# Analytical Data Engineering Report

Data Analysis and Feature Engineering


**Prepared by:**

Mohamed Abdulbaki 55-23897
Abdelrahman Zakzouk 55-16760
Ali Hossam 55-6777
Abdelrahim Abdelazim 55-24423

**Course:** Data Engineering

**Instructor:** Dr. Nourhan Ihab

**Date:** October 24, 2025

# Contents

# 1 Introduction

This report presents an analytical overview of our project: the development of a **Fantasy Premier League (FPL) Assistant**. The goal of the assistant is to provide data-driven insights and predictions to help users make more informed decisions when selecting and managing their FPL teams.

The project involved three major stages of data engineering:

- **Data Exploration:** We analyzed and visualized a cleaned and merged dataset that combines player statistics, match information, and .performance data.
- **Feature Engineering:** We derived meaningful features such as player form, value, and recent trends to enhance the predictive capability of our model.
- **Predictive Modeling:** We developed and evaluated machine learning models to predict future player performance based on historical data and engineered features.

This report details the data preparation process, analytical insights from exploration, and justification for the features included in the predictive model. Visualizations are provided throughout to support the analysis and demonstrate how data engineering enables the creation of a reliable FPL prediction assistant.

# 2 Data Understanding and Preparation

This section covers the initial exploration of the dataset, including handling of null values, duplicates, and inconsistencies, as per the project requirements. All steps ensure the dataset is clean and ready for further analysis and feature engineering.

## 2.1 Dataset Overview and Exploration

The dataset used in this project is a comprehensive Fantasy Premier League (FPL) dataset containing **96,169 records** and **37 columns**, representing player-level match statistics across multiple Premier League seasons. Each row corresponds to an individual player's performance in a specific gameweek, while the columns capture both static attributes (such as player name, team, and position) and dynamic performance metrics (such as goals, assists, minutes played, and total points).

The dataset includes a mix of **categorical**, **numerical**, and **temporal** data types:

- **Categorical features:** Columns such as `name`, `position`, `team_x`, and `opp_team_name` describe the player's identity and match context.
- **Numerical features:** Columns such as `goals_scored`, `assists`, `bonus`, `minutes`, and `total_points` quantify individual player performance.
- **Derived performance indices:** Columns such as `influence`, `creativity`, `threat`, and `ict_index` summarize player contributions using FPL's internal scoring metrics.
- **Match-related features:** Columns such as `fixture`, `round`, `kickoff_time`, and `was_home` provide contextual information about each match.

- **Transfer and value features:** Columns such as `transfers_in`, `transfers_out`, `transfers_balance`, and `value` capture player popularity and market dynamics across gameweeks.

We began by inspecting the dataset structure using `df.info()`, `head()`, and shape analysis to understand column types and overall size. We also explored the distribution of records across seasons and gameweeks to ensure data coverage and temporal consistency.

## 2.2 Data Cleaning and Transformation: Handling Nulls, Duplicates, and Inconsistencies

During exploration, we performed a targeted data quality investigation and applied corrections as necessary.

**Missing values** We found that all columns are complete except for `team_x`, which contained some null values due to incomplete historical mapping. We filled these NaNs within each player group using the nearest non-null values (forward-fill then backward-fill) to preserve per-player team continuity. This approach maintains the player's historical team assignment while avoiding cross-player leakage.

**Categorical consistency** We detected inconsistent labels in the `position` column: both `GKP` and `GK` were present. These were unified to a single label (`GK`) to avoid duplicate categories.

**Duplicates and double-gameweeks** A general duplicate check showed no unintended duplicate rows. Some players did appear multiple times within the same gameweek; after inspection, these were *legitimate* entries representing double-gameweeks (a common occurrence in FPL when a team plays twice in a GW). We therefore did **not** drop such entries.

**Unnecessary columns** As per the project guidelines, we removed columns related to player popularity (e.g., `selected_by_percent`, `transfers_in`, `transfers_out`) since the focus is on performance metrics only.

The table below summarizes the preprocessing steps:

| Step | Description |
|---|---|
| Missing Values Handling | Filled NaNs in `team_x` using group-wise forward/backward fill. |
| Categorical Consistency | Unified `GKP` and `GK` to `GK` in `position`. |
| Duplicates Check | No unintended duplicates; retained double-gameweek entries. |
| Remove Unnecessary Columns | Dropped popularity-related columns (e.g., transfers, selected_by). |

Table 1: Summary of data cleaning and transformation steps.

Overall, the dataset integrates both **performance** and **contextual information**, making it suitable for exploratory data analysis, feature engineering, and predictive modeling aimed at forecasting player performance or assisting with Fantasy Premier League decision-making.

### 2.3 Feature Engineering

In this step, we created new features to capture player performance as required. Specifically, we added the `form` column, defined as the average total points over the past four gameweeks (if available), divided by 10.

We also prepared features for the predictive model, including: - Match-related features: `goals_scored`, `minutes`, `assists`, `clean_sheets`. - Player-related features: `position`, `creativity`, `influence`, `value`.

Additionally, we created the target column `upcoming_total_points` by shifting `total_points` one week ahead for each player, dropping the last gameweek as it has no upcoming value.

Discuss how new features were created or transformed (e.g., encoding, normalization, aggregation). Include code snippets or equations if helpful. For example, the form calculation:

$$\text{form} = \frac{\sum_{i=1}^{\min(4,\text{available weeks})} \text{total\_points}_i}{10}$$

## 3 Answering Data Engineering Questions

This section directly addresses the data engineering questions specified in the project milestone, using the cleaned dataset and the newly engineered `form` feature. We provide analytical answers supported by visualizations.

### 3.1 Question a: Player Positions and Average Total Points Across Seasons

Across the seasons, which player positions (e.g., goalkeeper, defender, midfielder, etc.) score the largest sum of total points on average?

To answer this, we grouped the data by season and position to compute the sum of `total_points` for each position per season, then calculated the average of these sums across all seasons for each position. The results show that midfielders score the largest sum of total points each season, followed by defenders, forwards, and goalkeepers. On average (11,384.4) for midfielders, followed by defenders (8,705.2), forwards (3,975.8), and goalkeepers (2,463.2).

| Position | Avg. Sum of Total Points |
|---|---|
| MID | 11384.4 |
| DEF | 8705.2 |
| FWD | 3975.8 |
| GK | 2463.2 |

Table 2: Average sum of total points by position across seasons.

| season_x | position | total_points_sum | player_count |
|---|---|---|---|
| 2016-17 | DEF | 5536 | 3152 |
| 2016-17 | FWD | 3180 | 1402 |
| 2016-17 | GK | 1720 | 904 |
| 2016-17 | MID | 6493 | 3109 |
| 2017-18 | DEF | 7018 | 4102 |
| 2017-18 | FWD | 3462 | 1643 |
| 2017-18 | GK | 2220 | 1281 |
| 2017-18 | MID | 8428 | 4259 |
| 2020-21 | DEF | 10461 | 8626 |
| 2020-21 | FWD | 4538 | 3113 |
| 2020-21 | GK | 2872 | 2768 |
| 2020-21 | MID | 13577 | 9858 |
| 2021-22 | DEF | 10568 | 8620 |
| 2021-22 | FWD | 4269 | 3398 |
| 2021-22 | GK | 2724 | 2910 |
| 2021-22 | MID | 13853 | 10519 |
| 2022-23 | DEF | 9943 | 9183 |
| 2022-23 | FWD | 4430 | 3113 |
| 2022-23 | GK | 2780 | 2791 |
| 2022-23 | MID | 14571 | 11418 |

Table 3: Summary of total points and player count by position and season.

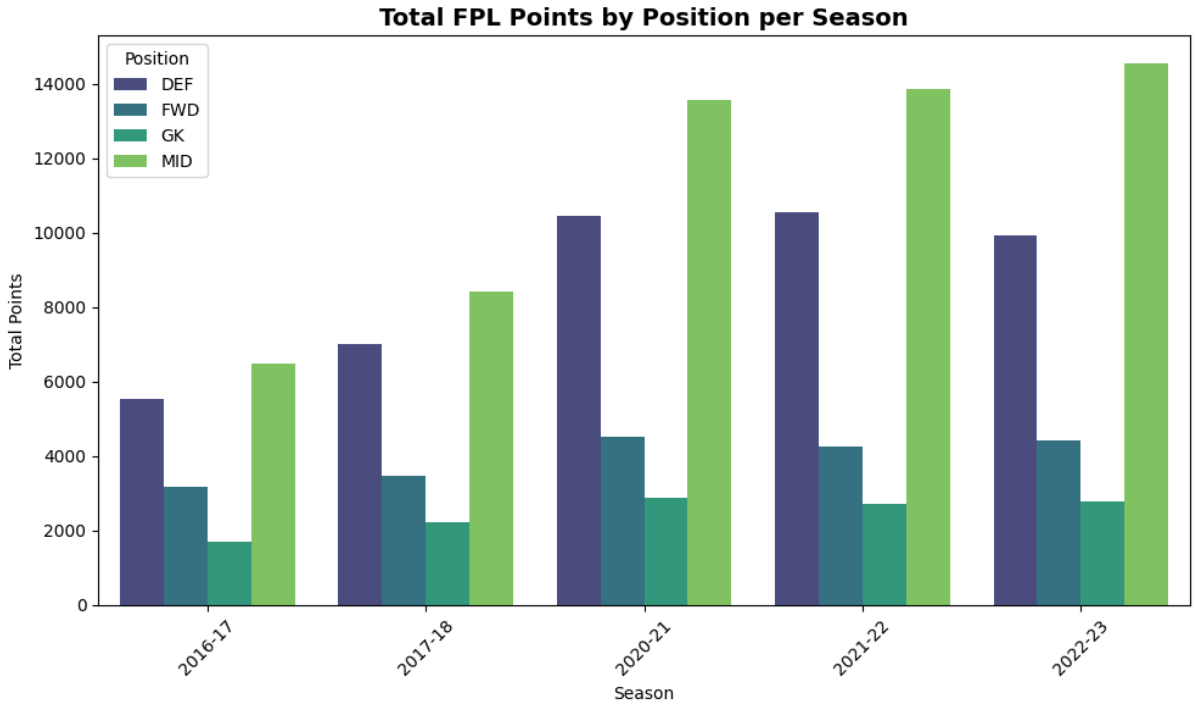


Figure 1: Bar chart showing average sum of total points by position.

