# Predicting Player Potential using FIFA Attributes: A Comparison of Neural Networks and Benchmark Models

**GitHub Repository**

Jonas Schrade (01/1080887) - Niklas Bacher (01/1081503)

## Introduction

Data-driven player evaluation has become a central tool in modern football analysis. This project uses the European Soccer Database with a focus on player-level data, particularly attributes derived from the FIFA video game series. These ratings summarize players' technical skills, physical traits, and overall ability as assessed by professional scouts. Although originating from a simulation game, FIFA attributes are widely used as standardized measures of player quality and have been shown to correlate strongly with real-world performance. They therefore provide a practical and accessible information source for player evaluation.

Predicting player performance based on these attributes has clear practical relevance. Clubs can use such predictions to inform recruitment, contract decisions, and talent development, while analysts may employ them to compare players across leagues and seasons. Understanding the predictive power of FIFA attributes is thus directly relevant for real-world decision-making in professional football.

The aim of this project is to examine whether complex models such as deep neural networks outperform simpler statistical learning approaches in linking player characteristics to potential ratings. Player potential is a latent concept reflecting expected future development rather than directly observed outcomes. More flexible models may better capture position-specific differences not explicitly encoded in the data. To test this, a standard benchmark model is compared to a neural network using identical predictors and outcomes, allowing an assessment of whether deep learning adds value beyond simpler models in identifying high-potential players.

## Analysis

### Data Preparation

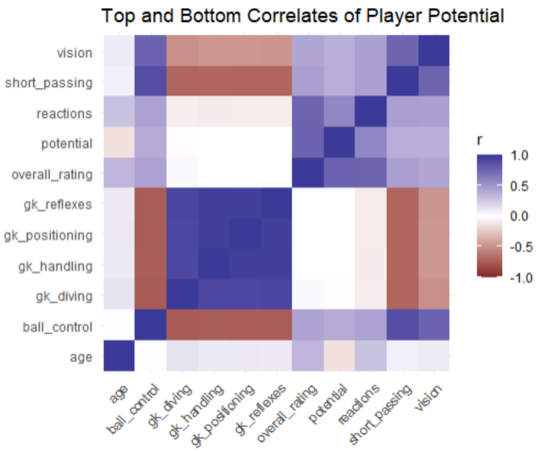
Several preprocessing steps were applied prior to analysis. Measurement dates and birthdates were used to compute each player's age in full years at the time each attribute snapshot was recorded. Players younger than 16 were excluded to remove implausible or non-comparable observations identified through manual screening. Non-informative identifiers and metadata, such as player names, internal IDs, and FIFA-specific API keys, were dropped to retain only predictive variables.

Observations with missing values were removed to ensure compatibility with the modeling framework. Overall missingness is low and limited to a small number of technical attributes. Given the large sample size and the focus on prediction rather than inference, a complete-case approach was used. When modeling player potential, the overall rating was excluded from the feature set to avoid mechanically related information. The final dataset consists of 178,364 observations and 34 features from 10,582 players.

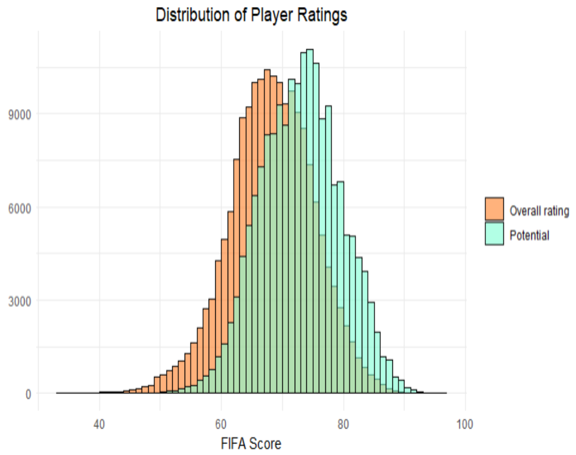
### Variable Structure and Correlations

The dataset mainly consists of continuous numeric predictors bounded between 0 and 99, reflecting FIFA's attribute scoring system, along with a small number of discrete or binary variables such as preferred foot. Correlation analysis, which can be found in the code document for reference, shows that player potential is strongly related to technical and cognitive attributes such as vision, short passing, reactions, and ball control.

