Football Player Market Value Prediction

Zhansaya Akhmetova	3259293
Albina Cyuzuzo Ntivuguruzwa	3284830
Martina Favata	3258750
Jakob Lutz	3253325
Martina Russo	3213229

1 Introduction

In the world of professional football, a player's market value reflects not only their current performance but also their potential, reputation, and perceived contribution to a club's success. In this project, we aim to predict the market value of players in Euros (value_eur) using a rich dataset of over 15,000 footballers and 76 features that describe their skills and more. The primary goal is to build a machine learning model capable of estimating player value based on measurable characteristics, with performance evaluated through Root Mean Squared Error (RMSE) on unseen test data. We begin by exploring and cleaning the data, identifying the most informative features, and examining their correlations with market value. We then test and compare several modeling approaches, starting with linear regression and moving to more advanced algorithms such as Random Forests and LightGBM. The final report presents our methodology, findings, and model performance, providing insight into what drives player valuation in the football economy.

2 Data Processing

2.1 Data Cleaning

To ensure the dataset was reliable and suitable for modelling, we carefully implemented a thorough data cleaning procedure. Our primary focus was on addressing missing values and removing unnecessary features. We began by examining the number of missing values in each column for both the training and test sets, paying close attention to the importance of each feature in predicting a player's market value. If a column had more than 50% missing values and was not crucial for our analysis, we removed it from the dataset. Missing numerical entries were filled using the median value for imputation, as this approach helps to minimise the effect of outliers and to preserve as much information as possible. Meanwhile, missing values were replaced by the label `unknown` for categorical features to avoid introducing bias. We also checked for consistency in data types across all columns and removed any duplicate records. In particular, we confirmed that the training and test sets included the same features, except for the target variable `value_eur`, which only appeared in the training set. In this way, we tried to ensure that our model would function correctly when applied to new data.

2.2 Distribution Analysis

After cleaning the data, we examined the main numerical features using box plots and histograms to better understand their distributions and potential impact on modelling. Using both visualizations gave us a fuller picture: histograms showed us the general shape and skeweness of the features, while box plots quickly highlighted the presence of outliers and the spread of these values. Although features such as age, overall rating, and potential were more evenly distributed, we noticed a much broader spread when looking at the value_eur feature, which represents each player's estimated market value in euros. In this case, the graphs made it easy to see that player market values vary a lot in professional football. Most players have moderate values, but there are a few top players whose values are much higher than the rest. By integrating these observations with our cleaning process, we aimed to develop a model that accurately represents the structure of the football world.

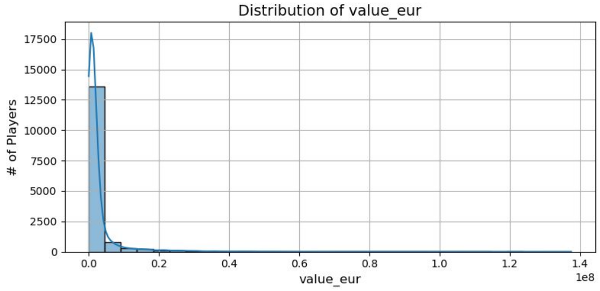

Figure 1: Histogram of value_eur distribution

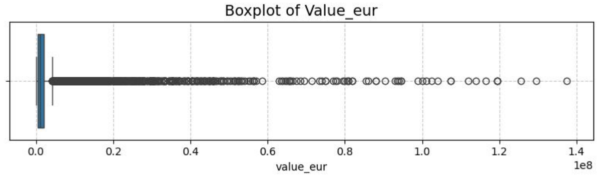

Figure 2: Boxplot of value_eur distribution

2.3 Correlation Matrix

To further improve model focus and reduce potential noise, we removed a small set of features deemed uninformative or redundant for prediction. These included identifiers (`id`), raw text fields (`long_name`, `short_name`), and difficult-to-use date fields like `dob` and `club_joined`. We also excluded high-cardinality categorical variables such as `nationality_name`, which would require extensive encoding but offer limited predictive contribution in a baseline model. After this refinement, the final shape of our cleaned datasets is: Train: (15,391, 64) and Test: (3,848, 63), ready for feature analysis and modeling. To understand the relationship between features and the target variable `value_eur`, we computed a full correlation matrix and selected the top 40 most positively correlated features. However, some of these features were also strongly correlated with each other, a condition known as multicollinearity. Multicollinearity can destabilize coefficient estimates in linear models like Linear Regression, reducing model interpretability and generalization. To address this, we applied a filtering step that drops one of any two features with a correlation above 0.80. This step was only applied when using linear models. For tree-based models like Random Forest or LightGBM, which are robust to multicollinearity, we retained the full feature set without reduction.

