UNIVERSIDAD SAN FRANCISCO DE QUITO
USFQ

Colegio de Ciencias e Ingenierías

Predicting Premier League Match Outcomes: Machine Learning Application for Football Match Analysis

Joaquín Orbe Hidalgo

Ingeniería Industrial

Trabajo de fin de carrera presentado como requisito
para la obtención del título de
INGENIERO INDUSTRIAL

Quito, 8 de mayo de 2025

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingenierías

HOJA DE CALIFICACIÓN
DE TRABAJO DE FIN DE CARRERA

Proyecto Integrador

Joaquín Orbe Hidalgo

María Gabriela Baldeón Calisto PhD

Quito, 8 de mayo de 2025

© DERECHOS DE AUTOR

Nombres y apellidos: Joaquín Orbe Hidalgo

Código: 00320820

Cédula de identidad: 1721480182

Lugar y fecha: Quito, 8 de mayo de 2025

ACLARACIÓN PARA PUBLICACIÓN

Nota: El presente trabajo, en su totalidad o cualquiera de sus partes, no debe ser considerado como una publicación, incluso a pesar de estar disponible sin restricciones a través de un repositorio institucional. Esta declaración se alinea con las prácticas y recomendaciones presentadas por el Committee on Publication Ethics COPE descritas por Barbour et al. (2017) Discussion document on best practice for issues around theses publishing, disponible en http://bit.ly/COPETheses.

UNPUBLISHED DOCUMENT

Note: The following capstone project is available through Universidad San Francisco de Quito USFQ institutional repository. Nonetheless, this project – in whole or in part – should not be considered a publication. This statement follows the recommendations presented by the Committee on Publication Ethics COPE described by Barbour et al. (2017) Discussion document on best practice for issues around theses publishing available on http://bit.ly/COPETheses.

RESUMEN

Predecir los resultados de partidos de fútbol es un desafío debido a los factores impredecibles del deporte que influyen en el resultado. El presente estudio desarrolla un modelo basado en datos para predecir partidos de la Premier League inglesa como victorias, empates o derrotas, combinando algoritmos de aprendizaje automático y aprendizaje profundo utilizando datos históricos de las últimas ocho temporadas. Para adquirir los datos necesarios, se crea un modelo de web scraping para extraer información de un sitio web de datos futbolísticos. Con estos datos, se implementan dos algoritmos de aprendizaje automático: Random Forest y XGBoost; y un algoritmo de aprendizaje profundo: TabNet, los cuales luego son comparados en función de métricas de evaluación específicas. Posteriormente, estos modelos se combinan en un único modelo de ensamble para unir las fortalezas de cada uno y lograr una mayor precisión. Se realiza un análisis de importancia de características para identificar variables clave que influyen en el resultado de un partido. Finalmente, las predicciones del modelo final se comparan con fuentes externas, como una inteligencia artificial y una casa de apuestas.

Palabras clave: Aprendizaje automático, aprendizaje profundo, web scraping, Random Forest, XGBoost, TabNet, fútbol, resultados de partidos, predicción.

ABSTRACT

Predicting football match results is a challenge due to the sport's unpredictable factors that influence the result. The present study develops a data-driven model to predict English Premier League matches as wins, draws, or losses, by combining machine learning and deep learning algorithms using historical data from the last eight seasons. In order to acquire the necessary data, a web scraping model is created to extract information from a football data website. With these data, two machine learning algorithms: Random Forest and XGBoost; and a deep learning algorithm: TabNet, are implemented and then compared based on specific evaluation metrics. These models are later combined into a single ensemble model to unite the strengths of each model and achieve a higher accuracy. A feature importance analysis is conducted to identify key variables that influence a match's result. Finally, the final model's predictions are compared with external sources such as artificial intelligence and a betting house.

Keywords: Machine learning, deep learning, web scraping, Random Forest, XGBoost, TabNet, football, match results, prediction.

Table of Contents

INTRODUCTION

Football is the world's most popular sport, known because of its beauty, its complicated tactics and techniques, its players and all the great tournaments. Among all the professional leagues, one of them stands out as one of the best or even the best national league in the world: the English Premier League. The Premier League stands out as the most competitive league due to the amount of top-tier teams, financial backing, and quality players from all over the world. All this amounts to the most interesting fact of the league: the amount of surprising match outcomes. Such unpredictability, along with the fluctuating performance of teams, presents as a great opportunity for analysts and fans attempting to forecast match results.

Despite progress in sports analytics, accurately predicting football match outcomes remains extremely complicated due to the many unpredictable factors that influence a game's result. However, with the increase of available data, along with advances in machine learning algorithms, there is potential to explore and develop a way to improve prediction accuracy and provide valuable insights that have different applications. In sports analytics, teams are able to identify patterns in both the team's and the opponent's performance and provide insights for coaches, analysts, and fans. In the world of sports betting, more accurate forecasts allow for more informed betting strategies, resulting in higher profits. Finally, teams can optimize their performance by using data to make adjustments in lineups, trainings, and other strategic decisions.

