Using Machine Learning and Data Analytics to Predict Fantasy Premier League Points

Moses Crasto
School of Computing
Dublin City University
Dublin, Ireland
moses.crasto2@mail.dcu.ie

Ananya Das
School of Computing
Dublin City University
Dublin, Ireland
ananya.das5@mail.dcu.ie

Jaslyn D'Souza
School of Computing
Dublin City University
Dublin, Ireland
jaslyn.dsouza2@mail.dcu.ie

Reeve Barreto
School of Computing
Dublin City University
Dublin, Ireland
reeve.barreto38@mail.dcu.ie

Abstract—**This paper investigates the potential of machine learning models to predict Fantasy Premier League (FPL) player points for each gameweek. We explore feature engineering techniques, data pre-processing steps, and compare the performance of various models including Linear Regression, Random Forest, XGBoost, and Neural Networks. The proposed models achieve a Root Mean Squared Error (RMSE) of around 1.97 and a Mean Absolute Error (MAE) of around 1.03, outperforming existing approaches found in the literature. These findings suggest that machine learning can be a valuable tool for informed decision-making in FPL.**

Keywords—Linear Regression, Random Forest, XGBoost, Neural Network, Fantasy Premier League

I. Introduction

Fantasy Premier League (FPL) has become a global phenomenon, captivating millions of football fans through its unique blend of strategy and chance. In this free-to-play online game, participants act as virtual managers, tasked with building a squad of real-life Premier League footballers. The ultimate goal: to collect the most points throughout the season, a feat heavily dependent on the on-field performances of their chosen players. Every goal scored, assist provided, clean sheet kept, and even crucial save contributes to a player's FPL point tally, making informed player selection paramount to success.

Traditionally, FPL managers have relied on a mix of intuition, player popularity, and historical statistics to navigate the complexities of team selection. This often involved meticulously analyzing statistics, scrutinizing past performance trends, and identifying the "hot" players of the moment. However, the ever-increasing availability of detailed player data has opened the door to a new frontier: data-driven player selection strategies powered by machine learning (ML).

The intricacies of the FPL draft process, highlight the strategic considerations managers face due to:

- Positional Requirements: FPL mandates selecting a specific number of players for each position (e.g., Goalkeepers, Defenders, Midfielders, Forwards).
- Team Restrictions: The limit on the number of players chosen from a single Premier League club, preventing managers from stacking their squads with players from a single high-performing team.
- Budgetary Constraints: Each player carries a price tag reflecting their perceived value and recent performance. Managers must carefully navigate this system, balancing high-priced star players with more affordable options to build a well-rounded squad within the total budget constraint.

This study investigates the effectiveness of ML models in predicting FPL player points for each gameweek. We explore various feature engineering techniques to extract meaningful insights from historical player data. We then engineer new features and pre-process the data to ensure its suitability for model training. Finally, we compare the performance of different ML models, including Linear Regression, Random Forest, XGBoost, and Neural Networks, in predicting player points. By evaluating these models through metrics like Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE), we aim to determine the most effective approach for maximizing FPL point prediction accuracy.

This research builds upon existing work in FPL player prediction by incorporating feature engineering to enhance the predictive power of ML models. We compare the performance of our models against existing approaches found in the literature to demonstrate the potential improvement achievable through feature engineering. Our findings aim to contribute to the growing body of research on applying ML to fantasy sports and provide valuable insights for FPL participants seeking data-driven player selection strategies.

II. Related Work

The application of machine learning (ML) in fantasy sports, particularly Fantasy Premier League (FPL), has gained significant traction in recent years. Several studies have explored the use of ML models to predict player points and optimize team selection strategies. Here, we review relevant literature to understand existing approaches and identify potential areas for improvement

M. Bangdiwala et al. [1] utilise Linear Regression, Decision Tree, and Random Forest models for point prediction. The study employs historical data without feature engineering, achieving the best results with Linear Regression. However, the lack of feature engineering might limit the model's ability to capture complex player performance factors. Additionally, the paper does not discuss their data-splitting strategy for training and testing. This is crucial, as a player's performance cannot be accurately predicted using future game data points that were not available during training.

