# Football Match Outcome Prediction Using Machine Learning



By: Eitan Bluer

Course: DS18

Date: 30.3.2025

**1. Introduction**

Football is one of the most unpredictable sports, yet it also generates vast amounts of structured data that can reveal hidden patterns. In recent years, advanced metrics such as xGoals (expected goals), PPDA (pressing intensity), and deep completions have become essential tools for professional analysis.

This project explores the use of machine learning (ML) to **predict football match outcomes** (home win, draw, away win) by integrating these modern statistical features. Using real historical match data, we aim to transform raw team and player performance indicators into a predictive system that can be applied before matches to forecast results.

The outcome is focused on **game-level predictions**, using both home and away team features, and targets not just prediction accuracy, but also **insight into what drives success on the field**.

**What is the probability that a home team will win, lose, or draw a football match based on match statistics and pre-match statistics (Historical)?**

This project aims to:

* Identify the most important performance metrics that correlate with match results.
* Build and evaluate machine learning models that classify matches into one of three outcomes.
* Create a deployable system to make predictions in real-time, potentially supporting coaches, analysts, sports media, and betting platforms.

The key focus is not only achieving high predictive power, but also **understanding the “why” behind each result** — revealing the tactical patterns hidden within data.

See in Append 1 the Files structure of the model.

**3. Project Design & Methodology**

**3.1 Data Journey**

The data used in this project comes from Kaggel Football Database[[1]](#footnote-1) 7 CSV files ( appearance, games, leagues, players, shots, teams, teamstats) that can be divided into three categories::

* Players performance
* Teams statistics
* Games statistics

These tables were merged and cleaned to construct a unified match-level dataset. Each row represents a single match, with features engineered for both the home and away teams.

**3.2 Data Preparation**

Notebook: Football\_data\_preperation\_13\_3.ipynb

We merged and cleaned the appearances, shots, games, and team performance datasets to build a match-level dataset for modeling. Players were accurately linked to teams using position order, substitution timing, and goal data.

Key team-level features were engineered—such as xGoals, ppda, shots on target, assists, and key passes—with separate values for home and away teams. The final structured dataset contains one row per match, combining raw and engineered features for both teams.

The final file (team\_results.csv) contain 12,860 games

**3.3 Exploratory Data Analysis (EDA)**

Notebook: EDA\_ML\_Football.ipynb

We explored the relationships between match outcomes and key performance metrics using visual tools and statistical tests to identify influential features.

* Included markdown commentary on the class balance of match outcomes (home win, draw, away win).

A graph with different colored rectangles

AI-generated content may be incorrect.

* Checked for normality using skewness and kurtosis tests, confirming that many features are skewed.

A collage of blue graph

AI-generated content may be incorrect.

A graph of a graph

AI-generated content may be incorrect.

A screenshot of a graph

AI-generated content may be incorrect.

* Presented correlation heatmaps and pairplots, revealing strong positive correlations between **xGoals\_h**, **teamgoals\_h**, **total\_assists\_h**, and match outcomes.

A screenshot of a computer

AI-generated content may be incorrect.

The resulting correlation heatmap revealed several pairs of variables with **high correlation coefficients (ρ > 0.8)**, including:

* xGoals\_h and shots\_on\_target\_h
* xAssists\_total and key\_passes\_h
* xGoals\_chain and xThreat\_diff
* Chi-square test with the match results showed **red cards** (redCards\_h, redCards\_a) significantly impact match results (p < 0.001).

A graph showing a number of data

AI-generated content may be incorrect.

* Used visual exploration and ANOVA to highlight influential features like **xGoals**, **deep completions**, **PPDA**, and **shots on target**.
* Found that **xGoals\_h** is significantly higher in wins compared to losses (p < 0.001); **shots\_h** also differ by result (p < 0.001).

A graph of different colored squares

AI-generated content may be incorrect.

**Key findings:**

* **xGoals** and **shots on target** are significantly higher in wins.
* Performance Metrics Drive Results: **Goals, assists, xGoals** and **shots on target** are strong predictors of match outcomes.
* Disciplinary Factors Matter: **Red cards** and **pressing metrics (PPDA)** strongly influence outcomes.