Important works have been presented with the objective of predicting football matches using machine learning algorithms. Machine learning is a subset of artificial intelligence (AI), which, according to [1], is used to learn from the data to predict an output variable. The majority of these studies, like [2] and [3], suggest that tree-based models, such as Random Forest and XGBoost, are some of the top performers when predicting football match results. Although these models are considered extremely powerful in this application, it has been demonstrated that ensemble models, where multiple models are combined, tend to provide a better prediction. In terms of data, most approaches incorporate match-level data such as goals in favor, goals against, shots, possession, etc., and contextual attributes such as venue, to know whether the team is playing at home or away, the team's recent form. or league standing. In addition to this, player statistics such as the passing accuracy of the player, the expected goals, etc. are also used in some studies, indicating that datasets with more data can refine predictive capabilities. Overall the studies highlight the importance of decision tree-based and ensemble algorithms because of their high performance.

In contrast to machine learning models, deep learning models have proved to work well in sports analytics too. Deep Learning is a machine learning technique that uses artificial neural networks to learn from the data. TabNet, specifically, is specified in [4] as a deep learning architecture optimized for tabular data. Unlike traditional machine learning models, TabNet employs sequential attention to selectively process and select relevant features, improving both prediction accuracy and interpretability. One of the key advantages of TabNet is its ability to perform instance-wise feature selection, meaning that it adapts dynamically to different inputs. This contrasts with traditional models such as XGBoost, which require external interpretability methods to

explain predictions. Performance comparisons reveal that TabNet outperforms top tree-based models such as XGBoost and LightGBM across multiple datasets. For example, on one of the papers' dataset, TabNet achieved an accuracy of 96.99%, significantly surpassing XGBoost's 89.34% and the other models. Similarly, in another one of the paper's dataset, TabNet reached an accuracy of 99.2%, compared to XGBoost's 71.1%. TabNet also benefits from self-supervised learning, where unsupervised pre-training enhances performance in scenarios with limited labeled data. This characteristic allows for faster convergence and better generalization, making it particularly useful for large-scale datasets.

The present study focuses on developing a machine learning model capable of predicting Premier League match outcomes (win, draw, or loss) with the best accuracy as possible. In order to accomplish this goal, specific objectives have been specified: implement a web scraping model to successfully access and extract match data, evaluate and compare different machine learning models' performance, identify key features that most significantly influence the results, and evaluate each of the models' capacities to forecast matches considering the variations in teams' performance across seasons. To ensure robust predictions, historical data from the Premier League is gathered using web scraping techniques. The data will be extracted from the internet, and will cover matches starting in the 2017/18 season up to the actual date, amounting to 6,020 records currently in the dataset. Some of the relevant predictors that will be taken into account include venue, opponent, date, team's formations, and key in-game statistics such as goals, shots and possession. The study covers the implementation of decision tree-based algorithms and recent deep learning advances to provide a comprehensive and comparative analysis. The models to be implemented are: Random Forest, XGBoost,

and TabNet. And, with these models, an ensemble model will be created with the objective of using each model's advantages and combining them, consequently improving the final accuracy of the predictions. Furthermore, a comparison between the ensemble model and outside predictions will be made. The outside predictions will come from sources including a betting house and Artificial Intelligence to provide a general vision of how well the model is performing. In this way, the research aims to provide a comprehensive analysis of Premier League match outcomes and contribute to the growing field of sports analytics by implementing machine learning algorithms.

LITERATURE REVIEW

In [5], the authors investigate the application of Decision Tree models to predict football match outcomes, incorporating predictors such as home/away performance, recent form, and league ranking. The research employs Decision Trees as the primary classifier, along with Naïve Bayes and Random Decision Trees for comparative analysis. The study finally shows that Decision Trees performs best for single-league predictions, while Random Decision Trees demonstrates superior performance in multileague scenarios. The research suggests integrating player-level data and external statistics improve the accuracy of the predictions. The findings have significant practical applications in football analytics, including prediction of the outcome of matches, strategic planning for teams, and analysis of the betting market.

The paper [2] explores the application of machine learning to predict football match outcomes and develop profitable betting strategies by analyzing a dataset of 47,856 matches from the top five European leagues and their second divisions, including data starting in the 2006 season and ending in the 2018 season. Player data is incorporated alongside match data and betting odds from leading bookmakers. Four machine learning models are evaluated: Random Forest, Boosting, Support Vector Machines, and Linear Regression. Among these, Random Forest demonstrates to have the best performance, with an accuracy of 81.26%, while an ensemble model combining all algorithms has the highest accuracy of 81.77%. The ensemble model, also proves to outperform strategies like random betting or always favoring the home team. The study highlights the effectiveness of decision tree-based models in capturing the complexity

of football data and demonstrates that combining models enhances both robustness and profitability.

