**Football Match Result Prediction**

**Group A**

**Group Members:**

|  |  |  |
| --- | --- | --- |
| **First name** | **Last Name** | **Student number** |
| Abhishek | Naithani | c0871411 |
| Karishma | Shirsath | c0871245 |
| Tejaswi | Kalla | c0852124 |
| Venkata Sai Charan | Yerapasetty | c0863224 |

**Submission date: *[13-08-2023]***

Contents

[**Abstract** 3](#_Toc142851904)

[**Introduction** 3](#_Toc142851905)

[**Methods** 4](#_Toc142851906)

[**Results** 11](#_Toc142851907)

[**Conclusions and Future Work** 21](#_Toc142851908)

[**References** 22](#_Toc142851909)

# **Abstract**

Data Science has become a game-changer in the constantly changing world of sports, providing fresh perspectives and predictions. This report demonstrates data processing and modeling to dive into the world of football match prediction. This project aims to create precise predictive models that predict the results of football matches by closely examining a wide range of match statistics, player qualities, history performance, and contextual elements. Although they may appear to have nothing in common, data science and sports are closely intertwined. The fusion of data science and sports has fundamentally changed how we perceive, evaluate, and connect to sporting and athletic activities. Each match data from the 2022-2014 seasons of the Spanish league, La Liga’ is retrieved, formatted, and validated for Exploratory data analysis and building supervised prediction models to predict the match outcome of a given football match.

# **Introduction**

Football is a fascinating subject for study because it is thought to be significantly more diversified and complex. The widespread application of machine learning techniques and rapid access to the Internet have fuelled sports betting and research. The Primera Division, commonly known as La Liga in Europe, has many renowned players and teams. Some famous clubs (Real Madrid, FC Barcelona, etc.) are a part of this league. Football, soccer, cricket, tennis, and hockey are among the prominent sports spectators watch worldwide. Due to the significant financial stakes, bigger teams continuously look for weaknesses in their opponent's defence.

Sports analytics benefits numerous sports organizations, and the market for these analytics is growing by billions. With organizations and management becoming more receptive to leveraging data to gain an advantage over competitors, the economy has also changed over time. Sports clubs and committees could invest in significant analytical data which can be used to assess opponents' performance, as opposed to restricting their in-depth study to simply noticing how particular teams play. This statistical and predictive analysis of such data sheds light on success rates, player performance, team analysis, fan voting, athlete safety, sports gambling, and many more potential contributors that would change the game and engage millions of football fans.

This report demonstrates our work on the methodology used for data gathering, preprocessing, wrangling, and visualization. It also discusses building a predictive model for the outcome of a football match by implementing a Random Forest Classifier, Extreme Gradient Boosting Classifier, and Logistic Regression model with 3-way and 2-way (binary) predictions.

# **Methods**

The below are the methodologies we used for our prediction.

**Data Collection:**

The list of all the matches from 2022-2014 of the La-Liga leagues was scrapped through the ‘Beautiful Soup’ library from the **‘fbref’** website and formatted into a Pandas data frame, having the dimensions of 6080 records and 26 attributes. This data is converted and saved into a CSV file for further analysis.

**Data Pre-processing:**

The collected data is checked for different data types, description statistics such as Mean, different percentiles, maximum, and minimum values of each column, and duplicated values.

1. **Data Validation:**

Data Validation or Data Cleaning is an essential step for quality assurance and meaningful results of the data analysis. It is crucial to guarantee the accuracy and integrity of the dataset before developing any Machine Learning models. Missing values are frequently present in datasets, which can have a negative effect on the precision and dependability of prediction models. This stage tries to examine the level of missing values in each column to determine the completeness of the data and pinpoint any potential problem areas.

We have set a threshold of 20 percent to filter out the columns having any missing or null values above this percentage relative to the overall dataset. The columns ‘xg’, ‘xga’, ‘notes’, and ‘notes’ represent expected goals to be scored by the targeted team, expected goals against the opposite team, note points, and average distance of each shot, respectively. These columns have been dropped since they do not carry relevant values useful for the prediction.

A yellow and white rectangles

Description automatically generated

Fig 1: Columns with missing values greater than 20%

Along with the above-mentioned columns, some other columns having the least relevant data (attendance, captain, match report, referee) were also dropped and the cleaned dataset with (6080, 18) dimensions is made ready for visualization.

