MACHINE LEARNING IN FOOTBALL: PREDICTING TEAM PERFORMANCE USING TACTICAL FEATURES

PANAGIOTIS SIARAFERAS

THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE IN DATA SCIENCE & SOCIETY
AT THE SCHOOL OF HUMANITIES AND DIGITAL SCIENCES
OF TILBURG UNIVERSITY

STUDENT NUMBER

u340068

COMMITTEE

dr. Paris Mavromoustakos Blom
dr. Giacomo Spigler

LOCATION

Tilburg University
School of Humanities and Digital Sciences
Department of Cognitive Science &
Artificial Intelligence
Tilburg, The Netherlands

DATE

December 2nd, 2024

WORD COUNT

8778

ACKNOWLEDGMENTS

I want to sincerely thank my supervisor, Dr. Paris Mavromoustakos Blom, for his support, motivation, and guidance throughout my thesis. His advice and feedback helped me stay on track and complete this work successfully. I am also grateful to Tilburg University for providing the courses and resources that gave me the knowledge and skills I needed for this research. A big thank you to my family and friends for always being there for me and giving me emotional support. Your encouragement made a big difference during this journey. Finally, I want to thank my fellow students for their friendship and support.

MACHINE LEARNING IN FOOTBALL: PREDICTING TEAM PERFORMANCE USING TACTICAL FEATURES

PANAGIOTIS SIARAFERAS

Abstract

This study explores how machine learning models can predict football team performance across six European leagues using tactical features. It focuses on three tasks: predicting goals scored (regression), classifying matches as over or under 2.5 goals (binary classification), and predicting match outcomes as win, draw, or lose (multi-class classification). Unlike previous studies that often rely on traditional features or single-league data, this research integrates tactical factors to improve predictions and test model generalizability across leagues, using a dataset with tactical features, models such as Random Forest and XGBoost were applied and evaluated. The findings show that tactical features play a key role in prediction accuracy, particularly in tasks involving match outcomes and goal classifications. However, challenges remain in predicting exact goals and generalizing models to leagues with different playing styles.

1 DATA SOURCE, ETHICS, CODE, AND TECHNOLOGY STATEMENT

This thesis uses a publicly available dataset from Kaggle, titled "Football Data: Expected Goals and Other Metrics". The dataset includes tactical data from six major European football leagues (2014–2019). The preprocessing step included removing irrelevant or highly correlated features, creating new features through interactions, and converting categorical variables to numerical values. No data was collected from human participants or animals, so no ethical review was required. The code written for this thesis is available on GitHub. All figures and visualizations were created by the author, and no external images were used. The dataset's original owner retains all rights, and the author acknowledges adherence to the dataset's usage guidelines. Various libraries and tools were utilized in this

study, and their names along with corresponding sources are detailed in Appendix A (page 40).

2 INTRODUCTION

2.1 *Motivation and relevance*

As technology has grown, the amount of data being created has increased rapidly, leading to new ways to analyze and understand large amounts of information (Rydning et al., 2018). Machine learning (ML) has become a powerful tool in sports, helping to improve performance and develop strategies. In sports, Machine Learning models are used to predict match outcomes, evaluate player performance, and study team tactics (Beal et al., 2019). By analyzing large datasets—like player movements, match statistics, and past performances—ML provides useful insights that help coaches and analysts make better decisions (Brefeld et al., 2018).

Football is one of the most popular sports globally (Cotta et al., 2016). Sports analysts utilize statistics and advanced metrics to predict outcomes and the performance of both teams and individual players (Pantzalis & Tjortjis, 2020). While Expected Goals is an advanced metric that estimates the likelihood of a shot resulting in a goal (Green, 2012), their focus on probabilistic outcomes provides limited insight into the tactical dynamics of the game (Spearman, 2018). Recent advancements in football analytics emphasize the importance of tactical features, such as pressing intensity, pass effectiveness, and team dynamics, as key determinants of match performance (Zhou et al., 2023). Pressing intensity, for instance, captures the degree of defensive pressure applied by a team, while passing efficiency reflects the ability to create scoring opportunities from possession. By using these tactical elements, analysts can gain a deeper understanding of how strategic decisions influence match outcomes. However, the integration of tactical variables into predictive models remains underexplored, presenting an opportunity to enhance our ability to predict team performance using machine learning techniques.

This project investigates the performance of machine learning models in predicting football team performance across six major European leagues. Unlike traditional methods, this research focuses on predicting actual goals and match outcomes instead of just probabilities, using tactical features as key inputs. The project uses three key ML methodologies: regression models to predict the number of goals scored, binary classification models to determine if the actual goals scored are under/over 2.5, and a broader classification task to predict match outcomes (win, draw, or loss). By comparing models like Linear Regression, Logistic Regression, Random

Forest, and XGBoost, the research seeks to uncover the most effective techniques for tactical feature-based predictions.

While the dataset represents post-match statistics, the analysis highlights crucial elements, such as team dynamics and pressing intensity, that significantly influence scoring outcomes. These findings are essential for improving team evaluations, strategy planning, and player performance assessments. Moreover, the methods developed in this thesis can be adapted for pre-match scenarios, enabling teams to design more effective strategies for the upcoming matches. By advancing the understanding of goal-scoring dynamics, this project delivers practical tools to improve team-level analysis in football.

2.2 *Societal Relevance*

From societal relevance this research provides useful tools to help coaches, analysts, and clubs improve team performance and plan better strategies. The findings can change how football is played and studied, making it easier to develop new approaches for training and tactics. Off the pitch, this work enhances fan experiences by offering deeper insights into the game, supports fairness in betting markets through better predictions, and drives improvements in sports entertainment. By connecting data science with football, this study helps make the sport more exciting, competitive, and accessible for everyone.

2.3 *Scientific Relevance*

From, scientific relevance this research makes an important contribution to sports analytics by focusing on the ability of tactical features to predict team performance, rather than relying on traditional metrics like Expected Goals (xG). Using machine learning, the study examines how factors such as pressing effectiveness, defensive actions, and the intensity of pressing between teams influence match outcomes. For example, features like the balance between attack and defense or the number of defensive actions in key areas provide valuable insights into team strategies. This approach improves the accuracy of predictions and helps us better understand the factors that impact team success. In doing so, this research adds to the growing field of sports analytics and introduces new methods for analyzing team tactics and performance.

2.4 *Research Questions*

Predicting the performance of football teams using machine learning has become a growing focus in sports analytics. However, most studies have primarily concentrated on top leagues or limited metrics, leaving room to explore the predictive power of tactical features across diverse leagues. This research aims to address these gaps by evaluating the effectiveness of machine learning models and tactical features in predicting match outcomes and team performance.

To what extent can machine learning models accurately predict the performance of football teams across six European leagues using tactical features?

This main research question is supported by the following sub-questions:

1. **Which machine learning model performs best for predicting team performance in terms of goals scored and match outcomes?** This question aims to identify the most effective machine learning model for predicting team performance. The analysis will evaluate the performance of various models, including regression and classification approaches, using metrics such as Mean Absolute Error (MAE), R^2, classification accuracy, and F1-score.

2. **How do feature selection and tactical variables, such as pressing intensity and home/away status, impact the accuracy of these predictions?** This question investigates the role of tactical features and feature selection techniques in improving model performance. Key tactical metrics, such as pressing intensity, home/away status, and defensive actions, will be analyzed to understand their contribution to predictions. The study will explore how different combinations of features and transformations enhance the accuracy and interpretability of the models. The objective is to determine the most impactful variables for building robust and explainable predictive models.

3. **How well do the developed models generalize across different leagues?** This question investigates whether machine learning models trained on data from one league can accurately predict outcomes in another league. By examining cross-league predictions, the study seeks to determine if certain leagues share tactical similarities that enhance model transferability. For example, a model trained on one league might perform better in a tactically similar league compared to a very different one. This analysis will include metrics such as prediction accuracy feature importance, and feature distribution to evaluate how well the models adapt to new leagues and to identify clusters of leagues with comparable tactical characteristics.

2.5 *Findings*

The primary goal of this study was to predict team performance using tactical features exclusively. Key findings demonstrate that advanced models like Random Forest and XGBoost effectively capture the relationships between tactical variables and target outcomes. Additionally, this research highlights the critical role of tactical data in forecasting team performance and reveals significant differences in playing styles across leagues.

3 RELATED WORK

The application of machine learning (ML) and big data analytics has transformed sports science (Bai & Bai, 2021), enabling deeper insights and improving predictive capabilities. This section reviews key studies in ML applications in sports and football, with a focus on team performance and their evolution. By examining existing gaps in the literature, this review highlights the need for a more comprehensive approach to predicting actual goals and match outcome using match and team-level data.

3.1 *Machine Learning in Sports*

Machine learning has become a powerful tool in sports analytics, offering advanced methods to analyze performance metrics, optimize strategies, and engage fans. In a comprehensive review, (Brefeld et al., 2018) explored how big data and ML have revolutionized sports, enabling innovative ways to analyze player and team performance. (Richter et al., 2024) further emphasized the role of ML in addressing challenges such as predictive accuracy and complex data management in sports science. These studies demonstrate how ML enhances player evaluation, tactical planning, and injury prevention. However, most current applications focus on individual performance metrics rather than team-level dynamics, highlighting a gap in broader strategic analyses.

3.2 *Machine Learning in Football Analytics*

The use of machine learning (ML) in football analytics has become increasingly prominent, enabling advancements in the prediction of match outcomes and player performance. For example, (Choi et al., 2023) explored logistic regression and random forests for predicting match results. Their study highlighted the critical role of feature selection and rigorous model evaluation in achieving high prediction accuracy.

Deep learning techniques have also shown promise in football analytics. (Rahman, 2020) developed a deep learning framework that leverages historical match data and player statistics to predict match outcomes. Their findings demonstrated how neural networks effectively capture complex patterns in football data, achieving superior results compared to traditional ML models.

Machine learning is not only used to predict match outcomes but also to evaluate player performance. (Wisdom & Javed, 2023) applied regression models to assess metrics such as passing accuracy and defensive actions, linking individual performance to team success. This approach offers valuable insights for player evaluation and coaching strategies.

While these studies underscore the effectiveness of ML in football analytics, a key limitation remains: many approaches focus on traditional features or top leagues, neglecting tactical elements such as pressing intensity or defensive actions. This study seeks to fill this gap by integrating tactical metrics into predictive models to enhance accuracy and generalizability.

3.3 *Tactical Features and Their Role in Performance Prediction*

Tactical features, such as pressing intensity, defensive actions, and team dynamics, play a significant role in football analytics. Several studies have explored these aspects, highlighting their potential to improve performance predictions and tactical understanding.

For instance, (Goes et al., 2021) examined tactical performance using position tracking data, extracting spatiotemporal features based on offensive principles. By training a machine learning classifier on these features, the authors achieved fair to good accuracy in predicting match outcomes. Their work demonstrates how tactical metrics derived from tracking data can inform team-level analyses and match predictions, but it leaves room for further exploration of defensive and pressing features, particularly across multiple leagues.

Similarly, (García-Aliaga et al., 2021) utilized machine learning and computer vision to analyze players' technical-tactical behaviors during matches. Their study provided valuable insights into how player positions and actions contribute to tactical behavior, demonstrating the potential of AI to characterize on-field dynamics. However, the focus was largely on individual player actions, leaving the integration of broader team-level tactical metrics relatively unexplored.

In another study, (Bauer & Anzer, 2021) introduced a supervised machine learning approach to detect counterpressing situations using synchronized positional and event data. By focusing on specific tactical scenarios,

this research highlighted how combining tactical features with ML models can enhance predictive accuracy. Nevertheless, the study was limited to counterpressing situations and did not evaluate broader predictive tasks such as match outcomes or goals scored.

Finally, (Forcher et al., 2024) conducted an in-depth analysis of defensive pressure and pressing characteristics in German Bundesliga, utilizing tracking data to identify the key factors that lead to successful ball recovery. Their research provided valuable tactical insights, such as the spatial and temporal dynamics of pressing, and highlighted the importance of defensive pressure in shaping match outcomes. However, the study primarily employed statistical methods to analyze these tactical features, focusing on descriptive patterns rather than predictive modeling. This approach, while insightful, limits its applicability to broader contexts, such as predicting match outcomes or team performance. By not integrating machine learning, the study misses the opportunity to explore how these defensive metrics could contribute to predictive tasks, such as forecasting goals scored or identifying match results.

3.4 *Research Gaps and Project Contributions*

Although football analytics has advanced, there are still important gaps in how tactical features are used in predictive models. Many studies focus on individual performance or specific tactics like counterpressing, without looking at a wide range of tactical elements such as pressing intensity, defensive actions, and team coordination. Most research also focuses on top leagues and traditional data, ignoring how tactical differences across leagues affect predictions. Additionally, many approaches rely on basic statistical methods instead of machine learning, which can better analyze and predict outcomes like goals scored or match results. This project aims to fill these gaps by using machine learning to include diverse tactical features in models, improving prediction accuracy and making the models work better across different leagues. This will provide deeper insights into team performance and better tools for football analysis.

