European Football Games and Teams Analysis

Nisha Mohan Devadiga
Masters in Data Science
San Jose State University
San Jose, US
nishamohan.devadiga@sjsu.edu

Akanksha Rawat
Masters in Data Science
San Jose State University
San Jose, US
akanksha.rawat@sjsu.edu

Karishma Kuria
Masters in Data Science
San Jose State University
San Jose, US
karishma.kuria@sjsu.edu

***Abstract*—This electronic document is a "live" template and already defines the components of your paper [title, text, heads, etc.] in its style sheet. **CRITICAL: Do Not Use Symbols, Special Characters, Footnotes, or Math in Paper Title or Abstract.* (*Abstract*)**

Keywords—component, formatting, style, styling, insert (*key words****)***

Introduction

Football is a trendy and most engaging game worldwide. There is no surprise that it is the most followed sport worldwide, having around 3.5 billion followers. It was invented in England in the nineteenth century and is also known as soccer. We grew up watching this sport, supporting our favorite teams, and cheering our favorite players. Being a sports enthusiast, it used to be mostly the subject of discussion about which team will win according to pre-half statistics. Could he have played center-back instead of left-back? Will he miss or hit the penalty? Slowly we saw how sports channels used to show the prediction of match-winner according to the historical analysis. We did not realize it back then, but we used to do lots of research and forecasts. It inspired us to start this project as it would be great to combine skills and interests. What could be more exciting and helpful to know about our favorite sport?

An application would be developing a system to identify the best and worst-performing teams based on their games - (Red Card Vs. Goal) using clustering technique and identifying the "GOLDEN CLUSTER." Another example would be Soccer Match Outcome/Goals Prediction based on their style of play. We will also cover the analysis/ questions below using the best classifier to compare the performance matrix and make predictions.

General statical analysis questions that we will try to analyze and answer with this project are below.

1. What are the most resultative leagues?
2. What are the most resultative teams?
3. What are the most missing teams?
4. Which players are the best finishers?
5. Analyzing Goals Scored?
6. Analyzing Yellow/Red Cards?
7. Analyzing Penalties?
8. Which players have the most "expected goals"?
9. Which players are the worst at deciding their shots?

Experiment

1. Data understanding and fundamental analysis, including detailed data visualization with univariate and bivariate feature analysis.
2. Feature transformation and engineering: categorical features via encoding and numerical feature scaling. We transformed features and added new features to the dataset via amalgamations.
3. We have made dimensionality reduction implementation via PCA and showed team similarity.
4. Define a Golden cluster and use Fractal Clustering to find it based on the business case you formulate.
5. Implemented two amalgamations and made data enrichment.
6. Feature importance with varying methods like SHAP values and GINI score.
7. We have researched latent variables which could help the dataset to get better performance and Sharpe ratio for clustering more understandable.
8. We have run classification/ regression with muller loop and evaluated the performance with different performance metrics.
9. We concluded by performing Cross fold validation for optimization and visualizing the Confusion metrics for performance evaluation.

10. We have used Experience Dashboard to give an interactive experience for both Classification and Regression results.
11. Interactive Dashboard experience for the results of both Regression and Classification algorithms used.
12. The dashboard has multiple tabs to show the model's performance, f1 score, confusion matrix, SHAP results, decision tree, etc.
13. It also contains various methods used to check the feature importance such as using SHAP values and picking the top features relevant for the prediction.
14. Smote has been used to do up-sampling and down-sampling of data.

Data

The dataset results from a tedious web-scraping effort and integrating different data sources. The central element is the text commentary. All events were derived by reverse-engineering the text commentary using regex. It includes events from more than 7,000 games from the top 5 European Leagues.

The dataset provides a granular view of 9,074 games, totaling 941,009 events from the most significant 5 European football (soccer) leagues: England, Spain, Germany, Italy, France from 2011/the 2012 season to 2016/2017 season as of 25.01.2017.

The dataset is organized into 3 files:

1. events.csv contains event data about each game. Text commentary was scraped from: bbc.com, espn.com, and onefootball.com
2. ginf.csv includes metadata and market odds about each game. Odds were collected from oddsportal.com
3. dictionary.txt contains a dictionary with the textual description of each categorical variable coded with integers.

We have used a second dataset for classification and regression, as with the first dataset, we were not achieving good results. The first dataset was chosen because it has the highest level of detail and the lowest levels of missing values and erroneous data. The second dataset includes all matches played in six European countries, including Portugal, spanning five years from 2012 to 2017.

Scraped Dataset: The game information from 2009 - 2019 from datahub.io is also scraped to better analyze the clustering and regression problem.

Exploratory Data Analysis and Visualization

The features from the event dataset are pre-processed to include problem statement-specific features to attain the project objective.

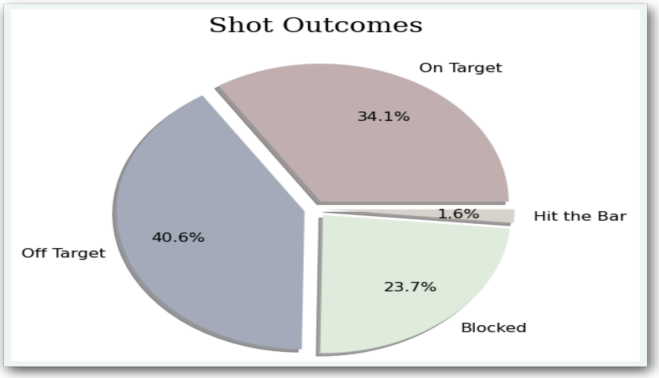


Fig 1

In the pie chart (fig 1), on analyzing the shot outcome results, it can be observed that most of the shots are off-target, but the difference with on-target shows is marginal. Only a fraction of those on-target shots end up as goals, since goalkeeper tries to prevent goals on the other side.

Identify applicable funding agency here. If none, delete this text box.

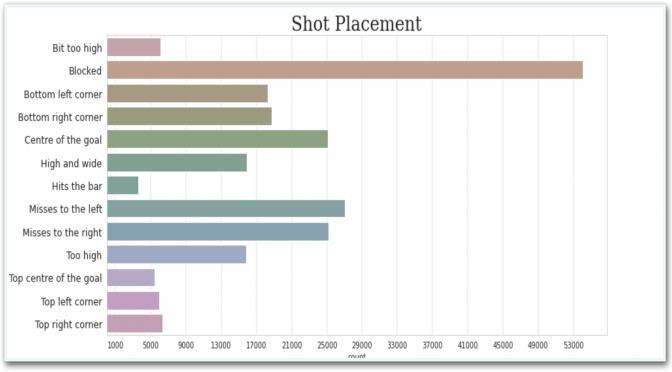


Fig 2

Looking at shot placements in fig 2, it can be noted that most of the shots are blocked.

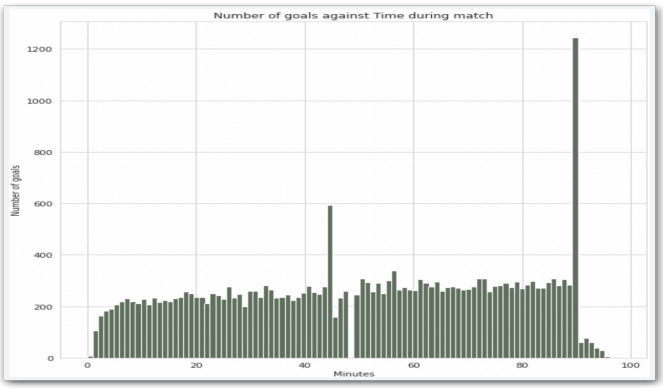


Fig 3

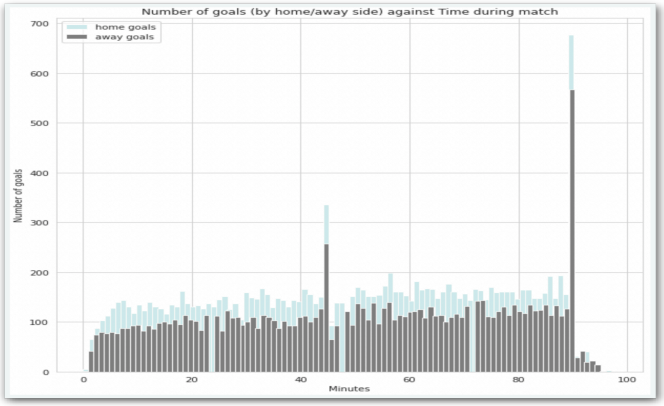


Fig 4

For number of goals against time in a match, it is found that most goals are scored around the Half-Time (45mins + extra time) and around Full-Time (90mins + extra time) (Fig 3). For every minute, most of the goals scored are by the home side. This supports the general notion that the home side has a statistical advantage. (Fig 4).

