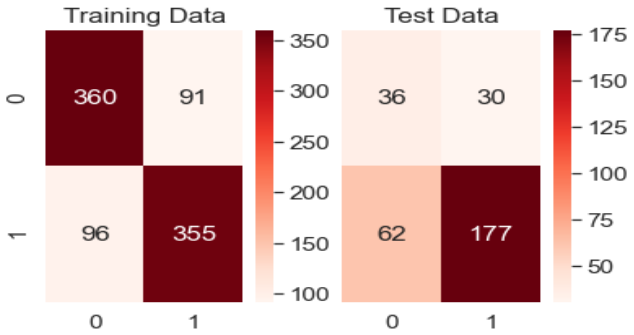
OUTPUT FOR MODEL RANDOM FOREST CLASSIFIER (RFC) FOR T20 DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

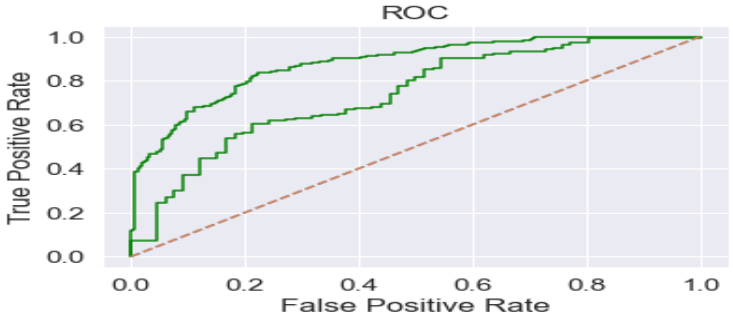
TRAIN SET	precision	recall	f1-score	support
0	0.79	0.8	0.79	451
1	0.8	0.79	0.79	451
accuracy			0.79	902
macro avg	0.79	0.79	0.79	902
weighted avg	0.79	0.79	0.79	902

TEST SET	precision	recall	f1-score	support
0	0.37	0.55	0.44	66
1	0.86	0.74	0.79	239
accuracy			0.7	305
macro avg	0.61	0.64	0.62	305
weighted avg	0.75	0.7	0.72	305

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 0.877

AUC TEST DATA: 0.738

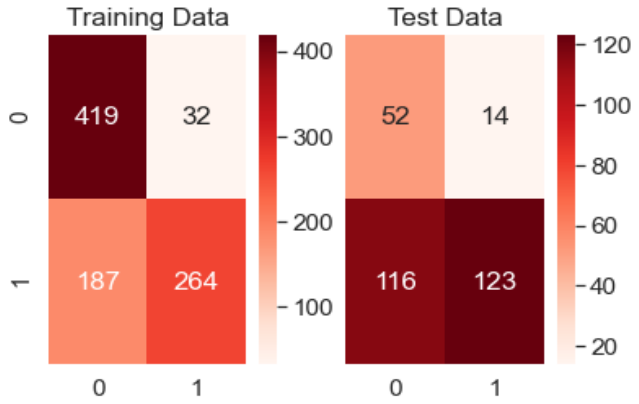
OUTPUT FOR MODEL NEURAL NETWORK CLASSIFIER (NNC) FOR T20 DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

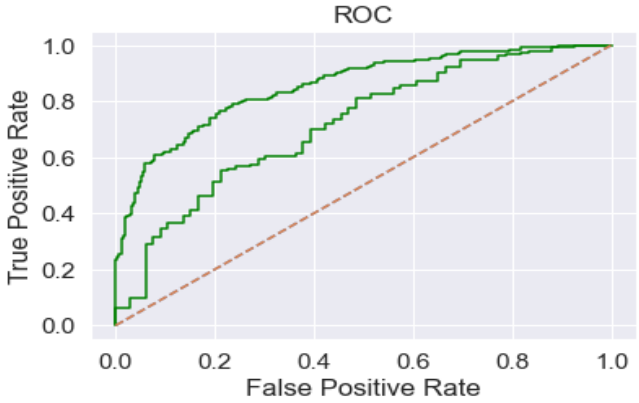
TRAIN SET	precision	recall	f1-score	support
0	0.69	0.93	0.79	451
1	0.89	0.59	0.71	451
accuracy			0.76	902
macro avg	0.79	0.76	0.75	902
weighted avg	0.79	0.76	0.75	902