4 METHOD

This section outlines the process for predicting actual goals and perfomance of teams across six European Leagues using tactical features. The methodology focuses on comparing various machine learning algorithms, including both regression and classification methods, to determine the most effective model. The pipeline starts with pre-processing raw data, followed by feature engineering, where feature interactions are created to improve

prediction accuracy. Next, models are trained, optimized through hyper-parameter tuning, and evaluated using performance metrics, as shown in Figure 1.

4.1 *Dataset*

The dataset, available at Kaggle, employed in this project encompasses match-level tactical data from six major European football leagues, La Liga from Spain, English Premier League (EPL), BundesLiga from German, Serie A from Italy, Ligue 1 from France and Russian Premier League (RFPL). This data spans from 2014 to 2019 and consists of 24,580 entries and 29 columns. It includes a mix of both numerical and categorical variables. The dataset can be divided into five categories, each providing distinct insights into various aspects of football matches, Figure 2.

The first category is match context, which provides essential information about each game. This includes league details, the season year, participating teams, the date of the game and whether a team played at home or away. Features in this category "league", which identifies the league name, "year", indicating the season year, "h_a", marking home or away matches, "team", representing the team name, and "date", specifying the match date.

The second category is expected performance metrics. These features estimate match outcomes based on statistical models, evaluating the likelihood of scoring or conceding goals. Key metrics such as xG (expected goals) and xGA (expected goals against) are derived from shot quality and contextual factors. Refined metrics like npxG (non-penalty expected goals) and npxGA (non-penalty expected goals against) exclude penalties and own goals, offering a more nuanced view. Differential measures such as xG_diff (difference between expected and actual goals) and xpts_diff (difference between expected and actual points) highlight discrepancies, providing insights into over- or under-performance.

The third category focuses on actual performance metrics, capturing match outcomes. These include the number of goals scored and conceded, as well as points earned during a match. Features like scored and missed reflect the goals scored by and against a team, respectively, while the result indicates the match outcome as a win, draw, or loss. Additional indicators, such as wins, draws, and losses, provide binary classifications of match results. The pts column records points earned, offering a comprehensive view of a team's performance during the season.

Pressing and defensive metrics form the fourth category, providing insights into defensive intensity and pressing efficiency. These features measure actions like passes allowed per defensive action in the oppo-

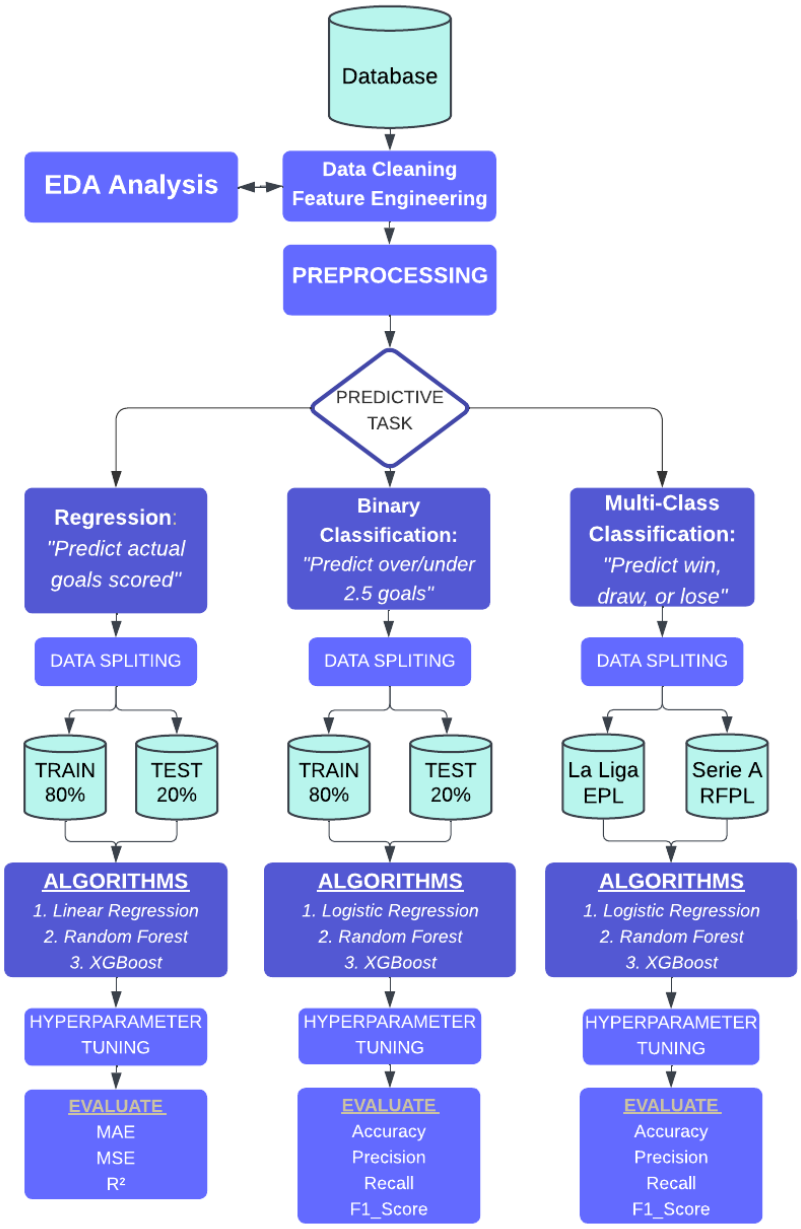

Figure 1: Research methodology flowchart showing steps for regression, binary classification, and multi-class classification tasks.

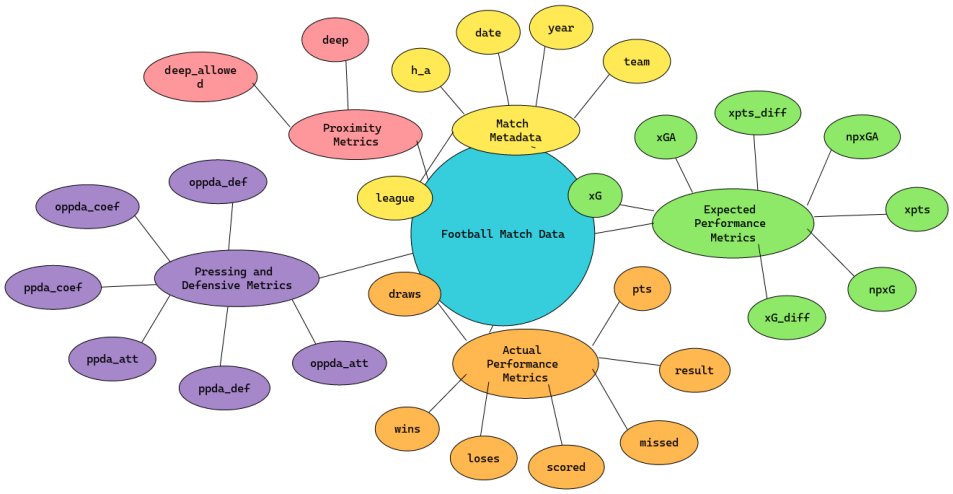

Figure 2: Dataset Overview

sition's half, a crucial indicator of pressing strategies. Metrics such as "ppda_coef" capture pressing effectiveness, while attributes like "ppda_att" and "ppda_def" measure attacking and defensive contributions to pressing. Similarly, the opponent's defensive efficiency is reflected through "oppda_coef", with supporting metrics such as "oppda_att" and "oppda_def".

Finally, proximity metrics focus on territorial dominance near the goal area. These features analyze ball progression and defensive vulnerability, emphasizing actions close to the opponent's or team's goal. Metrics such as "deep" measure the number of passes completed within 20 yards of the opponent's goal, while "deep_allowed" counts the passes allowed in the same zone near a team's goal. These metrics provide valuable insights into offensive penetration and defensive resilience.

4.2 *Preprocessing*

In this section, the preprocessing steps applied to the dataset are described. The dataset initially included numerous features directly related to scoring outcomes, such as expected goals "xG", non-penalty expected goals "npxG", and expected goals against "xGA". Additionally, several features captured post-match results, including points earned by the team "pts", expected points "xpts", and match outcomes (win, draw, or loss).

Since the goal of this study was to predict actual goals scored and team performance based on tactical features, it was inappropriate to include post-match features that directly reflected the outcomes being predicted. Therefore, all features related to scoring or match results were excluded.

Features	Description
ppda_coef	Passes allowed per defensive action in the opposition half (power of pressure)
oppda_coef	Opponent passes allowed per defensive action in the opposition half (power of opponent's pressure)
deep	Passes completed within an estimated 20 yards of goal (crosses excluded)
deep_allowed	Opponent passes completed within an estimated 20 yards of goal (crosses excluded)
ppda_att	PPDA attacking actions
ppda_def	PPDA defensive actions
oppda_att	OPPDA attacking actions
oppda_def	OPPDA defensive actions
h_a	Home or Away game
league	League

Table 1: Features retained after the removal of all goal-related features.

The remaining features shown in Table 1, which focus on tactical aspects of the match, were retained to evaluate their impact on team performance and outcomes. These tactical features included pressing metrics, such as passes allowed per defensive action in the opponent's half "ppda_coef" and the corresponding metric for the opponent "oppda_coef". Additional metrics related to pressing contributions, such as attacking and defensive pressing actions "ppda_att" and "ppda_def", as well as the opponent's pressing equivalents "oppda_att" and "oppda_def", were also included.

Proximity metrics, which provide insights into territorial dominance near the goal area, were incorporated as well. These included the number of passes completed within 20 yards of the opponent's goal "deep" and passes allowed in the same zone near a team's goal "deep_allowed". Contextual features, such as whether the team played at home or away "h_a" and the league in which the match took place "league", were also included to provide additional context for the tactical evaluation. These features and their descriptions can be found in Table 1.

The next step in the preprocessing pipeline was handling outliers in the target variable, "scored." Outliers can affect machine learning models, especially those sensitive to extreme values, such as linear regression. To address this, the Interquartile Range (IQR) method was used, as it is well-suited for non-normal distributions like the left-skewed distribution observed in "scored." The IQR method identifies outliers based on the

interquartile range, calculated as the difference between the 75th percentile (Q3) and the 25th percentile (Q1), (Vinutha et al., 2018). Data points below

$$Q1 - 1.5 \times \text{IQR}$$

or above

$$Q3 + 1.5 \times \text{IQR}$$

were classified as outliers and removed. After applying this method, 160 outliers were detected and removed, representing approximately 0.65% of the entries. The final shape of the dataset was 24,420 rows and 13 columns, ensuring the dataset was clean and free from extreme values that could skew model performance.

The final step in the preprocessing pipeline was encoding categorical variables to make them suitable for machine learning algorithms, which require numerical input. For this purpose, one-hot encoding was applied to the categorical features "h_a" (home/away status) and "league." One-hot encoding was chosen because it creates binary columns for each category, preserving all information without imposing ordinal relationships that do not exist in these variables. This method is particularly suitable for the models used in this study, such as Random Forest and XGBoost, as they can efficiently handle binary features without requiring further scaling (Manai et al., 2023).

4.3 *Feature engineering*

Feature engineering was a critical step in preparing the dataset for predicting team performance. The focus was on deriving meaningful tactical metrics that capture pressing intensity, defensive performance, and territorial dominance, which are essential for understanding team performance. Table 2 presents the final features used in the models. Below are the features engineered and the reasoning behind their inclusion:

1. **ppda_efficiency** and **oppda_efficiency**: These metrics measure the ratio of pressing actions in attacking and defensive scenarios for both the team and its opponent. Pressing efficiency is a strong indicator of a team's ability to disrupt the opponent's build-up play, which directly impacts scoring opportunities.

2. **relative_ppda_efficiency**: The ratio of the team's pressing efficiency to the opponent's pressing efficiency. This captures the balance of pressing dominance between the two teams, helping to contextualize a team's tactical effectiveness relative to its opponent.

3. **ppda_intensity** and **oppda_intensity**: Calculated by multiplying the pressing coefficient (ppda_coef) with pressing actions (ppda_att for attacking and ppda_def for defensive). Pressing intensity represents the overall pressure a team exerts during a match, which is often linked to goal-scoring opportunities or defensive vulnerabilities.

4. **intensity_diff**: The difference between the team's pressing intensity and the opponent's pressing intensity. This highlights the tactical imbalance in pressing, which can reveal whether a team has the upper hand in controlling the match.

5. **deep_x_ppda_intensity**, **deep_x_oppda_intensity**, and **deep_x_intensity_diff**: Interaction terms between territorial dominance (deep) and pressing metrics. Interaction features account for scenarios where pressing strategies near the opponent's goal (or defensive zone) might affect the likelihood of scoring.

