Research Article ISSN Online: 2783-4859 Computational Mathematics and Computer Modeling with Applications

Received 13 June 2023
Accepted 24 October 2023

DOI: 10.48308/CMCMA.2.1.24
AMS Subject Classification: 68-XX; 68Txx

Positioning Soccer Players for Success: A Data-Driven Machine Learning Approach

Mahdi Nouraie[a] **and Changiz Eslahchi**[b]

Determining a player's proper position in football is critical for maximizing their impact on the field. In this study, we propose a scientific and analytical approach to address this issue using machine learning models. We use the FIFA dataset to identify the correct positions for players and show that the logistic regression model provides the most accurate predictions, with an average accuracy of 99.84% on test data across the all positions. To further refine player positioning, we use the Recursive Feature Elimination (RFE) method to identify the most important features associated with each position. The top five features identified through RFE are used to evaluate players' suitability for their correct positions and we illustrate that the average Mean Squared Error (MSE) is 1.166 on a scale of 100, indicating high accuracy in predicting their suitability scores. Overall, our results suggest that the logistic regression model is an effective tool for accurately determining player positions, and that the selected features can be used to evaluate players' suitability for a given position with high accuracy. Our approach provides a data-driven solution to help teams make better decisions in player selection and positioning, potentially leading to improved team performance and success.

Keywords: Football tactical analysis; Team formation; Player positioning; Football team composition, Machine learning.

1. Introduction

Determining the best position for a player in football is a critical aspect of team success. Traditionally, football coaches rely on their intuition and experience to evaluate players and assign them to positions. However, with the advancements in machine learning, there is an opportunity to use data-driven approaches to help teams make better decisions in player selection and positioning.

Previous research has explored different methods for determining player positioning and team formation. Tavana et al. (2013) proposed using fuzzy logic to determine player positioning and team formation [7]. They introduced weights for the performance of each forward, midfielder, and defender positions. Abidin (2021) combined these weights with machine learning techniques to develop an algorithm for selecting the best players for the three field positions [1]. Uzochukwu and Enyindah (2015) used neural networks to determine player performance based on four feature groups: physical, technique, speed, and resistance [3]. Evwiekpaefe et al. (2020) demonstrated that artificial neural network models are capable of identifying forward players [4]. Garcia-Aliaga et al. (2021) used the t-SNE[†] and UMAP[‡] algorithms to describe player positions based on their features, while Frey et al. (2019) used player tracking and GPS data to assign players to positions using convolutional neural networks (CNN), random forest, and gradient boosting XGB [6, 5]. Apostolou and Tjortjis (2019) used machine learning algorithms to predict player positions with an accuracy rate of 81.5% [2].

Despite the progress made in previous research, there is still a need to improve the accuracy and effectiveness of player positioning models. In this study, we propose a data-driven approach to accurately determine a players' optimal positions by utilizing machine learning algorithms on the FIFA dataset. We aim to identify the key attributes that contribute to a players'

[a] *Department of Statistics, Shahid Beheshti University, Tehran, Iran,*
[b] *Department of Computer and Data Sciences, Shahid Beheshti University, Tehran, Iran.*
[*] *Correspondence to: C. Eslahchi. Email: ch-eslahchi@sbu.ac.ir.*

[†]t-Distributed Stochastic Neighbor Embedding is a statistical nonlinear unsupervised dimension reduction method used for data exploration and visualization. In simpler terms, t-SNE provides the user with an understanding of how data is organized in a high-dimensional space. This method was introduced by Laurens van der Maatens and Geoffrey Hinton in 2008.
[‡]UMAP is a dimensionality reduction technique that can be used to visualize data in a low dimension similar to t-SNE; This method can be used for non-linear dimension reduction. This method was introduced by Leland McInnes, John Healy and James Melville for the first time in 2018.

success in their positions and provide a practical tool for coaches and teams to make informed decisions on player selection and positioning.

In this paper, we aim to develop a model that evaluates players' suitability to play in any football position based solely on their physical and technical attributes using a classic and publicly available dataset. Such a model can provide coaches and trainers with a better understanding of players' skills, which can aid in player development and team strategy.

Previous research has focused on classification problems with multiple classes, but no study has considered all of the football positions. Adjacent positions on the football field require comparable skills, so it is unjustifiable to assume that a player can only play at one position and not at others.

To achieve our goal, we will compare the results of support vector machine, random forest, deep neural network, and logistic regression models. Additionally, we will employ a feature selection procedure for each position using the Recursive Feature Elimination (RFE) method based on the random forest algorithm. This procedure will help us identify the most important characteristics for each position.

Our goal is to create a model that coaches and team analysts can use to comment on players' qualities in a scientific way. This can aid in player development and team strategy, ultimately leading to better team performance.