**3.4 Data Cleansing – Outliers & Missing Values**

Notebook: EDA\_Football\_ML\_outliers\_and\_missing\_values.ipynb

We addressed data quality by identifying and handling outliers and missing values to ensure robust model training.

* Detected and removed outliers from 12 features (e.g., **xGoals**, **deep**, **fouls**, **corners**, **blocked/saved shots**) using the **IQR method**.



* Visualized before/after effects with **boxplots**, **histograms**, and **correlation analysis**, confirming improved distribution consistency.

Present 12 feature that we drop their outliers

A collage of graphs

AI-generated content may be incorrect.A group of graphs showing different sizes and shapes

AI-generated content may be incorrect.

A group of graphs showing different sizes and shapes

AI-generated content may be incorrect.A group of blue and white graphs

AI-generated content may be incorrect.

A group of blue and white graphs

AI-generated content may be incorrect.A group of blue and white graphs

AI-generated content may be incorrect.

* Missing values were found across several **numeric** features and one **categorical** variable. Though they had minimal distributional impact, we imputed them using **KNN imputation** to preserve the dataset’s structure and avoid losing rows.

A screenshot of a graph

AI-generated content may be incorrect.

**3.5 Feature Engineering & Feature Selection**

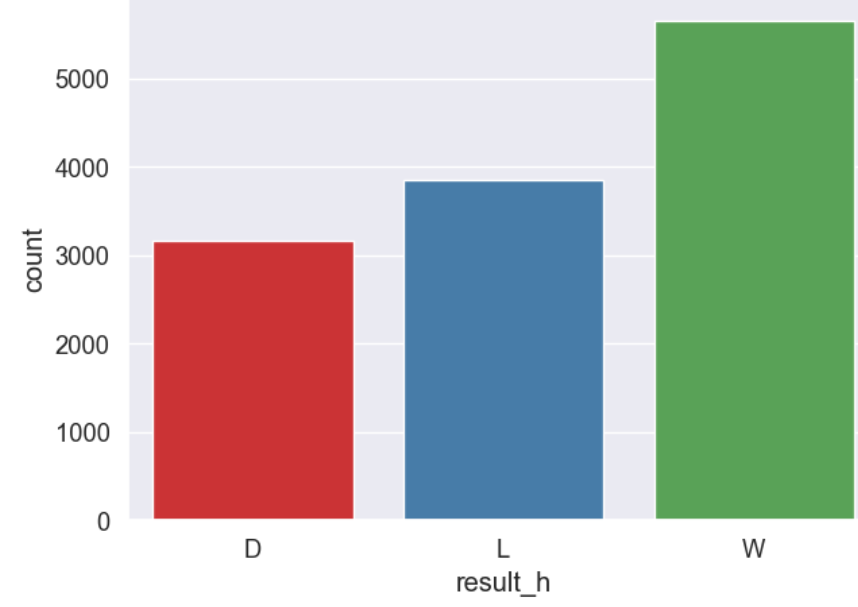
Notebook for **pre-game prediction**: ML\_Football\_Feature\_Engeniring\_and\_feature\_selection.ipynb

Notebook for **in-game prediction**: Prediction\_during\_the\_ganeML\_Football\_Feature\_Engeniring\_and\_feature\_selection.ipynb

This step was divided into two modeling scenarios: **in-game prediction** (using match statistics if the current game without the goals) and **pre-game prediction** (using features derived from the last 5 matches).

* Applied **label encoding** and **one-hot encoding** for categorical variables such as team names and venues.
* Engineered key features including:
  + **goal\_difference**
  + **xThreat\_diff**
  + **xGoals / xThreat ratio**
  + **xGoals\_chain**, **xAssists\_total**
* Introduced **rolling averages and ratios** over the last 5 games (e.g., home\_xGoals\_h\_rolling5, home\_win\_rate\_5) to capture team form and momentum.
* Conducted **PCA** for exploratory purposes, but retained original features to maintain interpretability.
* No external data sources were used; all features were derived from internal match and player statistics.