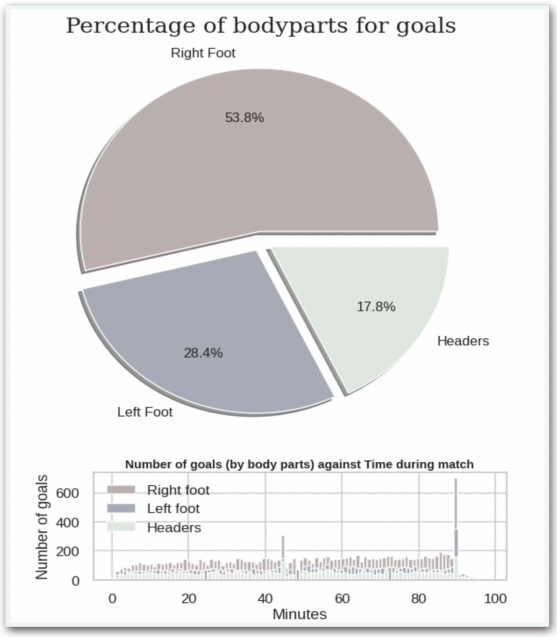


Fig 5

Fig 5. Most of the goals scored are by Right Footed, then followed by Left Footed and lastly, by Headers. Perhaps, this might be because most humans are right-footed and, hence, most players are right-footed.

It is also not surprising that most goals have been scored by foot not head, as after all, soccer is meant to be played by foot.

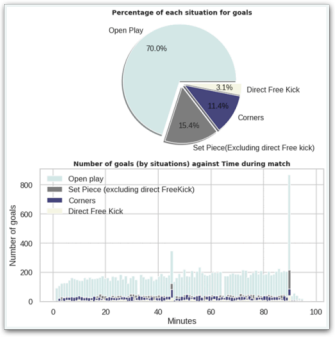


Fig 6

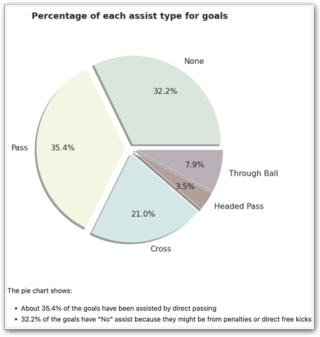


Fig 7

Fig 6, Nearly 70.8% of the goals scored are from Open Play. The pie chart in Fig 7 shows:

About 35.4% of the goals have been assisted by direct passing 32.2% of the goals have "No" assist because they might be from penalties or direct free kicks.

Feature Analysis

Univariate Analysis

We use the simplest form of analysis - univariate analysis to see out information of one feature at a time.

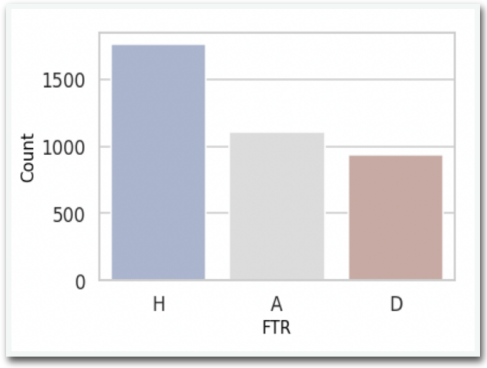
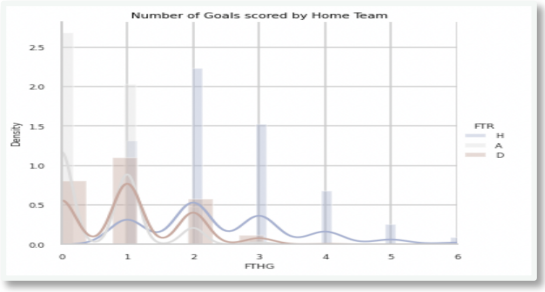
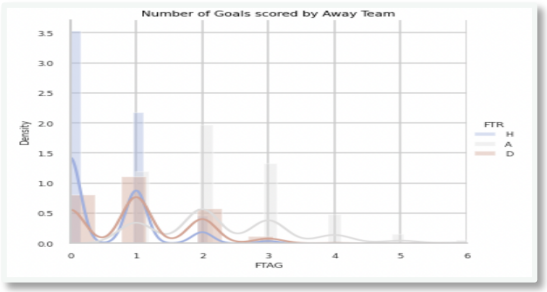


Fig 10

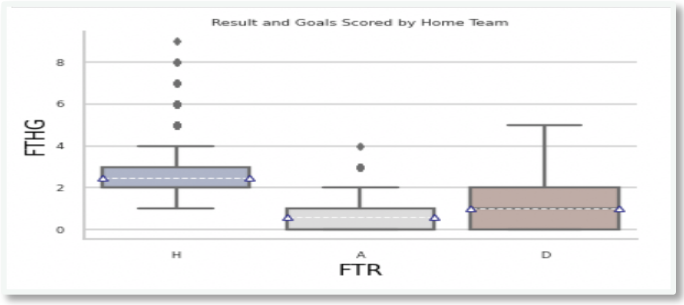
Using univariate analysis, we can observe that features such as:- Full-time Home Goal, Full Time Away Goal, Half time Away Goal, Half Time Home Goal, Total shot ratio, Home Hit Rate, and Away Hit Rate are Gaussian distributed. Hence

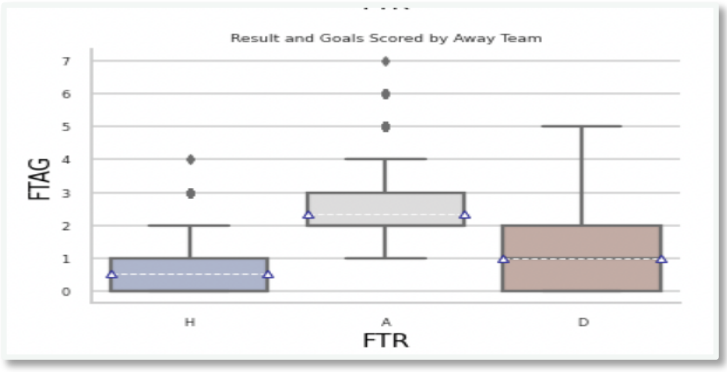


MinMax scaling is required.
Most of the time, Both the Home and Away Team score 1 goal, Frequent scores are (1,0,2 in this sequence). The away team is



slightly ahead here. However, when it comes to more than 2 goals, Home Teams are ahead.



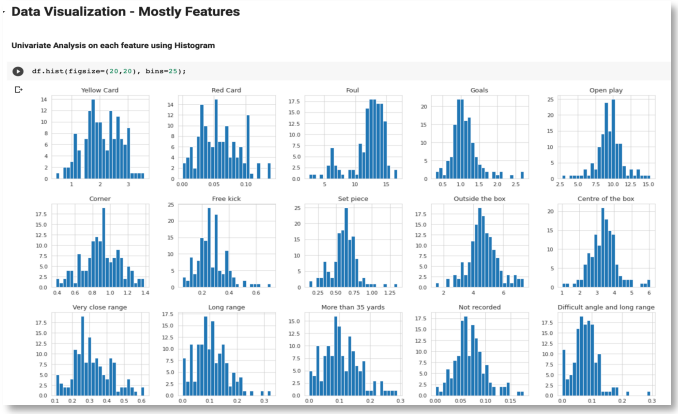


Home Team:

1. While winning score mean of 2.5 Goals.
2. When Drawn mean is 0.9 Goals
3. When Lost Mean is 0.5 Goals.

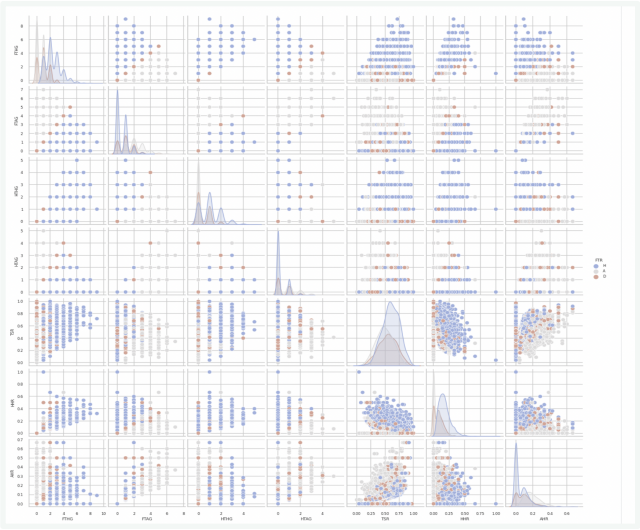
Away Team:

1. While winning score mean of 2.1 Goals.
2. When Drawn mean is 0.9 Goals.
3. When Lost Mean is 0.6 Goals -> Overall Home Team is Scoring more goals, while will be a huge factor in winning the game.