In contrast, goalkeeper-specific attributes exhibit weak or negative correlations with potential in the full sample, reflecting the strong positional structure of the data. The correlation heatmap in Figure 1 highlights this divide and suggests that interactions between position and skill attributes may be relevant for prediction, even when position is not explicitly modeled.



(a) Figure 1: Correlations among relevant predictors and potential score



(a) Figure 2: Distribution of overall and potential player ratings

### Distributions of Attributes and Ratings

The marginal distributions of player attributes are generally unimodal and concentrated around mid-to-high values. Goalkeeper attributes display different distributions, with substantial mass at low values, reflecting the small share of goalkeepers in the dataset. The preferred foot variable is highly imbalanced, with right-footed players dominating.

The distributions of overall rating and potential (Figure 2) highlight the difference between current performance and expected future ability. Although closely related, potential is systematically higher than overall rating, indicating that many players are assessed as having greater long-term upside. This gap motivates the focus on potential as a separate prediction target and underscores the challenge of modeling a concept that is inherently forward-looking and position-dependent.

# Methods

## Neural Network

The neural network follows a sequential architecture consisting of an input layer matching the feature dimensionality, three fully connected hidden layers with ReLU activations, and a single-unit output layer with linear activation. Each hidden layer incorporates $\ell_2$ (ridge) regularization with a decay of 0.001, batch normalization, and dropout to constrain weights, accelerate convergence, and enhance robustness, respectively.

For model training, features are organized in a numeric design matrix to ensure consistent encoding of categorical variables, with the target variable stored separately. Training and evaluation employ nested, group-aware cross-validation, grouping folds by unique player identifiers to prevent cross-sectional data leakage. Within each training fold, hyperparameters are selected via grid search using an internal train-validation split. The grid varies hidden-layer sizes, dropout rates, and the learning-rate annealing factor (two values each; reported in the code document), while batch size (128), maximum epochs (50), and early stopping with a patience of five epochs are held constant across all folds and configurations. The Adam optimizer with a default initial learning rate of 0.001 is used throughout, enabling fast and stable convergence. This approach balances robust model validation with computational efficiency and reproducibility.

Before training, 20% of the training data are held out as a validation set to guide optimization and hyperparameter selection. The normalization scaler is fitted exclusively on the inner training set and subsequently applied to the validation and test sets.

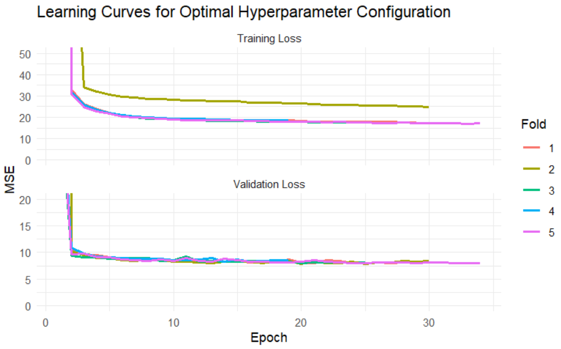


Figure 3: Learning curves of the optimal hyperparameter configuration across five outer folds.

Across five folds, 128 hyperparameter combinations are evaluated, resulting in 640 training runs. The learning curves of the optimal configurations exhibit a sharp initial decline in both training and validation loss, followed by a gradual decrease and a clear plateau in the validation loss around epoch 30, which triggers early stopping. The convergence patterns, early-stopping points, and final validation losses are similar between folds, indicating robust and stable training behavior under different data splits. Notably, in fold 2 the training loss stabilizes more rapidly after the initial decline, suggesting some greater structural complexity in the training data. Finally, the optimal model is tested against the test fold with evaluation metrics reported below.