**3.6 Imbalanced Data**

****

The target (result matches) seems balance for our needs

**3.7 Model Selection & Fine-tuning**

Notebook for **pre-game prediction**: Classification Models and Hyperparameter Finetuning.ipynb

Notebook for **in-game prediction**: Prediction\_during\_game\_Classification Models and Hyperparameter Finetuning.ipynb

The Data split in to train (80%) set and test set (20%). We trained several model Models( Logistic Regression Decision Tree Random Forest AdaBoost Gradient Boosting XGBoost SVM Extra Trees) and test their estimation on the test set. In the part blow presents the result for **in-game prediction** and **pre-game prediction**.

1. **in-game prediction result**

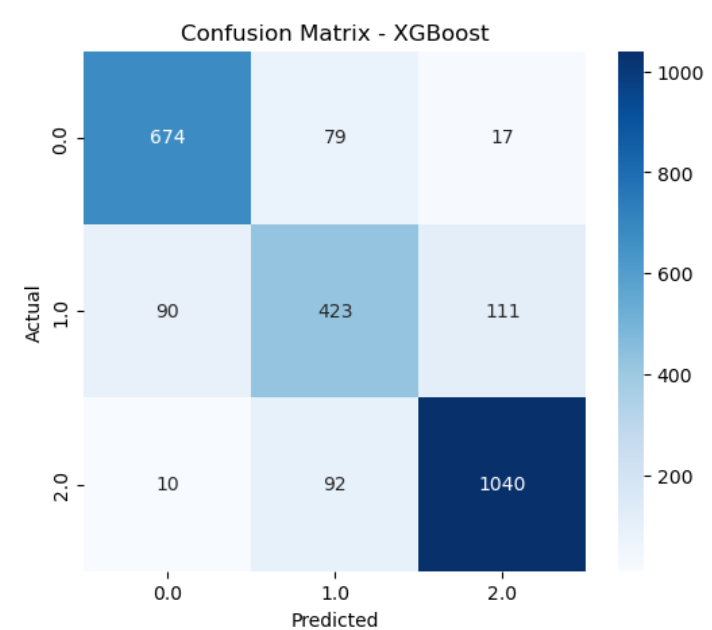
This model takes the features that are part of the current game (e.g total assists category, shot on target, XGoals etx.) with out the goals (that have strong relation with the score) and try to predict the game result.

We tested multiple models:



The best model was **XGBoost**, with the following results:

* **Accuracy**: 82 %
* **AUC**: 0.942
* **F1-score**: 0.793
* **Log-loss**: 0.391
* **Confusion matrices** for each model to identify per-class prediction quality. The best model confusion matrix below



Hyperparameters were tuned via **RandomizedSearchCV**, and model performance was visualized through confusion matrices and feature importance charts.

* Best Parameters: {'subsample': 0.9, 'n\_estimators': 300, 'max\_depth': 7, 'learning\_rate': 0.05, 'gamma': 1, 'colsample\_bytree': 0.7}



1. **pre-game prediction result**

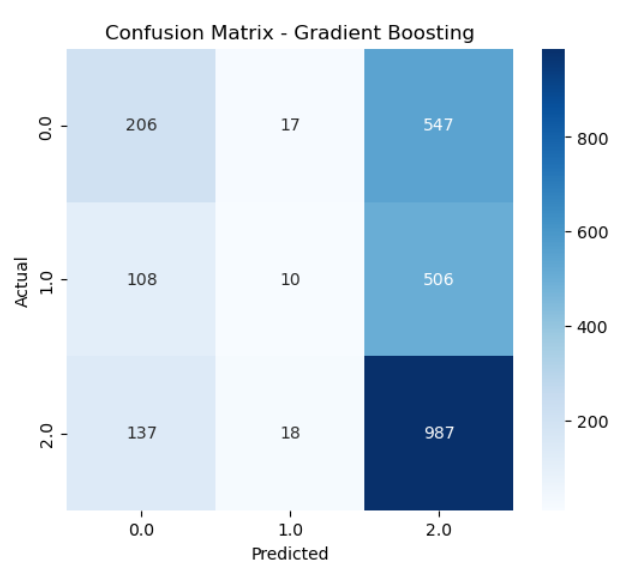
This model takes the features that are Historical statistics (e.g average of 5 games for shots, shots on target, Key pass etc. ) and try to predict the game result before the game started.