A dataset formed by 1,900 matches from the English Premier League across five seasons (2013/2014 to 2018/2019) is presented in [3], incorporating match statistics like goals, corners, and free kicks, and player attributes such as passing accuracy, agility, etc. A feature selection technique is implemented using the Boruta algorithm, narrowing the dataset to 18 key predictors. Several machine learning algorithms are tested and compared: Random Forest, Support Vector Machines, XGBoost, Naive Bayes, and Artificial Neural Networks. Models are trained on four seasons and tested on the fifth. The best-performing model, Random Forest, demonstrates an accuracy of 65.26% and a profit margin of 26.78%. SVM and XGBoost also show competitive results. The analysis shows the importance of implementing both match and player statistics in order to have high accuracy predictions.

In [6] the authors examine how various football performance metrics influence expected goals (xG) by using a dataset which includes Arsenal Football Club's matches from four English Premier League seasons (2019-2023). The study applies 19 predictor variables such as possession, passing accuracy, dribble success rate, and shots on target. Different machine learning techniques are applied to identify the most influential factors, concluding that one of those factors, formation stability, correlates with improved team performance.

METHODOLOGY

The study implements four principal stages in the methodology section: data collection, data processing, model implementation, and creation of an ensemble model. These phases guide the transition from raw online data to a predictive model for football matches. In the following subsections, each stage is presented and analyzed.

Data Collection

In the data collection phase, first, it is essential to obtain a webpage from which to gather the necessary information. The data is extracted from the Premier League's section from a webpage using a web scraping model. According to [7], web scraping is a solution to extract data from the web efficiently, quickly, and in an automated manner. Here, using python, HTML requests are made using python's requests library to retrieve the HTML content, and the library BeautifulSoup, used to parse the retrieved pages in search of specific information in the HTML file.

It is important to consider that the webpage will block the user if it realizes that data from the website is being extracted using web scraping methods, meaning that too much data is extracted in a short amount of time. To prevent this problem, a few techniques are applied: a list with user-agent strings is created to simulate different users and browsing patterns, and also, random pauses are introduced between requests. These methods prevent the webpage from perceiving that the data gathering process is extremely automated, resulting in the webpage blocking the requests.

The initial webpage shows the Premier League table for the actual season. In this table, when a click is made on each of the team's names, the webpage redirects the user

to a different link where the team's data for the actual season is stored. In this initial webpage, the HTML file is read and a reference to each team's link is found. This way, the webpage redirects the user to the desired team's statistics. Here, the desired match tables are located and extracted using the .read_html() method from the pandas library, which looks for the first table that has a specific keyword on its header, which is the title for the initial extracted table. The secondary table, is stored in a separate link accessible via a hyperlink that points to the desired section, so, as done before, the reference to the desired link is found and the webpage redirects the user to the new link, where the .read_html() method looks for a table with the new title on its header, thus requiring a second request to capture the second table.

Each team's data, consisting of the two extracted tables, once gathered, is combined into a single dataset; then, each team's dataset is combined with the other teams' datasets, creating a new dataset formed by the 20 teams' individual datasets for that specific season. Finally, this season's dataset, is then combined with the other seasons' datasets, creating the final dataset consisting of the teams' individual datasets for each one of the desired seasons (2017/18 - 2024/25).

The final dataset, saved as a .csv file, contains 6,020 records to date, which translates to 3,010 Premier League matches viewed from both the home and away teams’ perspectives. The data is shown for a total of 31 teams that have played in the Premier League in at least one of the last eight seasons. This final dataset contains exactly 28 variables, including the team, opponent, and performance statistics such as goals in favor, shots on target, possession, etc. It is essential to take into account that the

dataset needs to be frequently updated, which is why the number of matches is subject to change.

Data Processing

The data processing phase consists of several critical steps to ensure that the data to be applied in the machine learning algorithms is clean and balanced. The first step to achieve this is to remove the noninformative columns; then, the data type for some variables is corrected, like the date, which is transformed to a date type as it is initially identified as text; and, finally, correct the format for some variables, like the team's and opponent's formation, because Excel identifies specific formations, like 4-3-3, as a date (04/03/2003). Once this is done, it is necessary to identify all the missing values, which are handled depending on the variable's data type. Fortunately, there are only two variables with missing values, and as these are both numeric variables, the missing values are replaced with the respective mean values.

Additionally, the variables used as predictors in the machine learning models are identified: team, opponent, venue, day of the week, matchweek, formation, and opponent's formation. These variables, identified as nominal categorical variables, are converted into dummy variables through one-hot encoding to facilitate the algorithm's understanding of the dataset, creating a purely numerical feature set. Next, performance metrics are also added as predictors: goals in favor, goals against, shots, shots on target, average shot distance, free kicks, penalty kicks, expected goals in favor, expected goals against, and possession. These metrics are incorporated by calculating their rolling averages over each team's last six matches, capturing a notion of recent team's performance in the league. When a new season starts, the algorithm considers the match statistics from the last six matches from the previous season. Finally, the match outcome

is encoded into three categorical classes: 0 for losses, 1 for draws, and 2 for wins. This encoding facilitates the implementation and interpretation of the machine learning algorithms.

Table 1 shows the predictor and output variables along with a description and possible values.

TABLE 1: Variable description and values.