2. Materials and Methods

2.1. Dataset

This study utilized the FIFA dataset that EA Sports releases annually and makes available online[§]. The dataset contains information on the vast majority of adult male and female soccer players worldwide, as well as their scores in various soccer-related abilities (such as shooting, passing, ball controlling, heading, etc.), contract information, etc, totally 110 attributes. Since the purpose of this study is to evaluate players solely on the basis of their physical and technical attributes, we only consider the players' physical and technical skill-related variables. 44 of the 110 variables in this dataset are related to the technical skill, age, height, and weight of the player. The other variables were not considered. On the eddwebster GitHub website, one can access and download the archive of this dataset from the 20142015 season to the 20212022 season[¶]. Several helpful variables were added to this dataset since 2016 that had not been included in previous years. Therefore, we considered data from the 20162017 season through 20212022. For each season, each player in this dataset is assigned a club position, indicating the position in which they appeared most frequently.

There are 29 positions in this data set: LWB, LB, LCB, CB, RCB, RB, RWB, LM, LCM, CM, RCM, RM, LDM, CDM, RDM, LAM, CAM, RAM, LW, RW, LS, ST, RS, LF, CF, RF, GK, RES, SUB. (Left Wing Back, Left Back, Left Center Back, Center Back, Right Center Back, Right Back, Right Wing Back, Left Midfielder, Left Center Midfielder, Central Midfielder, Right Center Midfielder, Right Midfielder, Left Defensive Midfielder, Central Defensive Midfielder, Right Defensive Midfielder, Left Attacking Midfielder, Central Attacking Midfielder, Right Attacking Midfielder, Left Winger, Right Winger, Left Striker, Striker, Right Striker, Left Forward, Central Forward, Right Forward, Goal Keeper, Reserve, Substitute). We exclude the players with no assigned position, goalkeepers, reserves, and substitutes.

There were fewer than 100 players at each of the following positions in the dataset compiled from six consecutive seasons: LF, CF, RF, LAM, and RAM. In recent years, there has been a shift in football position terminology, which has led to this issue. After excluding these positions from the analysis, we ultimately determined a total of 21 standard positions. The most recent season of a soccer player is used as a benchmark because they are included in the dataset for multiple seasons. Consequently, the final collection contains information on 18,034 players and 48 attributes. Table 1 contains the names and brief descriptions of the dataset variables utilized in this investigation.

Table 1 demonstrates that the dataset contains an overall variable that assigns players a score out of 100 based on their suitability for their club position. Although this column will not be used for modeling, it will be utilized during the feature selection phase.

2.2. Methodology

As mentioned in the introduction, the process used in this study can be broken down into two distinct steps. In this section, we will talk about each step separately.

2.2.1. First step: Designing and training machine learning models :

In this step, we aim to design and train 21 models capable of evaluating a player's suitability for a particular position. We will employ four important machine learning algorithms for this purpose.

[§]Players FIFA 23. (2023, January 30). Sofifa.com.[https://sofifa.com/]
[¶]Webster, E. (n.d.). Football_analytics. GitHub.[https://github.com/eddwebster/football_analytics/tree/master/data/fifa/raw]

Table 1. This table contains the names and descriptions of variables extracted from the FIFA dataset.

Description	Feature
Player position	club_position
Player ID on sofifa website	sofifa_id
Full name of the player	long_name
The score of the player in the position he plays	overall
player's predicted overall rating	potential
Age of the player	age
player height(cm)	height_cm
Player weight(kg)	weight_kg
The player's dominant foot	preferred_foot
The level of mastery of the non-dominant foot	weak_foot
Player movement skills	skill_moves
Player activity level on the field	work_rate
Player's body shape	body_type
Player speed	pace
Player's shooting ability	shooting
Player's passing ability	passing
Player's dribbling ability	dribbling
Player defensive ability	defending
Physical status of player	physic
Attacking crossing ability	attacking_crossing
Attacking finishing ability	attacking_finishing
Accuracy of attacking heading	attacking_heading_accuracy
Attacking short pass ability	attacking_short_passing
Air-borne strikes ability	attacking_volleys
Player's dribbling skills	skill_dribbling
Ability to curve the ball when passing and shooting	skill_curve
Skill in free kicks	skill_fk_accuracy
Skill in long passes	skill_long_passing
Ball control skills	skill_ball_control
Movement acceleration of the player	movement_acceleration
The ability to move at the highest speed	movement_sprint_speed
The player's agility	movement_agility
The ability of the player to react	movement_reactions
The player's ability to maintain balance	movement_balance
Player's shot power	power_shot_power
Player's jumping ability	power_jumping
Physical endurance of the player	power_stamina
Physical strength of the player	power_strength
Long shot power	power_long_shots
Player aggression	mentality_aggression
Ability to prevent penetration	mentality_interceptions
Positioning ability	mentality_positioning
Intuition about the position of other players	mentality_vision
Ability in penalty kicks	mentality_penalties
The ability to withstand the pressure of the opposing player	mentality_composure
The ability to recognize, track and block the attacker	defending_marking_awareness
Standing tackle ability	defending_standing_tackle
Sliding tackle ability	defending_sliding_tackle

Using the one-hot encoding method, three features; body type, work rate, and preferred foot are encoded to enter the models and create indicator variables. These features are converted into the respective 9, 8, and 1 binary variables. Using the Min-Max transformation, the rest of the model input variables were also transformed to the range 0 to 1.