We tested multiple models:



The best model was **Gradient Boosting**, with the following results:

* **Accuracy**: 47 %
* **AUC**: 0.575
* **F1-score**: 0.331
* **Log-loss**: 1.046
* **Confusion matrices** for each model to identify per-class prediction quality. Matrix results below (**Gradient Boosting model)** indicate that there is likely a bias in variables entered into the model towards win and loss data than draw results.



Hyperparameters were tuned via **RandomizedSearchCV**, and model performance was visualized through confusion matrices and feature importance charts.

* Best Parameters: {'subsample': 0.9, 'n\_estimators': 300, 'min\_samples\_split': 2, 'min\_samples\_leaf': 5, 'max\_depth': 7, 'learning\_rate': 0.1}



**4. Deployment of Model**

**4.1 Deployment Strategy**

This project resulted in two distinct predictive models aimed at forecasting football match outcomes:

1. **In-Game Prediction Model (XGBoost)**
2. **Pre-Game Prediction Model (Gradient Boosting)**

Each model serves a different purpose and is deployed based on the timing of prediction needs.

**A. In-Game Prediction Model**

* **Use Case**: Real-time match analysis platforms, coaching support tools, and live media commentary.
* **Inputs**: In-match statistics excluding the actual goals (e.g., xGoals, PPDA, key passes, shots on target).
* **Model**: XGBoost Classifier (optimized with RandomizedSearchCV)
* **Deployment Options**:
  + **Web API Endpoint**: Deployed via Flask or FastAPI, allowing real-time match statistics to be sent and predictions returned instantly.
  + **Integration**: Can be embedded into existing analytics dashboards for clubs or broadcasters.
  + **Inference Speed**: Fast prediction time suitable for real-time inference every few minutes during a match.

**B. Pre-Game Prediction Model**

* **Use Case**: Pre-match forecasting for coaches, analysts, betting markets, and sports journalism.
* **Inputs**: Historical team performance metrics (rolling averages over the past 5 matches).
* **Model**: Gradient Boosting Classifier
* **Deployment Options**:
  + **Batch Prediction System**: Generates predictions for upcoming fixtures daily or weekly.
  + **Web Dashboard**: Enables users to select matches and view predicted outcomes along with top influencing features.
  + **Integration**: Ideal for preview shows, betting recommendation engines, and coaching staff game prep.

Both models were wrapped with joblib and exported as .pkl files to be served in a lightweight environment using Docker for consistency across platforms.

**4.2 Model Monitoring & Maintenance**

* **Performance Tracking**: Implemented monitoring hooks to log key metrics (accuracy, AUC, log-loss) per match/week.
* **Retraining Triggers**: Periodic retraining is recommended every 10 gameweeks to account for form, injuries, and lineup changes.
* **Feature Drift Monitoring**: Tracked distribution of top features (e.g., xGoals, red cards) to identify data drift over time.

**5. Conclusions**

This project demonstrated that **machine learning can be a powerful tool for understanding and predicting football match outcomes**, leveraging advanced performance metrics and structured historical data.

**Key Takeaways:**

**In-Game Prediction Model**

* Achieved high predictive accuracy (**82%**) and strong class separation (AUC **0.94**) using real-time match features.
* Top predictors included **xGoals**, **shots on target**, **PPDA**, and **red cards**, aligning well with tactical intuition.
* Highly suitable for real-time analytics, especially in second-screen or coaching applications.

**Pre-Game Prediction Model**

* Achieved moderate performance (**47% accuracy**, AUC **0.575**), reflecting the inherent uncertainty in forecasting based on historical form alone.
* Showed promise in identifying team momentum through features like **rolling xGoals**, **key pass averages**, and **win rates**.
* Could benefit from richer data sources in the future (e.g., lineup data, weather, player injuries).

