PREDICTIVE ANALYTICS OF FOOTBALL

Salil Chincholikar[1], Niharika Sadul[2], Parth Thakkar[3], Ashutosh Amrutkar[4], Mr. Nikhil Dhavase[5]

[1,2,3,4]*(Student, Dept. of Information Technology, MMCOE, Pune, Maharashtra)*
[5]*(Asst. Professor, Dept. of Information Technology, MMCOE, Pune, Maharashtra)*

---***---

Abstract - Collective analysis of viewer reach and its sources can help the world of football optimize the way in which they decide to broadcast their matches. It can allow us to create a completely different user experience to enhance football with the help of analytical techniques which would result in greater user engagement. Similar benefits have been observed in predicting the match scores, player performance and other predictable aspects. All this combined, helps in developing an excellent model of analysis and prediction using big data of football. It makes use of complex tools, technologies and a lot of detailed study in order to acquire the desired outcome of analysis and prediction. The analysis and prediction model of football matches is seemingly the most fascinating and interesting topic to be involved in for a whole project. This is so, because football is the most watched game in the world. Football has generated a lot of data over the timeline and also has many separate aspects to work on. The whole topic overall builds interest to the project developer group and has a huge scope for further development. The algorithms of machine learning that help to fetch output vary differently. Different algorithms give varied output accuracy. Depending on output we expect and better algorithms the finest execution method is being selected by us.

***Key Words*: Analysis and Prediction, Algorithms, Football, Machine learning, Match scores, Player performance.**

1. INTRODUCTION

Data analytics have come to play an important role in the football industry today. Clubs look to gain a competitive edge on and off the pitch, and big data is allowing them to extract insights to improve player performance, prevent injuries and increase their commercial efficiency. As we know football being such a diverse game it generates a huge amount of data. This data can be of various types for example: player data, team data, league data, fan engagement data, etc. This data includes historic data as well as current data. With the use of all these various types of data we can perform various analytics and predictions regarding almost every aspect of the game which includes prediction of match score, prediction of player ratings, analytics for goals scored with respect to various leagues and create comparisons amongst players, teams and leagues as well.

Football, which has always been a numbers game, is apparently driven by more and more Big Data. Clubs are now likely hiring fewer scouts and more computer analysts; TV, radio and newspapers drive more stats-based conversation about the performance of players, managers and teams than ever before. Numbers are also seeping out of real football and into the fantasy – the stats that surround players are not only used to measure their actual performance, but also to evaluate their contribution to fantasy football teams. It's fair to say that this Big Data revolution in football will only continue and change the whole experience of watching the most popular sport in the world.

2. SYSTEM DESCRIPTION

This system derives insights in the football world with the help of data analytics using machine learning techniques. This system will also be able to predict the outcome of football matches based on previous results of respective teams. It will also allow us to create a completely different user experience to enhance football with the help of analytical techniques which would result in greater user engagement. Predictive analytics makes use of many techniques from data mining, statistics, modeling, machine learning, and artificial intelligence to analyze current data to make predictions.

The goal of this system is to build analysis models in football games to study and predict the possible outcomes of matches. To make visualizations to get insights about almost every aspect in the game. The following steps are followed to obtain the outcome:

A. Gathering Data:

This first step here starts with gathering data required for prediction and visualization which is the desired task of the project. Obtaining data sets which include all the historic match data such as final score of the match, bookings in the match, fouls committed and man of the match. Data set with complete player attribute data is required.

B. Filtering and Parsing Data:

The data is filtered by removing the unwanted columns in the data set and also the factors which may lead to wrong prediction of results. Parsing is a method of breaking down the available data into parts so that it can be processed for predictive analysis.

C. Prediction and Visualization:

Prediction model for football matches based on the user's selection of playing 11 on each side of the team. Visualizations based on the available data which include player, league, team comparison.

3. METHODOLOGY

A. Obtaining the data set

There are various sources from which a data set can be obtained and it can be of different types. Two main datasets obtained from Kaggle are used in this system namely "FIFA 19 complete player dataset" which consist of detailed attributes for every player registered in the latest edition of FIFA 19 database. This dataset mainly contributes to the analytical part of the systems. Which is used to display multiple analyses like comparisons between different teams, leagues and players. And also, prediction of football matches.