G. Akhil et al. [2] (2017) utilize a hybrid approach combining AutoRegressive Integrated Moving Average (ARIMA) and Recurrent Neural Networks (RNNs) for time series prediction of player points. While their work explores the time series nature of player performance, it primarily focuses on creating the optimal team composition for the entire season, which differs from our objective of predicting individual player points. Nevertheless, this study provided valuable insights into data preprocessing techniques for time series modelling, which were beneficial for this research.

Rajesh et al. [3] focused on enhancing prediction accuracy by incorporating features like player form, return on investment (ROI), fixture difficulty, and points per game (PPG). Their work emphasizes the importance of considering diverse player performance indicators beyond traditional statistics. Notably, they introduced a moving window approach to capture dynamic player form, demonstrating its effectiveness. However, their exploration could be extended by engineering additional features using similar techniques.

These studies demonstrate the potential of ML for FPL player point prediction. However, a common limitation is the lack of emphasis on feature engineering. By creating new features that capture nuanced player performance aspects, we aim to improve the predictive power of ML models.

Machine Learning on Fantasy Premier League [4] explores various ML algorithms like Random Forest, Gradient Boosting, and Neural Networks for predicting FPL player points. Their findings suggest Random Forest outperforms other algorithms. This study provides a valuable reference for model selection, but a detailed exploration of feature engineering techniques is not included.

While existing research demonstrates the feasibility of ML for FPL, there is scope for improvement through advanced feature engineering. Our study aims to address this gap by incorporating feature engineering techniques to extract more comprehensive insights from player data and enhance the accuracy of ML models in predicting FPL player points.

III. Methodology

This section details the methodology employed for developing and evaluating machine learning models to predict FPL player points. Here's a breakdown of the key steps involved:

3.1 Data Acquisition and Preprocessing

The data for this study was obtained from an open-source repository maintained by Vaastav Anand [5]. This repository is updated periodically after each gameweek and contains historical player performance data for the past eight seasons (2016-2024). The dataset includes two categories of player data:

- Overview statistics: Season-specific data for each player.
- Gameweek-specific statistics: Data specific to each player's performance in a particular gameweek.

We first combined the datasets of each season into a single, merged dataset. Missing values in the "position" and "team" columns were addressed using an API.

To gain deeper insights from the player data, a new feature named "result" (indicating the outcome of the game for that player's team) was created. Duplicate team names were then removed before proceeding with feature engineering.

3.2 Feature Engineering

A function was implemented to generate features based on a moving window approach. This approach was used to create new features capturing player performance trends, including:

1. Player Form: This feature captures a player's recent performance by considering their total points and total influence over a rolling window of the last n games (typically 5). Higher values indicate a player in good form, consistently contributing to their team's performance.

$$\text{form} = \frac{\sum_{i=\text{gw-window}}^{\text{gw}} \text{points}_i + \text{ICT Index}_i}{\text{window}}$$

2. Foul Play: This feature captures a player's propensity for negative actions during a gameweek. It's calculated by taking the average number of yellow cards, red cards, goals conceded, and own goals per game over a window of games. Higher values indicate a player who is more likely to incur disciplinary actions or contribute negatively to their team's defensive performance.

$$\text{foul play} = \frac{\sum_{i=gw-\text{window}}^{gw} \text{yellow cards}_i + \text{red cards}_i + \text{own goals}_i}{\text{window}}$$

3. Goal Involvement: This feature measures a player's direct contribution to scoring goals. It's calculated by the average number of goals scored and assists provided per game over the window, further adjusted by a position-specific weight to account for the expected contribution from different positions.

$$\text{goal involvement} = \frac{\sum_{i=\text{gw}-\text{window}}^{\text{gw}} \text{goals}_i + \text{assists}_i}{\text{window}}$$

4. Defensive Contribution: This feature reflects a player's contribution to their team's defensive performance. It's calculated by the average number of clean sheets and saves per game over the window, further adjusted by a position-specific weight to account for the expected defensive contribution from different positions.