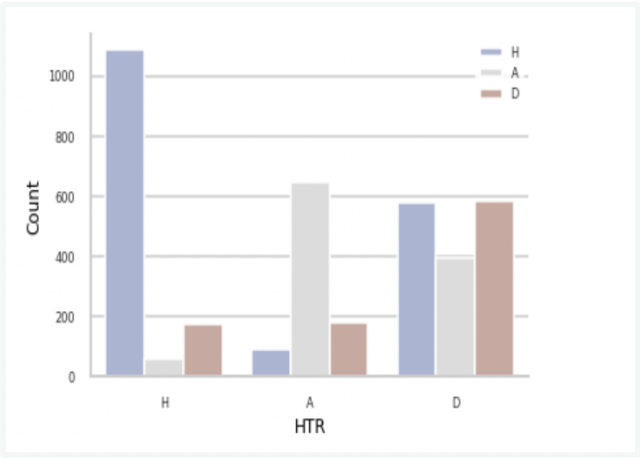


Bi-Variate Analysis

- Bivariate Analysis:



For Bivariate analysis, each feature is plotted again remaining features to understand the relationship between each other.
FTHG (Full-time Home Goal) and FTAG (Full time Away Goal) are the values that clearly indicate who will win.
So, studying these 2 variables can be the best way to predict FTR. Whichever feature has a higher value that the team wins which indicates that the team which scores more Goals at Full Time wins the match? Basically, this is how football works.



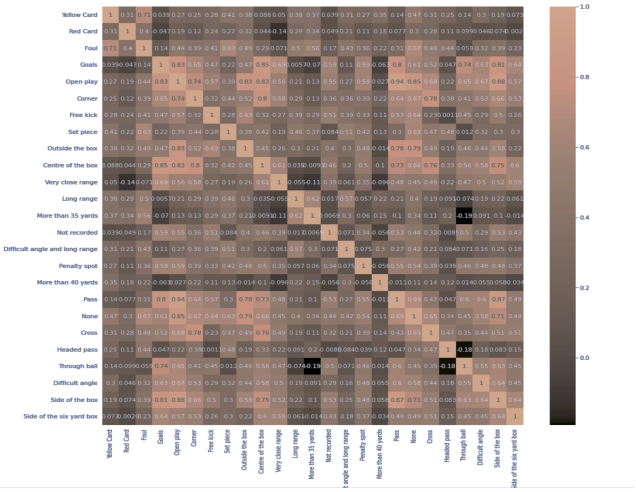
The following points is observed from the bar graph:

- The Team Leading at Half Time almost always goes on to win the game at Full time.
- If the game is level at Half Time, it is the more likely home team will win than the Away team. Although the most likely outcome is a Draw only.

So HTR is a very important variable to determine who wins at Full time.
And below mentioned points can be concluded: -

- There is a higher percentage of home team winning, so clearly, the team playing at Home has an advantage.
- Goals Scored at Full time (FTHG - Home Goals, FTAG - Away Goals) determine FTR - Full Time Result i.e., which team will go on to win the game, a team which scores more Goals at FT wins the match.
- The Home team usually scores more goals. Ex While winning home team scored a mean of 2.5 Goals as compared to 2.1 Goals by the Away team while winning.

  HTR (Half Time Result) is a very important variable to determine who wins at Full time. As we saw the Team winning at Half team does not usually end up Losing at Full time. So, this Variable can effectively predict who is likely to win at full time.



Data Cleaning

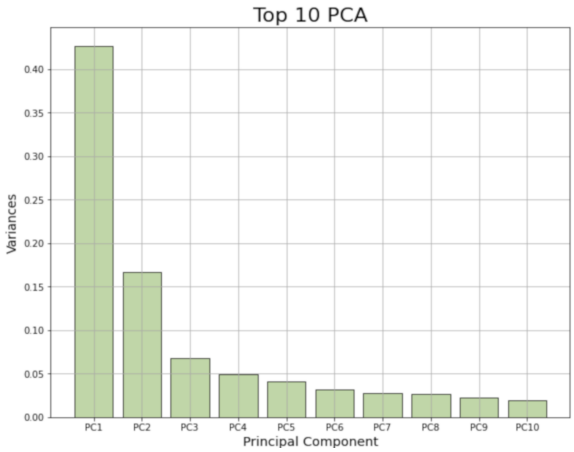
In the data cleaning step, the dataset for analysis is prepped by removing or modifying the data is incorrect, incomplete, irrelevant, duplicated or improperly formatted. Checking for missing or null values, converting the categorical value to

numeric is performed. Additional checking for if data is balanced or not (pair plot) or whether there is a need to whiten out the data (remove different weights in the data)

Feature Selection:

Dimension Reduction technique:

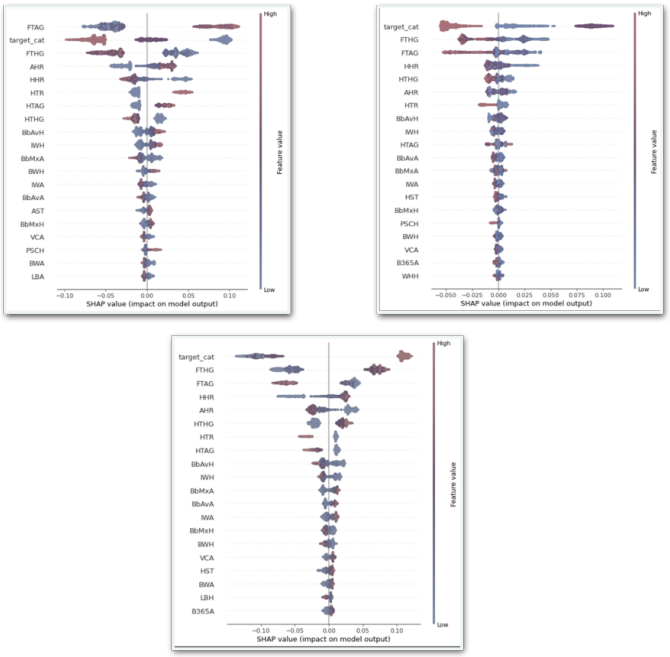
Principal Component Analysis:



Using PCA we will be losing important features for clustering. Hence dropping PCA and performing clustering based on our Problem statement

In lines to problem statement, we have focused on two important features - Red Cards and Goals for the clustering problem statement i.e., applying Fractal Clustering to identify best and worst football teams on these features.
In the first step of data pre-processing, event dataset is filtered on shot outcome as 'on target' to get all rows associated with goals and then, grouped on teams. The counts of yellow cards, red cards, fouls and goals each team during the match are calculated and used for further processing. Using Dictionary dataset, the features like 'situation','location','assist method' are renamed with more user-friendly names.
For classification problem, scraped dataset provides better accuracy and finer details for matches spanning from 2009 to 2019.

We use SHAP technique for Feature Selection by importing summary plot and TreeExplainer from SHAP libraries. Run one of the classifiers and input the fitted model to TreeExplainer to output SHAP values. Proceed to summary_plot that will show feature rankings based on SHAP values on a per class basis. For class 3 this will be



We can see the top Features on the top of the graph.