6. **deep_squared**, **deep_cubed**, **intensity_diff_squared**, and **intensity_diff_cubed**: Higher-order polynomial terms of deep and intensity_diff. Polynomial features capture non-linear relationships between these tactical metrics and goal outcomes, which linear models like Linear Regression cannot detect.

4.4 *Data split and models used*

The process of training and testing the machine learning models for predicting actual goals scored and team performance was designed to ensure robust evaluation. The dataset was divided into training and testing subsets, and the experiments followed a structured data science pipeline to maximize predictive performance. For models, Linear Regression, Logistic Regression, Random Forest, and XGBoost were employed to explore the relationships between tactical features and target variables. The dataset in four models was split into two subsets, the training set (80%) used to train machine learning models and the testing set (20%) used to evaluate the model performance on unseen data. A random seed of 42 was used to make sure the data split could be repeated consistently. To avoid any chance of information from future matches affecting the training process, the testing data was kept separate and followed the natural order of the matches. This approach ensured the models were tested fairly and could provide reliable results.

Additionally, a different approach was used for the multi-class classification task. Instead of splitting the dataset into training and testing sets by percentage, the training data came from one league, while the testing

Feature Name	Description
ppda_coef	Passes allowed per defensive action in the opposition half (power of pressure).
oppda_coef	Opponent passes allowed per defensive action in the opposition half (power of opponent's pressure).
deep	Passes completed within an estimated 20 yards of goal (crosses excluded).
deep_allowed	Opponent passes completed within an estimated 20 yards of goal (crosses excluded).
ppda_att	PPDA attacking actions.
ppda_def	PPDA defensive actions.
oppda_att	OPPDA attacking actions.
oppda_def	OPPDA defensive actions.
h_a	Home or Away game.
league	League.
ppda_efficiency	Ratio of pressing actions in attacking and defensive scenarios, indicating a team's ability to disrupt the opponent's build-up play.
relative_ppda_efficiency	Balance of pressing dominance between the team and its opponent.
ppda_intensity	Pressing coefficient multiplied by pressing actions, representing overall match pressure exerted by the team.
oppda_intensity	Opponent's pressing coefficient multiplied by pressing actions.
intensity_diff	Difference between the team's pressing intensity and the opponent's pressing intensity.
deep_x_ppda_intensity	Interaction between territorial dominance and pressing metrics near the opponent's goal.
deep_x_oppda_intensity	Interaction between territorial dominance and opponent's pressing metrics near the defensive zone.
deep_x_intensity_diff	Interaction term capturing tactical imbalances near critical areas of the pitch.
deep_squared	Higher-order polynomial term of deep to capture non-linear relationships.
deep_cubed	Third-order polynomial term of deep to capture complex non-linear relationships.
intensity_diff_squared	Higher-order polynomial term of intensity difference.
intensity_diff_cubed	Third-order polynomial term of intensity difference.

Table 2: Final Features Used in the Models.

data was taken from another league. This cross-league setup allowed us to assess how well the models could generalize to unseen data with potentially different tactical styles, providing additional insight into the robustness of the models.

Regarding the algorithms used to predict actual goals (regression approach), three models were selected. The first model, Linear Regression, was used as a baseline to evaluate the linear relationships between tactical features and actual goals scored. As a simple and interpretable algorithm, it served as a benchmark for comparison with more complex models. The second model, Random Forest, was chosen for its ability to model non-linear interactions and its robustness against overfitting. Finally, XGBoost, a gradient-boosting algorithm, was selected for its ability to handle structured data effectively and its strong performance in capturing complex relationships.

Three models were also applied to the algorithms used to predict Over/Under 2.5 Goals (a binary classification approach) and to predict match outcomes (a multi-class classification approach). The first, Logistic Regression, was selected as a probabilistic baseline for classification tasks. For the multi-class classification task, the multinomial Logistic Regression approach was used, which leverages the softmax function to predict probabilities across multiple classes. The second, Random Forest, was applied to capture non-linear relationships between features effectively. Lastly, XGBoost was employed for its good classification performance and computational efficiency.

4.5 *Hyperparameter tuning*

Optimizing hyperparameters is a critical step in developing effective machine learning models, particularly for tree-based algorithms and neural networks (Yang & Shami, 2020). This section presents the hyperparameter tuning strategies and optimal configurations used across the three predictive tasks: regression, binary classification, and multi-class classification.

Hyperparameter tuning for the regression task was conducted to optimize the performance of Random Forest and XGBoost. Key hyperparameters, including the number of trees (n_estimators), maximum depth (max_depth), and feature selection methods, were tested across defined ranges. For Random Forest, the optimal configuration included 200 estimators and a maximum depth of 10. Similarly, XGBoost achieved its best results with 100 estimators, a learning rate of 0.05, and a maximum depth of 6. Regularization parameters and sampling ratios were also fine-tuned to enhance generalization and prevent overfitting. Detailed settings and their optimal values are summarized in Table 3.

Model	Hyperparameter	Range Tested	Optimal Value
Random Forest	n_estimators	{100, 200}	200
	max_depth	{5, 10, 15}	10
	max_features	{'sqrt', 'log2'}	'sqrt'
	min_samples_leaf	{1, 5, 10}	5
	min_samples_split	{2, 5, 10}	10
XGBoost	n_estimators	{50, 100, 200}	100
	learning_rate	{0.01, 0.05, 0.1}	0.05
	max_depth	{2, 6, 9}	6
	subsample	{0.5, 0.8, 1.0}	0.8
	colsample_bytree	{0.6, 0.8, 1.0}	0.8
	reg_alpha	{0, 1, 10}	1
	reg_lambda	{0, 1, 10}	1

Table 3: Hyperparameter Tuning for Regression Task

For the binary classification task of predicting whether a match resulted in over or under 2.5 goals, hyperparameter tuning was conducted for Logistic Regression, Random Forest, and XGBoost. Logistic Regression, used as the baseline model, required minimal tuning. The maximum number of iterations (max_iter) was set to 2000 to ensure convergence, and the class_weight parameter was adjusted to 'balanced' to address the class imbalance. Random Forest achieved optimal results with 200 estimators and a maximum depth of 10, leveraging its ability to model non-linear relationships and handle imbalanced datasets effectively. Similarly, XGBoost performed best with 200 estimators, a learning rate of 0.1, and a maximum depth of 10. The model used the 'binary:logistic' objective to handle the classification task efficiently. Table 4 provides a summary of the tested ranges and optimal values for these models.

For the multi-class classification task of predicting match outcomes —win, draw, or lose— hyperparameter tuning was conducted for Logistic Regression, Random Forest, and XGBoost. Logistic Regression was tuned for three key hyperparameters: the maximum number of iterations (max_iter), which was set to 5000; the multi-class strategy (multi_class), where the multinomial approach was chosen and class weights (class_weight), which were balanced to account for the distribution of outcomes. Random Forest achieved its best performance with 200 estimators, a maximum depth of 10, and balanced class weights to address the class distribution. XGBoost performed optimally with 200 estimators, a learning rate of 0.1, and a maximum depth of 7. Subsampling ratios (subsample and colsample_bytree) were set to 0.8 to improve generalization. Additionally, SMOTE was applied across all models to address class

Model	Hyperparameter	Range Tested	Optimal Value
Logistic Regression	max_iter	{1000, 2000}	2000
	class_weight	{'balanced'}	'balanced'
Random Forest	n_estimators	{100, 200, 300}	200
	max_depth	{5, 10, 15}	10
	class_weight	{'balanced'}	'balanced'
XGBoost	n_estimators	{50, 100, 200}	200
	learning_rate	{0.01, 0.05, 0.1}	0.1
	max_depth	{5, 10, 15}	10
	objective	{'binary:logistic'}	'binary:logistic'

Table 4: Hyperparameter Tuning for Binary Classification Task

imbalances in the dataset. Table 5 provides a summary of the tested ranges and optimal values for each model.

Model	Hyperparameter	Range Tested	Optimal Value
Logistic Regression	max_iter	{1000, 2000, 5000}	5000
	multi_class	{'multinomial'}	'multinomial'
	class_weight	{'balanced'}	'balanced'
Random Forest	n_estimators	{100, 200, 300}	200
	max_depth	{5, 10, 15}	10
	class_weight	{'balanced'}	'balanced'
XGBoost	n_estimators	{100, 200, 500}	200
	learning_rate	{0.01, 0.05, 0.1}	0.1
	max_depth	{5, 7, 10}	7
	subsample	{0.5, 0.8, 1.0}	0.8
	colsample_bytree	{0.5, 0.8, 1.0}	0.8

Table 5: Hyperparameter Tuning for Multi-Class Classification Task

4.6 *Evaluation*

To evaluate how well the models performed, different metrics were used for regression and classification tasks. These metrics helped measure the accuracy, reliability, and overall performance of the models.

For the regression models, which predicted the number of goals scored, three metrics were used: Mean Absolute Error (MAE), Mean Squared Error (MSE), and R-squared (R^2). MAE calculates the average size of errors between the predicted and actual values, giving a clear idea of how

accurate the predictions are. MSE, on the other hand, squares these errors, which means larger mistakes are given more weight, making it useful for spotting big differences. R-squared shows how much of the variation in the actual goals is explained by the model, giving an overall sense of how well the model fits the data.

For the classification models, which predicted Over/Under 2.5 Goals and match outcomes (win, draw, or loss), additional metrics were used. Accuracy measured the percentage of predictions that were correct. While it is useful, accuracy alone does not always give the full picture, especially if the dataset is imbalanced. To address this, precision, recall, and F1 score were included. Precision shows how many of the predicted positive results were correct, which helps avoid too many false positives. Recall measures how many actual positives the model was able to find, which is important for reducing false negatives. The F1 score combines precision and recall into a single value, providing a balanced view of the model's performance, especially when both false positives and false negatives matter.

Confusion matrices were also used to better understand the results of the classification models. For binary classification (Over/Under 2.5 Goals), the confusion matrix showed how many predictions were correct (true positives and true negatives) and where the model made errors (false positives and false negatives). This breakdown helped highlight patterns in the model's mistakes. For multi-class classification (win, draw, loss), the confusion matrix was expanded into a 3x3 table to show how often each outcome was correctly predicted and where misclassifications occurred, such as predicting a win as a draw.

By combining these metrics and tools, the evaluation ensured a complete and clear understanding of how well the models worked. This approach helped identify the strengths and weaknesses of each model for both regression and classification tasks. To ensure the models were reliable and performed well on new data, cross-validation and testing on unseen data were used. These methods helped evaluate how well the models could generalize and whether they were overfitting or underfitting.

Cross-validation was performed using a five-fold approach. In this process, the training data was split into five parts, or folds. The model was trained on four folds and validated on the remaining fold. This process was repeated five times, with each fold taking a turn as the validation set. For regression tasks, the Mean Absolute Error (MAE) was used as the evaluation metric during cross-validation, and negative MAE values were converted to positive for easier interpretation. The average of these cross-validation scores provided a reliable estimate of the model's performance, ensuring it was evaluated on different subsets of the training data.

In addition to cross-validation, the models were tested on a separate test set that had not been used during training. This test set represented new, unseen data. Performance metrics, such as Mean Absolute Error (MAE), Mean Squared Error (MSE), R-squared (R^2), accuracy, precision, recall, and F1 score, were calculated for both the training and test sets. Comparing these results helped identify any significant differences in performance. A large gap between training and test metrics would indicate overfitting, while similar results across both sets confirmed the model's ability to generalize to new data.

By combining cross-validation and testing on unseen data, the evaluation ensured that the models were robust and capable of making accurate predictions, even on data they had not encountered before.

This research used Python and several open-source libraries to implement and evaluate the models. Pandas and NumPy were used for handling and processing the dataset. Machine learning tasks, including training and evaluation, were performed with Scikit-learn, while advanced models like XGBoost were implemented using the XGBoost library. For visualizations, such as confusion matrices and feature importance plots, Matplotlib and Seaborn were utilized to create clear and effective graphics. The code was written and executed in a Jupyter Notebook environment using Python 3.11, ensuring easy analysis and reproducibility. All tools used are open-source.

5 RESULTS

In this section, the outcomes of the three predictive tasks—predicting actual goals, over/under 2.5 goals, and match outcomes are presented. Performance metrics such as accuracy, precision, recall, F1 score, Mean Absolute Error (MAE), Mean Squared Error (MSE), R^2, and error pattern visualizations (e.g., confusion matrices for classification tasks) are included to provide a comprehensive view of the models' effectiveness. Where applicable, the results are compared across models (Logistic Regression, Random Forest, and XGBoost) to identify the best-performing approaches.

5.1 *Predicting actual goals (Regression task)*

The regression task aimed to predict the number of actual goals scored by a team in a match using Linear Regression as a baseline model, Random Forest, and XGBoost. Their performances are summarized in Table 6, which reports Mean Absolute Error (MAE), Mean Squared Error (MSE), and R^2 scores. These results are further supported by cross-validation metrics, residual analysis, and feature importance.