$$\text{defensive contribution} = \frac{\sum_{i=\text{gw}-\text{window}}^{\text{gw}} \text{clean sheets}_i + \text{saves}_i}{\text{window}}$$

5. Value for price: This feature represents a player's cost-effectiveness relative to their point production. It's calculated by the average number of points earned divided by their average cost over the window.

$$\text{value for price} = \frac{\sum_{i=\text{gw-window}}^{\text{gw}} (\text{points}_i / \text{value}_i)}{\text{window}}$$

6. Popularity: This feature captures the transfer market interest in a player. It's calculated by the average transfer balance in the game divided by the average number of times the player is selected by managers. Higher values indicate players with a higher transfer demand, potentially reflecting their perceived value.

$$\text{popularity} = \sum_{i=\text{gw}-\text{window}}^{\text{gw}} \frac{\text{transfer balance}_i}{\text{selected}_i}$$

7. Team Form: This feature reflects a team's recent performance. It's calculated by a weighted sum of wins (weighted by 1.5), draws (weighted by 0.5), and losses (weighted by -1) over a window of n games. Higher values indicate teams on a winning streak, while lower values suggest struggling teams.

$$\text{team form} = 1.5\left(\sum_{i=\text{gw}-\text{w}}^{\text{gw}} \text{wins}_i\right) + 0.5\left(\sum_{i=\text{gw}-\text{w}}^{\text{gw}} \text{draws}_i\right) - 1\left(\sum_{i=\text{gw}-\text{w}}^{\text{gw}} \text{losses}_i\right)$$

8. Team Offensive Score: This feature represents a player's team's overall attacking performance. It's calculated by the average number of goals scored by the player's team over the window.

$$\text{team offensive score} = \frac{\sum_{i=\text{gw-window}}^{\text{gw}} (\text{team goals scored}_i)}{\text{window}}$$

9. Team Defensive Score: This feature captures a player's team's overall defensive performance. It's calculated by the average number of goals conceded by the player's team plus the average number of clean sheets per game over the window.

$$\text{team defensive score} = \frac{\sum_{i=\text{gw-window}}^{\text{gw}} (\text{team goals conceded}_i)}{\text{window}}$$

3.3 Feature Selection

We employed correlation analysis with a threshold of 0.75 to identify highly correlated features, which were then flagged for potential removal to mitigate multicollinearity and its impact on model performance. Additionally, we used both Variance Inflation Factor (VIF) analysis (> 5) and Lasso Regression to identify and remove features with high VIF values and those with minimal contribution to prediction accuracy, respectively. This multi-pronged approach ensured a set of informative and non-redundant features for model training.

3.4 Data Preparation for Training

Categorical features like 'season', 'gameweek', 'was_home', 'position', and 'result' were label encoded for ordinal representation to avoid introducing dummy variable traps, while one-hot encoding was applied to categorical features like 'team', 'opponent', and 'next_opponent'.

The preprocessed data was then chronologically split into training and testing sets using an appropriate ratio (80/20 split) to ensure the model doesn't train on future games it's meant to predict. Finally, numerical features in both the training and testing sets were scaled using standard scaling. This scaling normalizes the features and helps ensure all features contribute equally to the model's learning process.

3.5 Machine Learning Model Training and Evaluation

This section details the process of training and evaluating various ML models for FPL player point prediction.

We compared the performance of four machine learning models i.e. Linear Regression, Random Forest Regressor, XGBoost Regressor, and a Neural Network. Each model was trained on the scaled training data.

To assess the performance of each model, we employed two key metrics: Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE). RMSE measures the average magnitude of the difference between predicted and actual values, while MAE measures the average absolute difference between these values. Lower values of both metrics indicate better model performance.

IV. Results and Discussion

This section analyzes the effectiveness of the implemented machine learning models in predicting FPL player points. Here's a table summarizing the key findings:

Model	**RMSE**	**MAE**
Linear Regression	1.9480	1.0726
Random Forest	1.9901	1.1144
XGBoost	1.9674	1.0435
Neural Network	1.9668	1.0214

Table 1: Performance Evaluation of Machine Learning Models for FPL Player Point Prediction.