3 Previous Models

3.1 Linear Regression Model

To establish a baseline for model performance, we trained a Linear Regression model using a refined set of numerical features selected for their strong correlation with the target variable `value_eur` and their overall predictive relevance. Originally, we considered 23 features but excluded `club_contract_valid_until` due to its non-numeric format, bringing the final model input to 22 columns. Using this feature set, the model was trained on a cleaned dataset of 15,391 player records and evaluated using a validation split. The model produced a Root Mean Squared Error (RMSE) of approximately 1,382,405 and an R^2 score of 0.9614, indicating that the model is able to explain over 96% of the variance in market value within the validation set. While these results appear strong numerically, they should be interpreted with caution. Linear Regression assumes a linear relationship between all predictors and the target variable. This high-bias structure limits the model's flexibility and its ability to capture the more complex, non-linear interactions that likely exist between a player's traits and their market value. As a result, the model risks underfitting—failing to fully leverage the richness of the data—and may perform poorly on unseen or edge cases. Additionally, Linear Regression is sensitive to multicollinearity, which can cause instability in coefficient estimates when correlated features are present. For these reasons, we use Linear Regression primarily as a baseline and turn to more flexible, non-linear models such as Random Forest and LightGBM to improve predictive accuracy and robustness.

3.2 Random Forest Regression Model

The Random Forest Regressor model was developed to predict player market value. Unlike Linear Regression, which assumes linearity between predictors and the target variable, Random Forest is a non-linear, ensemble-based approach that better captures complex interactions within the data. We trained the model using a set of numerical features after filtering out non-numeric columns, and evaluated performance using cross-validation with five folds. The model achieved a cross-validated Root Mean Squared Error (RMSE) of approximately 889,047.58, significantly outperforming the baseline Linear Regression model (RMSE of 1,382,405). This improvement demonstrates Random Forest's ability to reduce bias and capture non-linear relationships between player attributes and their market value. While the reduction in RMSE suggests a more accurate model, it is essential to consider the model's tendency to overfit, especially given the potential for high variance with the inclusion of a large number of features. Additionally, the Random Forest model is less interpretable compared to simpler linear models, making it challenging to identify the individual impact of each feature on predictions. Despite these limitations, the superior RMSE highlights the model's enhanced predictive performance and robustness when dealing with non-linear data structures. Our results could have been more accurate, considering all the factors. Therefore, we decided to change the model to LightGBM.

4 Final Model

To achieve the best possible performance, we employed a **gradient boosting framework (LightGBM)** combined with **hyperparameter optimization (Optuna)** and advanced **feature engineering** techniques. The main goal was to develop a model that not only minimizes the **prediction error** but also **generalizes well** without overfitting. This is particularly challenging given the **noisy and skewed nature of football player market valuations**.

4.1 Target Transformation

One of the key challenges in modeling player market values is handling the **highly right-skewed distribution**. A few players have exceptionally high market values, while the majority have relatively lower values. Predicting these skewed targets directly can cause the model to prioritize minimizing errors on high-value players, neglecting those with lower market values. To address this, we applied a `log1p` transformation to the target variable, which stabilizes variance and makes the distribution more **Gaussian-like**. Specifically, the original value y was transformed as:

$$y_{\log} = \log(1 + y)$$

This transformation improves model performance by reducing the influence of **outliers**. During inference, we **revert predictions to the original scale** using the `exponential` function, maintaining the interpretability of results. This transformation significantly reduces the effect of extreme values, allowing the model to make more **balanced predictions** across the range of player market values.

4.2 Cross-Validation

To ensure the model's robustness and to avoid overfitting, we used **five-fold cross-validation**. The dataset was divided into five subsets: one subset was used as the **validation set** while the other four formed the **training set**. This process was repeated five times so that each subset served as the validation set exactly once. The chosen evaluation metric was **Root Mean Squared Error (RMSE)** on the **log-transformed target**. RMSE was selected because it **penalizes larger errors more heavily**, making it suitable for evaluating models predicting high-value targets. A **low cross-validation RMSE** indicates that the model balances **bias and variance well**, implying that it is neither too simplistic (underfitting) nor overly complex (overfitting).

4.3 Feature Engineering

Feature engineering played a vital role in improving model accuracy. The goal was to create **higher-level features** that reduce dependency on **raw, noisy variables**. We derived the following new features:

- **Total Skill**: Aggregate of passing, shooting, and dribbling skills.
- **Attacking Skill**: Combined metric of finishing, volleys, and positioning.
- **Defending Skill**: Incorporates interceptions, defensive awareness, and standing tackle.
- **Goalkeeping Skill**: Summarizes key goalkeeper statistics like diving, handling, and kicking.
- **Age Squared**: A nonlinear transformation to account for the **diminishing returns** of age on market value.

These features capture **complex player traits** more effectively, improving the model's ability to predict market values accurately.