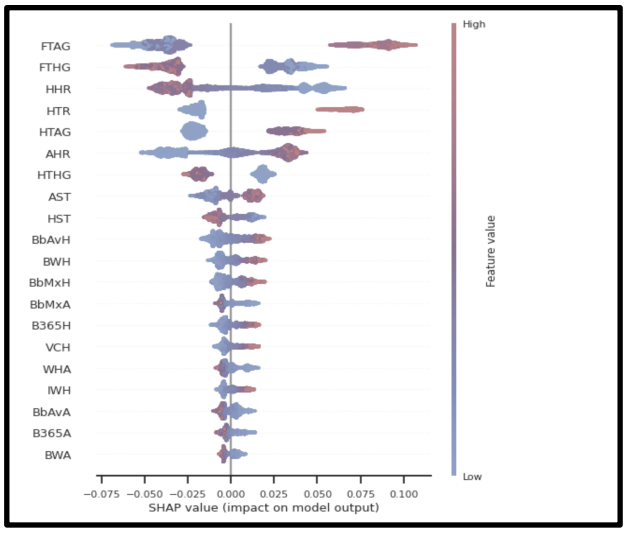
Although there are many features in the game information dataset, we will consider the elements related to the football games and discard the information about various betting scores for machine learning. Therefore, the following features are included - Full-Time Home Goal, Full Time Away Goal, Half time Away Goal, Half Time Home Goal, Full-Time Result, Half Time Result, Total Shot Ratio, Home Hit Rate, and Away Hit Rate.

```
importance = feature_importance(decisionTree, X)
importance[:]
```

	Importance
FTHG	0.435786
AHR	0.281177
FTAG	0.229547
HHR	0.053489
HTHG	0.000000
HTAG	0.000000
HTR	0.000000
TSR	0.000000

Algorithms we used to select features: Gini score and Shap values.

The below explanation shows features each contributing to pushing the model output from the base value (the average model output over the training dataset we passed) to the model output. Features pushing the prediction higher are shown in red. Those making the prediction lower are in blue.



Clustering

Clustering Techniques:
Now the dataset is ready for modeling. We will scale the dataset with MinMaxScaler and apply K-Means Clustering.

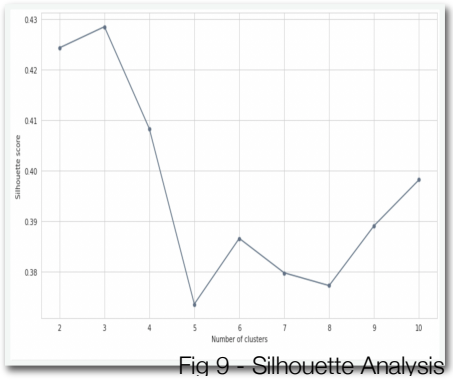
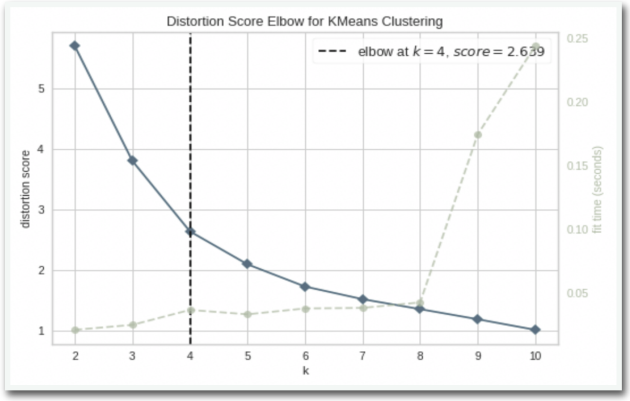


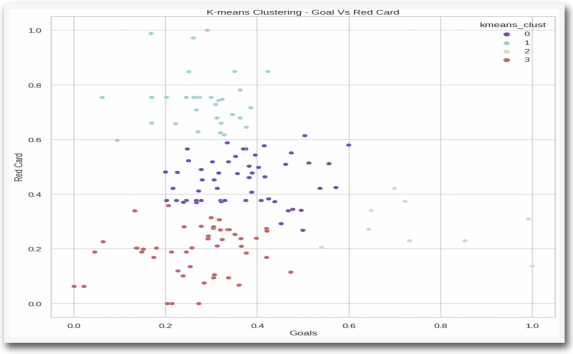
Fig 9 - Silhouette Analysis

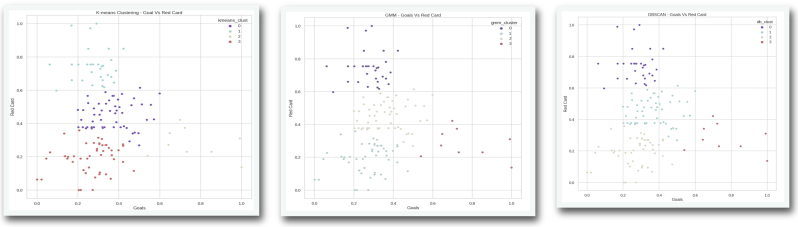


Elbow Method

K-Means is implemented with cluster k=4.

From the given scatterplot, the following clusters can be observed: -



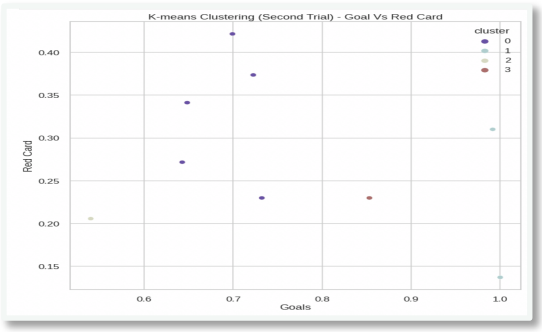


- Cluster 0 - Team with Average Goals and an Average number of Red Cards.
- Cluster 1 - Teams with Fewer Goals but the highest number of Red Cards. This team calls for special attention for a good coach.
- Cluster 2 - Teams with more Goals and more miniature Red Cards - This indicates a Good Team.
- Cluster 3 - Team with Less Goal and Less Red Cards.

We apply various clustering techniques - KMEANs, Gaussian, and DBSCAN - results.

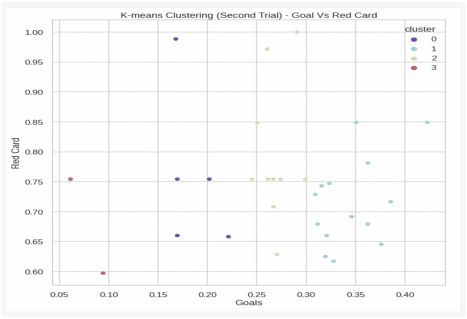
FRACTAL CLUSTERING

The next step is to apply Fractal Clustering to identify the best and worst-performing soccer team with a Good Goal score to Red Card ratio (aka Sharpe Ratio). i.e., GOLDEN CLUSTER.



Identify Top 2 Best Performing Soccer teams based on Goals scored to accumulated Red Cards ratio by applying K-means clustering repetitively on the resultant cluster till the objective is achieved.

From the plot, cluster 1 represents the group of best soccer teams in terms of the Goal-RedCard ratio.



Identify Top 2 Worst Performing Soccer teams based on Goals scored to accumulated Red Cards ratio by applying K-means clustering repetitively on the resultant cluster till a group of lousy performing teams are found.

From the plot, cluster 3 represents the wrong performing soccer team group regarding the Goal-RedCard ratio.

Latent Variables and Manifolds

Latent variables are introduced when it is determined that such features are not directly observable and potentially affect the response variable. Based on the selected football domain following three latent variables are identified.

1. Latent variable #1: The Total Shots Ratio (TSR) is used to determine how well teams fare in a match when taking and conceding shots.

   The following formula determines the TSR:

   TSR = Total Shots for / (Total Shots for + Total Shots against)

2. Latent Variable# 2: Home Hit Rate determines how home teams serve a goal against shots played.

   The following formula determines the HHR:

   Home Hit Rate = Full Time Home Goal / Home Shots

3. Latent Variable# 3: Away Hit Rate determines how away teams serve a goal against shots played.

   The following formula determines the AWR:

   Away Hit Rate = Full Time Away Goal / Away Shots.

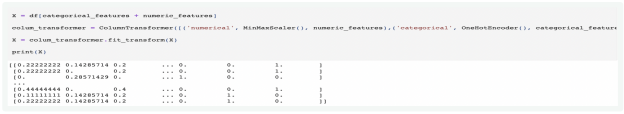
Regression

The main objective of classification and regression problems is as follows:

1. To find which team wins the match. So we have used Full-Time Result as the target variable for Regression and Classification Algorithms. For doing feature selection and model explainability, we have used SHAP values and selected the top-scoring feature for model training.
2. Predict if a shot is a goal or no goal.