As observed from Table 1, all machine learning models achieved comparable performance with a slight edge towards XGBoost and Neural Networks. The RMSE values hover around 1.97, indicating an average prediction error of approximately 2 points. Similarly, MAE values range between 1.02 and 1.11, suggesting an average absolute difference between predicted and actual points within this range.

While these results showcase the potential of machine learning for FPL player point prediction, it's important to consider limitations. The inherent randomness and variability in player performance can introduce a degree of uncertainty into the models' predictions. Additionally, the choice of features and hyperparameters can significantly impact model performance.

Despite its simplicity, Linear Regression achieved a competitive RMSE and MAE. This suggests that a linear relationship exists between some of the engineered features and player points. Random forest and XGBoost exhibited slightly higher RSME compared to Linear Regression. However, their ability to capture non-linear relationships between features might contribute to better generalizability to unseen data.

Although achieving the overall lowest RMSE and MAE, the Neural Network architecture might be susceptible to overfitting due to the limited training data. Further exploration with hyperparameter tuning and regularization techniques could potentially improve performance.

It's important to acknowledge that while the achieved RMSE and MAE values indicate a reasonable prediction accuracy, there is still room for improvement. Future work could explore alternative feature engineering techniques, hyperparameter tuning for the models, or incorporating additional data sources like fixture difficulty or player sentiment analysis.

4.1 Comparison with Existing Work

Several studies have explored machine learning for FPL player point prediction. However, many rely on basic statistical features without extensive feature engineering. Our approach, incorporating a wider range of features along with advanced techniques like sliding window analysis, contributes to improved model performance. Our results demonstrate that XGBoost and Neural Networks when combined with feature engineering, achieve a competitive edge over simpler models found in the existing literature.

V. Limitations and Future Work

The limitations of this study include the reliance on historical data, which may not perfectly reflect future performance due to factors like injuries or transfers.

Future work could explore the incorporation of additional data sources, such as injuries, weather conditions, and opponent strength to further enhance prediction accuracy.

Additionally, investigating advanced deep learning architectures specifically designed for time series forecasting like Long Short Term Memory (LSTM) or Recurrent Neural Networks (RNN) could potentially yield even better results.

VI. Conclusion

This study investigated the potential of machine learning models for predicting Fantasy Premier League (FPL) player points for each gameweek. We employed feature engineering techniques to create informative features capturing player performance nuances. The pre-processed data was then used to train and evaluate various machine learning models.

For the FPL player points prediction task, based on the similar performance across models and the benefits of interpretability, Linear Regression offers a compelling choice. It provides valuable insights into feature importance while achieving competitive accuracy.

The code and the datasets are available on GitHub for further reference [6].

VII. References

[1] "Using ML models to predict points in Fantasy Premier League," IEEE Conference Publication | IEEE Xplore, Aug. 26, 2022. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9909447

[2] A. Gupta, "Time series Modeling for Dream Team in Fantasy Premier League," arXiv.org, Sep. 19, 2019. https://arxiv.org/abs/1909.12938

[3] "Player Recommendation System for Fantasy Premier League using Machine Learning," IEEE Conference Publication | IEEE Xplore, Jun. 22, 2022. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9836260

[4] D. Mandrasa T., "Machine Learning on Fantasy Premier League," Machine Learning on Fantasy Premier League. https://rstudio-pubs-static.s3.amazonaws.com/577042_d4492e2e511848fb97ef839be077e9b8.html#content

[5] Vaastav, "GitHub - vaastav/Fantasy-Premier-League: Creates a .csv file of all players in the English Player League with their respective team and total fantasy points," GitHub. https://github.com/vaastav/Fantasy-Premier-League/tree/master

[6] Reeveboy. GitHub - reeveboy/fantasy-premier-league-analytics. GitHub. Retrieved from https://github.com/reeveboy/fantasy-premier-league-analytics/tree/main