Secondly the "International football results from 1872 to 2020" dataset which is also from Kaggle and consist of up-to-date dataset of over 40,000 international results and more than 9,000 events from these football games. This data set helped us to study and implement machine learning algorithms for match prediction.

B. Data set preprocessing

The data set obtained from multiple sources is considered as raw data with respect to the system. In order to make the data suitable for execution with respect to the system it is important to clean or preprocess the data. Issues related incomplete data, inconsistent data, lacking attribute values, lacking certain attributes of interest, or containing only aggregate data were solved. Noisy (containing errors or outliers) and Inconsistent (containing discrepancies in codes or names) data was eliminated prior processing.

C. Analysis & Visualization Phase

This system provides visualizations based on analysis of various components of the game. These components include player analysis, team analysis, league analysis, match analysis. Comparison of these components can be visualized in order to obtain greater insight of the game. Majority of this analysis and visualization is based on feature selection. In order to effectively model our data for visualization we performed feature selection with consideration to Reduced Overfitting (Less redundant data means less opportunity to make decisions based on noise), Improved Accuracy (Less misleading data means modeling accuracy improves), Reduces Training Time (Fewer data points reduce algorithm complexity and algorithms train faster). Extracting key attributes of a player and creating combined visualizations of the attributes for multiple comparison is implemented in this system. Analysis is obtained in various user desired formats like bar graph comparison, pie chart, heat map comparison, list comparison, line graph.

a. PLAYER ANALYSIS

We have defined a player scorecard which contains a collection of player attributes. It is an overall scorecard for key statistics of a player which include Shooting, Passing, Defending, Rating, General, Power, Mentality, Mobility. These stats are combined with a player's bio for international recognition.

Below is the code for defining a scorecard followed by an actual player scorecard.

CODE:

```
def ScoreCard(id = 0):
    if 0 <= id < len(data.ID):
        details(row = players.index[id],
                title = players['Name'][id],
                age = players['Age'][id],
                photo = players['Photo'][id],
                nationality = players['Nationality'][id],
                image = players['Flag'][id],
                logo = players['Club_Logo'][id],
                club = players['Club'][id])
    else:
        print('Index out of Range!')
```

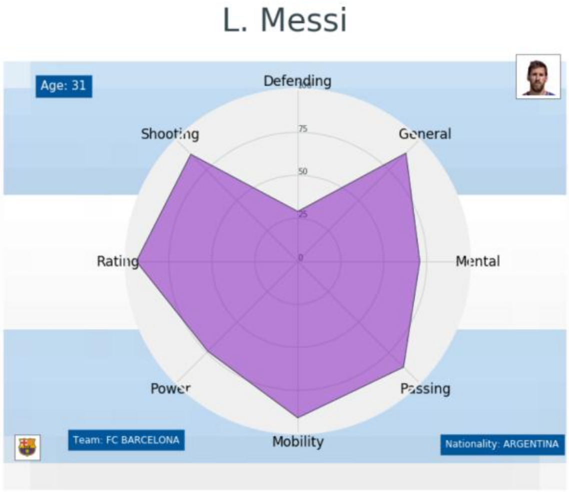

Fig I: Single Player Analysis

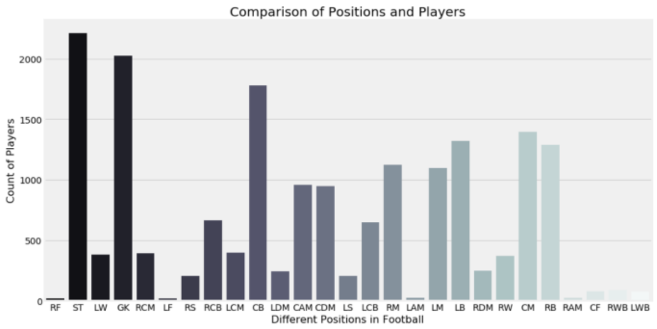

Fig II: Multiple Players Analysis

b. MATCH ANALYSIS

The football match analysis process is used to examine the actions performed during a match, and may concern one's team or the opponent, or even a single athlete. The data is collected and selected by means of the analysis of the events occurring in the different phases of the match. Multiple match analysis can be performed. Which include Goals scored during the match, whether the goals were home goals/ away goals, body parts used to score a particular goal, number of bookings received during a match i.e number of cards (Red, Yellow).