Before Data Modeling, encode the categorical feature with OneHot Encoding and scale numerical features using MinMax Scaler. Use Column Transformer as shown to transform all the columns before splitting the dataset into training and test set.

- For Regression , implement Muller Loop to run the training dataset against - “ "MLPRegressor", "LinearRegression", "RandomForestRegressor", "KNNRegressor", "LogisticRegression", "AdaBoost", "Multinomial Naive Bayes" and display their accuracy.
- Popular regression algorithms is selected as follows:



- MLP Regressor: it is a Multi-Layer Perceptron regressor that optimizes the square error using LBFGS or stochastic gradient descent.
- Linear Regressor: it is an ordinary least squares Linear Regression.

- KNN Regressor: This algorithm learning is based on K nearest neighbors of each query point, where k is an integer value specified by the user.
- Random Forest: It is basically a set of decision trees the randomly selected subset of the training dataset. Then it collects the votes from different decision trees to decide the final value or label.
- Logistic Regressor: This algorithm measures the relationship between the categorical dependent variables and one or more independent variables, by estimating the probability of occurrence of an event, using its logistics function.:
- AdaBoost: Is an optimized distributed gradient boosting algorithm popular for its efficiency and flexibility. It provides parallel tree boosting to solve many data science problems.
- Multinomial Naive Bayes algorithm is a probabilistic learning method that is mostly used in Natural Language Processing (NLP).

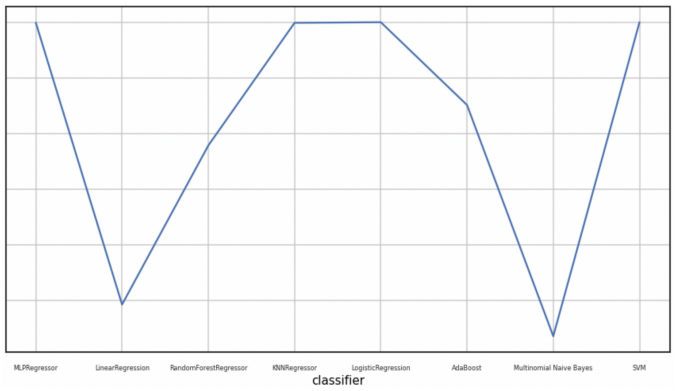
Regression before adding Latent Variables

The regression outcome after running the Muller loop:

Regression Classifier = MLPRegressor, Score (test, accuracy) = 99.96, Training time = 1.34 seconds
Regression Classifier = LinearRegression, Score (test, accuracy) = 74.62, Training time = 0.01 seconds
Regression Classifier = RandomForestRegressor, Score (test, accuracy) = 88.91, Training time = 0.33 seconds
Regression Classifier = KNNRegressor, Score (test, accuracy) = 99.94, Training time = 0.09 seconds
Regression Classifier = LogisticRegression, Score (test, accuracy) = 100.00, Training time = 0.17 seconds
Regression Classifier = AdaBoost, Score (test, accuracy) = 92.55, Training time = 0.75 seconds
Regression Classifier = Multinomial Naive Bayes, Score (test, accuracy) = 71.77, Training time = 0.01 seconds
Regression Classifier = SVM, Score (test, accuracy) = 100.00, Training time = 0.26 seconds

---

Best --> Regression Classifier = LogisticRegression, Score (test, accuracy) = 100.00



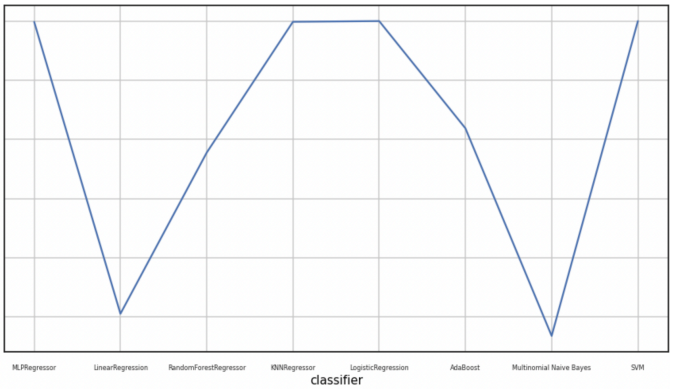
Regression after adding Latent Variables

The regression outcome after running the Muller loop:

Regression Classifier = MLPRegressor, Score (test, accuracy) = 99.93, Training time = 5.06 seconds
Regression Classifier = LinearRegression, Score (test, accuracy) = 75.28, Training time = 0.02 seconds
Regression Classifier = RandomForestRegressor, Score (test, accuracy) = 88.88, Training time = 1.61 seconds
Regression Classifier = KNNRegressor, Score (test, accuracy) = 99.94, Training time = 0.41 seconds
Regression Classifier = LogisticRegression, Score (test, accuracy) = 100.00, Training time = 1.11 seconds
Regression Classifier = AdaBoost, Score (test, accuracy) = 90.94, Training time = 4.37 seconds
Regression Classifier = Multinomial Naive Bayes, Score (test, accuracy) = 73.41, Training time = 0.02 seconds
Regression Classifier = SVM, Score (test, accuracy) = 100.00, Training time = 1.39 seconds

---

Best --> Regression Classifier = LogisticRegression, Score (test, accuracy) = 100.00



Inference:
On comparing the accuracy before and after adding latent variables, we can conclude the following:-

- AdaBoost showed improved accuracy from 71% to 90.94%.
- Linear regression and Multinomial Naive Bayes showed a slight increase in accuracy to 75.28% and 73%
- Accuracy for RandomForest, KNN and MLP regression remained constant.
- Logistic regression and SVM continue to score 100%

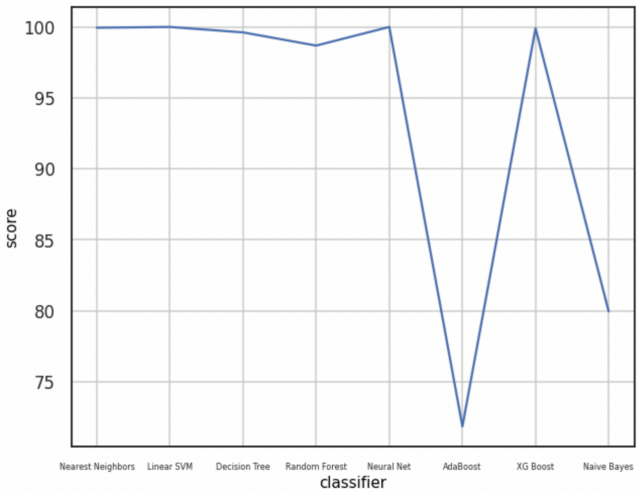
Classification

For Classification, we have selected the following algorithms:

- K Nearest Neighbors: It is used to create models which does the prediction based on similarity. It stores all the available cases provided in the training dataset and classifies the new data based on similarity.
- Gaussian Naïve Bayes: It is a probabilistic algorithm used for classification. It is based on probability models that incorporate strong independence assumptions.
- Random Forest: It is basically a set of decision trees the randomly selected subset of the training dataset. Then it collects the votes from different decision trees to decide the final value or label
- XGBoost: Is an optimized distributed gradient boosting algorithm popular for its efficiency and flexibility.It provides parallel tree boosting to solve many data science problems.
- Decision Tree: It is a Supervised Machine Learning Algorithm that uses a set of rules to make decisions, similarly to how humans make decisions.
- ADABoost: Ada-boost classifier combines weak classifier algorithm to form strong classifier.
- Linear SVM: Linear SVM is a generalization of Maximal Margin Classifier.
- Neural net: MLPClassifier trains iteratively since at each time step the partial derivatives of the loss function with respect to the model parameters are computed to update the parameters.

Ensemble learning: We are finally using the Stacking method which is an ensemble technique that uses predictions from multiple nodes to build a new model. The new model in our project corresponds to voting classifier which will decide the prediction label for the given data depending on voting.

Popular classification algorithms are applied to the reduced dataset, using classification techniques. The optimized dataset is split into testing and training datasets prior to applying the algorithm. The algorithm is then trained with the training dataset, and the trained classifier is applied in the testing phase. The objective is to use the algorithm/classifier with the best score and use it for crop prediction or to come up with an ensemble-based voting technique to classify the label for given parameters.