Model	MAE	MSE	R^2
Baseline (Linear Regression)	0.8989	1.2541	0.0942
Random Forest	0.8771	1.2065	0.1286
XGBoost	0.8757	1.2024	0.1315

Table 6: Regression Model Performance Metrics for Predicting Actual Goals

Metric	Value
Mean MAE	0.8733
Standard Deviation of MAE	0.0101
MAE Scores for Each Fold	0.8699, 0.8921, 0.8699, 0.8616, 0.8729

Table 7: Cross-Validation Metrics Summary for XGBoost Model (Regression task)

From Table 6, XGBoost emerged as the best-performing model, with the lowest MAE (0.8757) and MSE (1.2024), along with the highest R^2 (0.1315). Random Forest followed closely, with an MAE of 0.8771 and an R^2 of 0.1286, while Linear Regression, the baseline model, demonstrated weaker performance with an MAE of 0.8989 and an R^2 of 0.0942. These results indicate that XGBoost outperforms the other models in capturing the relationship between input features and actual goals scored.

Since XGBoost demonstrated slightly better performance compared to the other models, we conducted a more detailed analysis of its predictions and errors to better understand its strengths and limitations. The cross-validation results further validated XGBoost's robustness, as shown in Table 7. The model achieved a mean MAE of 0.8733 across folds, with a low standard deviation of 0.0101, demonstrating consistent predictive performance across five subsets of data. The cross-validated mean MSE was 1.2046, and the mean R^2 was 0.1286, closely aligning with the test set performance. These results enhance confidence in the model's reliability and its ability to generalize effectively.

Residual analysis revealed systematic trends in the errors, particularly for high-scoring matches. As shown in Figure 3, larger residuals were observed in games with high goal outcomes. Additionally, Figure 4 provides an error breakdown by goal ranges, highlighting that while the model performed well in predicting games with one to two goals, it struggled with higher goal ranges, especially in matches involving three or more goals and in games with no goals scored.

Figure 5, highlights the discrepancies between the predicted and actual goals scored in football matches. While some points align with the diagonal line, suggesting accurate predictions, there are significant inconsistencies across different goal ranges. For games with no goals scored (actual goals = 0), the model frequently overestimates, predicting higher goal counts

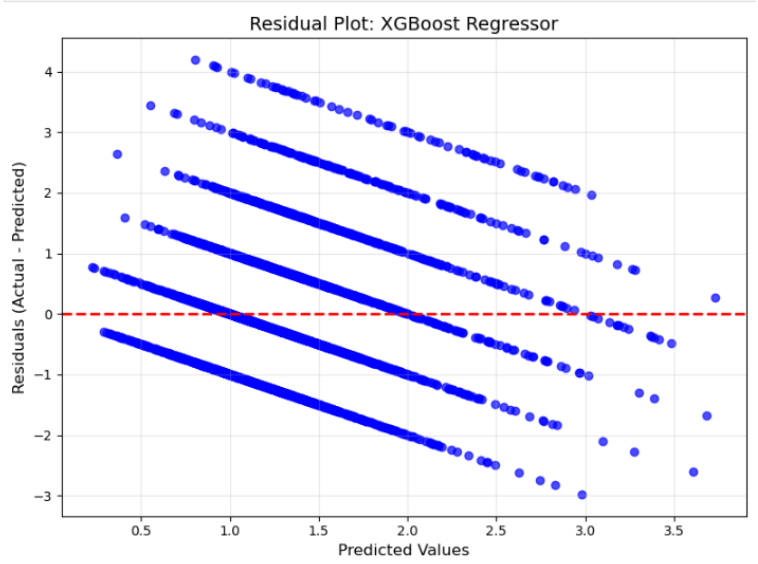

Figure 3: XGBoost Model, Residual Plot

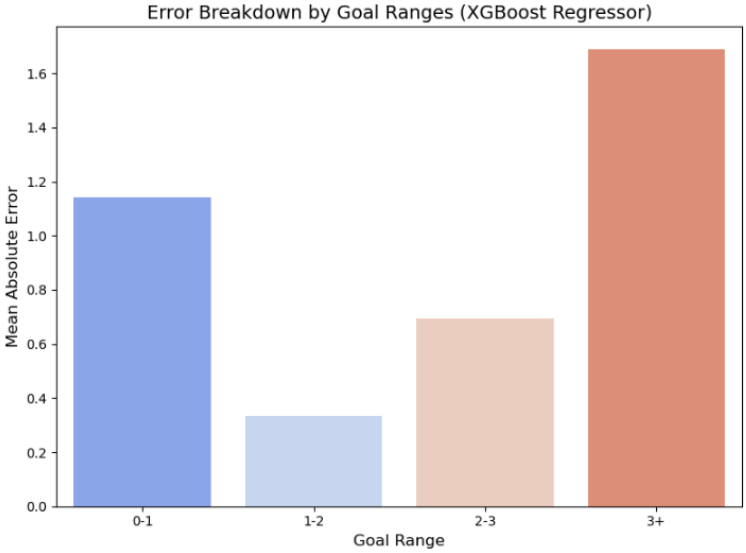

Figure 4: Error by Goal Ranges

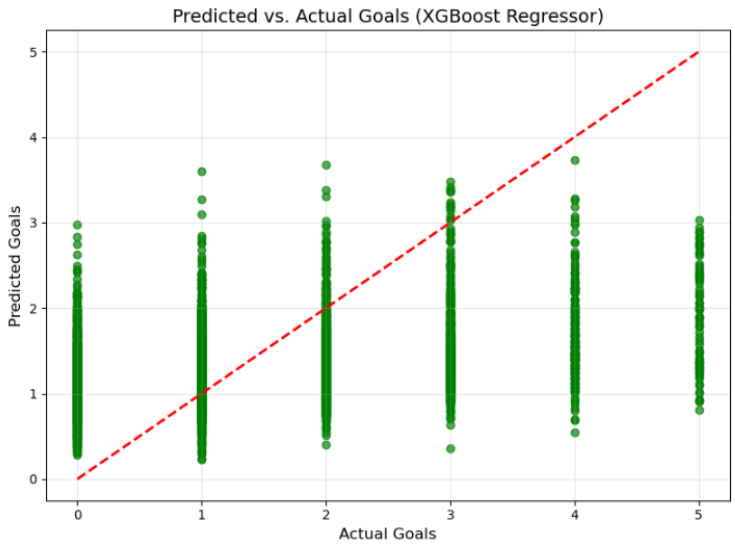

Figure 5: Predicted vs Actual Goals (XGBoost Model)

than observed. Similarly, for games with three or more goals, the model systematically underestimates, predicting lower goal counts than observed. This indicates that the model struggles to capture extreme cases of both low- and high-scoring games, showing that the model does not perform well in these cases.

Feature importance analysis, as shown in Figure 6, reveals that the feature "deep_squared" is the most significant, with a score of 0.196. This suggests that actions deeper in the opponent's area play a crucial role in predicting actual goals. Other influential features include "deep" (0.121) and "h_a_h" (0.068), highlighting the importance of match location (home or away). Conversely, features such as "league_La_liga" (0.015) and "league_Ligue_1" (0.016) had minimal importance, indicating that league-specific effects contribute less to the predictions.

Finally the SHAP summary plot (Figure 7) shows the key features that impact the XGBoost model's predictions. The feature 'deep' has the biggest influence, highlighting how actions near the opponent's goal are crucial for scoring. 'h_a_h' (home or away games) shows that playing at home often gives teams an advantage in scoring. 'deep_x_ppda_intensity' reflects how pressure near the opponent's goal, combined with territorial dominance, plays an important role in creating scoring opportunities. oppda_att' (opponent pressing actions in attack) plays an important role by showing how the opponent's pressure can disrupt a team's ability to create scoring chances. When opponents press aggressively, it becomes harder for teams to build up their play and score goals, making this feature highly relevant. In contrast, features like 'ppda_efficiency' and 'ppda_coef' show

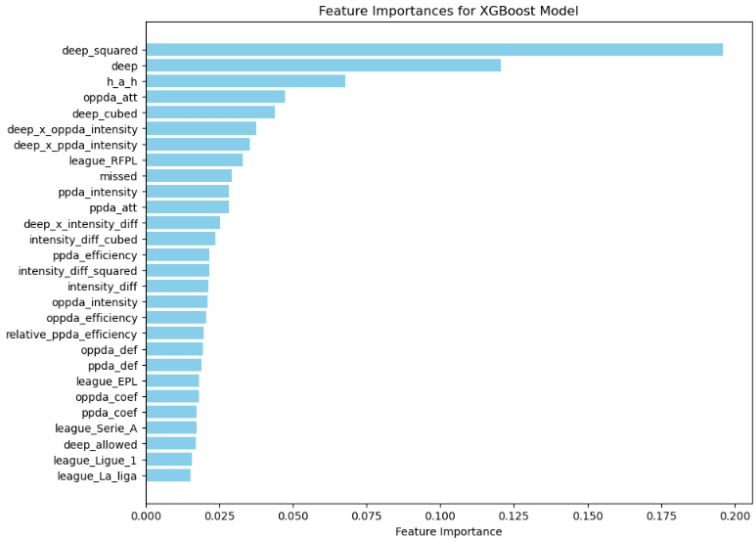

Figure 6: Feature Importance (Regression tasl - XGBoost model)

consistent but lower contributions to predictions, indicating their limited relevance to scoring outcomes."

Table 8 compares the performance of the optimized XGBoost regression model across different scenarios involving feature removal. In Scenario 1, where tactical metrics ('deep,' 'deep_allowed,' 'ppda,' 'oppda,' and their interactions) were removed, the model's performance slightly declined, with the MAE increasing to 0.9041 and the R^2 dropping to 0.0796. In Scenario 2, where both the tactical metrics and the 'h_a' (home or away) feature were removed, performance further worsened, with the MAE rising to 0.9129 and the R^2 falling to 0.0685. In contrast, Scenario 3, which involved removing only the 'h_a' feature, had a smaller impact, with the model achieving an MAE of 0.8798 and an R^2 of 0.1231.

Feature Removal Scenario	MAE	R^2
`Main Model`	0.8757	0.1315
`Scenario 1`	0.9041	0.0796
`Scenario 2`	0.9129	0.0685
`Scenario 3`	0.8798	0.1231

Table 8: Impact of Feature Removal on Regression Model Performance

5.2 *Predicting under/over 2.5 goals (Binary classification task)*

In this subsection, the results of three machine learning models—Logistic Regression, Random Forest, and XGBoost—are presented for the task of

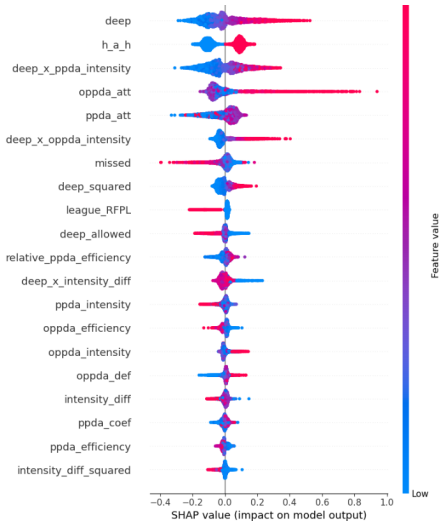

Figure 7: SHAP Summary Plot (Regression task)

Model	Accuracy	Macro Avg F1-Score	Weighted Avg F1-Score
Logistic Regression	0.6744	0.57	0.72
Random Forest	0.7457	0.60	0.76
XGBoost	0.8409	0.54	0.79

Table 9: Overall Performance Metrics (Binary classification task)

predicting whether a match will result in over or under 2.5 goals. Table 9 provides an overview of their overall performance metrics, including accuracy, macro-average F1-scores, and weighted F1-scores, while Table 10 details class-specific metrics, including precision, recall, and F1-scores for each class.

Among the models, XGBoost emerged as the best-performing algorithm, achieving an overall accuracy of 0.8409, as seen in Table 9. It showed strong performance in predicting the majority class (Under 2.5 goals), with a precision of 0.85 and a recall of 0.98, leading to a high weighted F1-score of 0.79 (Table 9). However, XGBoost struggled with the minority class (Over 2.5 goals), achieving a precision of only 0.48 and a recall of 0.10 (Table 10), reflecting challenges in correctly identifying matches with more than 2.5 goals. The confusion matrix for this binary classification task using XGBoost model, illustrating these strengths and limitations, is presented in Appendix (page 41).