A pie chart with a yellow triangle

Description automatically generated

Fig 2: Percentage of retained and dropped columns.

1. **Data Visualization**

A graph of a number

Description automatically generated with medium confidence

Fig 3: Total matches played by each team across the seasons.

* The number of matches per team is affected by the rule of ‘Relegation’ in the league. According to this rule, the teams in the last three positions of one division will be replaced by the top 3 teams of the other division, and so on. It varies from one season to the other and the bar graph depicts the scenario of the 2022-2014 seasons.

The below plots show the goals scored by each team and the goals scored against each team.

A graph of blue and white bars

Description automatically generated with medium confidence

Fig 4: Goals scored by each team.

A graph of red columns

Description automatically generated with medium confidence

Fig 5: Goals scored against each team.

.A colorful circle with numbers

Description automatically generated

Fig 6: Teams with most wins

In the above donut chart, we can see the team with most wins, we used the result column in the dataset to calculate the total wins of the each team.

1. **Data Transformation:**

* The ‘date’ column in the dataset is transformed into a date-time format using the ‘to\_datetime()’ function of the Pandas library. Analysing and comparing data over a specific time period such as Seasonal events, Time-series forecasting, Chronological sorting, etc, can be facilitated by this conversion.
* The categorical features, ‘Venue’ and ‘Opponent’ are transformed, and new features called ‘venue\_code’ and ‘opp\_code’ (code for each opponent team) are created to represent the corresponding numeric values.
* The match outcomes in the target feature, ‘result’ are encoded, and ‘win’, ‘loss’, and ‘draw’ are mapped to 1, 0, and 2, respectively. A new column, ‘target’, is created to store these numeric values.

**Applying ML Models:**

**Train and Test set:**

The dataset we have is a timeseries so instead of conventional train and test split of the data we performed the split based on the date column. The data before 2022-01-01 is taken as a train set and data after 2022-01-01 is taken as a test set. This allowed us to use data from a certain time to build and train a model, and the test the model performance on the different time periods (test set).

**Predictors:**

For predicting the match result, we have used few features as the input to the model, these are the features we used as predictors.

* venue\_code : Generally venue plays a important role during the football season
* opp\_code : we have used this feature as one of the input ,usually the team performance vary depending on whether the team is playing at home or away against a specific opponent,
* sot: This column has information about the shots on target (goal post) in a match which is a very important feature in predicting the result.

We wanted to try other feature “poss” as well which is a possession of the ball but not every time the team with high possession wins the game so we decided not to go with this feature as predictor.

**Three-way classification:**

The objective of the classification problem known as "three-way classification," also referred to as "multi-class classification," is to assign the fed data into one of three potential classes or categories. Each input is given a single class, out of the three possible classes, in multi-class categorization. Machine learning frequently uses this kind of classification. In our case we used three-way to classify the results as win(1), loss(0), draw(2).

**Two-way Classification:**

Data must be categorized into one of two possible classes or categories in a job known as two-way classification, often known as binary classification. The objective of binary classification is to give each data point a label indicating which of the two classes it belongs to. The classes can represent any two different outcomes of interest, though they are typically characterized as positive (class 1) and negative (class 0). We have classified our prediction as win(1) and loss(0).

After all the processing of the data we have applied below models to predict the results in three-way and two-way.

**Random Forest (RF) Classifier:**

RF classifier is a versatile and potent machine learning model that is frequently used for classification problems. Multiple decision trees are combined in this ensemble learning technique to produce predictions. The model is resistant to overfitting and noise since each decision tree is trained on a separate sample of the data and considers a subset of features. The RF classifier gathers the outcomes of various trees during prediction to produce a final prediction. This ensemble strategy improves the model's generalization and accuracy capabilities.

**Extreme Gradient Boosting (XgBoost) Classifier:**

XgBoost classifier is a more sophisticated algorithm that works on gradient-boosting decision trees. It is also widely preferred for its computation speed, scalability, and model performance on large datasets. XGBoost offers remarkable prediction performance by progressively integrating decision trees and utilizing ensemble learning. Its distinguishing characteristics

include automatic handling of missing values, insightful feature importance analysis, and normalization strategies to prevent overfitting. This classifier supports multiclass and binary classification.