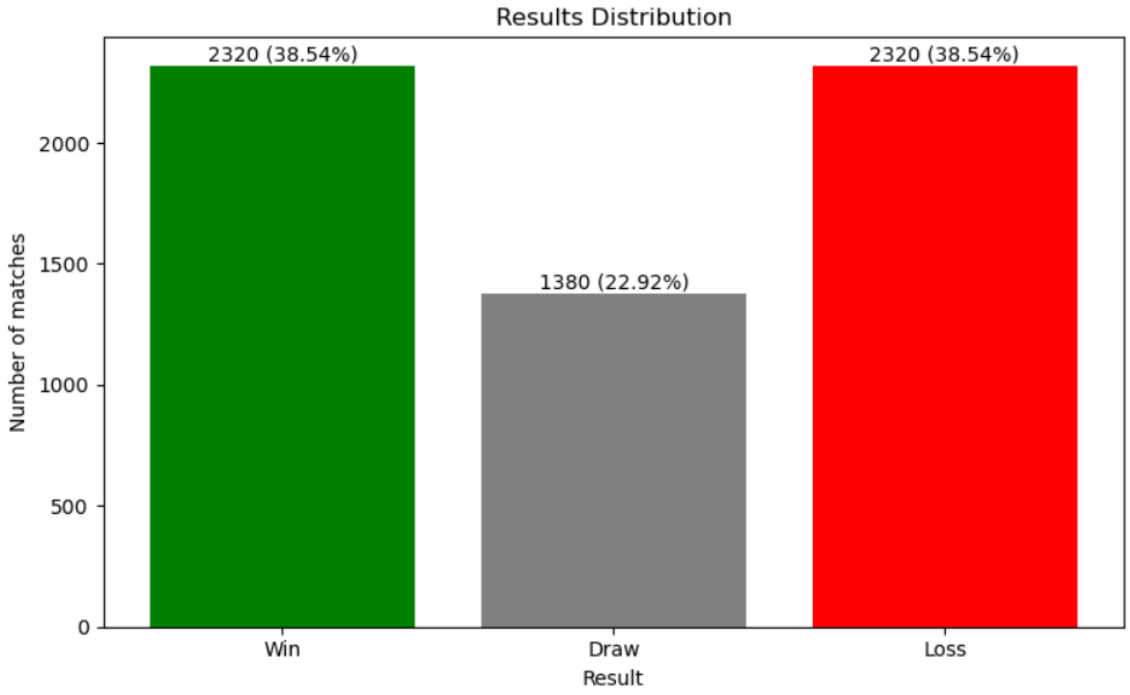
Variable	Description	Values
Team	Categorical nominal variable that represents the team that plays the game	Liverpool, Arsenal, Manchester City, etc. (31 teams)
Opponent	Opponent team that plays the game	Liverpool, Arsenal, Manchester City, etc. (31 teams)
Venue	Whether the team plays at home or away stadium	Home, Away
Day	Day of the week	Mon, Tue, Wed, Thu, Fri, Sat, Sun
Round	Premier League Matchweek	Matchweek 1 through 38
Formation	Team's formation	4-3-3, 4-2-3-1, etc.
Opp Formation	Opponent team's formation	4-3-3, 4-2-3-1, etc.

GF	Goals For	Discrete numerical variable
GA	Goals Against	Discrete numerical variable
Sh	Shots excluding penalty kicks	Discrete numerical variable
SoT	Shots on target excluding penalty kicks	Discrete numerical variable
Dist	Average distance from shots made (yards)	Discrete numerical variable
FK	Free kicks	Discrete numerical variable
PK	Penalty kicks made	Discrete numerical variable
PKatt	Penalty kicks attempted	Discrete numerical variable
xG	Expected goals for	Continuous numerical variable
xGA	Expected goals against	Continuous numerical variable
Poss	Team’s possession in the match	Discrete numerical variable
Result	Match result (win, draw or loss)	W, D, L

The dataset is divided into a training set and a testing set. The training set includes data from the first six seasons (2017/18 - 2022/23), totaling 4,560 records to date, representing approximately 80% of the dataset. The test set takes data from the previous and current seasons (2023/24 - 2024/25), with 1,460 records to date, representing approximately the remaining 20%. Since predictions rely on the average statistics from a team's last six matches, the number of usable records in both sets decreases, as the average cannot be calculated until a team has played at least six matches, meaning that valid predictions begin from the 7th match onward. Considering there are 29 teams in the first six seasons and 2 new teams in the testing period, the number of records in the training and testing sets is reduced to 4,386 and 1,448 records, respectively. It is also crucial to note that the dataset is continuously updated as the current season progresses, causing both sets to grow over time.

Figure 1 shows why a technique to balance the data has to be implemented, because the frequency of each class demonstrates that there is a minority class, draw, with only 22.92% of records showing this result, whereas 38.54% represent victories and the other 38.54% represent defeats.

Figure 1: Match result distribution.



This is why SMOTE (Synthetic Minority Over-sampling Technique) is applied to address the tendency of draws to be underrepresented in football match outcomes. SMOTE produces synthetic samples for the minority class based on the values that are actually present, lessening the model's bias against predicting draws.