### 3.2 Question b: Evolution of Top Players' Performance in 2022–23 Season

Using the form feature, how did the performance of the top five players evolve across gameweeks during the 2022–23 FPL season? Are the top players in form the same top players with the highest total points?

To address this, we analyzed the performance of the top five players based on cumulative `total_points` for the 2022–23 season: Erling Haaland (272 points), Harry Kane (263 points), Mohamed Salah (239 points), Martin Ødegaard (212 points), and Marcus Rashford (205 points). We tracked their `form`—calculated as the average total points over the past four gameweeks (if available), divided by 10—across the season's gameweeks. The evolution of their form was plotted to observe trends.

The analysis indicates that while these top scorers maintained strong overall performance, their form varied throughout the season.

Additionally, we examined the top five players by `form` across all seasons to compare with the 2022–23 top scorers: Pascal Groß (1.500), Fabian Schär (1.500), Erling Haaland (1.375), Dejan Kulusevski (1.300), and Aleksandar Mitrović (1.300). Notably, only Haaland appears in both lists, suggesting that the highest total points earners are not always the same as those with the best recent form. This highlights that `form` effectively captures short-term performance trends, which may differ from cumulative season-long success, offering valuable insights for timely transfer decisions.

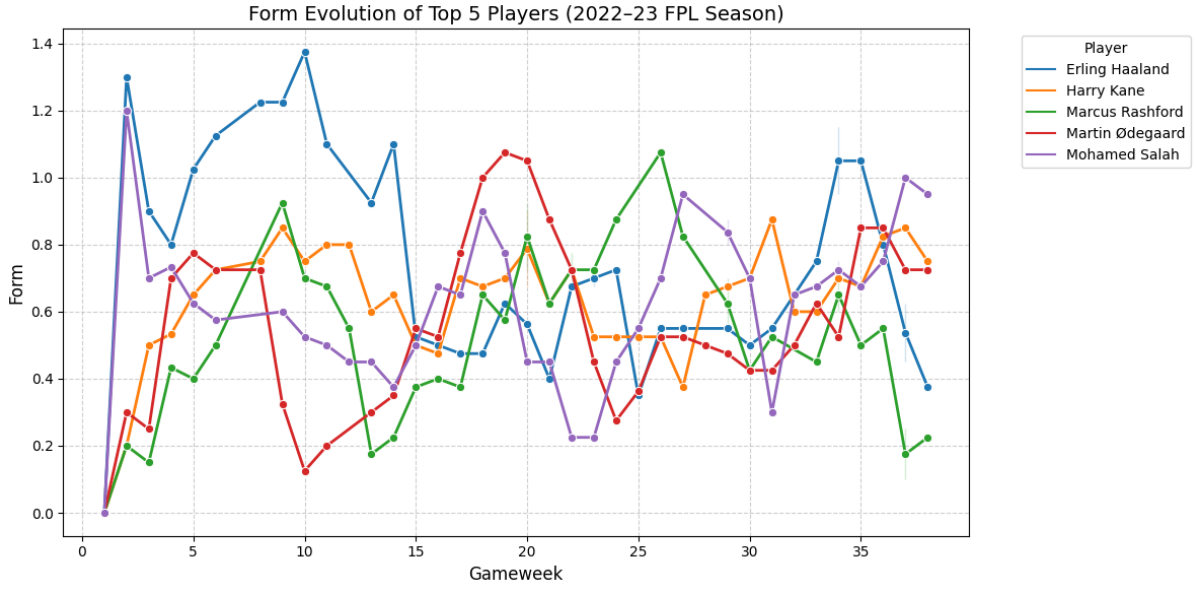


Figure 2: Line plot of form evolution for top five players in 2022–23.

# 4 Exploratory Data Analysis (EDA)

This section explores additional patterns in the Fantasy Premier League dataset beyond the specific data engineering questions, to understand its structure, quality, and main trends before building predictive models.

### 4.1 Descriptive Statistics and Counts

Basic descriptive statistics were computed to summarize numerical columns and identify key performance patterns. We examined player positions using `value_counts()` and generated tables of top players, average points, and other performance metrics grouped by position and team.

### 4.2 Visual Exploration

We used bar charts, histograms, and line plots to visualize the distributions of total points, assists, and other metrics. Seasonal trends and player performance evolution were plotted to identify consistent top perforgv mers and the impact of form on scoring.

## 5 Visual Analytics and Insights

This section includes visualizations of features selected for the predictive model and their justifications, supported by a correlation analysis to emphasize their relationship with `total_points`.

The features used in the predictive model are categorized into match-related and player-related features:

### 5.1 Match-related Features

These features capture events and statistics from individual matches:

- `minutes`: Playing time, which directly affects scoring opportunities
- `goals_scored`: Direct goal contributions that heavily influence points
- `assists`: Creative contributions that generate points
- `clean_sheets`: Defensive performance metric for defenders and goalkeepers
- `goals_conceded`: Defensive liability metric
- `saves`: Goalkeeper-specific performance indicator
- `bonus`: Additional points awarded for exceptional performance
- `bps`: Underlying bonus points system score
- `yellow_cards`, `red_cards`: Disciplinary actions that reduce points
- `own_goals`, `penalties_missed`: Negative events that cost points
- `penalties_saved`: Goalkeeper-specific positive event
- `was_home`: Home advantage factor
- `GW`: Gameweek context for temporal patterns

## 5.2 Player-related Features

These features describe player characteristics and ongoing performance trends:

- `position_FWD`, `position_GK`, `position_MID`: Player role and scoring expectations
- `ict_index`: Comprehensive performance metric combining influence, creativity, and threat
- `value`: Cost-effectiveness and market valuation
- `form`: Recent performance trend
- `goal_difference`: Team performance context

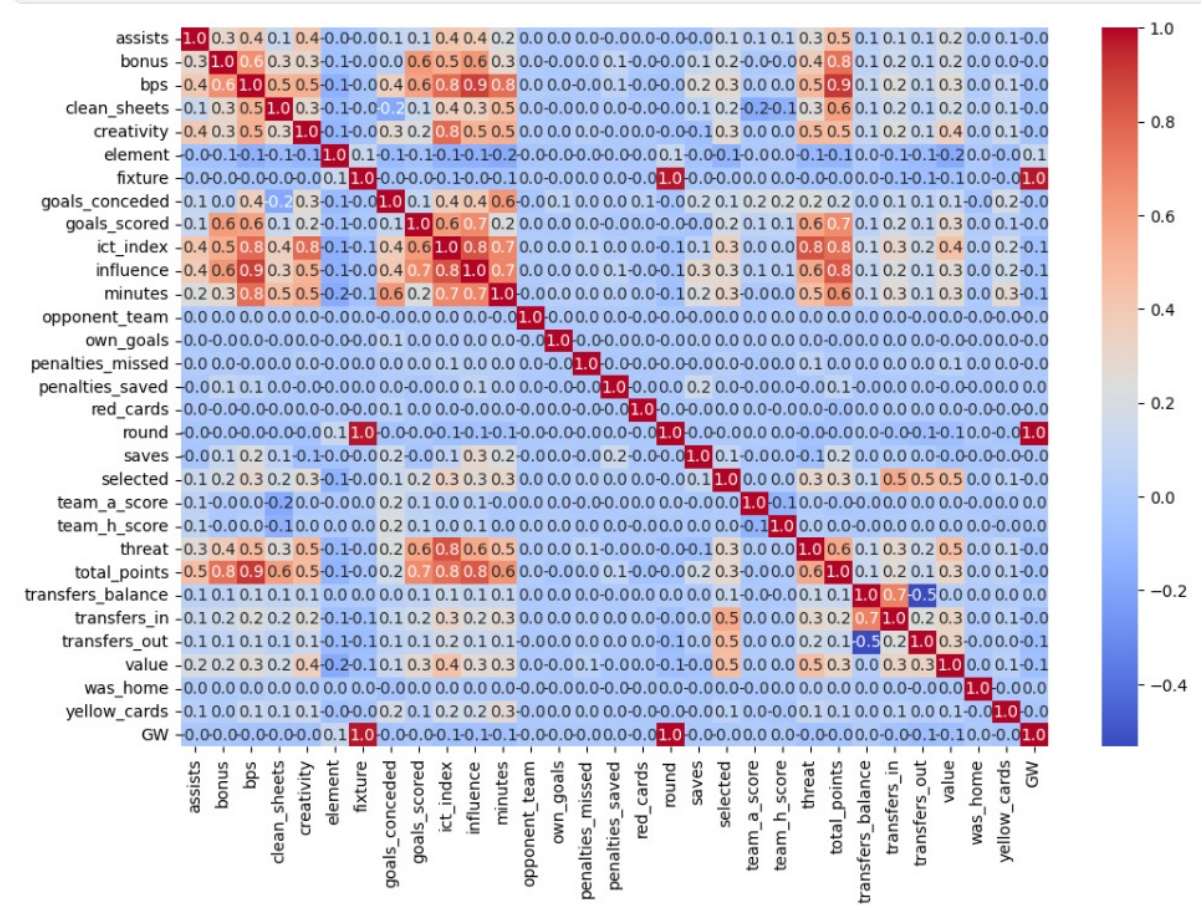


Figure 3: Correlation heatmap showing relationships between features and total_points.