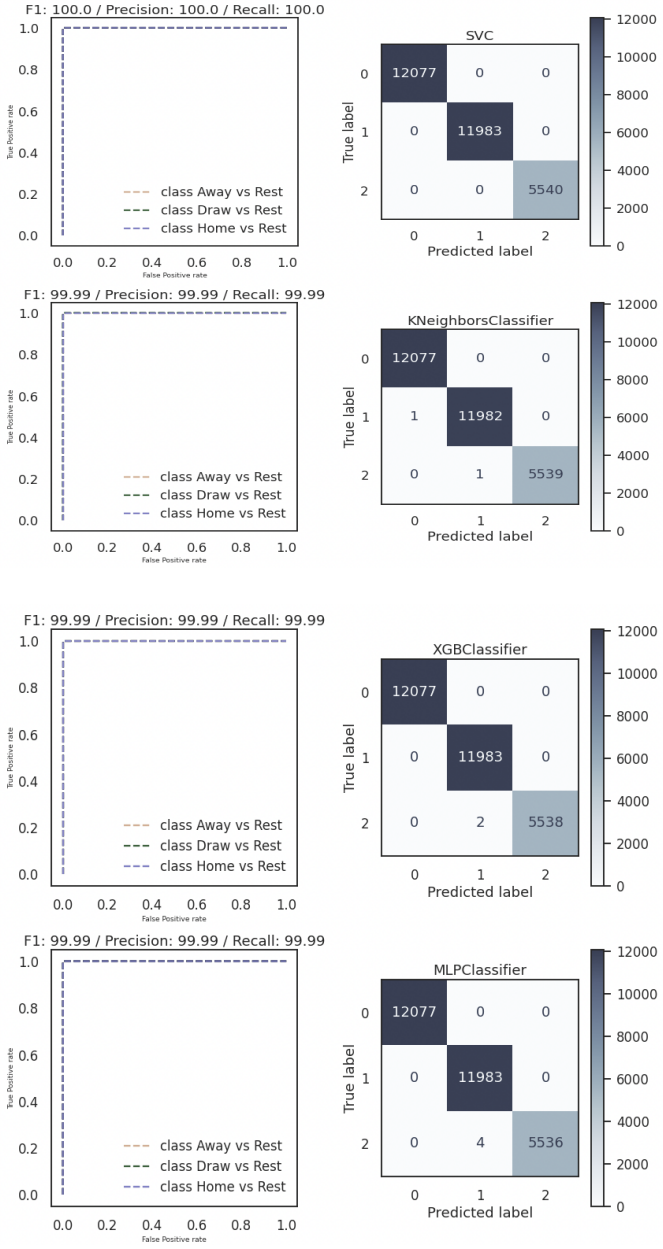


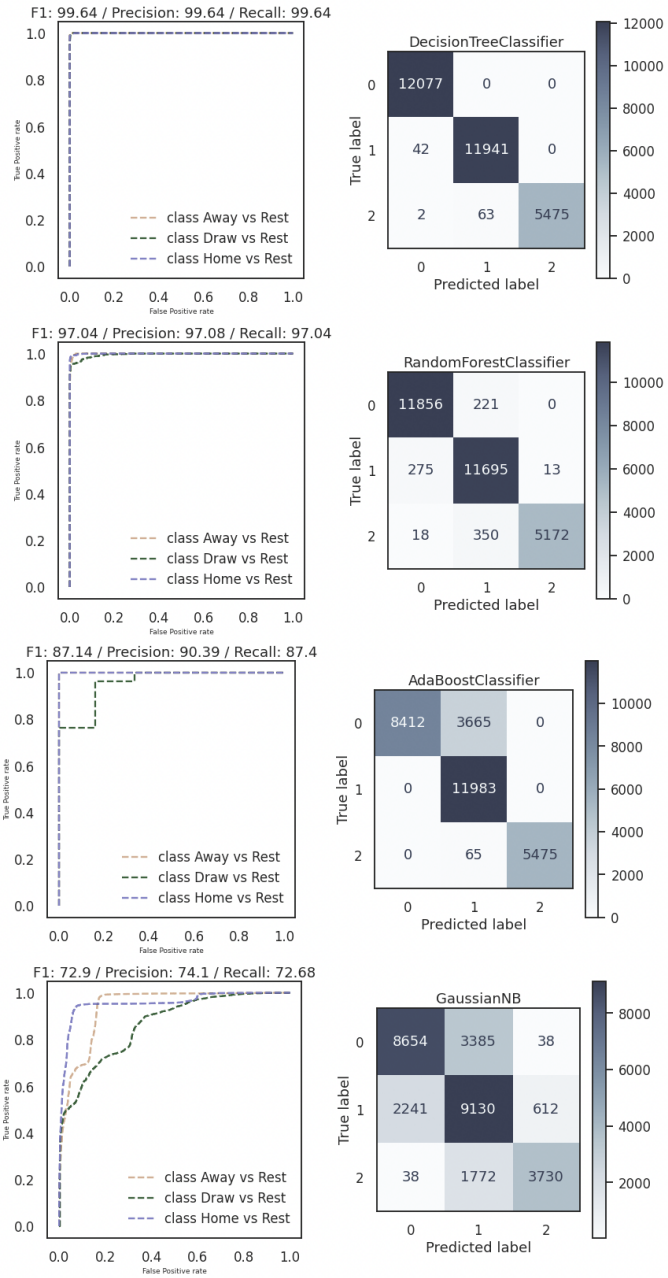
From the above plot, it is concluded that Linear SVM gives the highest accuracy.

Inference:

- Above stats shows that Neural Net and Linear SVM has 100% accuracy.
- Followed by KNN, Decision Tree, XG Boost and Random Forest which has accuracy around 99 %.
- Gaussian Naive Bayes has accuracy of 79.96 % and followed by Adaboost which has the least accuracy of 71%.85

```
          Classifier    Accuracy   ROC_AUC
1         Linear SVM  100.000000  1.000000
0  Nearest Neighbors   99.993243  1.000000
6            XG Boost  99.993243  1.000000
4          Neural Net  99.986486  1.000000
2      Decision Tree   99.638514  0.999917
3      Random Forest   97.037162  0.998121
5           AdaBoost   87.398649  0.985950
7        Naive Bayes   72.682432  0.919365
Best --> Classifier = Linear SVM, Score (test, accuracy) = 100.00
```





Inference:

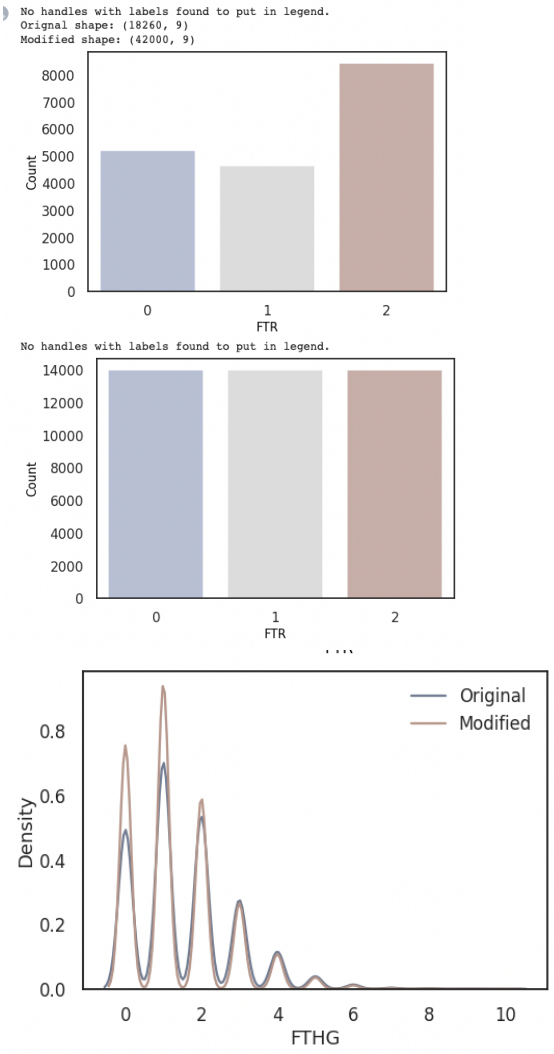
- Above stats shows that Neural Net and Linear SVM has 100% accuracy.
- Followed by KNN, Decision Tree, XG Boost and Random Forest which has accuracy around 99 %.
- Gaussian Naive Bayes has accuracy of 79.96 % and F1 score of 72.9 and followed by Adaboost which has the least accuracy of 71.85% and better F1 score of 87.14
- The above confusion matrix shows the result for various classification algorithms. It can be noted that

the model has provided good accuracy with all the algorithms used.