Model Implementation

The study focuses on the implementation of two machine learning models: Random Forest and XGBoost, and a deep learning algorithm: TabNet. Furthermore, for comparison, the final ensemble model is compared to a betting website and ChatGPT's predictions. It is worth noting that for each model, there are two training variations: the first variation trains the model only with the categorical variables, and the other trains the model not only with the categorical variables but with the statistical averages too. This allows for the variations to be compared and for analysis of the influence of statistical averages on the model.

Random Forest, a decision tree-based machine learning model that builds numerous decision trees and trains each one of them using random data samples, and then takes the majority vote of the decision trees' results as the final answer. The random forest algorithm includes: n_estimators, which represents the number of decision trees; min_samples_split, which represents the minimum number of samples a parent node needs in order to be divided into two child nodes; max_depth, representing the maximum depth or maximum number of child nodes in each decision tree; and class_weight, which in this case is specified as "balanced" so each class' weights are adjusted based on the class' frequency on the dataset.

XGBoost, is a decision tree-based model that follows a gradient boosting framework in which each new decision tree attempts to improve and correct the residuals of the prior decision tree, allowing for powerful predicting performance. The hyperparameters applied are: objective="multisoftmax" and num_class=3 to specify a multiclass problem consisting of three classes (loss, draw or win); n_estimators, for the number of decision trees; max_depth, representing the maximum depth for each decision tree, and learning_rate, which is responsible for the rate at which the algorithm learns from each iteration.

TabNet, the last algorithm implemented, is a deep learning model designed to work specifically with tabular data, able to identify relevant features in each step through a sequential attention-based mechanism. Key hyperparameters for TabNet include: optimizer_params, where the learning rate with which the model learns is specified; scheduler_params, where the model's step_size to adjust the learning rate in a staggered manner is set; max_epochs, to control the maximum number of times the

model trains the training set; batch_size and virtual_batch_size, which specify the number of samples processed before weight updates; and finally, the patience, which specifies the number of times where there is no accuracy improvement for the model to stop.

In order to improve each model's performance, GridSearch, a technique used for hyperparemeter optimization is applied. Here, a cross validation procedure is applied with fivefold partitions within the training data, which means that the dataset is trained in four partitions and validated on the fifth, and repeating this process five teams, which is why a validation set is not needed. TabNet's hyperparameters are not optimized using GridSearch due to the long processing time required to run this code, which is why hyperparameters are optimized manually by testing different values. Table 2 shows the algorithms' hyperparameters that are optimized and the chosen value.

TABLE 2: Hyperparameter optimization for Random Forest and XGBoost.

Algorithm	Hyperparameters	Values	Chosen
Random Forest	n_estimators	100, 200, 300, 500	300
	min_samples_split	10, 100, 200, 300	200
	max_depth	1, 10, 20, 30, 50	50
	class_weight	balanced	balanced
XGBoost	n_estimators	100, 200, 300, 500	200
	max_depth	1, 10, 20, 30, 50	1
	learning_rate	0.01, 0.1, 0.2	0.1

By using this technique, the dataset is trained using each possible combination for the specified hyperparameters, and the results are then compared based on the

desired metric, accuracy, with the objective of finding the best hyperparameters' values in order to maximize the model's accuracy.

Ensemble model

The implementation of different machine learning models means that one of them will have a better performance, but that does not mean that the rest of the models are bad, which is why it is crucial to find a way of leveraging each of the models' strengths by combining all three models, Random Forest, XGBoost, and TabNet, into an ensemble model so the final prediction model has more confidence in the results.

In order to create the ensemble model, a voting mechanism is created to combine the results of all three models, presented in the following pseudo code presented to choose a prediction:

1. *If all three models agree:*

   Any model chosen
2. *If none of the models agree:*

   Model with higher accuracy chosen
3. *If two models agree:*

   Majority class chosen

The prediction chosen using the voting mechanism is the final prediction.

RESULTS AND DISCUSSION

Model Evaluation

To evaluate the performance of each of the algorithms, several evaluation metrics are considered in order to compare the models and be able to understand how well they predict match outcomes compared to actual results in the test set. According to [8], the evaluation metrics to be analyzed are defined as:

- Accuracy: Measures the percentage of correct predictions compared to all predictions.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

- Precision: Represents the percentage of true positive predictions compared to all positive predictions.

$$Precision = \frac{TP}{TP + FP}$$

Where:

$TP = true\ positives,$

$TN = true\ negatives,$

$FP = false\ positives,$

$FN = false\ negatives.$

Two variations are considered: The first one considering only the categorical variables and the second one that considers both the categorical and statistical variables.

TABLE 3: Model accuracy for each algorithm variation.

Algorithm	1st Variation	2nd Variation

Random Forest	46.16%	49.86%
XGBoost	42.81%	51.52%
TabNet	38.63%	45.24%

Table 3 shows that in the first variation, where the models are trained only with the categorical variables, Random Forest, XGBoost and TabNet, achieved accuracies of 46.16%, 42.81% and 38.63% respectively. When applied the average statistics for the last six matches, the accuracy for each model improved significantly, reaching values of 49.86%, 51.52% and 45.24%. With these final results, the ensemble model considers each model and achieves a final accuracy of 52.83%, confirming that the combination of the three models is an important advantage when compared to each one of them separately.