TEST SET	precision	recall	f1-score	support
0	0.31	0.79	0.44	66
1	0.9	0.51	0.65	239
accuracy			0.57	305
macro avg	0.6	0.65	0.55	305
weighted avg	0.77	0.57	0.61	305

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 0.854

AUC TEST DATA: 0.716

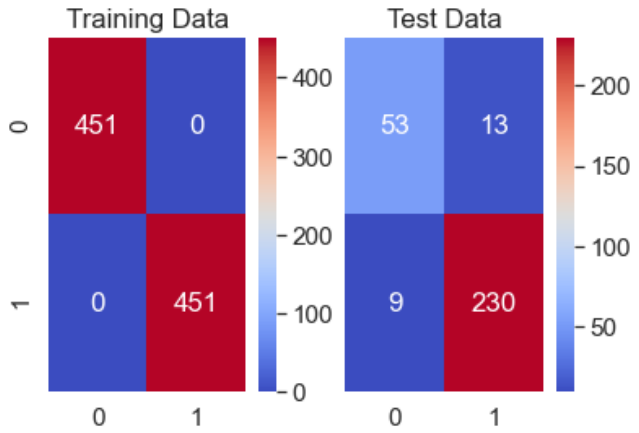
OUTPUT FOR MODEL TUNED ENSEMBLE RFC FOR T20 DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

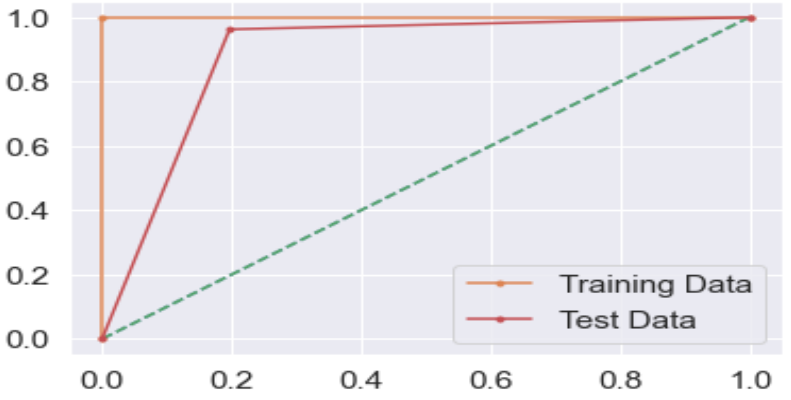
TRAIN SET	precision	recall	f1-score	support
0	1	1	1	451
1	1	1	1	451
accuracy			1	902
macro avg	1	1	1	902
weighted avg	1	1	1	902

TEST SET	precision	recall	f1-score	support
0	0.85	0.8	0.83	66
1	0.95	0.96	0.95	239
accuracy			0.93	305
macro avg	0.9	0.88	0.89	305
weighted avg	0.93	0.93	0.93	305

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 1.000

AUC TEST DATA: 0.883

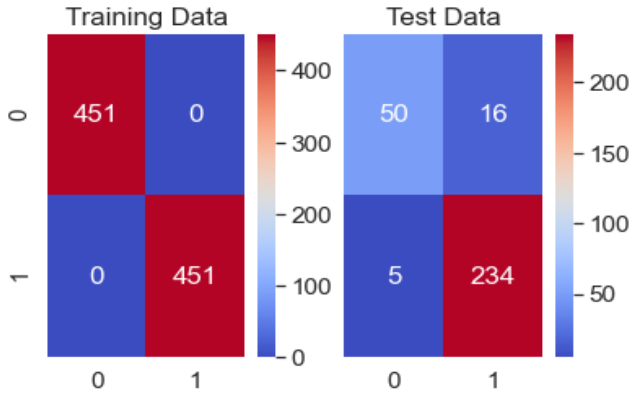
OUTPUT FOR MODEL TUNED BAGGING FOR T20 DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

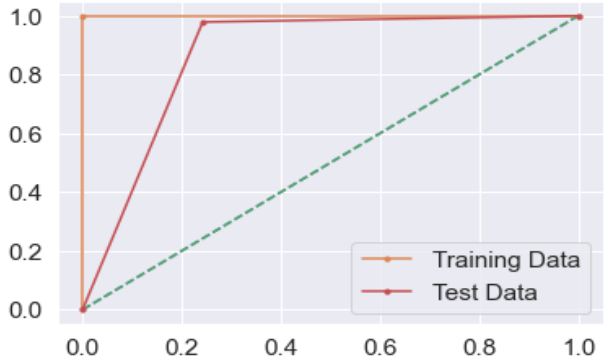
TRAIN SET	precision	recall	f1-score	support
0	1	1	1	451
1	1	1	1	451
accuracy			1	902
macro avg	1	1	1	902
weighted avg	1	1	1	902