**Limitations:**

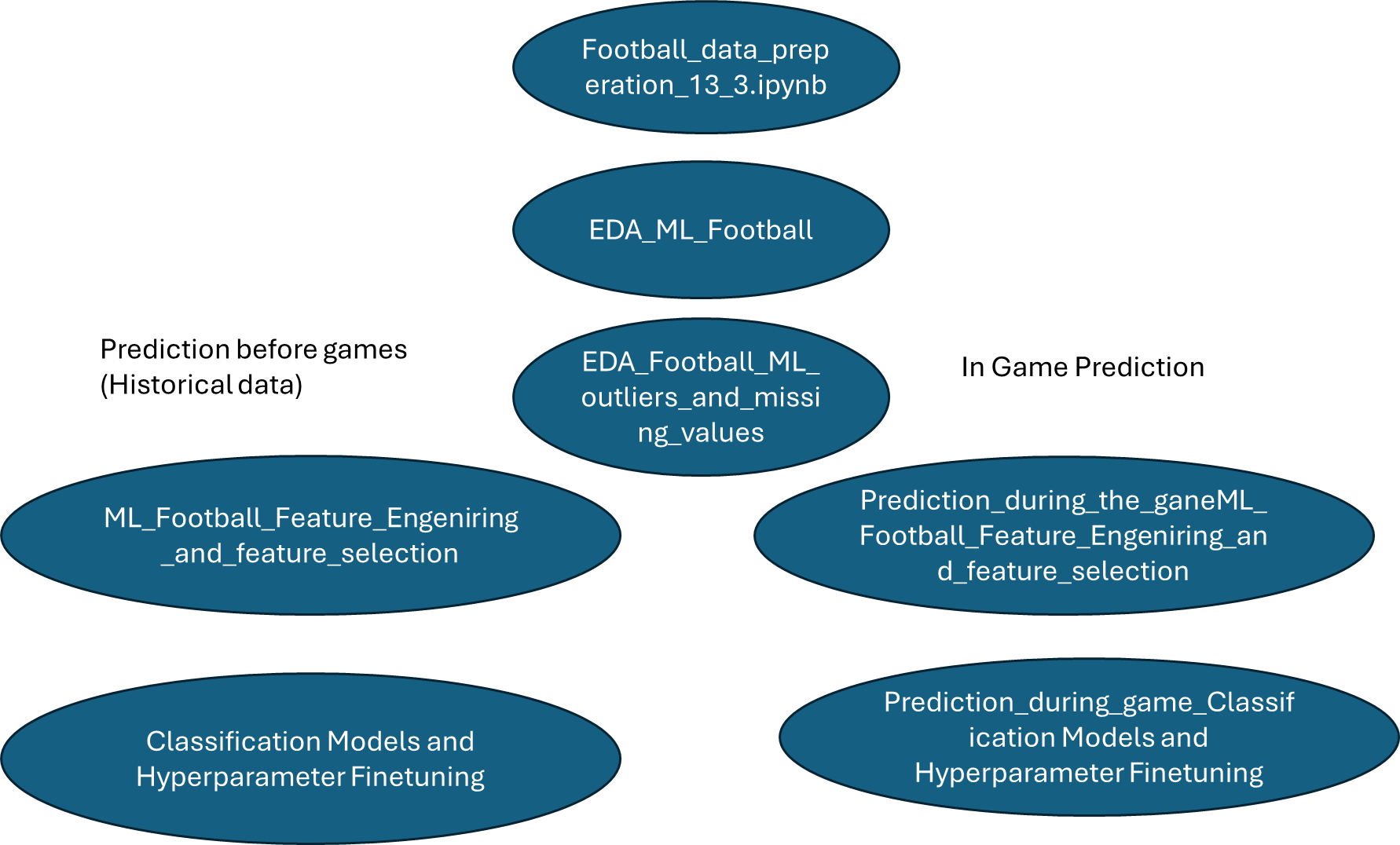
* The pre-game model struggled with predicting draws, likely due to class imbalance or insufficient tactical data.
* No external context (e.g., lineup strength, manager strategies, injuries) was used — incorporating these could boost performance.
* The dataset is historical and may not generalize to future seasons without regular retraining.

**Future Work:**

* **Ensemble models** combining both pre-game and in-game insights for dynamic prediction updates.
* **Explainability tools** (e.g., SHAP values) to better interpret model decisions and aid coaching staff.
* **Transfer learning** across leagues or competitions to improve generalizability.

**Append 1**

The model structure in the GoitHub folder



1. Football Database < https://www.kaggle.com/datasets/technika148/football-database?select=shots.csv https://www.kaggle.com/datasets/technika148/football-database?select=shots.csv> [↑](#footnote-ref-1)