### Baseline model: Lasso Regression

As a transparent benchmark, we estimate a linear prediction model for player potential using Lasso regression. The Lasso extends OLS by adding an $\ell_1$ penalty that shrinks coefficients toward zero and sets some exactly to zero, thereby regularizing the model and performing variable selection. As in the neural network setup, all predictors are converted into a numeric design matrix. Predictors are standardized during estimation so that the penalty treats all features comparably across scales.

Model performance is evaluated using grouped 10-fold cross-validation. As before, folds are constructed at the player level to prevent information leakage across training and test sets. We use 10-fold cross-validation instead of LOOCV for computational reasons, as 10-fold CV is known to provide a favorable bias-variance trade-off while being substantially less computationally intensive.

Within each training fold, the regularization parameter $\lambda$ is selected via internal cross-validation over a grid of candidate values. The model is then refit using the selected $\lambda$, and predictive performance is assessed using MSE and MAE on training and held-out test data.

Finally, we conduct diagnostic checks for the linear baseline, including assessments of linearity, homoskedasticity, and the residual distribution. While these conditions are not strictly required for predictive performance, particularly in the case of normally distributed errors, the diagnostics indicate that they are reasonably well satisfied and provide useful context for comparing the linear model to more flexible approaches.

## Results

Averaged across cross-validation folds, the evaluation metrics indicate strong predictive performance for both models, with mean absolute test errors of approximately 2 and 3 potential points, respectively. Additionally, for both models the training loss is only marginally below the test loss, providing no indication of overfitting.

| K-Average Evaluation Metrics | Train MSE | Test MSE | Train MAE | Test MAE |
|---|---|---|---|---|
| Neural Network | 7.36 | 7.91 | 1.94 | 2.01 |
| Lasso Regression | 15.83 | 15.90 | 3.07 | 3.08 |

Nevertheless, the neural network consistently outperforms the baseline regression model on both the training and held-out test sets. Although the difference in mean absolute error is relatively modest, the regression model exhibits a mean squared error that is nearly twice as large as that of the neural network, indicating that a small number of extreme prediction errors disproportionately drive its overall deviation. This suggests that the neural network is less sensitive to such extreme outliers, likely due to its capacity to model nonlinear relationships and interactions as well as the regularization and early-stopping mechanisms that constrain overfitting to atypical observations. Despite its higher computational costs, these results indicate that the neural network constitutes the preferable approach for predicting player potential.

## Reflection & Limitations

While the neural network appears better suited to modeling complex relationships between player attributes, age, and potential, this flexibility does not translate into superior predictive performance. Accordingly, our expectation that the regression model would be more strongly disadvantaged by the omission of positional interactions is not supported. A qualitative comparison of the ten players with the largest average overprediction and underprediction errors across both models does not reveal systematic differences by position, as no position predominates in either group. However, more granular positional information at the player year level would be required to enable a more thorough quantitative assessment of such underlying complexity. Nevertheless, this is consistent with the idea that neural networks can flexibly capture complex and high-dimensional patterns. Such patterns, however, may not translate into clear group-level regularities and are often difficult for researchers to identify or specify in parametric models.

Furthermore, the study design does not permit predictions of players' career development, as both the modeling approach and data splitting are strictly cross sectional. Consequently, no inferences can be made about individual career trajectories or whether a player ultimately realizes their true potential. Each player is treated solely as a unit of observation, with potential assessments derived from pooled historical data across multiple generations of players. This cross sectional inference further assumes temporal stability in the relationship between player features and potential, both in real world development processes and in FIFA's underlying assessment methodology across time.

A final limitation concerns the restricted amount of data available to train the neural network within each cross validation fold. Although each training fold contains four fifths of the dataset, the additional inner validation split reduces the effective training sample to 60 percent of the full data. We therefore did not retrain the model on the complete training fold in order to retain protection against overfitting via validation based early stopping, which may have resulted in some performance loss. Future work could instead refit the model on the full training fold while fixing the number of training epochs, as no early indications of overfitting were observed.