**Logistic Regression:**

Classification challenges are often handled through the fundamental statistical and machine-learning technique known as logistic regression. It is mainly used to forecast the likelihood that an instance will belong to a specific class. The logistic function, which converts any input value into a range between 0 and 1, is used to model this likelihood. To determine the influence of the input features on the result, logistic regression estimates the coefficients for each feature. This linear model is beneficial when there is a nonlinear relationship between the features and the goal variable. Predictions are based on whether the estimated likelihood exceeds a decision threshold that has been established.

# **Results**

True Positive (TP) : It is the number of correct predictions for the positive class.

False Positive (FP): represents the number of incorrect predictions for the positive class.

True Negative (TN): represents the number of correct predictions for the negative class.

False Negative (FN): represents the number of incorrect predictions for the negative class.

We have used the below metrics to evaluate how model performed in predicting the results.

* Precision: Precision = TP / (TP + FP)
* Accuracy: Accuracy = (TP + TN) / (TP + TN + FP + FN)
* Recall: Recall = TP / (TP + FN)
* F1-Score: F1-Score = 2 \* (Precision \* Recall) / (Precision + Recall)

**Three-way Prediction:**

**Test set:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model | Accuracy | Precision | Recall | F1-score |
| Random forest | 0.5129 | 0.46 | 0.47 | 0.46 |
| xgboost | 0.5441 | 0.48 | 0.49 | 0.46 |
| Logistic regression | 0.5311 | 0.36 | 0.47 | 0.40 |

**Train Set:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model | Accuracy | Precision | Recall | F1-score |
| Random Forest | 0.5885 | 0.57 | 0.56 | 0.55 |
| xgboost | 0.5664 | 0.54 | 0.53 | 0.50 |
| Logistic Regression | 0.5073 | 0.34 | 0.46 | 0.39 |

For Precision, Recall and F1-score we have taken macro average because it treats all classes equally irrespective of the frequency. We wanted to give equal importance to all the classes when predicting our results. On the other hand the weighted average was also almost equal to the macro average.

**Confusion Matrix:**

**Random Forest:**

**Test data:**

A blue squares with numbers and a number on them

Description automatically generated with medium confidence

Fig 7: Confusion Matrix of Random Forest for test data

**Train data:**

**A blue squares with numbers

Description automatically generated**

Fig 8: Confusion Matrix of Random Forest for train data

**Xgboost:**

**Test data:**

A blue squares with numbers

Description automatically generated

Fig 9 : Confusion Matrix of xgboost for test data

**Train data:**

A blue squares with numbers

Description automatically generated

Fig 10: confusion matrix of xgboost for train data

**Logistic Regression:**

**Test data:**

A graph of confusion matrix

Description automatically generated

Fig 11: confusion matrix of logical regression for test data

**Train data:**

A graph with blue squares

Description automatically generated

Fig 12: confusion matrix of logical regression for train data

In our analysis, we evaluate the performance of different models for football match result prediction. Starting with logistic regression, this model exhibited subpar performance. Notably, the precision, recall, and F1-scores indicate challenges in predicting draw outcomes, as evident from the confusion matrices (refer to Figure 11 and Figure 12) for both training and test data. This struggle with predicting draws affected the model's overall metrics.

Comparing the results of the Random Forest and XGBoost models, they both demonstrated similar predictive capabilities. However, a closer examination of the difference in scores between the test and train datasets reveals a key distinction. In the case of XGBoost, this difference is notably smaller, signifying better generalization performance. A smaller difference suggests reduced risk of overfitting, where the model becomes too closely tailored to the training data.

This observed trend highlights the technical advantage of XGBoost in handling unseen data, making it a superior choice when compared to the Random Forest model. While both models performed well, XGBoost's ability to generalize effectively is a critical factor that lends it a competitive edge in terms of robustness and reliability.