Random Forest also performed well, achieving an overall accuracy of 0.7457 and a weighted F1-score of 0.76 (Table 9). For the majority class, it achieved a precision of 0.89 and a recall of 0.80, translating to a strong

Model	Class	Precision	Recall	F1-Score
Logistic Regression	Under	0.90	0.69	0.78
	Over	0.26	0.57	0.36
Random Forest	Under	0.89	0.80	0.84
	Over	0.30	0.44	0.36
XGBoost	Under	0.85	0.98	0.91
	Over	0.48	0.10	0.16

Table 10: Class-Specific Metrics (Binary classification task)

Metric	Value
Mean Accuracy	0.8450
Standard Deviation of Accuracy	0.0021
Accuracy Scores for Each Fold	0.8449, 0.8487, 0.8433, 0.8425, 0.8451

Table 11: Cross-Validation Metrics Summary for XGBoost Model (Binary classification task)

F1-score of 0.84. However, like XGBoost, its performance dropped for the minority class, with a precision of 0.30 and a recall of 0.44, resulting in an F1-score of 0.36 (Table 10).

Logistic Regression, the baseline model, performed the weakest overall, with an accuracy of 0.6812 and a weighted F1-score of 0.72 (Table 9). While it maintained a high precision of 0.90 for the majority class, its recall was only 0.70, leading to an F1-score of 0.79. For the minority class, the model's metrics were lower, with a precision of 0.26 and a recall of 0.57, resulting in an F1-score of 0.36 (Table 10). These results highlight that while XGBoost outperformed the other models in terms of overall accuracy and majority class performance, the challenge of predicting the minority class remains across all models.

The cross-validation results for XGBoost, summarized in Table 11, further validated its reliability. The model demonstrated consistent performance with a mean accuracy of 0.8409 and a low standard deviation of 0.0021. This consistency across folds indicates that the model generalizes well to unseen data and reinforces its suitability for this classification task.

The feature importance analysis based on the XGBoost model, as depicted in Figure 8, highlights the key predictors for this task. The feature "deep" had the highest importance score (0.094), emphasizing its strong predictive contribution. This likely reflects the importance of attacking actions deep into the opponent's territory in determining the number of goals. Other influential features include "oppda_att" with a score of 0.056 and "league_RFPL" with a score of 0.053, demonstrating the significance of both tactical and contextual aspects in model performance.

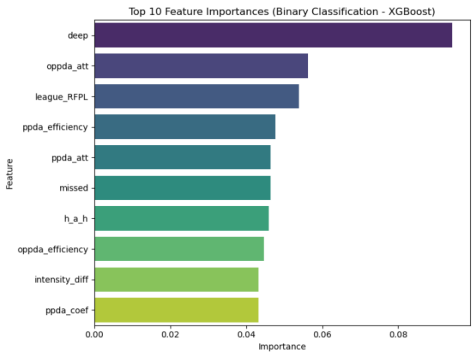

Figure 8: Feature importance - XGBoost model (Binary classification task)

The SHAP summary plot (Figure 9) shows which tactics impact team performance the most. The feature 'deep,' representing passes near the opponent's goal, has the strongest influence. This suggests teams should focus on controlling these areas to create more scoring opportunities through precise passing and positioning. 'h_a_h' (home or away) highlights the advantage of playing at home, where teams can adopt more aggressive strategies like high pressing and attacking with more players. 'ppda_att' (attacking pressing actions) shows the importance of pressing high to win the ball in dangerous areas and disrupt the opponent's defense. 'oppda_att' (opponent pressing actions) emphasizes the need to handle pressure effectively, using quick passes and movement to bypass pressing opponents. 'ppda_intensity' (overall pressing intensity) highlights the value of consistent pressure to force turnovers and create chances. To improve performance, teams should focus on dominating critical areas near the goal, make the most of home games, and use effective pressing strategies to control the game.

5.3 *Predicting team performance — win, draw, or lose (multi-class classification task)*

This section presents the results of Logistic Regression, Random Forest, and XGBoost for predicting team performance in a match (win, draw, or lose). The evaluation metrics include accuracy, precision, recall, and F1-scores, along with an analysis of important features for each model.

To establish a baseline, a simple model predicting only the majority class (win) was implemented. As shown in Table 12, this baseline model achieved an accuracy of 0.37 by correctly predicting all instances of the win class, but it failed to classify any instances of the draw or lose classes. This limitation is further illustrated in the confusion matrix (Figure 10), where no predictions were made for the other two classes. Consequently,

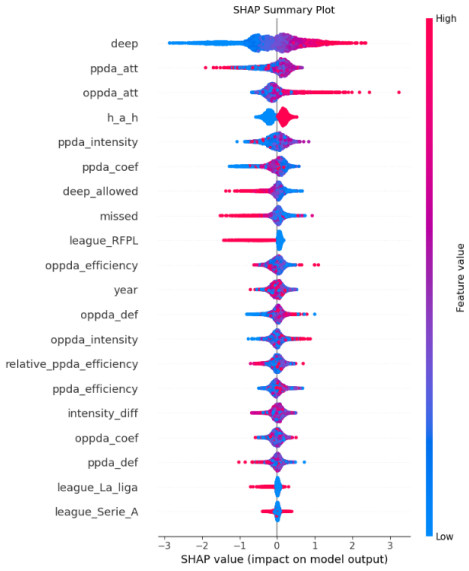

Figure 9: SHAP Summary Plot (Binary classification task)

the F1-score for the win class was 0.54, while the draw and lose classes had undefined metrics due to the absence of predictions (Table 13).

Building on this baseline, Logistic Regression achieved an accuracy of 0.45 (Table 12), demonstrating modest improvements over the baseline. This model showed balanced performance for the win and lose classes, with F1-scores of 0.48 for both, as shown in Table 13. However, it struggled to predict draws, achieving an F1-score of only 0.33. The confusion matrix (Figure 11) highlights that Logistic Regression frequently misclassified draws as either wins or losses, reflecting its challenges in handling this class.

Random Forest demonstrated better overall performance than Logistic Regression, achieving an accuracy of 0.51 (Table 12). The model exhibited consistent performance across all classes, with F1-scores of 0.51 for wins, 0.51 for draws, and 0.50 for losses, as detailed in Table 13. This balance is also evident in the confusion matrix (Figure 12), which shows a relatively even distribution of correct predictions across the three classes. Random Forest's ability to improve classification for the draw class, in particular, highlights its advantage over simpler models.

Finally, XGBoost closely followed Random Forest in overall performance, achieving an accuracy of 0.50 (Table 12). It demonstrated comparable F1-scores across all classes, with values of 0.49 for losses, 0.50 for draws, and 0.51 for wins (Table 13). The confusion matrix (Figure 13) reveals a performance pattern similar to that of Random Forest, with a balanced distribution of predictions but occasional misclassifications, particularly

Model	Accuracy	Macro Avg F1-Score	Weighted Avg F1-Score
Baseline	0.37	0.18	0.20
Logistic Regression	0.45	0.43	0.44
Random Forest	0.51	0.51	0.51
XGBoost	0.50	0.50	0.50

Table 12: Overall Performance Metrics (Multi-class classification task)

Model	Class	Precision	Recall	F1-Score
Baseline	Lose	0.00	0.00	0.00
	Draw	0.00	0.00	0.00
	Win	0.37	1.00	0.54
Logistic Regression	Lose	0.50	0.47	0.48
	Draw	0.30	0.35	0.33
	Win	0.50	0.47	0.48
Random Forest	Lose	0.50	0.51	0.50
	Draw	0.52	0.51	0.51
	Win	0.50	0.50	0.50
XGBoost	Lose	0.49	0.49	0.49
	Draw	0.50	0.50	0.50
	Win	0.50	0.51	0.51

Table 13: Class-Specific Metrics (Multi-class classification task)

between the win and draw classes. This balance underscores XGBoost's ability to identify patterns across all outcomes effectively.

The feature importance analysis for all models provides additional insights into the factors driving their predictions. According to Figure 14, XGBoost identified "deep_ratio," "attack_defense_ratio," and "ppda_att" as the most influential features, emphasizing the importance of offensive depth and the balance between attack and defense. Similarly, Random Forest highlighted "oppda_att," "ppda_att," and "pressing_effectiveness" as key predictors, illustrating the significance of attacking intensity and pressing actions in match outcomes. In contrast, Logistic Regression focused on "deep_allowed," "deep," and "oppda_coef," suggesting that defensive metrics and positional depth play a critical role in its predictions.

In addition, this approach was selected to assess the model's performance and patterns across different league combinations, focusing on the best and worst-performing cases. According to Figure 15, the best-performing pair is La Liga $\rightarrow$ Serie A, while the worst-performing pair is EPL $\rightarrow$ RFPL. For the best-performing pair, where the model was trained on La Liga data and tested on Serie A, an accuracy of 0.4763 was achieved as shown Table 14. The precision and recall metrics indicate that the model

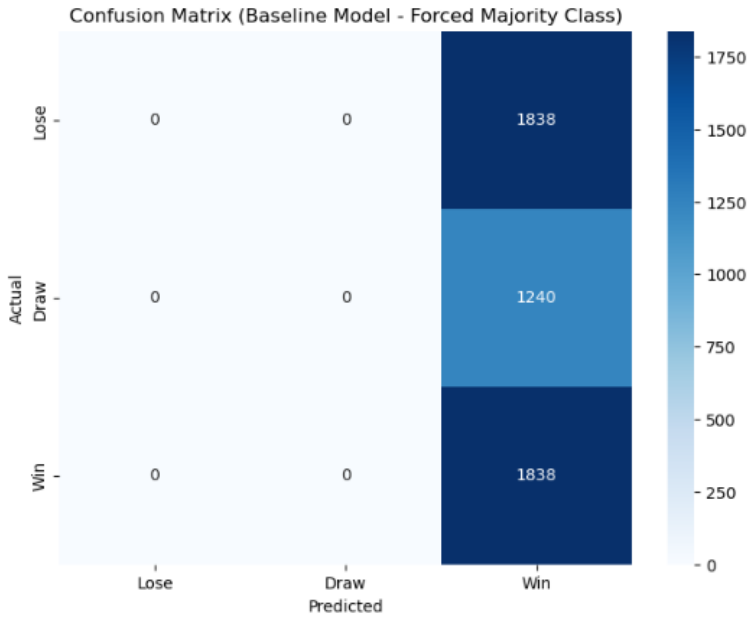

Figure 10: Confusion matrix (Baseline Model)

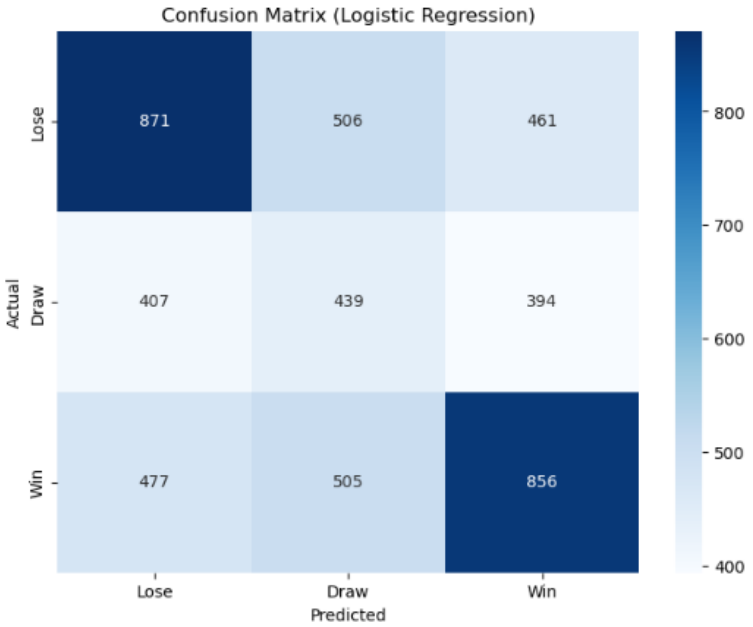

Figure 11: Confusion matrix (Logistic Regression Model)

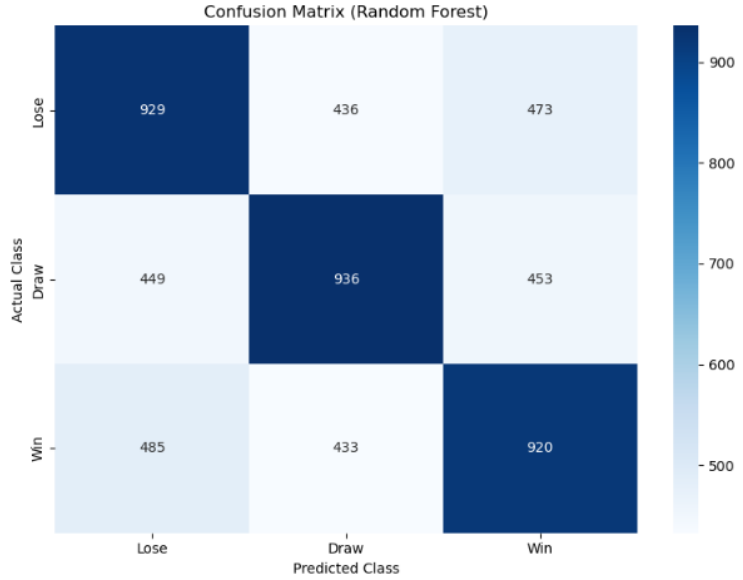

Figure 12: Confussion matrix (Random Forest Model)