Here is an example of the same; code for the number of red cards against the time during the match. Following analysis is performed on a data set named "Football Events" which consist of data from more than 900,000 events from 9,074 football games across Europe.

CODE:

```
sec_yellow = events[events["event_type"] == 5]

red = events[events["event_type"] == 6]

reds = [sec_yellow,red]

red_cards = pd.concat(reds)

fig = plt.figure(figsize=(8,6))

plt.hist(red_cards.time, width = 1,bins = 100, color =
"red")
plt.xlabel("Minutes")
plt.ylabel("Number of red cards")
plt.title("Number of red cards against Time during match",
fontname = "Times New Roman Bold", fontsize = 14,
fontweight = "bold")
```

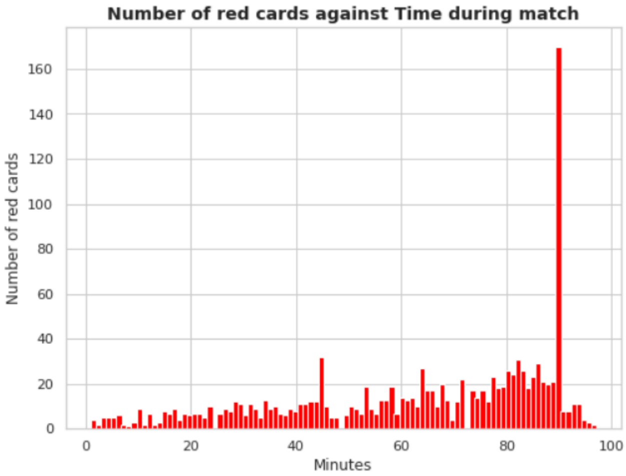

FIG III: Number of red cards against time

D. Prediction Model

European Football or soccer is the world's most popular team sport. It is played by over 150 million men and women of all ages in more than 200 countries. It is also one of the favorite sports for betting. The current estimations, which include both the illegal markets and the legal markets, suggest the sports match-betting industry is worth anywhere between 700bn and 1tn dollars a year. The aim of this system is to build a system to predict the outcome using machine learning algorithms like Support Vector Machine, Naïve Bayes and Logistic Regression.

The prediction model in this system consists of an interface which requires input from the user for prediction. The user input comprises the starting line-up of both the teams for which prediction is to be carried out. The prediction model of this system is an innovative approach of processing input data to predict results. Results are predicated using input features based on the attributes of the players of the match and with the information about past match results which is the historical data. Prediction for the match scores is done by taking the average of results obtained by applying multiple machine learning algorithms. To be precise results obtained from four algorithms (*K nearest neighbor, Random Forest, Logistic Regression, Support Vector Machines*) are averaged to obtain the final result. Results are basically composed of goals scored by each team which is the fundamental requirement which states the result of a football match.

Initially we are performing Exploratory Data Analysis on the dataset we got from the Kaggle website. This dataset is in SQLite format and has tables of Country, League, Match, Player, Player Attributes, Team, Team Attributes and sequences. It has information of more than 25000 matches, 10000 players, 11 European Countries with their lead championship from 2008 to 2016, Players and Teams attributes sourced from EA Sports' FIFA video game series, betting odds from up to 10 providers. We are only going to use the information from the country, match, league and team for the EDA. This is the information which would be useful to understand the game.

To predict the probability of winning the league, the first step is to calculate the entropy to determine predictability. Entropy is calculated to measure the disorder using bet365 odds. The higher the entropy value the more unpredictable are the results of the matches. Further plotting entropy of the leagues for visual representation.