Considering the confusion matrix for the ensemble model in figure 2, out of 1448 records, the model has correctly predicted 355 losses, 16 draws, and 394 wins.

Figure 2: Ensemble model confusion matrix.

final_pred	0	1	2
actual			
0	355	23	176
1	149	16	170
2	153	12	394

This means that each class has a precision of: 54.03% for losses, 31.37% for draws, and 53.24% for wins. Misclassifications show that 149 draws were predicted as losses and 170 were predicted as wins, confirming that even with SMOTE balancing, the draw class remains the hardest to learn and, consequently, the hardest to predict. However, the precision for each class is comparable to each individual model, as can be seen in table 4.

TABLE 4: Class precision for each model.

Model	Loss	Draw	Win
Random Forest	55.67 %	26.02%	53.85%
XGBoost	53.21%	26.32%	51.99%
TabNet	46.36%	26.23%	45.80%
Ensemble	54.03%	31.37%	53.24%

Table 4 shows that Random Forest achieved the highest precision when predicting losses, followed by the ensemble model, XGBoost, and TabNet. For draw predictions, the ensemble model outperformed the others considerably, succeeded by XGBoost, and then TabNet and Random Forest following closely behind. Finally, when predicting wins, the study shows that the model with the highest precision is Random Forest, again, followed by the ensemble model, XGBoost, and TabNet.

Feature Importance Analysis

A correlation analysis has been conducted using Pearson correlation coefficients to identify the key predictors that influence a football match's outcome. This analysis relies on both the Pearson correlation coefficient and the p-value. The p-value helps

understand whether the relationship between two variables is likely to be real (statistically significant) or just due to chance. Variables with a p-value less than 0.05 present strong evidence that the relationship with the match outcome is statistically significant. The Pearson correlation coefficient (r) measures the strength and direction of the relationship, where values closer to +1 or -1 indicate a stronger correlation, and values near 0 suggest there is no impact on the outcome. A high positive correlation means that as the variable increases, the likelihood of a win also increases; while, a high negative correlation suggests that an increase in the variable is associated to a decreased chance of winning, implying an increased chance of a loss.

The feature importance analysis result shows that there are six predictor groups that significantly impact the result: the team and the opponent, where correlation depends on the team; the venue, capturing the clear home-field advantage; formation and opponent formation, where correlation depends on the specific formation used; and finally, the six-match average for performance statistics-GF, GA, Sh, SoT, Dist, FK, PK, PKatt, xG, xGA, and Poss-which have shown a significant impact on the models' accuracies. Together, all these variables form the backbone of the model's predictions.

Figure 3 highlights the top 20 features that most significantly affect football match outcomes, considering only variables found to be statistically significant (p-value < 0.05). These variables are ranked based on both their statistical significance and strength of the relationship with the outcome.

Figure 3: Top 20 features that influence a football match.