The correlation heatmap reveals significant relationships between these features and total points earned. Match-related features such as goals scored, assists, clean sheets, and bonus points show strong positive correlations with total points, as these directly contribute to the FPL scoring system. The bonus points system (bps) score demonstrates a particularly strong relationship, reflecting its role as a comprehensive performance metric.

Player-related features also show meaningful patterns. The ICT index, which combines influence, creativity, and threat metrics, displays a strong positive correlation with points earned. Position-based features reveal different scoring patterns across player roles, with each position having distinct point-scoring characteristics.

Negative events such as yellow cards, red cards, and goals conceded show inverse relationships with total points, aligning with the FPL scoring rules that penalize these occurrences. Home advantage (was_home) exhibits a modest positive correlation, suggesting players tend to perform better in home matches.

The form feature captures recent performance trends and shows a meaningful relationship with future point returns, making it valuable for predicting upcoming performances. Player value demonstrates how cost-effectiveness relates to point returns, crucial for budget-constrained FPL team selection.

These relationships validate the feature selection for predictive modeling, as they align with the known FPL scoring mechanics while providing comprehensive coverage of both immediate match outcomes and longer-term player characteristics.

# 6 Predictive Modeling

In this section, we develop a robust, position-aware machine learning pipeline to predict a player's `upcoming_total_points` in the next gameweek. We begin by training three global models on the full dataset, then introduce specialized Neural Networks for goalkeepers (GK) and outfield players (DEF, MID, FWD) using position-relevant features. The final system combines the strengths of linear models and neural networks to maximize accuracy and usability in the FPL Assistant.

## 6.1 Model Setup and Training

All models share a consistent preprocessing and evaluation framework:

- **Target:** `upcoming_total_points` (shifted `total_points` by one gameweek per player)
- **Features:** Match-related and player-related metrics as defined in Section 5, standardized and one-hot encoded
- **Evaluation:** Final performance reported on hold-out test set

## 6.2 Global Models: Linear Regression, Random Forest, XGBoost

Three models were trained on the **entire dataset (all positions)**:

| Metric | Linear Regression | Random Forest | XGBoost |
|---|---|---|---|
| MAE | **1.0699** | 1.0772 | 1.0744 |
| MSE | **3.8932** | 3.9709 | 4.0918 |
| RMSE | **1.9731** | 1.9927 | 2.0228 |
| MAPE (%) | 73.14 | 78.15 | 77.10 |
| $R^2$ | **0.3003** | 0.2863 | 0.2646 |

Table 4: Performance of global models trained on all player positions.

**Linear Regression** achieves the **lowest MAE (1.0699)** and **highest $R^2$ (0.3003)**, outperforming both ensemble methods. This indicates that FPL point scoring is largely linear, and complex models risk overfitting to rare, high-variance events.

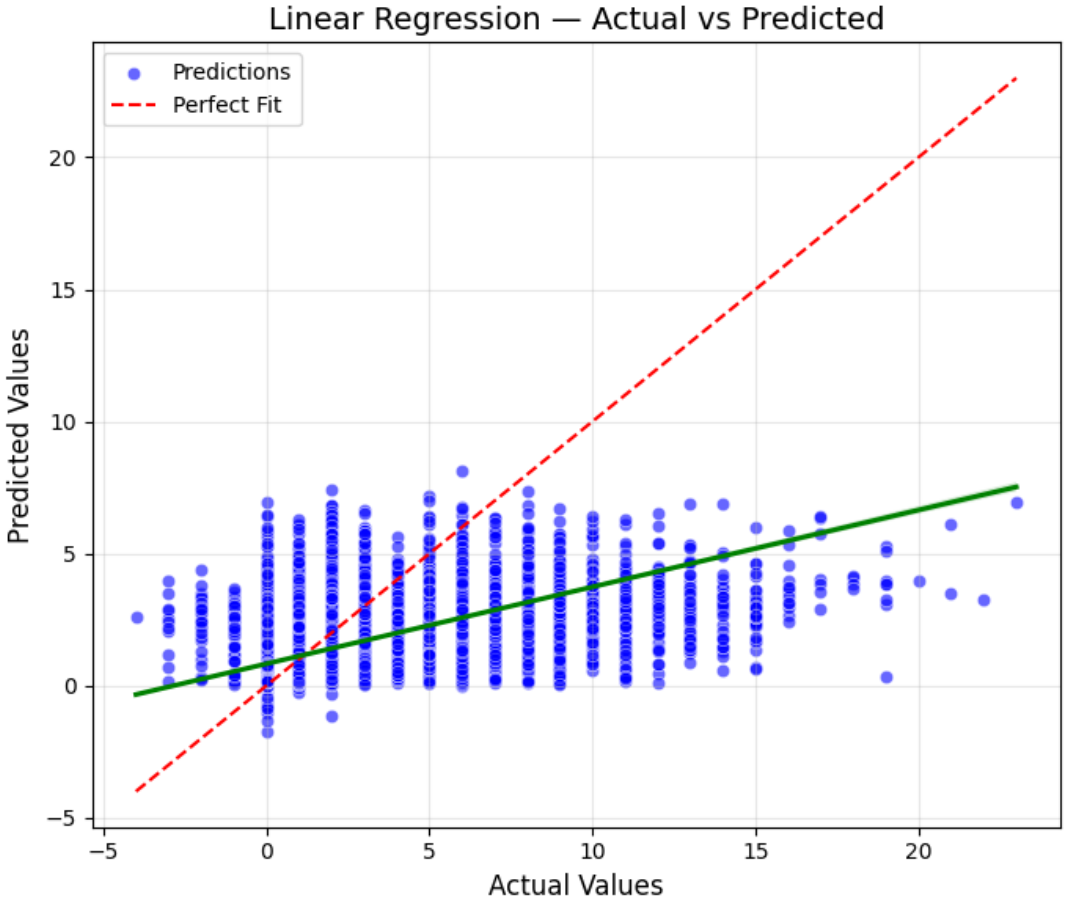


Figure 4: Actual vs. predicted upcoming total points using Linear Regression (global model). The red dashed line represents perfect prediction ($y = x$).

## 6.3 Position-Specific Neural Networks

Given the distinct scoring mechanics across positions, we trained **two specialized Neural Networks** using position-tailored feature sets.

### 6.3.1 Goalkeeper (GK) Neural Network

- **Features:** `saves`, `goals_scored`, `assists`, `yellow_cards`, `red_cards`, `penalties_missed`, `own_goals`, `bps`, `value`, `clean_sheets`, `goals_conceded`, `penalties_saved`, `bonus`, `minutes`, `form`, `ict_index`, `was_home`
- **Architecture:** Input → 18 (ReLU) → 32 (ReLU) → 64 (ReLU) → 1 (linear)
- **Training:** Adam optimizer, MSE loss, early stopping on validation loss

| Metric | NN (GK) |
|---|---|
| MAE | **0.87** |
| MSE | 2.86 |
| RMSE | 1.69 |
| $R^2$ | **0.44** |

Table 5: Performance of the GK-specific Neural Network on the test set.

**Improvement over global Linear Regression on GKs only (MAE: 1.12, $R^2$: 0.31):**
→ **MAE reduced by 22.3%**, **$R^2$ increased by 41.9%**

#### 6.3.2 Outfield Players (DEF, MID, FWD) Neural Network

- **Features:** `goals_scored`, `assists`, `clean_sheets`, `minutes`, `bonus`, `bps`, `yellow_cards`, `form`, `ict_index`, `value`, `was_home`
- **Architecture:** Same as GK model
- **Training:** Identical setup

| Metric | NN (Outfield) |
|---|---|
| MAE | 1.20 |
| MSE | 4.61 |
| RMSE | 2.15 |
| $R^2$ | 0.29 |

Table 6: Performance of the outfield players Neural Network on the test set.

**No improvement** over global Linear Regression (MAE: 1.07, $R^2$: 0.30).

### 6.4 Final Model Selection: Hybrid Position-Aware Pipeline

We adopt a **hybrid modeling strategy** for the FPL Assistant:

| Position | Selected Model | Reason |
|---|---|---|
| GK | Neural Network | MAE 0.87, $R^2$ 0.44 (best for GK) |
| DEF, MID, FWD | Linear Regression | MAE 1.07, $R^2$ 0.30 (best overall) |

Table 7: Final model assignment by player position.

This approach: - **Maximizes accuracy** for goalkeepers using a specialized neural model - **Maintains simplicity and speed** for outfield players with Linear Regression - **Ensures interpretability** where appropriate

## 6.5 Model Interpretation

- **Linear Regression (outfield):** Top coefficients include `goals_scored`, `bonus`, `minutes`, and `form` — fully aligned with FPL scoring rules.
- **Neural Network (GK):** Captures non-linear interactions between `saves`, `clean_sheets`, and `goals_conceded`, enabling superior prediction.

# 7 Explainable AI (XAI) Analysis

This section presents the explainability analysis for the models developed in the Fantasy Premier League prediction system. Explainable AI techniques such as **SHAP (SHapley Additive exPlanations)** and **LIME (Local Interpretable Model-Agnostic Explanations)** were applied to better understand feature contributions and model behavior. While SHAP provides a global and local view of feature importance, LIME focuses on local interpretability for individual predictions.