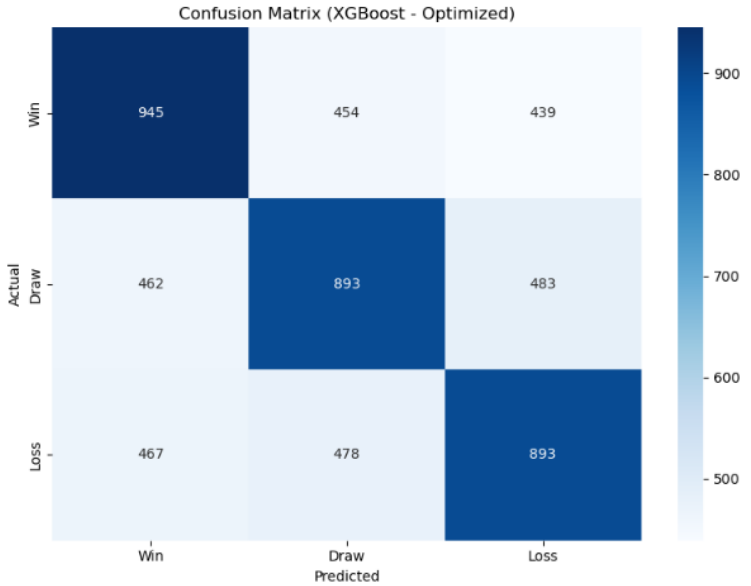

Figure 13: Confussion matrix (XGBoost Model)

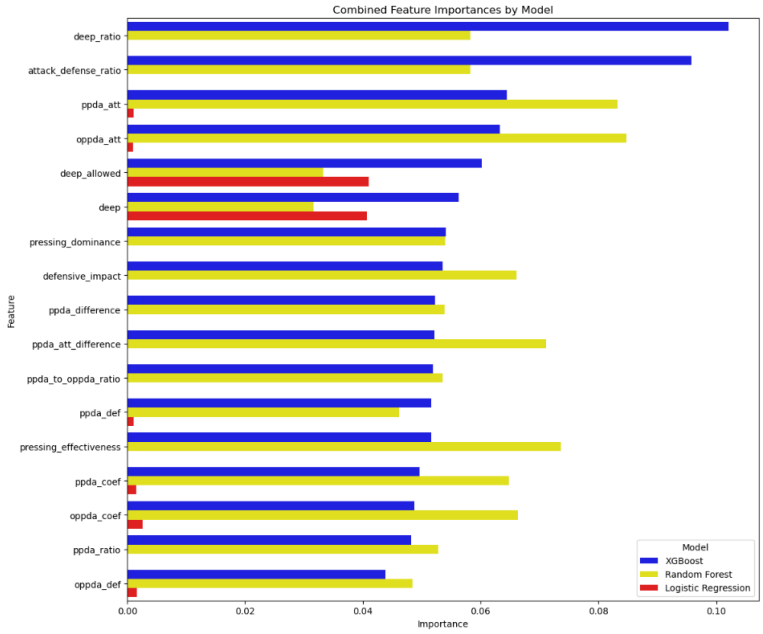

Figure 14: Combined Feature Importance

performed relatively better in predicting wins and losses compared to draws. Specifically, for wins, the model achieved a recall of 0.58 and a precision of 0.49, while for losses, it attained a recall of 0.58 and a precision of 0.51. However, for draws, the model struggled, with a low recall of 0.17 and precision of 0.32, as shown in Table 15.

In contrast, the worst-performing pair involved training on EPL data and testing on RFPL data, where the model got an accuracy of 0.4053 Table 14. The precision and recall results here also show challenges in classifying draws, with a precision of 0.29 and recall of 0.35. Losses had slightly better recall (0.44) and precision (0.47), while wins exhibited a precision of 0.46 and recall of 0.42 Table 15. The confusion matrices illustrating the model's performance for the best-performing league pair (La Liga → Serie A) and the worst-performing league pair (EPL → RFPL) are presented in Appendix D (page 44).

As illustrated in Figure 16, the feature importance analysis provides insights into the key features of the model's predictions across the best and worst-performing league pairs. For the best-performing pair (La Liga → Serie A), the most significant contributors to the predictions were 'ppda_att,' 'oppda_att,' and 'pressing_effectiveness'. Conversely, for the worst-performing pair (EPL → RFPL), features such as 'oppda_att,' 'ppda_att,' and 'attack_defense_ratio' emerged as the most influential, suggesting a continued dependency on attacking intensity and the balance between offense and defense.

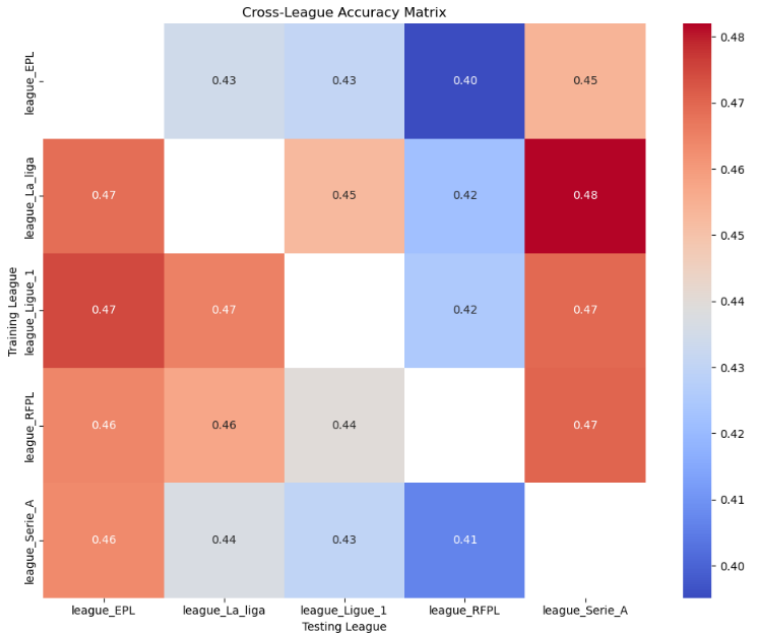

Figure 15: Cross-league accuracy matrix

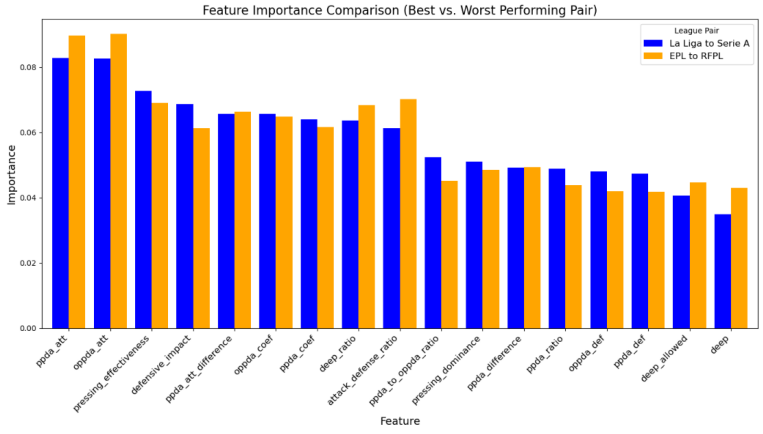

Figure 16: Feature Importance Comparison Across Pairs of Leagues

Metric	La Liga → Serie A (Best Pair)	EPL → RFPL (Worst Pair)
Accuracy	0.4763	0.4053
Macro Avg Precision	0.44	0.41
Macro Avg Recall	0.44	0.40
Macro Avg F1-Score	0.43	0.40
Weighted Avg Precision	0.46	0.42
Weighted Avg Recall	0.48	0.41
Weighted Avg F1-Score	0.46	0.41

Table 14: Overall Metrics for Best and Worst Performing League Pairs

Metric	La Liga → Serie A (Best Pair)			EPL → RFPL (Worst Pair)		
	Lose (-1)	Draw (0)	Win (1)	Lose (-1)	Draw (0)	Win (1)
Precision	0.51	0.32	0.49	0.47	0.29	0.46
Recall	0.58	0.17	0.58	0.44	0.35	0.42
F1-Score	0.54	0.22	0.53	0.45	0.32	0.44
Support	1705	1140	1682	1045	790	1027

Table 15: Class-Specific Metrics for Best and Worst Performing League Pairs

Continuing with the feature importance analysis between the two pairs, the research visualized the distributions of key features across leagues Figure 17 and Figure 18, providing additional clarity for comparing the tactical styles of these pairs. For La Liga and Serie A, distributions of features such as 'deep,' 'deep_allowed,' 'ppda_att,' and 'oppda_att' were relatively aligned, suggesting less variability in tactical styles, which might have contributed to better generalization. In contrast, EPL and RFPL feature distributions exhibited more noticeable variations, particularly in 'ppda_att' and 'oppda_att,' pointing to tactical differences that likely reduced the model's performance in this pair.

6 DISCUSSION

This study aimed to evaluate the performance of machine learning models in predicting team performance across three different tasks: predicting the number of goals scored by the team (regression), determining if the team scored over/under 2.5 goals (binary classification), and predicting if a team win, draw, or loss (multi-class classification). The results highlight the strengths and limitations of the models and reveal important tactical variables that influence the predictions.

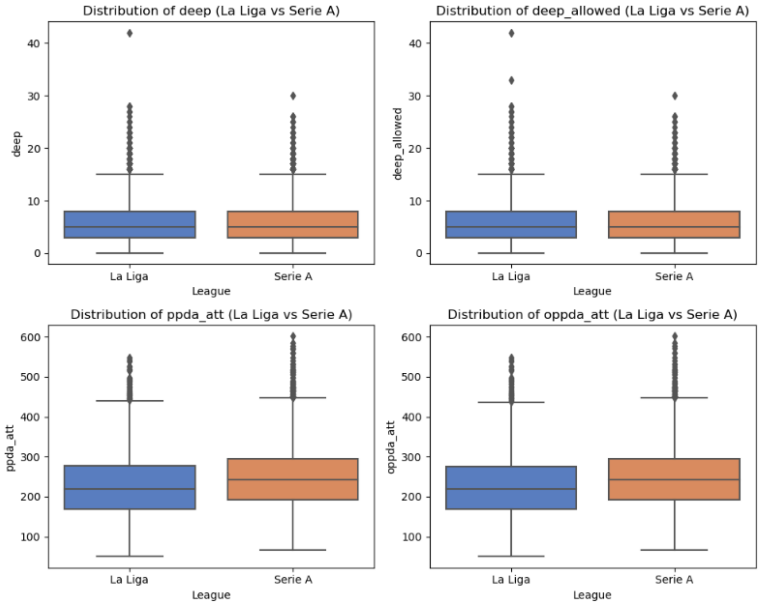

Figure 17: Feature distributions (La Liga vs Serie A)

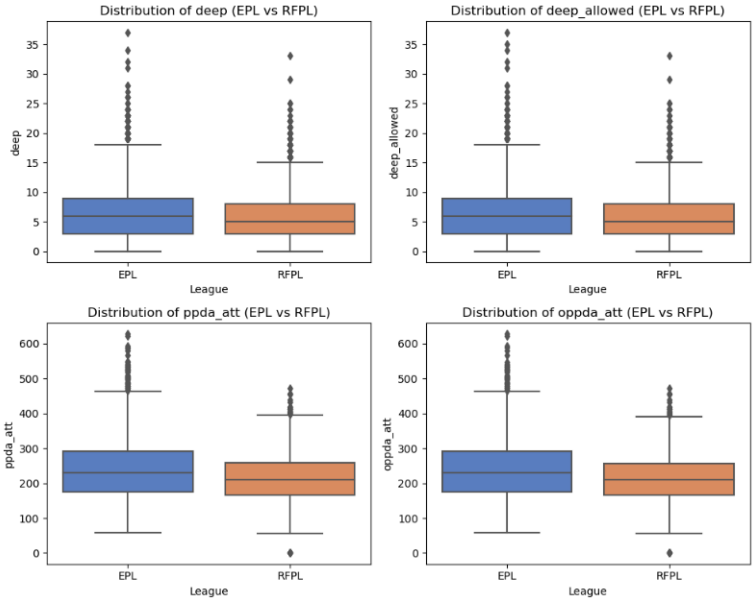

Figure 18: Feature distributions (EPL vs RFPL)

Main RQ: *To what extent can machine learning models accurately predict the performance of football teams across six European leagues using tactical features?*

Combining the results from different approaches in this research, Machine learning models showed different levels of success in predicting football team performance across six European leagues using tactical features. Advanced models like Random Forest and XGBoost performed well in general classification tasks, such as predicting outcomes or performance. However, they faced challenges in accurately predicting the exact number of goals scored, highlighting the difficulty of using only tactical features for detailed predictions.