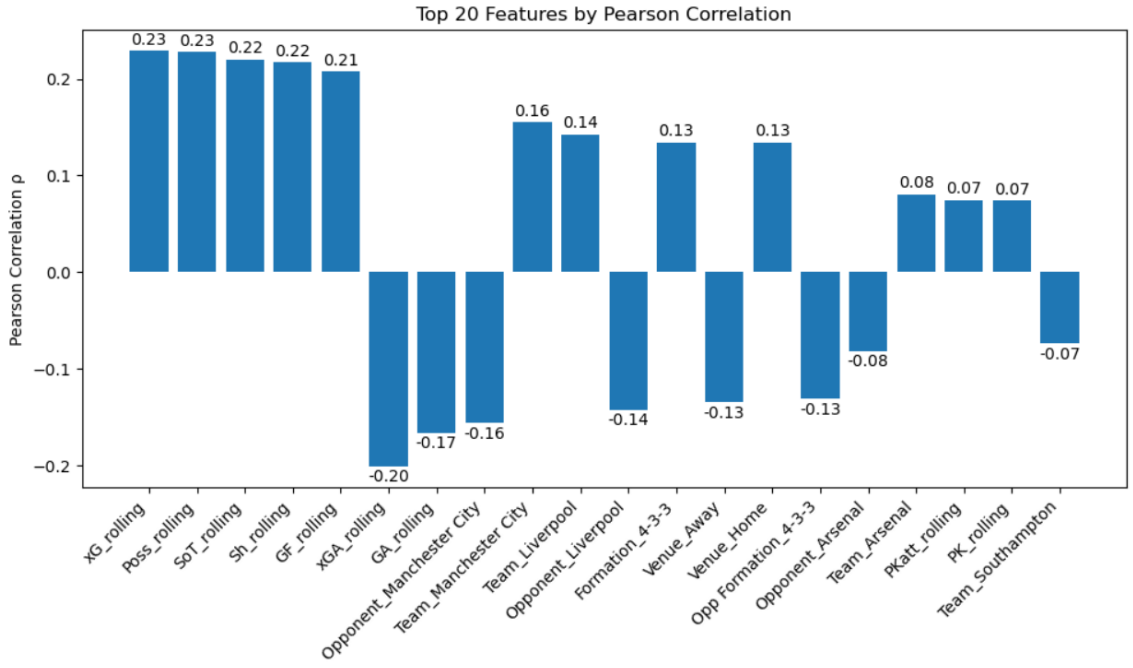


Figure 3 shows that the mentioned variables are the ones with the most influence on football match outcomes. Considering the results, it is evident that higher average values over the last six matches for metrics such as expected goals, possession, shots on target, and goals scored increase the chance of winning. Conversely, higher averages for expected goals against and goals conceded suggest a greater probability of losing. The analysis also reveals that teams like Manchester City and Liverpool are more likely to win matches; and consequently, teams that play against these teams are more likely to lose. Another key finding is the clear advantage of playing at home, which aligns with common expectations. Lastly, teams playing with a 4-3-3 formation have a higher probability of winning.

Comparison with external predictions

In addition to comparing the machine learning models and the final ensemble model, the present study also compares the ensemble's performance against outside sources like artificial intelligence (AI) and a betting house in order to examine the

strengths of the machine learning-ensemble predictions against traditional models and AI-driven methods.

- Betting House: Some betting houses share match predictions for bettors to have an idea what the match outcome will be and understand the betting quotas for each match. This outside source has been chosen because of the popularity of betting and because of the deep analysis every betting house needs to make in order to be profitable.
- Artificial Intelligence: AI has the power of creating an answer to the prompt provided, which is why ChatGPT has been chosen as external source. The given prompt is: “I want you to predict the following Premier League matches as an expert using historical and actual data. For each match I want you to give me just the teams along with the predicted result: W (win), D (draw) or L (loss). Give me the results with the highest possible accuracy please. These are the matches with some data:”, and then the match data along with information for each of the categorical variables are included.

Table 5 shows the calculated accuracy based on the predicted results to the date, consisting of 60 matches viewed from both perspectives, accounting for a total or 120 records.

TABLE 5: Prediction accuracy for each source.

Source	Accuracy
Ensemble model	48.33%

Betting House	51.67%
ChatGPT (AI)	51.67%

The comparison results show that at this point, both the betting house and AI have similar results, showing an accuracy of 51.67%, while the ensemble model presents an accuracy of 48.33%. Even though the last one has a lower accuracy compared to the other two, it is worth noting that the ensemble model achieved an accuracy of 52.83% in the test set. On a day-to-day basis, there is not a model that outperforms any of the other two, but instead there are matchweeks where one might be better than the other two. In general, the ensemble model relates to the outside sources in this aspect of variability. This comparison also shows that the predictions in which all three sources agree have a better chance of occurring.

CONCLUSIONS AND FUTURE WORK

The study shows that combining each individual model-Random Forest, XGBoost, and TabNet-through a voting-mechanism ensemble model provides the most accurate predictions, achieving an overall accuracy of 52.83%. This performance, while seemingly modest, outperforms simple heuristics such as choosing a random result, picking the home team, or always going for the win, and it is done so in the Premier League, which is arguably the best and most volatile local competition in world football. The comparison of class precision reveals that Random Forest performs best than the other models when predicting both losses and wins, while the ensemble model achieves the highest precision when predicting draws. The correlation analysis concludes that the variables with the most significant impact on Premier League match outcomes include team, opponent, venue, each team's formation, and the six-match average of performance statistics. Among these, rolling averages, specifically for expected goals, possession, shots on target, goals scored, expected goals against, and goals conceded, stand out as the most influential predictors. Out of the three individual models, XGBoost has proven to be better because of its constant higher accuracy, followed by Random Forest, while TabNet has proven to be more volatile as some matchweeks shows a higher accuracy, but always being less than or equal to Random Forest's. Comparing the ensemble model with external sources' predictions, specifically from a betting house and artificial intelligence, further boosts confidence because prediction agreements coincide with higher hit rates.

Despite these positives, some limitations present an obstacle to the results. One key issue is that the model occasionally predicts logically impossible results by

assigning the same outcome (win or lose) to both teams, which should only happen when predicting draws. This is a consequence of treating each team's record separately. Another important finding, which highlights a key area for improvement, is that draws remain especially hard to predict even after applying SMOTE balancing. Reliance on a single webpage makes the web scraping model vulnerable to changes depending not only on the web page's existence but also on any changes made on the webpage. Some information regarding individual player statistics, injuries, transfers, coach transfers, etc. is not taken into account, which might impact the model in the aspect of leaving meaningful information out of the equation. Together, these limitations help explain why roughly half of the predictions are inaccurate.

Future work should therefore focus on four specific aspects. First, building a robust, multi-source web scraping model with automated pipeline rotation in order to avoid getting the IP address blocked at any point. Then, enriching the dataset with more data, specifically more team performance statistics because of ease of use. Also, there is still a lot of ground to cover regarding model optimization, this means implementing other machine learning or deep learning models and finding the ones that perform best and also through hyperparameter optimization. Finally, evaluate betting strategies under realistic conditions in order to improve predictions and create a practical tool not only for sport bettors, but also for the teams themselves, analysts, and fans.