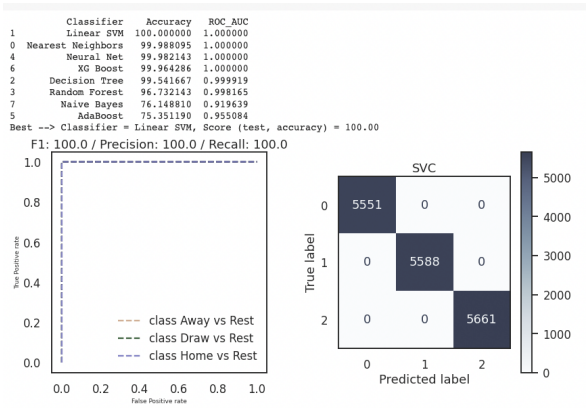
Understanding the significance of data distribution by sampling

Steps

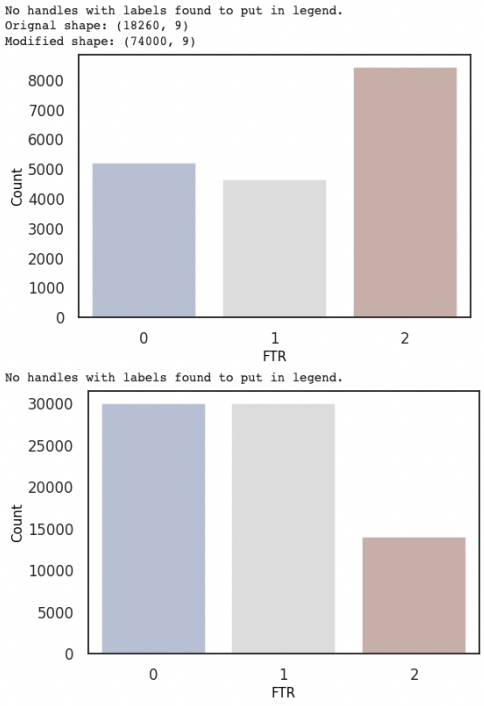
1. Upsampling the dataset to have each classes to have 14000 outcomes.
2. After up sampling the dataset to have each classes to have 14000 outcomes, we will train the dataset in a muller loop for classification and store all the performance metrics including - f1 score for all the algorithms.

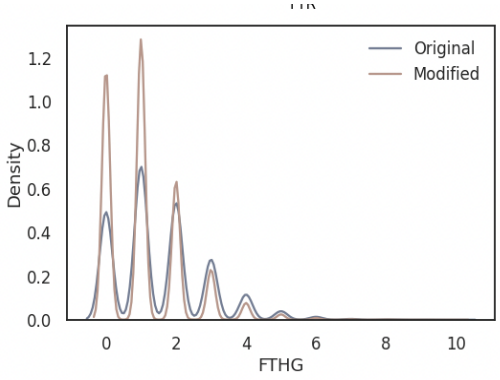


Result:

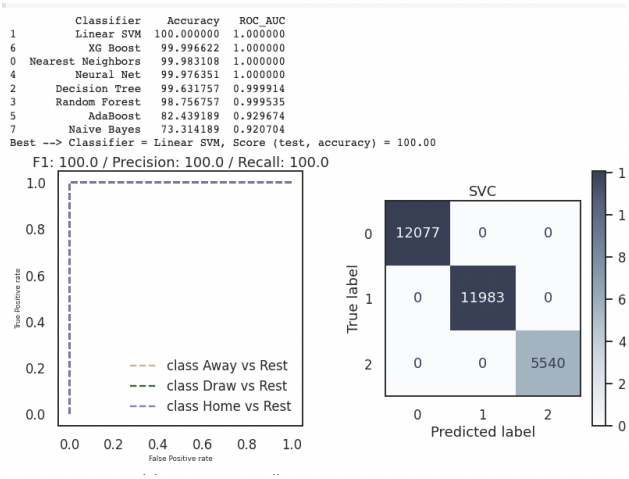


3. Now we will change the data distribution.
4. We will re-train the dataset in a muller loop for classification and store all the performance metrics including - f1 score for all the algorithms.





Result:

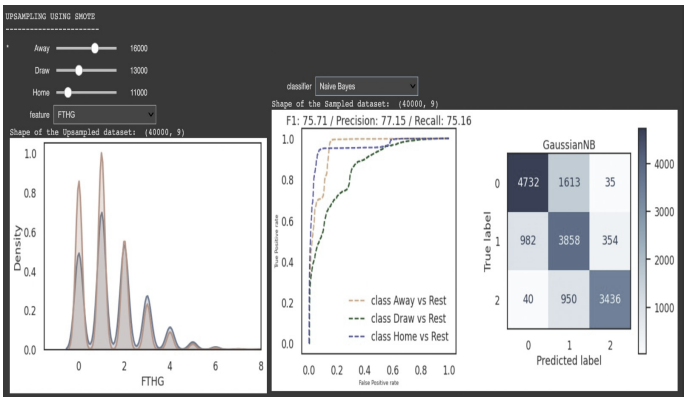


Inference:

- SVC, KNN, Neural Net, XG Boost, Decision Tree and Random Forest continue to show good F1 scores and better prediction capability.
- Whereas performance of Gaussian NB decreased to F1 score of 73.53 and Performance of AdaBoost increased to F1 score of 82.48.

We will repeat the same step for each feature (1-3) using IPYWIDGETS.

Up-sampling Dashboard

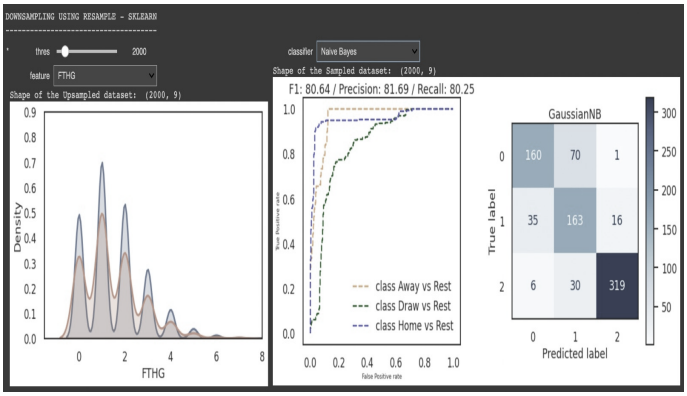


Inferences

- We can see that without Up-sampling, the F1 score of KNN is 100% and with the down-sampling, the F1 score slightly decreases. The similar trend is noticed with other algorithms - DecisionTree, Random Forest, Neural Network and Gaussian NB.
- Surprisingly, Gaussian Bayes shows improvement in F1 score of 77.7.
- This similar trend is also noticed when threshold is leaned more towards "Draw" or "Home Win" class.

On further analysis, the equally same up-sampled classes does not show any significant improvement in the algorithm performances except for ADABOOST where the performance increases gradually.

Down-sampling Dashboard



Inferences

- We can see that without Down-sampling, the F1 score of KNN is 100% and with the down-sampling, the F1 score slightly decreases. The similar trend is noticed with

other algorithms - DecisionTree, Random Forest, Neural Network and Gaussian NB.

- Surprisingly, Gaussian Bayes shows improvement in F1 score of 77.7

The equally same down-sampled classes does not show any significant improvement in the algorithm performances.

Data Narrative:

We have used the unsupervised learning technique Clustering to identify the groups of soccer teams similar to each other concerning their gaming styles.

We apply Fractal Clustering to identify best and worst-performing teams based on their games - Red Card Vs. The goal uses Clustering and identifying the "GOLDEN CLUSTER".

We observed that for every minute, most of the goals scored are by the home side This supports the general notion that the home side has a statistical advantage.

The maximum number of goals were scored between 40 to 50 minutes in a match by both the Home and Away Team.

Most of the goals scored are by Right Footed, then followed by Left Footed, and lastly, by Headers. Perhaps, this might be because the majority of humans are right-footed and, hence, most players are right-footed.

It is also not surprising that most goals have been scored by foot not head, as after all, soccer is meant to be played by foot.

About 35.4% of the goals have been assisted by direct passing

32.2% of the goals have "No" assist because they might be from penalties or direct free kicks

Clearly Lionel Messi and Ronaldo are the most Offensive players in the top 10 most offensive teams in the league.