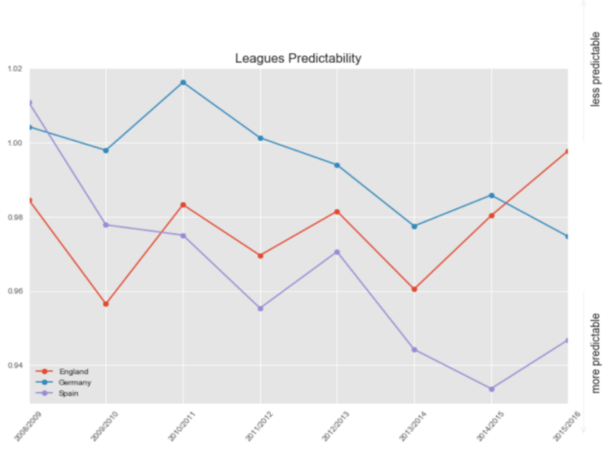

FIG IV: League Predictability

Since 2008 Bayern Munich and Barcelona have won Bundesliga and La Liga 6 times each. The top three teams in La Liga are the same since 2012. The competition for the title race is always between Barcelona, Real Madrid and Atlético Madrid, where Real Madrid won twice and Athletico won once since 2008.From the 2012-13 season Bayern Munich has won Bundesliga 5 times in a row. Borussia Dortmund also performs well and have won the league twice in 09-10 and 10-11 seasons. Since 2008-09 in Premier League Manchester United have won the league thrice (08-09,10-11,12-13), Manchester City has won twice (11-12,13-14), Chelsea has won thrice (09-10,14-15,16-17) and Leicester City once (15-16). Leicester City was promoted to premier league in the previous season and it was placed 14th at the end of the season. Still it won the league the very next season, so we can see that it is very difficult to predict the Premier League.

In an attempt to more precisely predict outcomes of football matches, we selected required features like full time goals, halftime goals, corners, free kicks, fouls, yellow cards, red cards, total shots, shots on target etc from a dataset and put it in one dataframe. In this system ee calculated attacking and defensive strengths of each team at home and away from goals scored and conceded at home and away. Also using the shots, corners, set pieces conversion rate. The algorithms implemented to predict match results in this system are;

a. Support Vector Machine

Support vector machines (SVMs) are a set of supervised learning methods used for classification, regression and outlier's detection.

The advantages of support vector machines are; Effective in high dimensional spaces. Still effective in cases where the number of dimensions is greater than the number of samples. Uses a subset of training points in the decision function (called support vectors), so it is also memory efficient. Versatile: different Kernel functions can be specified for the decision function. Common kernels are provided, but it is also possible to specify custom kernels. The disadvantages of support vector machines include; If the number of features is much greater than the number of samples, avoid over-fitting in choosing Kernel functions and regularization term is crucial. SVMs do not directly provide probability estimates, these are calculated using an expensive five-fold cross-validation.

b. Logistic Regression

The dependent variable should be dichotomous in nature (e.g., presence vs. absence). There should be no outliers in the data, which can be assessed by converting the continuous predictors to standardized scores, and removing values below -3.29 or greater than 3.29. There should be no high correlations (multicollinearity) among the predictors. This can be assessed by a correlation matrix among the predictors. Tabachnick and Fidell (2013) suggest that as long correlation coefficients among independent variables are less than 0.90 the assumption is met. At the center of the logistic regression analysis is the task estimating the log odds of an event. Mathematically, logistic regression estimates a multiple linear regression function defined as:

logit(p)

for i = 1…n .

c. Multinomial Naive Bayes

The multinomial Naive Bayes classifier is suitable for classification with discrete features (e.g., word counts for text classification). The multinomial distribution normally requires integer feature counts. However, in practice, fractional counts such as tf-idf may also work. It is easy and fast to predict the class of test data sets. When assumption of independence holds, a Naive Bayes classifier performs better compared to other models like logistic regression and you need less training data. It performs well in case of categorical input variables compared to numerical variable(s). Naive Bayes is so called because the independence assumptions we have just made are indeed very naive for a model of natural language. The conditional independence assumption states that features are independent of each other given the class.

However, just using the attributes of 22 players which are the starting 11 of each team indicates losing potentially vital information in the form of the team formation. There exist various approaches in formations of teams when strong teams play comparatively weak teams our system also considers such types of instances. We have therefore chosen to model each starting 11 with a vector of size 18. In each 18-dimensional vector, the first component is for the rating of the goalkeeper, the next six components are for the ratings of defenders — if there are only four defenders, two of these components are left as zero.

The diagram below can help visualize what this 18-dimensional vector might look like given a team lineup.

89	81	82	82	82	86	0	85	88	0	0	0	0	0	91	84	84	0

FIG V: 18-dimensional vector

Using similar methodology, seven components exist for the midfielders and four for the forwards. This input structure allows some inference to be drawn from the formation. Basically, the formation of the team changes which components of the vector are left as 0 and allows the neural network to draw inferences from this. Furthermore, when we pass these to the neural network, we will pass them as one large 36-dimensional vector — the home team occupying the first 18 dimensions and the away team occupying the final 18 dimensions. By using this structure, the network also accounts for home advantage. Finally, the output of the model will be the 1X2 odds of the match.