1st RQ: *Which machine learning model performs best for predicting team performance in terms of goals scored and match outcomes?*

For predicting goals scored by a team in a match, XGBoost emerged as the best model, achieving the lowest Mean Absolute Error (MAE) of 0.8757 and an R^2 score of 0.1315, demonstrating its ability to capture patterns in tactical features effectively. In contrast, for predicting match outcomes (win, lose, draw), Random Forest performed slightly better than Logistic Regression and XGBoost, achieving an accuracy of 0.51 and consistent F1-scores across all classes. For the binary classification task of predicting over/under 2.5 goals, XGBoost was again the top performer, achieving an accuracy of 0.8409, highlighting its capacity to differentiate high and low goal-scoring matches. Overall, XGBoost and Random Forest showed robust performance, with their suitability depending on the specific predictive task.

Comparing these findings with the related literature, this study aligns with and extends findings in the literature, with some differences. For predicting goals scored, (Stübinger et al., 2019) found that Random Forest gave the best results with the lowest RMSE and MAD. In comparison, this research showed that XGBoost performed best, achieving the lowest MAE of 0.8757. For predicting match outcomes (win, lose, draw), (Baboota & Kaur, 2019) reported gradient boosting as the best model, followed by Random Forest, while (Eryarsoy & Delen, 2019) achieved 74% accuracy using Random Forest. In comparison, this study observed Random Forest performing slightly better than XGBoost and Logistic Regression, achieving an accuracy of 51%.

2st RQ: *How do feature selection and tactical variables, such as pressing intensity and home/away status, impact the accuracy of these predictions?*

To answer this research question, this study uses the results from the regression approach (predicting actual goals).

The results from this research demonstrate how tactical factors, like actions near the opponent's goal and pressing intensity, are important for predicting team performance. They show how teams create and defend

scoring chances. Actions close to the opponent's goal, such as passes completed within 20 yards, are very influential because they represent a team's ability to break through defenses and create opportunities to score. Teams that can consistently operate in these dangerous areas are much more likely to score. This also shows that controlling key areas near the goal is essential in football, as even small improvements in this can lead to bigger scoring chances.

When tactical factors like pressing intensity are removed, the model's performance drops slightly because these features capture unique match dynamics. Pressing intensity shows how aggressively a team pressures the opponent to win the ball back in dangerous areas. Teams that press effectively can force mistakes and create quick chances to score, without needing long build-up play. On the other hand, removing whether a team is playing at home or away has a smaller impact. This suggests that while home advantage matters, it doesn't directly change tactical decisions as much as other factors like pressing or positioning.

The SHAP analysis gives more insight by showing that actions near the opponent's goal and whether a team plays at home or away have a strong influence on predictions in different match situations. This confirms that tactical factors, like how teams attack and position themselves, are important for improving accuracy. However, the model still struggles with extreme cases, like very high-scoring matches or unusual game situations. Tactical factors are important for making better predictions but may need to be combined with other information to handle these outliers more effectively.

3st RQ: *How well do the developed models generalize across different leagues?*

The results of the multi-class classification task show that the generalization of the models across leagues varies significantly based on the tactical styles of the leagues involved. The best-performing pair, La Liga $\rightarrow$ Serie A, achieved an accuracy of 47.63%, while the worst-performing pair, EPL $\rightarrow$ RFPL, had an accuracy of 40.53%. These differences suggest that the similarity or disparity in playing styles between leagues heavily influences the model's ability to generalize.

The better generalization between La Liga and Serie A can be attributed to their relatively aligned tactical styles. Both leagues exhibit similar distributions of key features, such as actions near the goal (deep, deep_allowed) and pressing metrics (ppda_att, oppda_att). This alignment likely reflects comparable approaches to attacking and defensive strategies, making it easier for the model to transfer patterns learned in one league to the other. For example, the emphasis on controlled build-up play and consistent pressing intensity in both leagues creates a more predictable tactical environment, enabling the model to generalize better.

In contrast, the poor generalization between the EPL and RFPL reflects greater variability in tactical styles. The feature distributions for pressing (ppda_att) and attacking intensity (deep) showed notable differences between these leagues. For instance, RFPL teams exhibited lower median values for pressing and attacking metrics, indicating a less aggressive and more defensive playing style compared to the EPL. This mismatch makes it challenging for the model to recognize similar patterns across the leagues, resulting in lower accuracy. Moreover, the EPL's higher pace and physicality, contrasted with RFPL's more cautious and defensive approach. This makes it even harder for the model to adapt, as the tactical styles in these leagues are different.

This shows that a model's ability to generalize depends not just on its quality but also on how similar the leagues are in their playing styles. When leagues share tactical patterns, the model performs better. But when leagues are very different, the model struggles to adapt. These findings highlight the importance of considering tactical differences when designing models for cross-league predictions. More broadly, this reflects how football tactics are shaped by cultural and strategic factors unique to each league. The results suggest that football is highly context-dependent, and models need to account for this.

6.1 *Scientific and societal impact*

This study contributes to football analytics by showing how machine learning models can predict team performance across tasks and leagues using tactical features. It highlights the influence of variables like pressing intensity and home/away status on match outcomes and goals scored, adding to current knowledge about the role of tactics in football. The findings also provide insights into how differences in playing styles across leagues affect model performance, offering a framework for cross-league analysis. These results can help coaches refine strategies, improve training, and better prepare for international competitions by understanding opponents from different leagues and countries.

6.2 *Limitations and Future Directions*

While most of the models showed good results, their performance was limited because they relied only on tactical features. This made it harder to predict the exact goals scored and reduced their ability to generalize across leagues with different playing styles. Future studies should include more features like shooting metrics (e.g., shot accuracy, shot location, and shot quality), and player-specific data to improve predictions. Adding factors

like player psychology, and real-time match conditions could also help make the models more accurate and reliable.

7 CONCLUSION

This study highlights the importance of key tactical factors in football, such as offensive actions near the opponent's goal ("deep"), pressing intensity ("ppda_att"), in predicting goals and match outcomes. To score more, teams should focus on pushing the ball deeper into the opponent's area, using intense pressing to disrupt their play, and taking advantage of playing at home to create more opportunities. While models like XGBoost and Random Forest use these factors effectively, predicting extreme results and adapting to different league styles remain challenging. Overall, this research provides valuable insights into how teams can improve their tactics and performance analysis.

REFERENCES

Baboota, R., & Kaur, H. (2019). Predictive analysis and modelling football results using machine learning approach for english premier league. *International Journal of Forecasting*, *35*(2), 741–755.

Bai, Z., & Bai, X. (2021). Sports big data: Management, analysis, applications, and challenges. *Complexity*, *2021*(1), 6676297.

Bauer, P., & Anzer, G. (2021). Data-driven detection of counterpressing in professional football: A supervised machine learning task based on synchronized positional and event data with expert-based feature extraction. *Data Mining and Knowledge Discovery*, *35*(5), 2009–2049.

Beal, R., Norman, T. J., & Ramchurn, S. D. (2019). Artificial intelligence for team sports: A survey. *The Knowledge Engineering Review*, *34*, e28. https://doi.org/10.1017/S0269888919000225

Brefeld, U., Davis, J., Van Haaren, J., & Zimmermann, A. (2018). Machine learning and data mining for sports analytics. *Cham: Springer*.

Choi, B. S., Foo, L. K., & Chua, S.-L. (2023). Predicting football match outcomes with machine learning approaches. *MENDEL*, *29*(2), 229–236.

Cotta, L., de Melo, P., Benevenuto, F., & Loureiro, A. (2016). Using fifa soccer video game data for soccer analytics. *Workshop on large scale sports analytics*.

Eryarsoy, E., & Delen, D. (2019). Predicting the outcome of a football game: A comparative analysis of single and ensemble analytics methods.

Forcher, L., Forcher, L., Altmann, S., Jekauc, D., & Kempe, M. (2024). The keys of pressing to gain the ball–characteristics of defensive

pressure in elite soccer using tracking data. *Science and Medicine in Football*, *8*(2), 161–169.

García-Aliaga, A., Marquina, M., Coteron, J., Rodríguez-González, A., & Luengo-Sanchez, S. (2021). In-game behaviour analysis of football players using machine learning techniques based on player statistics. *International Journal of Sports Science & Coaching*, *16*(1), 148–157.

Goes, F., Kempe, M., Van Norel, J., & Lemmink, K. (2021). Modelling team performance in soccer using tactical features derived from position tracking data. *IMA Journal of Management Mathematics*, *32*(4), 519–533.

Green, S. (2012). Assessing the performance of premier league goalscorers. *OptaPro Blog*.

Harris, C. R., Millman, K. J., Van Der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., et al. (2020). Array programming with numpy. *Nature*, *585*(7825), 357–362.

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing in science & engineering*, *9*(03), 90–95.

Manai, E., Mejri, M., & Fattahi, J. (2023). Impact of feature encoding on malware classification explainability. *2023 15th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, 1–6.

McKinney, W., et al. (2010). Data structures for statistical computing in python. *SciPy*, *445*(1), 51–56.

Pantzalis, V. C., & Tjortjis, C. (2020). Sports analytics for football league table and player performance prediction. *2020 11th International Conference on Information, Intelligence, Systems and Applications (IISA*, 1–8.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, *12*, 2825–2830.

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

Richter, C., O'Reilly, M., & Delahunt, E. (2024). Machine learning in sports science: Challenges and opportunities. *Sports Biomechanics*, *23*(8), 961–967.

Rydning, D. R.-J. G.-J., Reinsel, J., & Gantz, J. (2018). The digitization of the world from edge to core. *Framingham: International Data Corporation*, *16*, 1–28.

Spearman, W. (2018). Beyond expected goals. *Proceedings of the 12th MIT sloan sports analytics conference*, 1–17.

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.

Vinutha, H., Poornima, B., & Sagar, B. (2018). Detection of outliers using interquartile range technique from intrusion dataset. *Information and decision sciences: Proceedings of the 6th international conference on ficta*, 511–518.

Waskom, M. L. (2021). Seaborn: Statistical data visualization. *Journal of Open Source Software*, *6*(60), 3021.

Wisdom, C., & Javed, A. (2023). Machine learning for data analytics in football: Quantifying performance and enhancing strategic decision-making. *Available at SSRN 4558733*.

Yang, L., & Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing*, *415*, 295–316.

Zhou, W., Yu, G., You, S., & Wang, Z. (2023). An improved passing network for evaluating football team performance. *Applied Sciences*, *13*(2), 845.

APPENDIX A

This appendix lists the software and tools used in this research. The main programming language is Python 3.11 (64-bit).

Library	**Reference**
Pandas	(McKinney et al., 2010)
Numpy	(Harris et al., 2020)
Matplotlib	(Hunter, 2007)
Seaborn	(Waskom, 2021)
Scikit-learn	(Pedregosa et al., 2011)

Table 16: Libraries and their references

Additionally, tools such as Thesaurus and AI-based applications like ChatGPT, Grammarly, and Perplexity were used to check spelling and grammar errors and to assist in improving the overall flow and clarity of the research.

APPENDIX B

This appedix presents visualizations showing the relationships between features and their relevance to predicting team performance. The figures

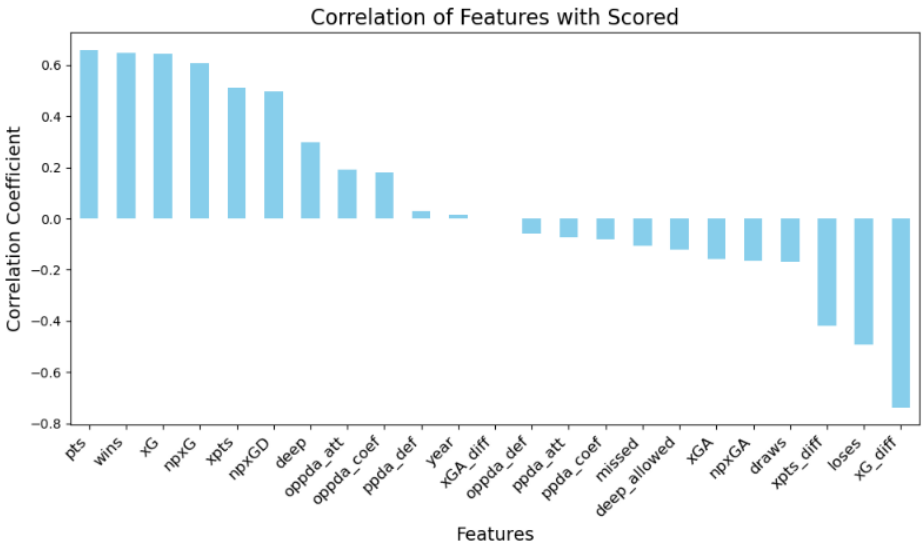

Figure 19: Correlation of Features Before Removing Goal-Related Features

highlight the impact of feature engineering and preprocessing on the dataset.