4.4 Modeling with LightGBM

We selected **LightGBM** for its outstanding performance in gradient boosting tasks. Its key strengths include:

- **High efficiency**: Handles large datasets swiftly.
- **Categorical support**: Natively processes categorical variables.
- **Regularization techniques**: Uses **L1 and L2 penalties** to mitigate overfitting.
- **Non-linear modeling**: Capable of capturing complex relationships between features and the target variable.

4.5 Hyperparameter Tuning with Optuna

To optimize the model, we used **Optuna**, an automatic hyperparameter optimization framework. It leverages **Bayesian optimization** and advanced sampling methods to efficiently explore the parameter space. The following hyperparameters were tuned:

- **Learning rate**: Controls the step size in each iteration.
- **Number of leaves**: Influences the model's complexity.
- **Maximum depth**: Restricts tree depth to prevent overfitting.
- **Subsample ratio**: Determines the fraction of data used for each iteration.
- **Column sample by tree**: Selects features randomly to enhance generalization.
- **Regularization terms (L1 and L2)**: Prevent overfitting by adding penalty terms.

Optuna iteratively evaluated parameter combinations via **cross-validation**, and the best configuration was chosen based on **minimized RMSE**. We set the maximum number of trials to **30**, balancing computational cost and model accuracy.

4.6 Model Performance

During the tuning process, the LightGBM model reached its best iteration with the following results:

- **Training RMSE**: 0.0160901
- **Validation RMSE**: 0.058501

The small gap between training and validation RMSE indicates that the model generalizes well without significant overfitting.

After applying the **inverse transformation** to the predictions, the calculated **real-world RMSE** is 597,195.11 Euros, representing a significant improvement over previous models. This final model demonstrates a strong ability to **generalize well** to unseen data. Figure 3 shows that the predicted player values closely align with the actual values across the dataset.

By integrating **data cleaning**, **feature engineering**, **advanced ensemble modeling (LightGBM)**, and **automated hyperparameter tuning (Optuna)**, we achieved a robust model capable of accurately predicting player market values. The final **RMSE of 597,195.11 Euros** marks a significant improvement from the previous, more naive approaches.

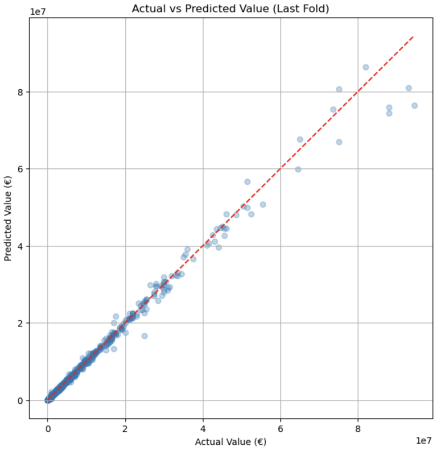

Figure 3: Actual vs Predicted Player Values

4.7 Model Interpretation

To better understand our LightGBM model's predictions, we performed a SHAP (SHapley Additive exPlanations) analysis. The SHAP summary plot in Figure 4 highlights that features such as `release_clause_eur`, `overall`, and `potential` have the greatest influence on the player value predictions in our model. By leveraging SHAP values, we can quantitatively assess the contribution of individual features to the final prediction, allowing for a transparent interpretation of the model's decision-making process. This insight helps validate model assumptions and identifies key factors influencing player market value.

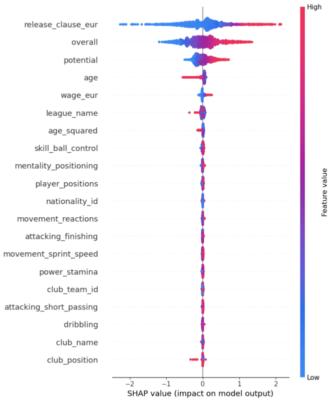

Figure 4: SHAP Summary Plot for Feature Impact

5 Conclusion

In this project, we developed a machine learning model to predict the market value of professional football players using a wide set of characteristics. We began by cleaning the data to ensure consistency and informativeness across different models. By examining distributions and correlations between features, we prepared a structured and meaningful dataset. Our initial model, based on Linear Regression, provided a foundational benchmark but exposed its limitations in capturing complex, non-linear relationships.

To address these limitations, we transitioned to more powerful tree-based models—first Random Forest, which significantly reduced RMSE, and ultimately LightGBM, which delivered the best performance. We enhanced our model by implementing target transformation, conducting careful feature engineering, and performing hyperparameter tuning with Optuna to extract meaningful patterns and minimize prediction error. Our final model achieved a cross-validation RMSE of approximately 597,195.11€, marking a substantial improvement over more naive approaches. This result underscores the effectiveness of ensemble learning, feature engineering, and targeted parameter optimization in predicting values for new, unseen data. Throughout this project, we not only aimed to produce strong predictive results but also provided insights into the factors influencing player valuation in the dynamic football economy.