REFERENCES

[1] Dey, A. (2016). Machine learning algorithms: a review. International Journal of Computer Science and Information Technologies, 7(3), 1174-1179.

[2] Stübinger, J., Mangold, B., & Knoll, J. (2019). Machine learning in football betting: Prediction of match results based on player characteristics. *Applied Sciences*, *10*(1), 46.

[3] Rodrigues, F., & Pinto, Â. (2022). Prediction of football match results with Machine Learning. *Procedia Computer Science*, *204*, 463-470.

[4] Arik, S. Ö., & Pfister, T. (2021, May). Tabnet: Attentive interpretable tabular learning. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 35, No. 8, pp. 6679-6687).

[5] de Stefano, E., de Oliveira Farroco, L., Lima, G. B. A., Parrancho, A., Gavião, L. O., & Principe, V. A. (2020). Decision Trees for the Prediction of Outcome of Soccer Games-Historical Data Analysis. *Brazilian Journal of Development*, *6*(1), 4719-4732.

[6] Rumsey, T. G. (2024). A Statistical Look into how Common Soccer Metrics Influence Expected Goal Measures in the Professional Game.

[7] Kumar, S., & Roy, U. B. (2023). A technique of data collection: web scraping with python. In *Statistical Modeling in Machine Learning* (pp. 23-36). Academic Press.

[8] Sathyanarayanan, S., & Tantri, B. R. (2024). Confusion matrix-based performance evaluation metrics. African Journal of Biomedical Research, 4023-4031.

[9] Malikov, D., & Kim, J. (2024). Beyond xG: A Dual Prediction Model for Analyzing Player Performance Through Expected and Actual Goals in European Soccer Leagues. *Applied Sciences*, *14*(22), 10390.

[10] Rico-González, M., Pino-Ortega, J., Méndez, A., Clemente, F., & Baca, A. (2023). Machine learning application in soccer: a systematic review. *Biology of sport*, *40*(1), 249-263.

[11] Zhao, Q., Bie, Z., & Li, Y. (2025). Exploration of Sports Data Analysis and Fitness Effect Optimization Strategies using XGBoost.

[12] Barron, D., Ball, G., Robins, M., & Sunderland, C. (2018). Artificial neural networks and player recruitment in professional soccer. *PloS one*, *13*(10), e0205818.

[13] Joseph, A., Fenton, N. E., & Neil, M. (2006). Predicting football results using Bayesian nets and other machine learning techniques. *Knowledge-Based Systems*, *19*(7), 544-553.

[14] Van Roy, M., Robberechts, P., Yang, W. C., De Raedt, L., & Davis, J. (2021). Leaving goals on the pitch: Evaluating decision making in soccer. *arXiv preprint arXiv:2104.03252*.

[15] Martínez de la Rosa, C. (2024). *Aplicación de algoritmos de aprendizaje automática y ciencia de datos para la predicción de resultados de partidos de fútbol* (Doctoral dissertation, Universitat Politècnica de València).

[16] Anjum, S., & Fatima, A. (2023). Predictive Analytics For FIFA Player Prices: An ML Approach. *Journal of Scientific Research and Technology*, 204-212.

[17] Davis, J., Bransen, L., Devos, L., Jaspers, A., Meert, W., Robberechts, P., ... & Van Roy, M. (2024). Methodology and evaluation in sports analytics: challenges, approaches, and lessons learned. *Machine Learning*, *113*(9), 6977-7010.

[18] Rahimian, P., Mihalyi, B. M., & Toka, L. (2024). In-game soccer outcome prediction with offline reinforcement learning. *Machine Learning*, *113*(10), 7393-7419.

[19] Berrar, D., Lopes, P., & Dubitzky, W. (2024). A data-and knowledge-driven framework for developing machine learning models to predict soccer match outcomes. *Machine Learning*, *113*(10), 8165-8204.

[20] Rahman, M. A. (2020). A deep learning framework for football match prediction. *SN Applied Sciences*, *2*(2), 165.

[21] Cui, K., Li, X., & Yang, S. (2024). Intelligent Prediction of the Sport Game Outcome Using a Hybrid Machine Learning Model. *Tehnički vjesnik*, *31*(6), 2167-2175.

[22] Salman, H. A., Kalakech, A., & Steiti, A. (2024). Random forest algorithm overview. *Babylonian Journal of Machine Learning*, *2024*, 69-79.

[23] Zhang, P., Jia, Y., & Shang, Y. (2022). Research and application of XGBoost in imbalanced data. *International Journal of Distributed Sensor Networks*, *18*(6), 15501329221106935.

[24] Brandt, J., & Lanzén, E. (2021). A comparative review of SMOTE and ADASYN in imbalanced data classification.

[25] Belete, D. M., & Huchaiah, M. D. (2022). Grid search in hyperparameter optimization of machine learning models for prediction of HIV/AIDS test results. *International Journal of Computers and Applications*, *44*(9), 875-886.