**Two-way Prediction:**

**Test Data:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model | Accuracy | Precision | Recall | F1-score |
| Random Forest | 0.6955 | 0.68 | 0.65 | 0.66 |
| xgboost | 0.7093 | 0.69 | 0.67 | 0.68 |
| Logistic Regression | 0.6894 | 0.67 | 0.64 | 0.64 |

**Train Data:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model | Accuracy | Precision | Recall | F1-score |
| Random Forest | 0.7475 | 0.73 | 0.71 | 0.71 |
| xgboost | 0.7254 | 0.71 | 0.69 | 0.69 |
| Logistic Regression | 0.6917 | 0.67 | 0.64 | 0.64 |

**Confusion Matrix:**

**Random Forest:**

**Test Data:**

A blue and white box with numbers

Description automatically generated

Fig 13: Two-way classification of random forest for test data

**Train data:**

A blue squares with white text

Description automatically generated

Fig 14: Two-way classification of random forest for train data

**Xgboost:**

**Test Data:**

A blue squares with white text

Description automatically generated

Fig 15: Two-way classification of xgboost for test data

**Train Data:**

A blue squares with white text

Description automatically generated

Fig 16: Two-way classification of xgboost for train data

**Logistic Regression:**

**Test Data:**

A blue and white graph

Description automatically generated

Fig 17: Two-way classification of logistic regression for test data

**Train Data:**

A blue squares with white text

Description automatically generated

Fig 18: Two-way classification of logistic regression for train data

The results show notable enhancements in the prediction outcomes when transitioning from three-way to two-way predictions. In the two-way approach, losses and draws were treated as losses, significantly contributing to the increased accuracy of the predictive models. Comparing the various models, there is a general trend of improved performance, with their evaluation metrics closely aligned.

Particularly, Logistic Regression demonstrated substantial progress due to its inherent suitability for binary classification. The model showcased uniform precision, recall, and F1-score values, signifying strong generalization capabilities. Conversely, both Random Forest and XGBoost yielded nearly identical scores. However, upon closer examination, XGBoost demonstrated a more controlled potential for overfitting, rendering it a superior choice between the two.

These improvements can be attributed to the finer distinctions made in the two-way approach, leading to more accurate predictions and a better alignment between model performance and real-world outcomes.

# **Conclusions and Future Work**

In conclusion when we transitioned from three-way to two-way predictions resulted in significant advancements in our models. By combining losses and draws into a single category, the accuracy of predictions improved significantly. When comparing model performances, a consistent trend of enhanced results emerged, showcasing their adaptability to this refined approach. Logistic Regression, inherently suitable for binary classification, demonstrated remarkable progress with consistent precision, recall, and F1-score values. Although Random Forest and XGBoost shared similar scores, XGBoost displayed a more controlled susceptibility to overfitting. This strategic shift from three-way to two-way predictions not only improved model accuracy but also underscored the importance of nuanced approaches in enhancing predictive outcomes.

As a future work we aim to incorporate rolling averages and expand the predictor pool to enhance predictive results. Our vision extends to establishing a dedicated web interface, showcasing match outcomes in terms of win, loss, and draw percentages. This future trajectory of the project encompasses the integration of advanced techniques and interactive platforms to provide comprehensive insights into match predictions and outcomes.

# **References**

[1] C. Ebejer, “What is La Liga? A look at Spain’s top league,” Field Insider, [https://fieldinsider.com/what-is-la-liga-spains-top-league/](https://fieldinsider.com/what-is-la-liga-spains-top-league/%20) (accessed Aug. 13, 2023).

[2] Simplilearn, “What is xgboost? an introduction to XGBoost algorithm in Machine Learning: Simplilearn,” Simplilearn.com, [https://www.simplilearn.com/what-is-xgboost-algorithm-in-machine-learning-article#:~:text=cost%20to%20accuracy!-,What%20is%20XGBoost%20Algorithm%3F,optimize%20their%20machine%2Dlearning%20models.](https://www.simplilearn.com/what-is-xgboost-algorithm-in-machine-learning-article%23:~:text=cost%20to%20accuracy!-,What%20is%20XGBoost%20Algorithm%3F,optimize%20their%20machine%2Dlearning%20models.) (accessed Aug. 13, 2023).

[3] A. Shafi, “Random Forest classification with Scikit-Learn,” DataCamp, <https://www.datacamp.com/tutorial/random-forests-classifier-python> (accessed Aug. 13, 2023).

[4] “Football statistics and history,” FBref.com, [https://fbref.com/en/](https://fbref.com/en/%20) (accessed Aug. 13, 2023).