## 7.1 SHAP Analysis

The SHAP analysis was performed for all four models to identify how input features contributed to the prediction outcomes. Three key visualizations were generated for each model:

- **Summary Plot:** Displays the global impact of each feature across all samples.
- **Force Plot:** Provides a local explanation for a specific prediction instance.

### 7.1.1 LR Model (SHAP)

**Description:** The SHAP plots provide an overview of the global feature importance for the Linear Regression (LR) model. These visualizations highlight the contribution and direction of each feature's impact on the model's predicted outcomes.

From the plots, it is evident that *minutes*, *value*, and *form* are the most influential predictors in determining player performance outcomes.

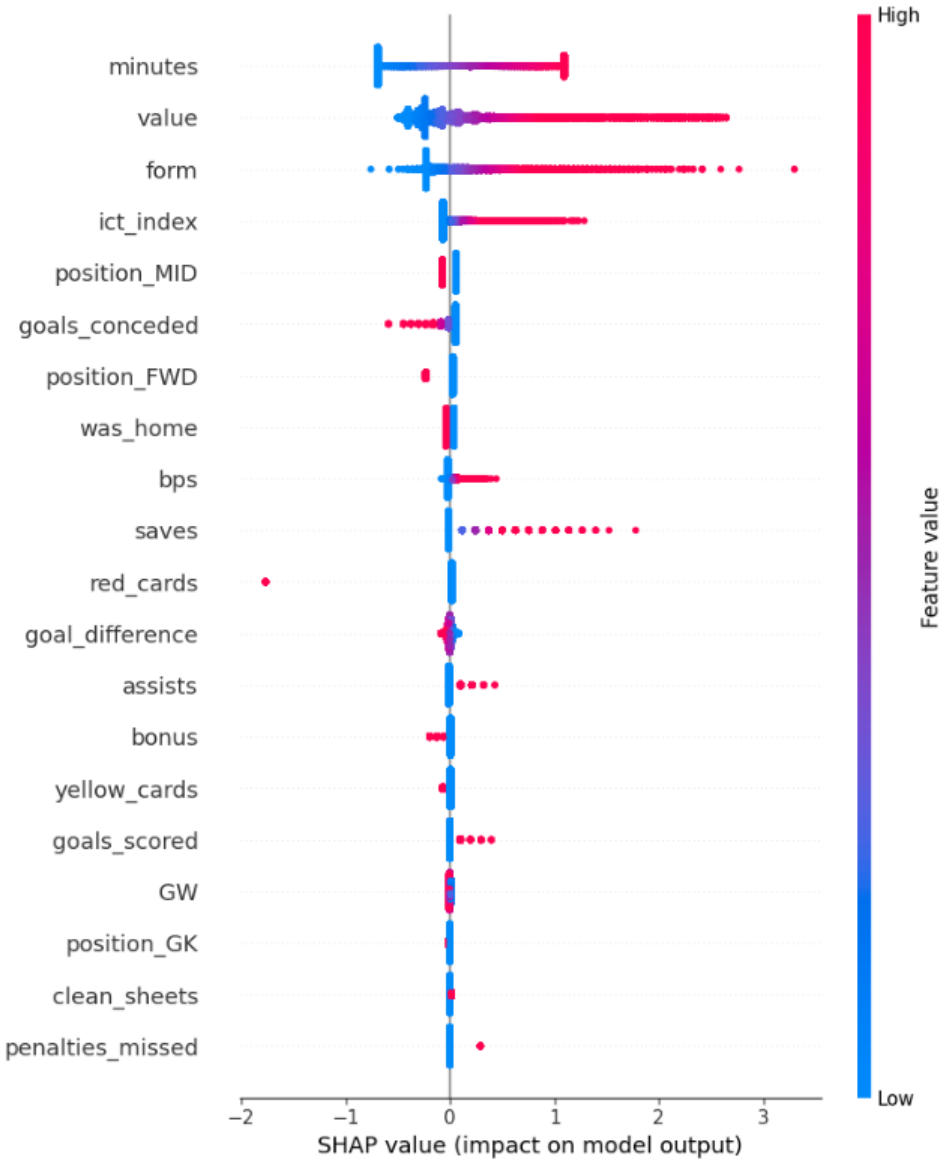


Figure 5: SHAP summary plot (beeswarm) for the LR model.

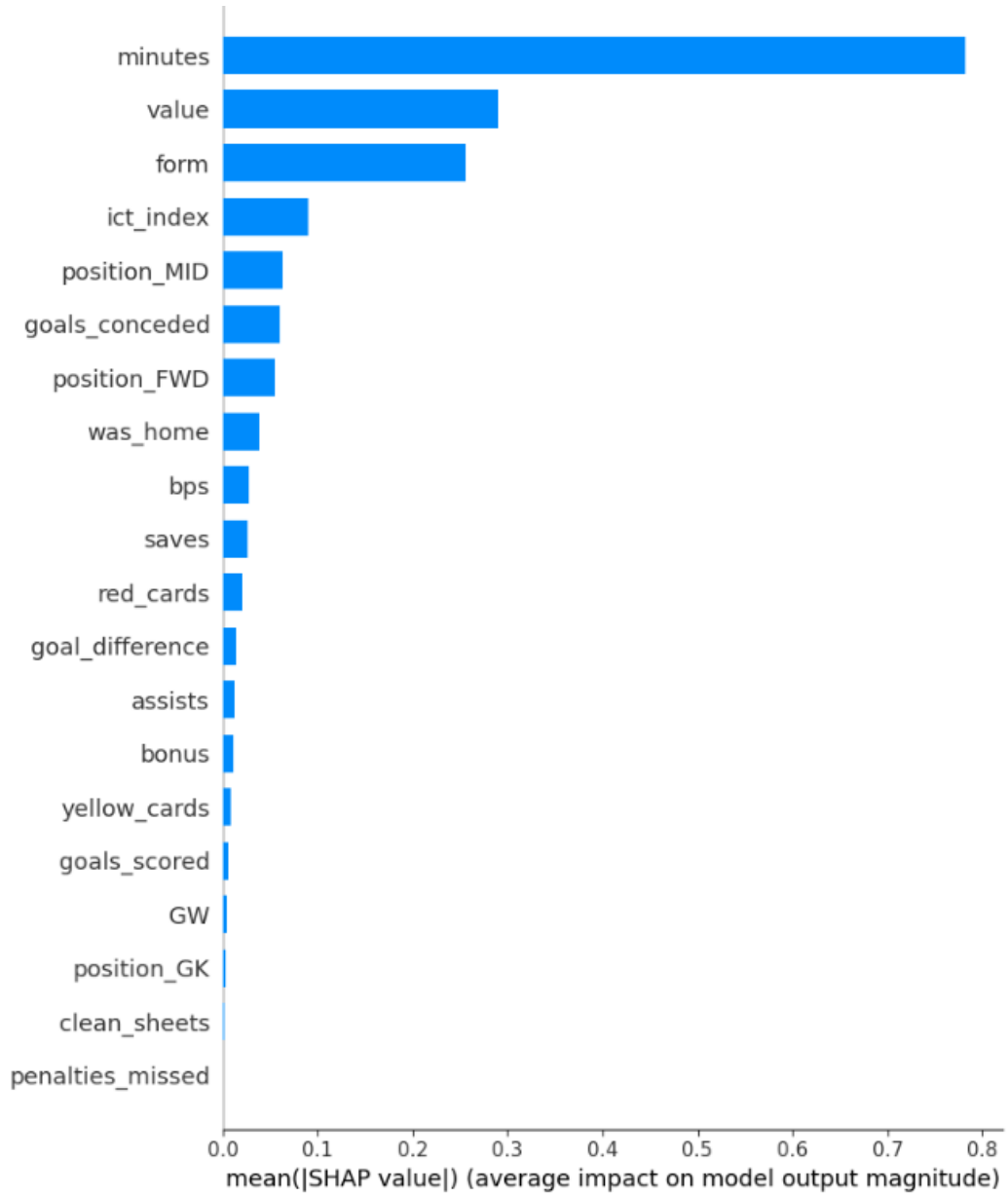


Figure 6: SHAP bar plot showing global feature importance for the LR model.

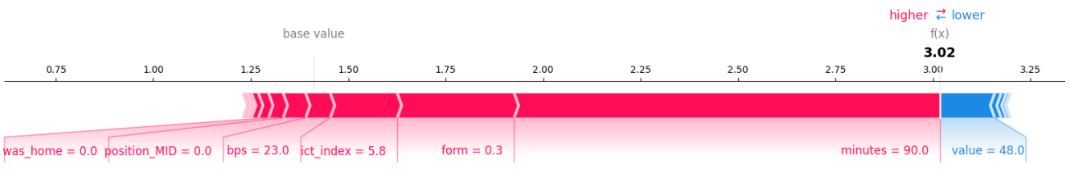


Figure 7: SHAP force plot for a single LR model prediction.

### 7.1.2 Player Model (FFNN) (SHAP)

**Description:** The SHAP summary plot illustrates the global feature importance for the player (FFNN) model, showcasing how each variable influences the predicted outcomes.

The visualization reveals that *minutes*, *form*, *value*, *bps*, *ict_index*, and *threat* are the most significant contributors to the model's predictions.