TEST SET	precision	recall	f1-score	support
0	0.91	0.76	0.83	66
1	0.94	0.98	0.96	239
accuracy			0.93	305
macro avg	0.92	0.87	0.89	305
weighted avg	0.93	0.93	0.93	305

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 1.000

AUC TEST DATA: 0.868

OUTPUT FOR MODEL LOGISTIC REGRESSION FOR TEST DATA

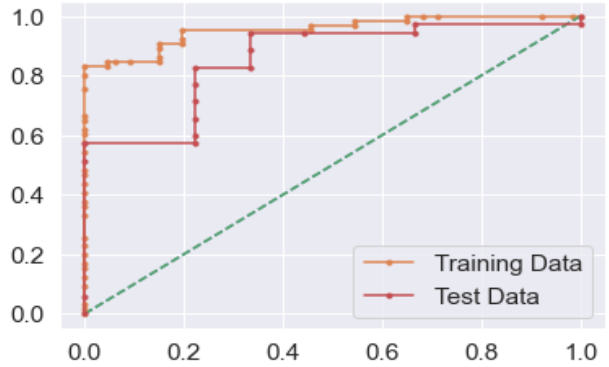
CLASSIFICATION METRICS OF TRAIN AND TEST SET

TRAIN SET	precision	recall	f1-score	support
0	0.85	0.85	0.85	66
1	0.85	0.85	0.85	66
accuracy			0.85	132
macro avg	0.85	0.85	0.85	132
weighted avg	0.85	0.85	0.85	132

TEST SET	precision	recall	f1-score	support
0	0.6	0.67	0.63	9
1	0.91	0.89	0.9	35
accuracy			0.84	44
macro avg	0.76	0.78	0.77	44
weighted avg	0.85	0.84	0.84	44

CONFUSION MATRIX OF TRAIN AND TEST SET

ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 0.956

AUC TEST DATA: 0.857

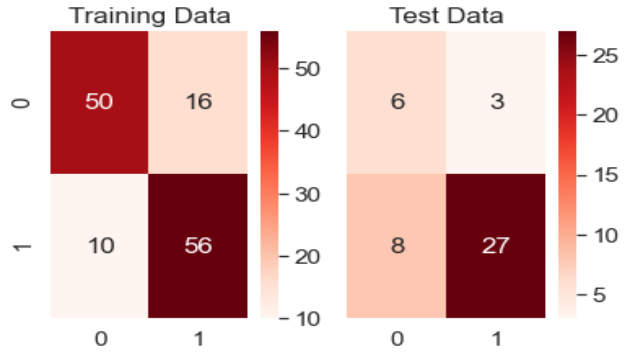
OUTPUT FOR MODEL NEURAL NETWORK CLASSIFIER (NNC) FOR TEST DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

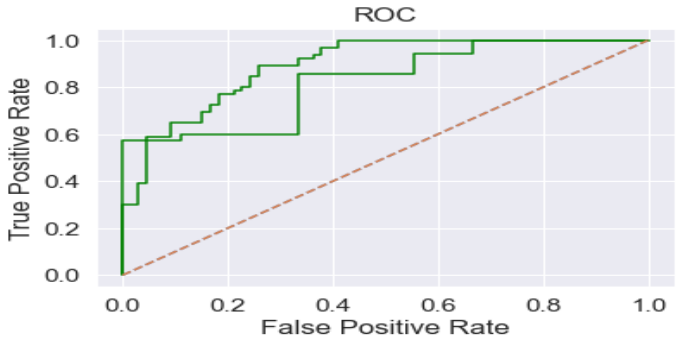
TRAIN SET	precision	recall	f1-score	support
0	0.83	0.76	0.79	66
1	0.78	0.85	0.81	66
accuracy			0.8	132
macro avg	0.81	0.8	0.8	132
weighted avg	0.81	0.8	0.8	132

TEST SET	precision	recall	f1-score	support
0	0.43	0.67	0.52	9
1	0.9	0.77	0.83	35
accuracy			0.75	44
macro avg	0.66	0.72	0.68	44
weighted avg	0.8	0.75	0.77	44