4. TOOLS

A. Matplotlib

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK+. There is also a procedural "pylab" interface based on a state machine, designed to closely resemble that of MATLAB, though its use is discouraged. SciPy makes use of Matplotlib. Pyplot is a Matplotlib module which provides a MATLAB-like interface. Matplotlib is designed to be as usable as MATLAB, with the ability to use Python, and the advantage of being free and open-source.

2. Scikit-learn

Scikit-learn is largely written in Python, and uses NumPy extensively for high-performance linear algebra and array operations. Furthermore, some core algorithms are written in Cython to improve performance. Support vector machines are implemented by a Cython wrapper around LIBSVM; logistic regression and linear support vector machines by a similar wrapper around LIBLINEAR. In such cases, extending these methods with Python may not be possible. Scikit-learn integrates well with many other Python libraries, such as matplotlib and plotly for plotting, NumPy for array vectorization, pandas dataframes, SciPy, and many more.

3. Jupyter Notebook

Jupyter Notebook is a web-based interactive computational environment for creating Jupyter notebook documents. The "notebook" term can colloquially make reference to many different entities, mainly the Jupyter web application, Jupyter Python web server, or Jupyter document format depending on context. A Jupyter Notebook document is a JSON document, following a versioned schema, and containing an ordered list of input/output cells which can contain code, text, mathematics, plots and rich media, usually ending with the ".ipynb" extension. A Jupyter Notebook can be converted to a number of open standard output formats (HTML, presentation slides, LaTeX, PDF, ReStructuredText, Markdown, Python) through "Download As" in the web interface, via the nbconvert library or "jupyter nbconvert" command line interface in a shell.

5. CONCLUSIONS

Machine learning methods can be applied to different fields, including sports. In the example of Football Leagues, it is shown that it is possible to find a classifier that predicts the outcome of Football matches with the precision of more than 70%. This system is a part of machine learning and big data analysis which involves analyzing player and team attributes along with past match results to predict results of football matches and also obtain data visualizations for comparisons of players teams and leagues. This system does not make any claims to be used for betting purposes, this project is implemented for fan engagement and for better understanding of the game.

REFERENCES

[1] K. Sujatha, T. Godhavari, and N. P. Bhavani, "Football match statistics prediction using artificial neural networks," International Journal of Mathematical and Computational Methods, vol. 3, 2018.

[2] J. Shin and R. Gasparyan, "A novel way to match prediction," Stanford University: Department of Computer Science, 2014.

[3] D. Prasetio et al., "Predicting football match results with logistic regression," in Advanced Informatics: Concepts, Theory and Application

(ICAICTA), 2016 International Conference On. IEEE, 2016, pp. 1–5.

[4] https://sofifa.com/players

[5] https://www.kaggle.com/karangadiya/fifa19

[6]https://www.kaggle.com/secareanualin/football-events

[7] http://football-data.co.uk/data.php

[8] Pena J and Touchette H 2012 Sports Physics: Proc. 2012 Euromech Physics of Sports Conference 517–528

[9] Heckerman D, Geiger D and Chickering D 1995 Machine Learning 20 197–243

[10] Heaton J 2013 Forecasting & Futurism 7 6–10

[11] Pearl J 1985 Bayesian Networks: A Model of Self-Activated Memory for Evidential Reasoning (No: CSD-850021)

[12] Kersting K and Raedt L 2000 Bayesian Logic Programs (Report No. 151)

[13] Rue H and Salvesen O 2000 Journal of the Royal Statistical Society: Series D (The Statistician) 49 399–418

[14] Maher M 1982 Statistica Neerlandica 36 109–118

[15] Dixon M J and Coles S G 1997 Journal of the Royal Statistical Society. Series C: Applied Statistics 46 265–280

[16] Min B, Kim J, Choe C, Eom H and (Bob) McKay R I 2008 Knowledge-Based Systems 21551–562

[17] Hvattum L M and Arntzen H 2010 International Journal of Forecasting 26 460–470

[18] Leitner C, Zeileis A and Hornik K 2010 International Journal of Forecasting 26 471–481

[19] Rotshtein A P, Posner M and Rakityanskaya A B 2005 Cybernetics and Systems Analysis 41 619–630