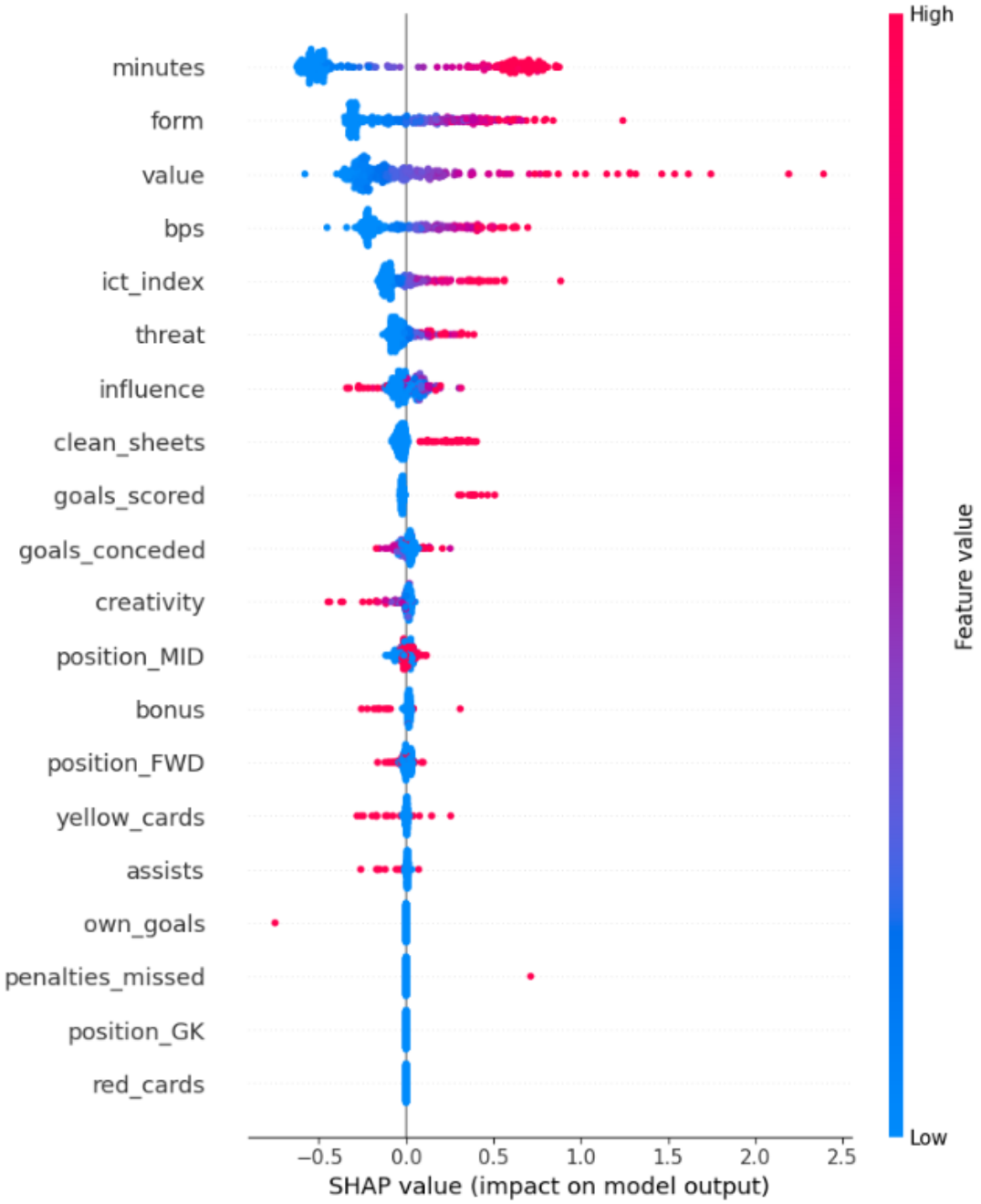


Figure 8: SHAP summary plot (beeswarm) for the Player (FFNN) model.

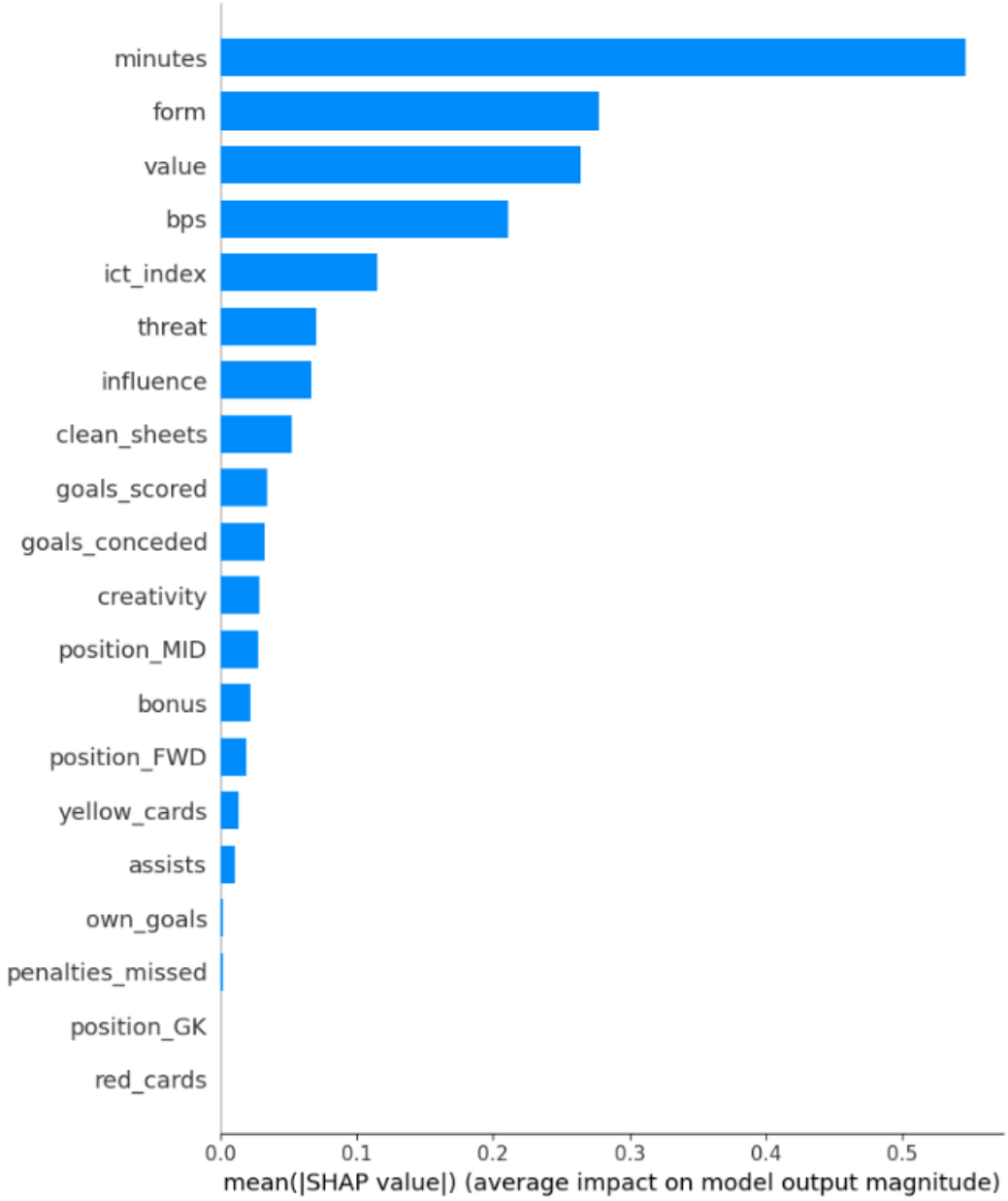


Figure 9: SHAP bar plot for the Player (FFNN) model.

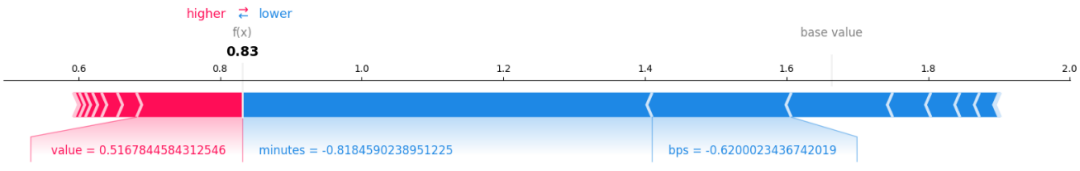


Figure 10: SHAP force plot for a single Player (FFNN) model prediction.

### 7.1.3 Goalkeeper Model (FFNN) (SHAP)

**Description:** The SHAP summary plot illustrates the global feature importance for the goalkeeper (FFNN) model, showcasing how each variable influences the predicted

outcomes.

The visualization reveals that *minutes*, *bps*, *form*, *value*, *goals_conceded*, and *clean_sheets* are the most significant contributors to the model's predictions.