Observations about Home Team and Away Team

Home Team :

- While winning score mean of 2.5 Goals.
- When Drawn mean is 0.9 Goals
- When Lost Mean is 0.5 Goals

Away Team :

- While winning score mean of 2.1 Goals.
- When Drawn mean is 0.9 Goals.
- When Lost Mean is 0.6 Goals

Overall Home Team is Scoring more goals, While will be a huge factor in winning the game.

Objective Functions:

1. Identified and grouped Soccer Teams based on their Goals and Red Cards received throughout their games.
2. Identified the Top 3 Best Performing Soccer teams based on Goals scored and accumulated Red Cards.
3. Identified Worst Performing Soccer teams based on Goals scored and accumulated Red Cards.

Also, Regression and Classification technique would be applied to model a solution to predict whether the Home team or Away Team will win the match.

We also look into the questions like below using the best classifier to compare through the performance matrix and make predictions:-

- What are the most resultative leagues?-
  With the mean value, we can see that EPL(England's league) is not as resultative as Bundesliga(Germany's league), and we took average because there are 20 teams in EPL, but only 18 in 1st Bundesliga. So the mean value is a better score to compare.

- What are the most resultative teams?

  Real Madrid 604.0
  Barcelona 602.0
  Bayern Munich 469.0
  Manchester City 458.0
  Paris Saint-Germain 454.0

- What are the most missing teams?

  Rayo Vallecano 360.0
  Werder Bremen 356.0
  Granada 341.0
  Sunderland 317.0
  Aston Villa 316.0
  Hamburg SV 315.0
  TSG Hoffenheim 312.0

- Which players are the best finishers?

The best free kickers

```
free_kicks = goals[goals.situation == 4]
best_kickers = free_kicks.groupby('player')
best_kickers.head(20)
```

```
player
lionel messi            14
cristiano ronaldo       13
miralem pjanic          13
andrea pirlo            12
```

- Analyzing Penalties?

Penalties analysis

```
penalties_goals = goals[goals.location
penalties_scored = penalties_goals.gro
penalties_scored.head(20)
```

```
player
cristiano ronaldo       43
zlatan ibrahimovic      35
lionel messi            30
edinson cavani          20
```

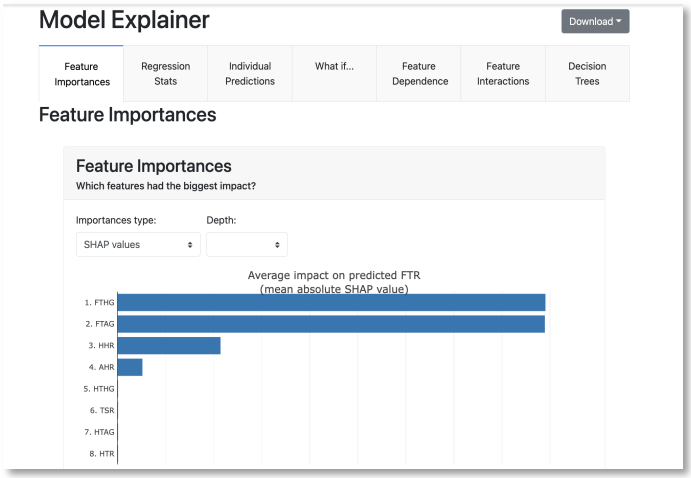
- Which players have the most “expected goals”?

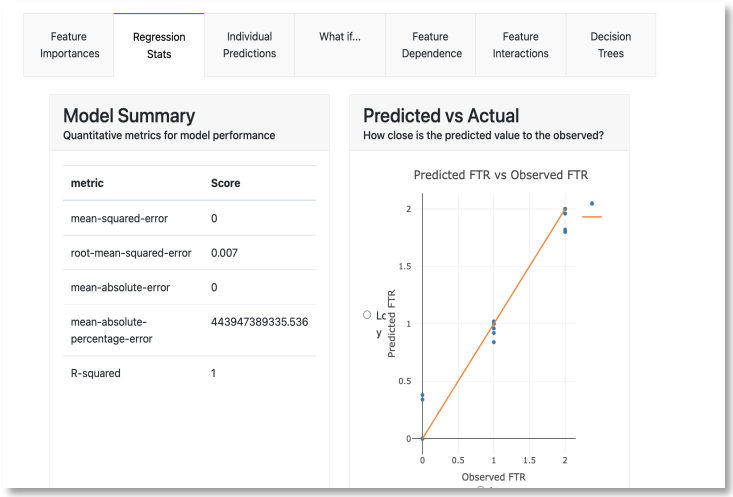
Most Attempts

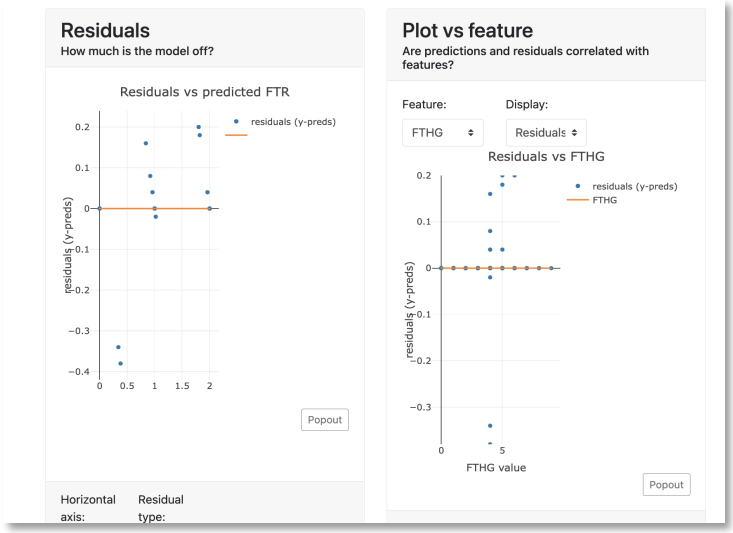
```
# By player
attempts = events[events.event_type
attempts.groupby('player').player.
```

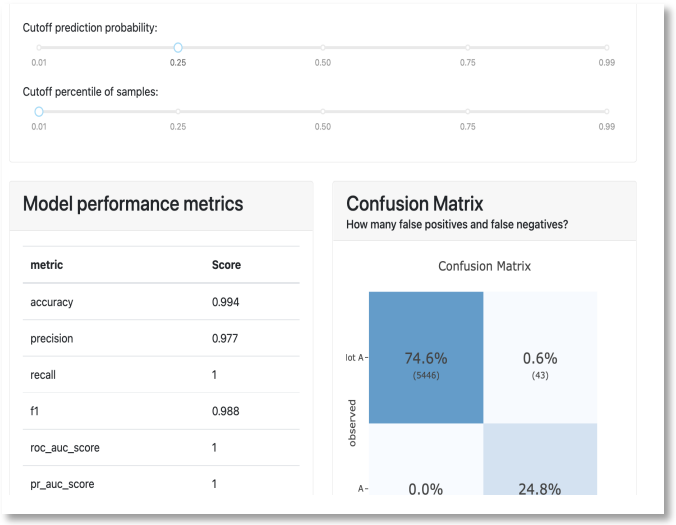
```
player
cristiano ronaldo           1190
lionel messi                 914
zlatan ibrahimovic           774
robert lewandowski           633
edinson cavani               623
```

EXPLAINER DASHBOARD:









Conclusion

The empirical result shows that we can classify the outcome of the football match accurately by applying the given predictive model on the game info provided by the Sports News channel. This study can also be extended to include player performance if the feature extraction is done with a large

Acknowledgment

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks ...”. Instead, try “R. B. G. thanks...”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

References

[1] Applying Data Mining Techniques to Football Data from European Championships- https://paginas.fe.up.pt/~prodei/comic06/p6.pdf

[2] SPORTS DATA MINING - https://eller.arizona.edu/departments-research/centers-labs/artificial-intelligence/research/previous/sports-data-mining

[3] Data Mining in Sports: A Systematic Review - https://www.researchgate.net/publication/323198458_Data_Mining_in_Sports_A_Systematic_Review

[4]

[5] https://scikit-learn.org/

[6]]kaggle.com

[7] datahub.io

[8] scikit-learn.org

[9] https://towardsdatascience.com/

[10] https://medium.com/

IEEE conference templates contain guidance text for composing and formatting conference papers. Please ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove template text from your paper may result in your paper not being publish