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 0.893

AUC TEST DATA: 0.825

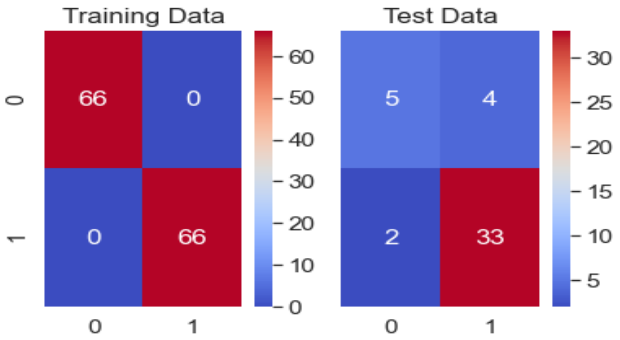
OUTPUT FOR MODEL TUNED GRADIENT BOOSTING FOR TEST DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

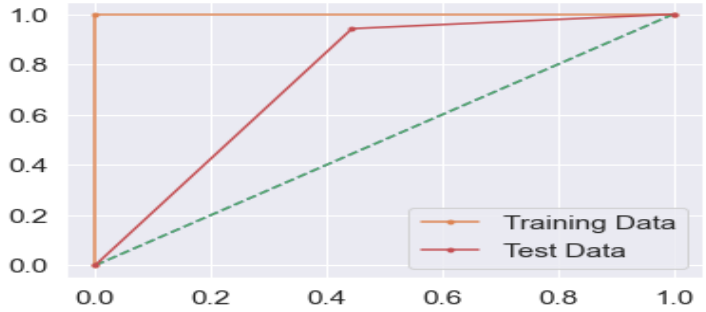
TRAIN SET	precision	recall	f1-score	support
0	1	1	1	66
1	1	1	1	66
accuracy			1	132
macro avg	1	1	1	132
weighted avg	1	1	1	132

TEST SET	precision	recall	f1-score	support
0	0.71	0.56	0.63	9
1	0.89	0.94	0.92	35
accuracy			0.86	44
macro avg	0.8	0.75	0.77	44
weighted avg	0.86	0.86	0.86	44

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 1.000

AUC TEST DATA: 0.749

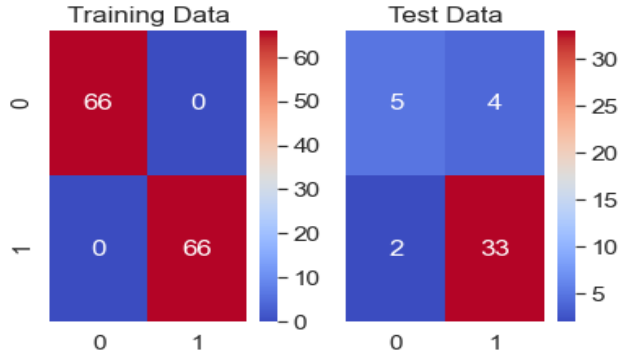
OUTPUT FOR MODEL TUNED ADABOOST FOR TEST DATA

CLASSIFICATION METRICS OF TRAIN AND TEST SET

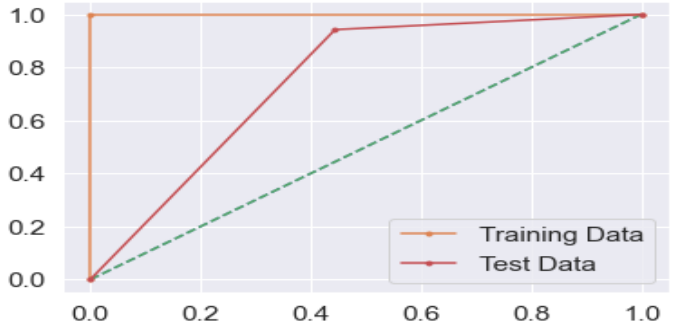
TRAIN SET	precision	recall	f1-score	support
0	1	1	1	66
1	1	1	1	66
accuracy			1	132
macro avg	1	1	1	132
weighted avg	1	1	1	132

TEST SET	precision	recall	f1-score	support
0	0.71	0.56	0.63	9
1	0.89	0.94	0.92	35
accuracy			0.86	44
macro avg	0.8	0.75	0.77	44
weighted avg	0.86	0.86	0.86	44

CONFUSION MATRIX OF TRAIN AND TEST SET



ROC OF TRAIN AND TEST SET



AUC TRAIN DATA: 1.000

AUC TEST DATA: 0.749