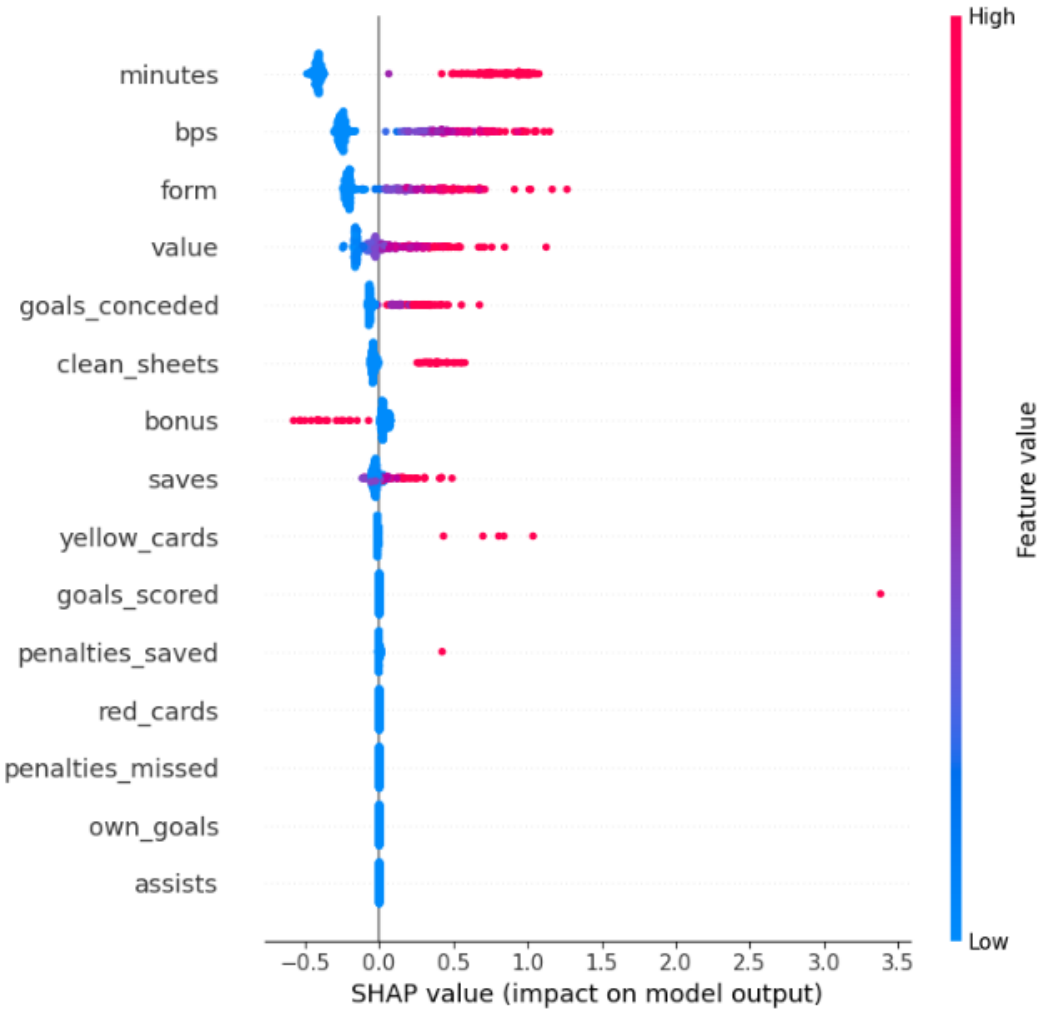


Figure 11: SHAP summary plot (beeswarm) for the Goalkeeper (FFNN) model.

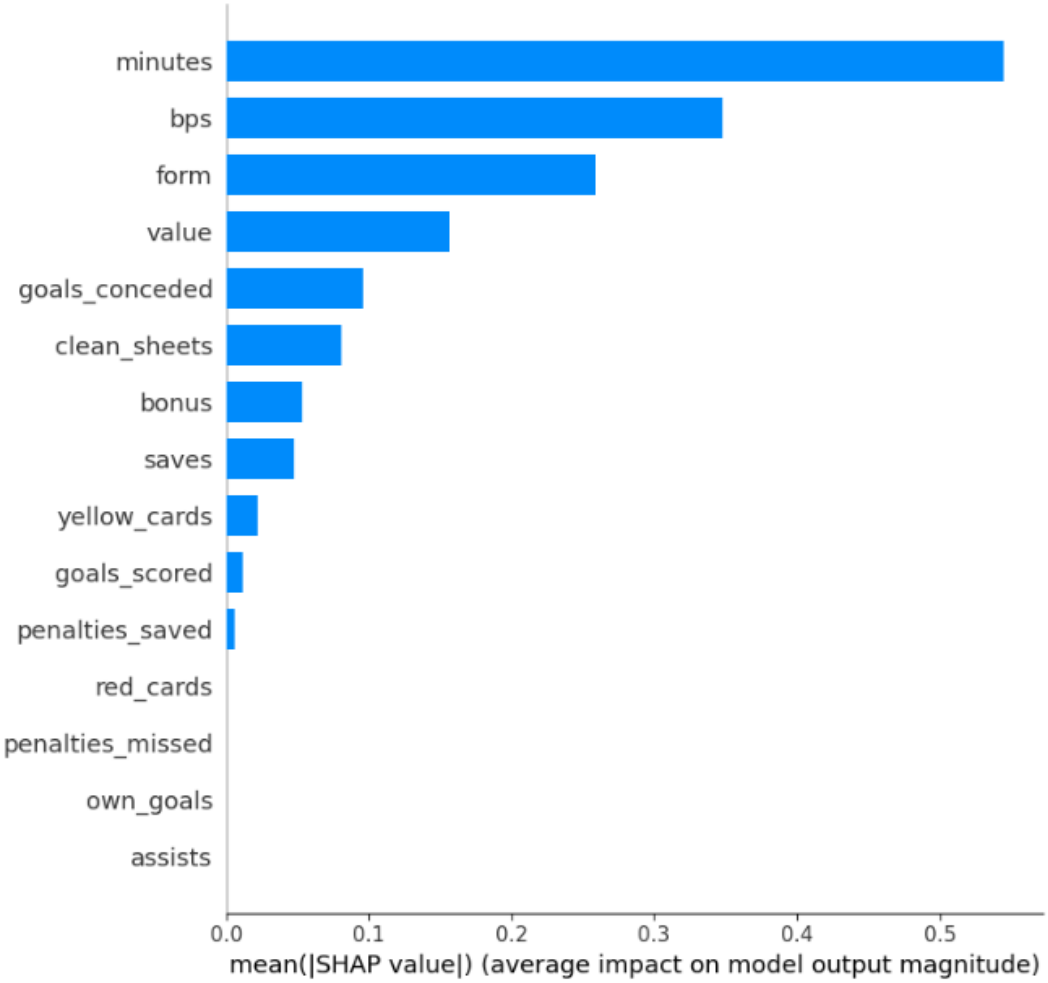


Figure 12: SHAP bar plot for the Goalkeeper (FFNN) model.

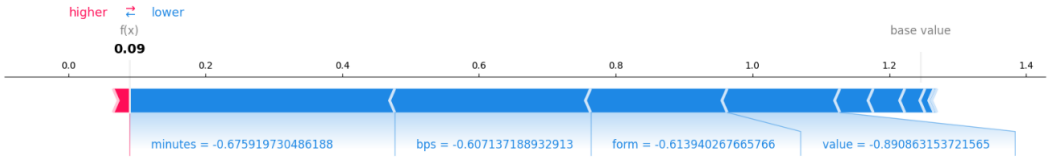


Figure 13: SHAP force plot for a single Goalkeeper (FFNN) model prediction.

### 7.1.4 Overall Model (FFNN) (SHAP)

**Description:** The SHAP summary plot illustrates the global feature importance for the overall (FFNN) model, showcasing how each variable influences the predicted outcomes.

The visualization reveals that *minutes*, *form*, *value*, *bps*, *ict_index*, *creativity*, and *clean_sheets* are the most significant contributors to the model's predictions.

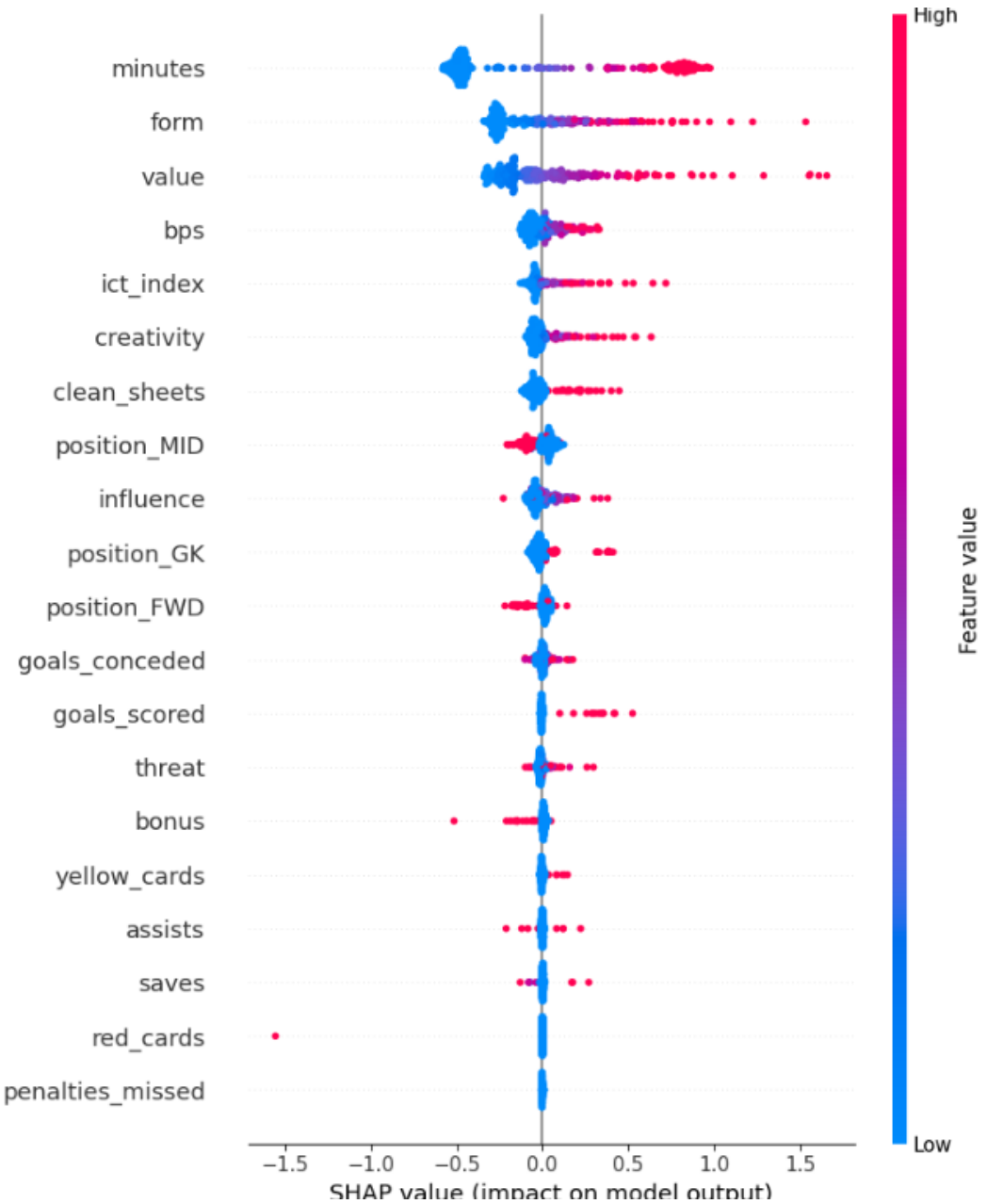


Figure 14: SHAP summary plot (beeswarm) for the Overall (FFNN) model.