Figure 19 displays the correlation coefficients of all features, including goal-related variables, with the scored target variable. It highlights the dominance of goal-related metrics in predicting outcomes.

Figure 20 shows the correlation of features with the scored variable after removing goal-related metrics and incorporating engineered features. The new features emphasize tactical and positional metrics relevant to predictions.

Figure 21 shows pairwise correlations among the final features used in the project, highlighting relationships between tactical, engineered, and categorical variables. Only correlations above 0.2 are displayed to focus on significant connections and reduce redundancy.

APPENDIX C

This appendix presents the confusion matrix generated for the binary classification model (XGBoost) used in this project. The matrix provides a detailed breakdown of the model's predictions, highlighting the true positives, true negatives, false positives, and false negatives. It offers insights into the model's strengths and limitations in distinguishing between the "Over 2.5" and "Under 2.5" goal categories, as shown in Figure 22.

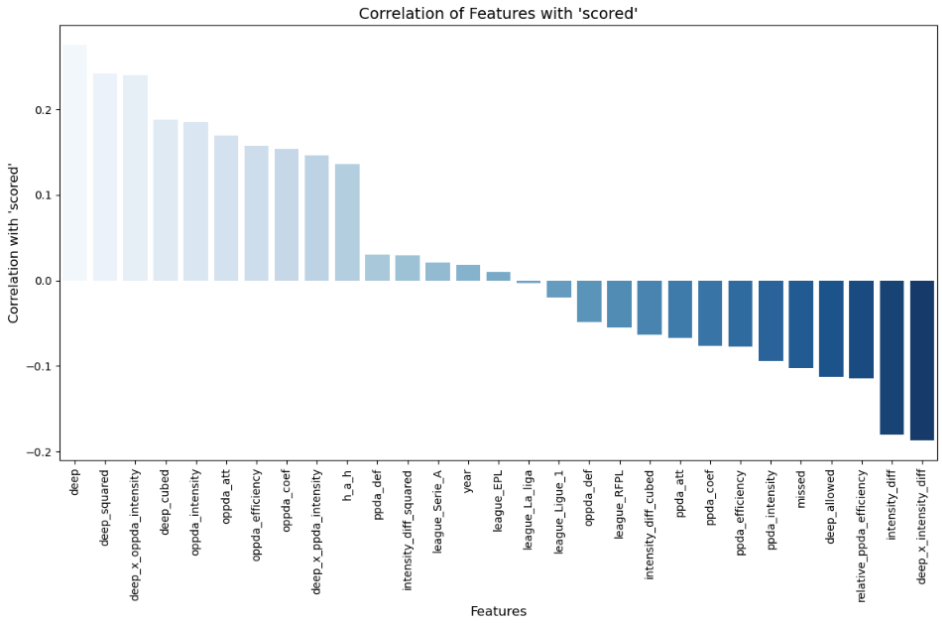

Figure 20: Correlation of Features After Removing Goal-Related Features and Adding Feature Engineering

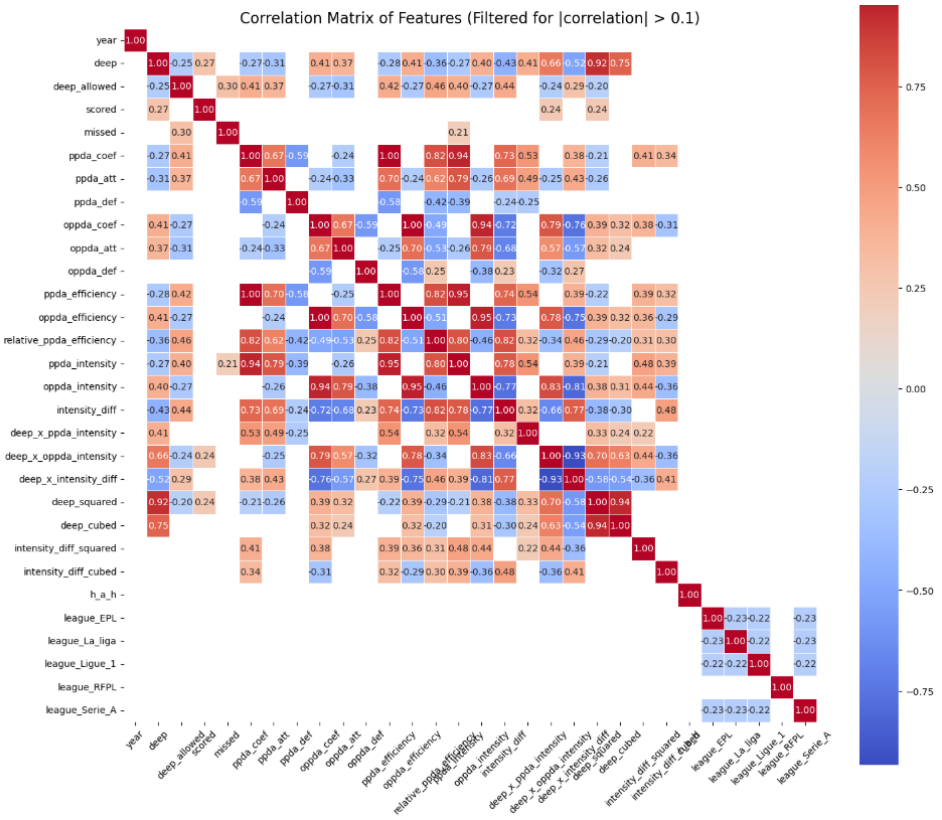

Figure 21: Correlation Matrix of Final Features

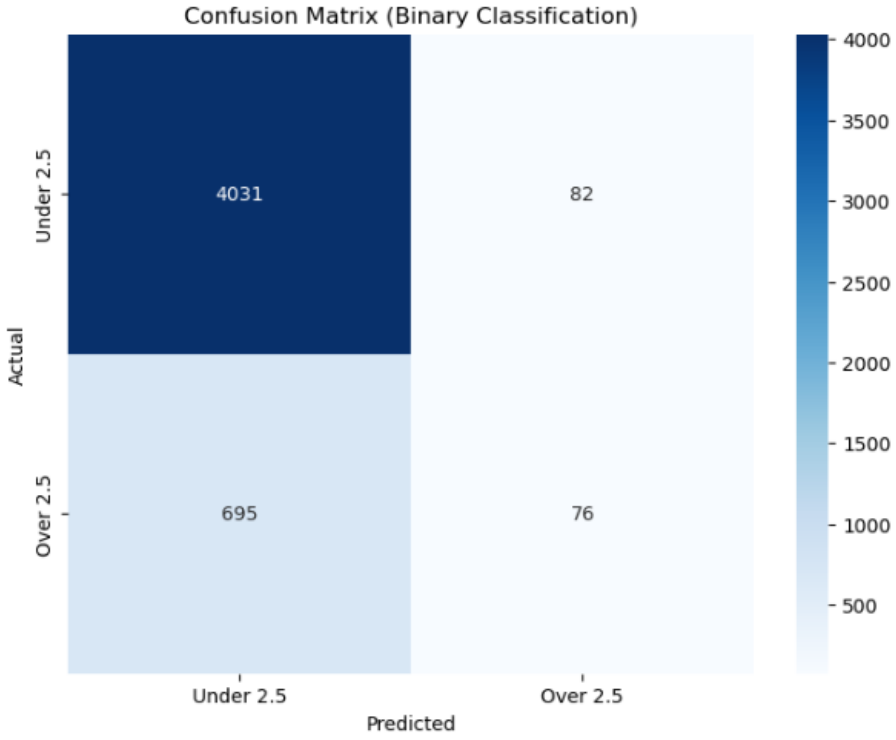

Figure 22: Confusion Matrix (XGBoost Model) for the Binary Classification task.

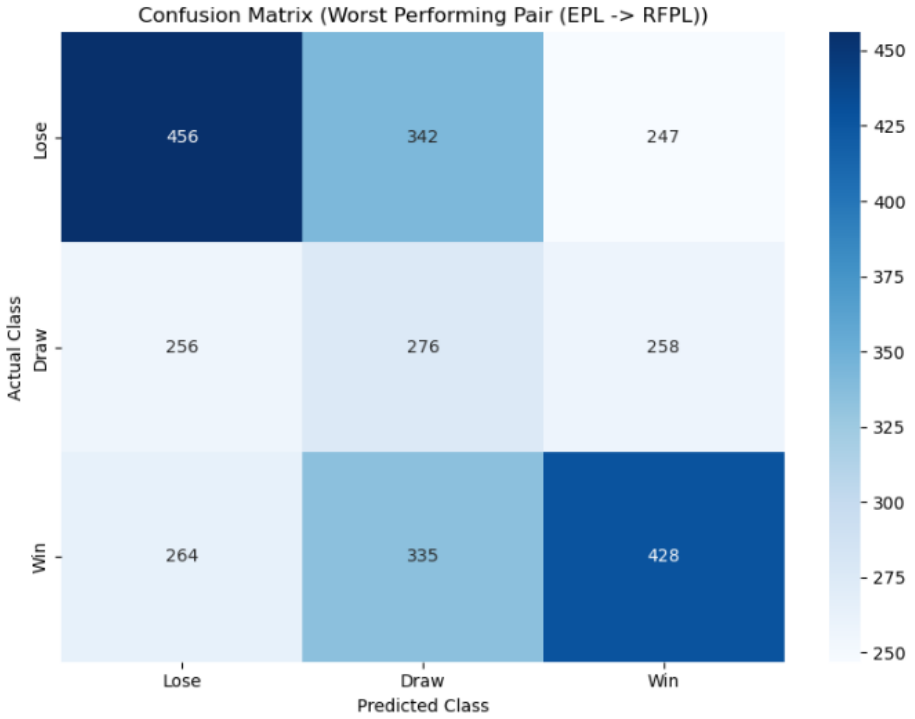

Figure 23: Confusion Matrix (EPL → RFPL)

APPENDIX D

Appendix D, includes confusion matrices for the multi-class classification task, showing model performance when trained on one league and tested on another.

Figure 23 illustrates the Worst-Performing Pair (EPL → RFPL): The matrix reveals significant misclassifications, especially for draws, reflecting challenges in adapting to the tactical differences between EPL and RFPL.

Figure 24 illustrates the Best-Performing Pair (La Liga → Serie A): The matrix shows better generalization, with fewer misclassifications, likely due to similar tactical styles between La Liga and Serie A, though draws remain challenging to predict.

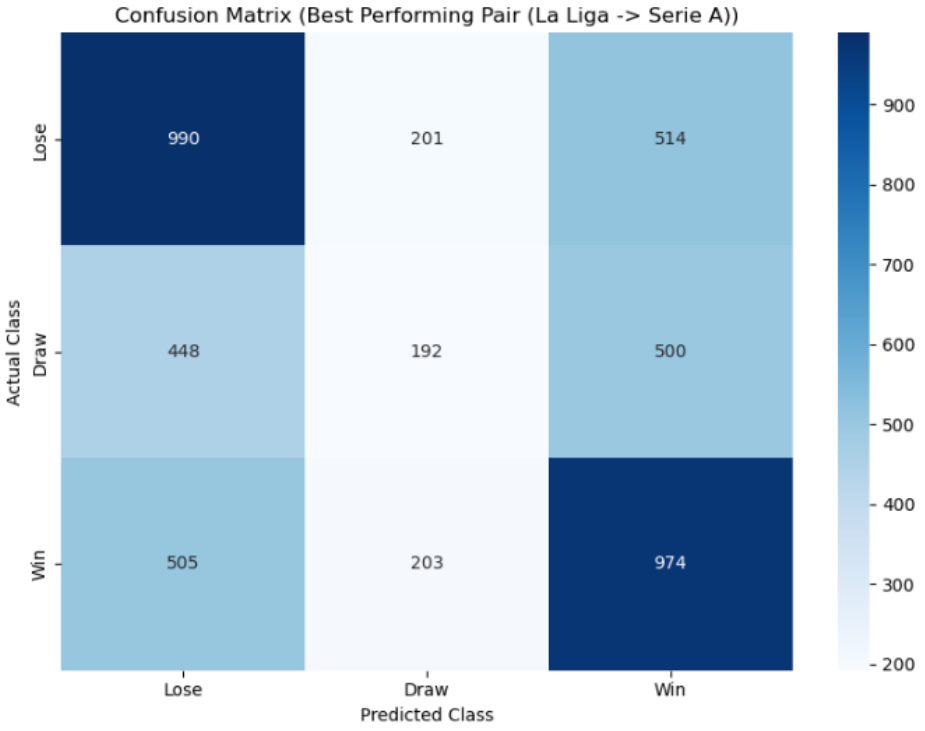

Figure 24: Confusion Matrix (La Liga → Serie A)