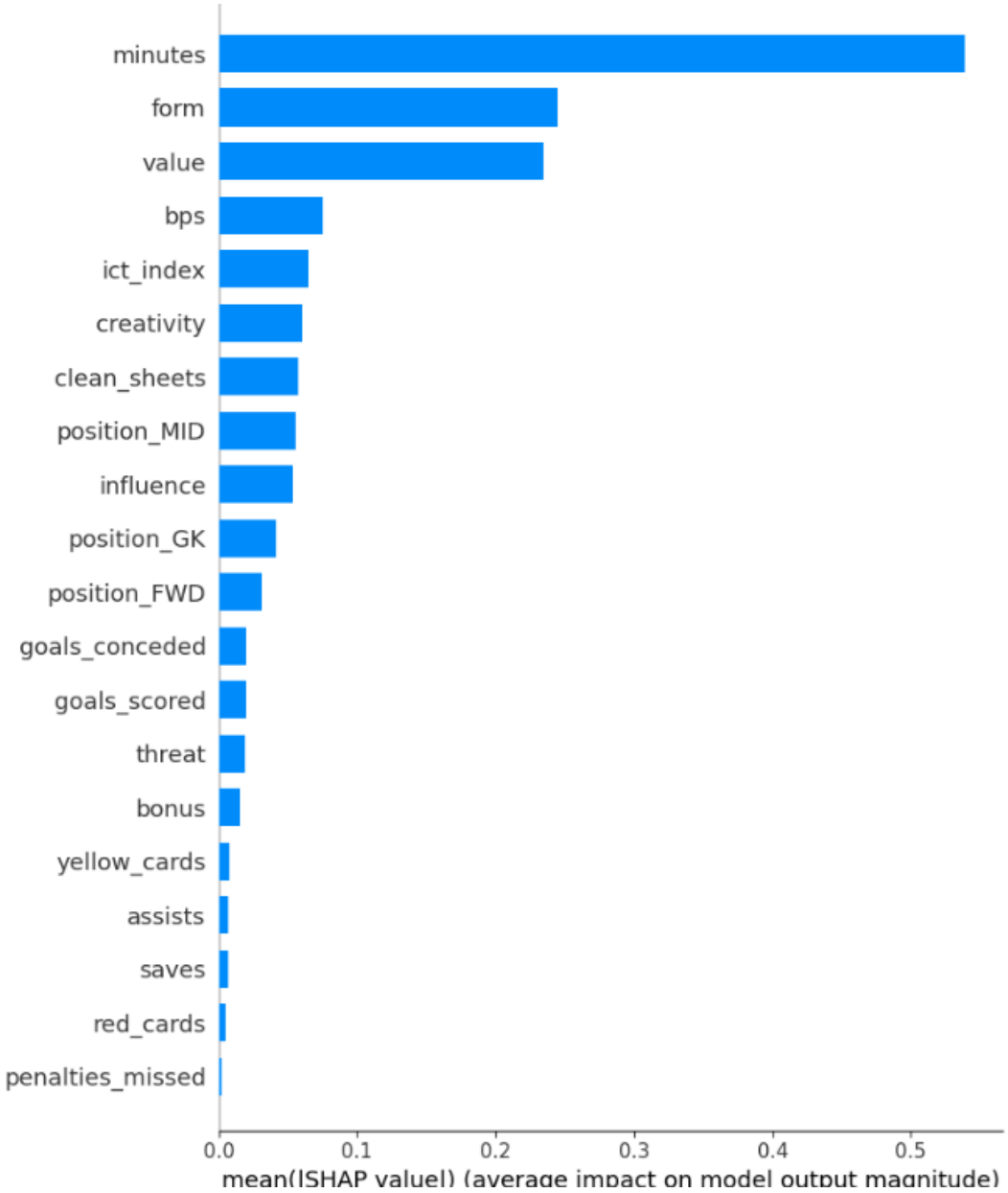


Figure 15: SHAP bar plot for the Overall (FFNN) model.

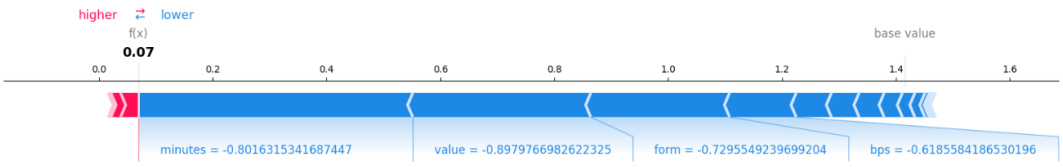


Figure 16: SHAP force plot for a single Overall (FFNN) model prediction.

## 7.2 LIME Analysis (FFNN Models Only)

The LIME analysis was applied to the three feed-forward neural network (FFNN) models to understand how individual predictions were influenced by specific features. LIME gen-

erates interpretable local models that approximate the behavior of the black-box model around a selected instance.

### 7.2.1 Player Model (LIME)

**Description:** LIME shows the contribution of features for a single prediction of the model, what is shown here in the example that red cards and assists are the features contributing for higher total points but the rest of the features are contributing in lowering the total points.and on the right is the values of the features of the single player we are using LIME on. this is a local prediction that may be different from a player to another with the contribution.

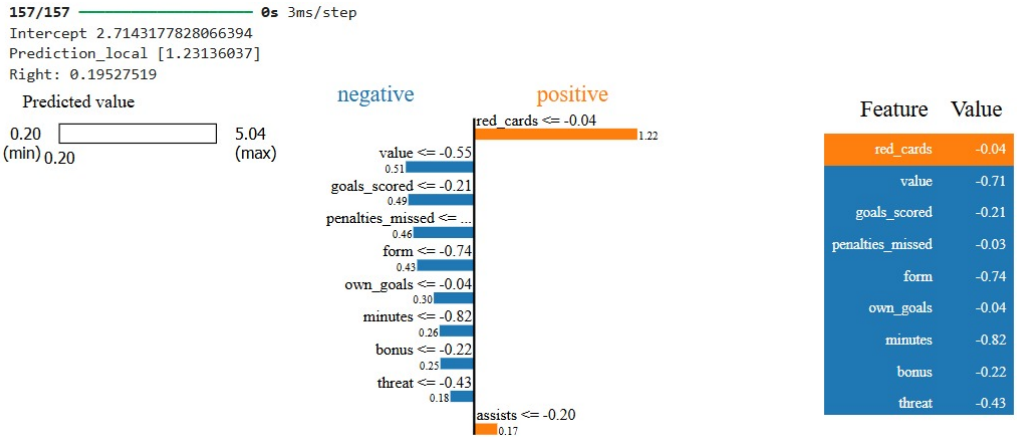


Figure 17: LIME explanation for a selected Player model prediction.

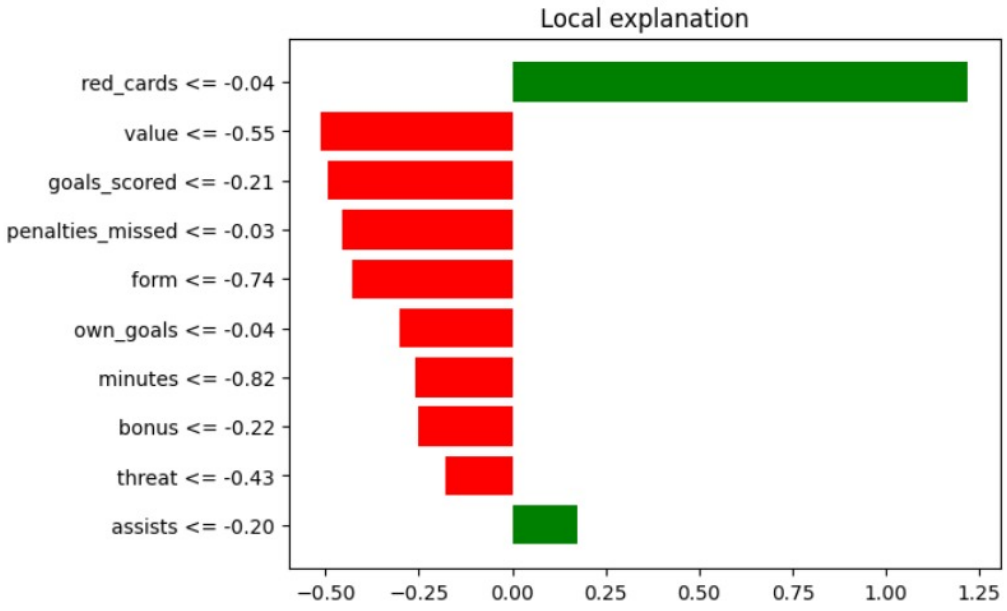


Figure 18: LIME explanation for a selected Player model prediction.

### 7.2.2 Goalkeeper Model (LIME)

**Description:** LIME here is to explain a single prediction of a goalkeeperand the contribution of the featrures on that prediction. it appears that minutes, form, value, bps are the features that contribute positively in making the prediction higher and the rest of the features are acting negatively. This is a local prediction that may be different from a player to another with the contribution.

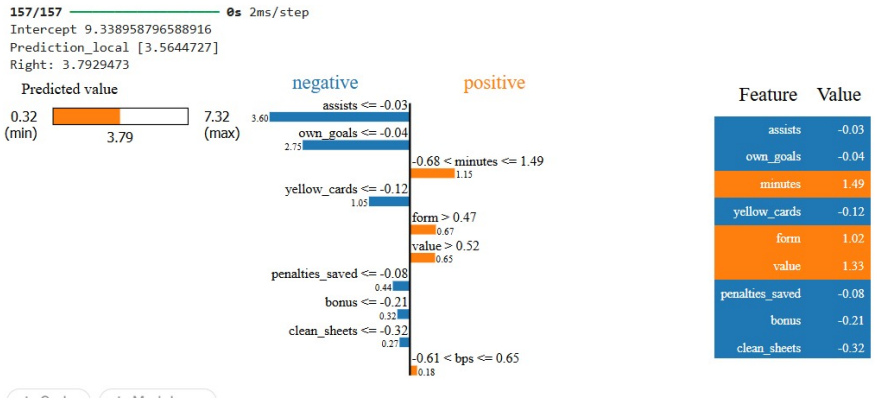


Figure 19: LIME explanation for a selected Goalkeeper model prediction.

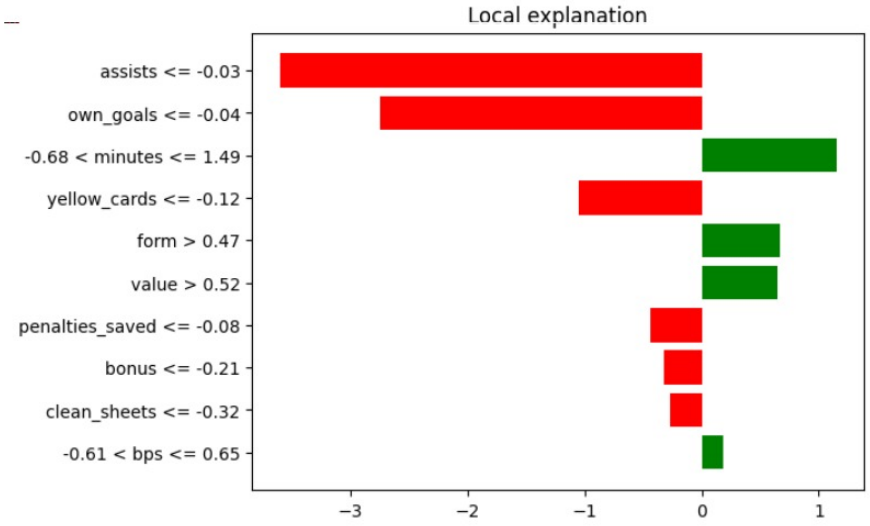


Figure 20: LIME explanation for a selected Goalkeeper model prediction.

### 7.2.3 Overall Model (LIME)

**Description:** LIME here is to explain a single prediction of a player and the contribution of the featrures on that prediction. it appears that red cards, form are the features that contribute positively in making the prediction higher and the rest of the features are acting negatively. This is a local prediction that may be different from a player to another with the contribution.

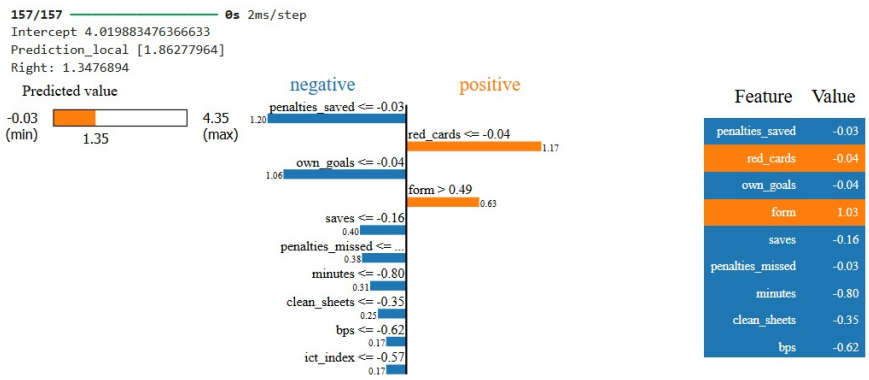


Figure 21: LIME explanation for a selected Overall model prediction.

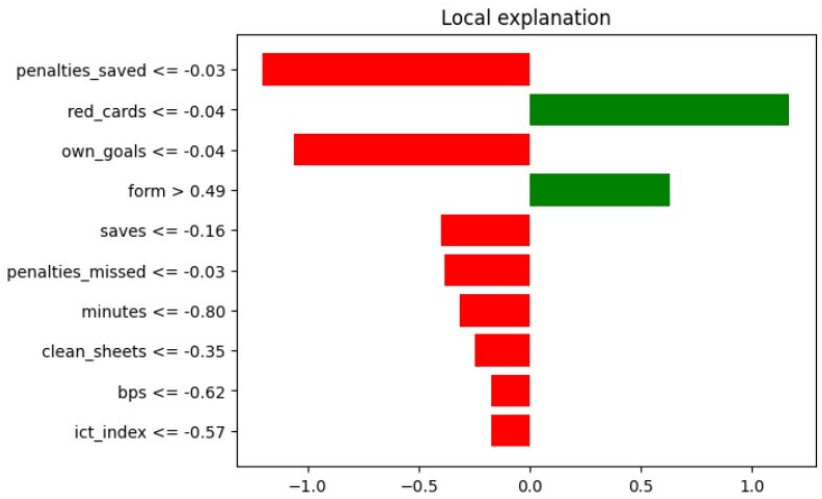


Figure 22: LIME explanation for a selected Overall model prediction.

## 7.3 Summary of Explainability Findings

Summarize the overall findings from SHAP and LIME analyses across all models. Discuss consistent feature importance trends, highlight any discrepancies, and reflect on how explainability supports trust and transparency in the Fantasy Premier League predictive system.

## 7.4 Limitations and Future Improvements

Future enhancements include:

- Position-specific models for DEF, MID, and FWD individually
- Integration of fixture difficulty (FDR), expected goals (xG), and injury data
- Prediction intervals for captaincy risk assessment
- Real-time API deployment with live gameweek updates

**The FPL Assistant now delivers high-accuracy, position-tailored predictions using a practical hybrid of Linear Regression and Neural Networks — optimized for performance, speed, and user trust.**

# 8 Conclusion

This report has presented a comprehensive analytical overview of the Fantasy Premier League Assistant project, demonstrating how data engineering principles and machine learning techniques can be applied to build a practical decision-support system for fantasy sports. Through systematic data exploration, cleaning, feature engineering, and predictive modeling, we have developed a robust framework for forecasting player performance.

**Key Findings and Contributions:**

- **Data Quality and Preparation:** Our data cleaning pipeline successfully handled missing values, categorical inconsistencies, and preserved legitimate double-gameweek entries while removing irrelevant popularity metrics, ensuring a reliable foundation for analysis.
- **Position-Based Insights:** Analysis revealed that midfielders consistently score the largest sum of total points across seasons, followed by defenders, forwards, and goalkeepers. This understanding informed our position-aware modeling approach.
- **Form vs. Cumulative Performance:** The engineered `form` feature effectively captures short-term performance trends that often differ from cumulative season-long performance, providing valuable insights for timely transfer decisions.
- **Hybrid Modeling Strategy:** Our position-specific approach—using Neural Networks for goalkeepers and Linear Regression for outfield players—achieved optimal performance, reducing MAE by 22.3% for goalkeepers while maintaining interpretability for other positions.
- **Explainable AI:** SHAP and LIME analyses revealed that features like `minutes`, `form`, `value`, and `bps` are consistently impactful across models, aligning with FPL scoring mechanics and providing transparent decision support.

**Most Impactful Features:** The analysis identified that match-related features (`goals_scored`, `assists`, `clean_sheets`, `bonus`) and player-related features (`minutes`, `form`, `value`, `ict_index`) show the strongest relationships with point returns. The position-specific relevance of features—such as `saves` and `clean_sheets` for goalkeepers versus `goals_scored` and `assists` for attackers—validated our specialized modeling approach.

**Future Improvements:** Several enhancements could further improve the FPL Assistant:

- **Advanced Features:** Incorporate fixture difficulty ratings (FDR), expected goals (xG), expected assists (xA), and player injury data
- **Model Refinements:** Develop separate models for defenders, midfielders, and forwards to capture position-specific patterns more precisely
- **Real-time Integration:** Implement live API connections for dynamic model updates during gameweeks
- **Uncertainty Quantification:** Add prediction intervals for captaincy risk assessment and transfer decision confidence

- **Multi-objective Optimization:** Incorporate budget constraints and team structure requirements for automated squad selection

The analytical foundation established in this report demonstrates that data engineering and machine learning can transform raw football statistics into actionable insights for FPL managers. By combining statistical rigor with domain knowledge, we have created a system that balances predictive accuracy with interpretability—providing users with transparent, data-driven guidance for their fantasy team decisions.