For each position, we first consider all available players. Then, from all players who do not play in that position, a sample with the size as the number of players who do play in that position is drawn at random. Then, we combine this random sample with players who play the position in question. The dataset gains a column titled "label." We assign a value of 1 to the label for

Table 2. The size of the training and testing dataset by position.

RWB	LWB	CB	RB	LB	RCB	LCB	Position
332	324	569	2329	2296	2768	2884	**Training**
84	82	143	583	574	692	722	**Testing**
RDM	**LDM**	**CDM**	**CAM**	**RCM**	**LCM**	**CM**	**Position**
1075	1012	689	1334	1825	1836	345	**Training**
269	254	173	334	457	460	87	**Testing**
ST	**RS**	**LS**	**RW**	**LW**	**RM**	**LM**	**Position**
2070	995	979	729	710	1864	1880	**Training**
518	249	245	183	178	466	470	**Testing**

players of that position and a value of 0 to all other players. We now anticipate having a model for each position that can predict the label value. For each position, we allocate 80% of the samples for model training and 20% for model testing. Table 2 displays the number of instances obtained by the aforementioned method for each position. Table 2 demonstrates that the quantity of data varies significantly across different positions, with some positions having much less data due to their less frequent use in lineups.

As was previously mentioned, nominally discrete variables are encoded as binary variables. Some of these indicator variables have only one possible value in certain positions. The logistic regression model and support vector machine have difficulties estimating model parameters and making predictions in this situation. In the context of machine learning, nominal variables are variables that represent categories without a clear order or hierarchy. For example, "preferred_foot" in the FIFA dataset can take on the values "Left" or "Right", but there's no inherent order to these categories.

To use nominal variables in a machine learning models, they need to be converted to a format that can be processed by the algorithms. One common approach is to use one-hot encoding, which creates binary variables for each category in the nominal variable. For example, if we use one-hot encoding on "preferred_foot", we would create two binary variables: "preferred_foot_Left" and "preferred_foot_Right". If a player's preferred foot is left, then the "preferred_foot_Left" variable would be set to 1 and the "preferred_foot_Right" variable would be set to 0.

However, in some cases, a nominal variable may only have one possible value for certain positions. For example, the "preferred_foot" variable may always be "Right" for goalkeepers. In these situations, the corresponding binary variable will also always have a value of 0 or 1, which can cause issues for some machine learning algorithms like logistic regression and support vector machine. These algorithms work by estimating the parameters of a mathematical function, and having a variable with only one value can cause problems for this estimation process. To address this issue, one approach is to simply omit the single-valued variables for certain positions during model training and testing. This means that the model won't take these variables into account when making predictions for those positions. This issue was considered during the training and testing of the logistic regression model and the support vector machine, and single-valued variables were omitted from different positions.

The structure of the deep neural networks used in this study follows a general architecture. It begins with an input layer consisting of 59 neurons, followed by three hidden layers with 30, 20, and 10 neurons, respectively. Finally, there is an output layer with a single neuron that predicts a value of either 0 or 1, indicating whether the player is suitable for the given position or not. In the layer with 10 neurons, the Rectified Linear Unit (ReLU) activation function is used. ReLU is a commonly used activation function in deep neural networks because it is computationally efficient and helps prevent the vanishing gradient problem, which can occur when using other activation functions.

The use of the binary cross-entropy loss function and the Adam optimizer is a common practice in deep learning for binary classification problems. The validation data was used to monitor the model's performance during training and prevent overfitting, which occurs when the model becomes too complex and fits the training data too well but fails to generalize to new data. The number of epochs for training the network refers to the number of times the entire training dataset is presented to the network during training. Increasing the number of epochs can improve the model's performance, but it can also lead to overfitting if the model becomes too complex. The decision to increase the number of epochs or add more layers for certain positions suggests that the model may have had difficulty learning the patterns in the data for those positions with fewer samples. The addition of a layer of 40 neurons for the CDM position suggests that the model required additional complexity to effectively capture the relevant features for that position. Overall, these adjustments demonstrate the importance of careful model tuning and monitoring to achieve optimal performance.

Random forest is a popular ensemble learning algorithm that combines multiple decision trees to improve prediction accuracy and reduce overfitting. The number of trees in the forest is a hyperparameter that can be tuned to achieve better performance. Support vector machine (SVM) is a popular classification algorithm that finds the best hyperplane to separate classes in a linearly separable dataset. SVM can also be used with non-linear data by applying a kernel trick that transforms the data to a higher-dimensional space, where it can be linearly separable. SVM aims to find the maximum margin between the classes, which can improve the generalization of the model to new data.

In the results section, we will compare the outcomes of logistic regression, deep neural network, support vector machine, and random forest models on five positions from different areas of the field (forward, midfielder, and defender).

2.2.2. Second step: Finding the most important features of each position :

This step seeks to identify the most vital and effective characteristics for each football position. By this method, coaches and team analysts can talk about the fit of players for different positions in a more objective and scientific way.

Initially, the objective of our research was to develop deep learning models capable of accurately determining the optimal football player positions. However, we also recognized the difficulties inherent in implementing these models in actual coaching situations.

A significant obstacle was the potential unfamiliarity of football coaches with machine learning techniques, which could hinder their trust and adoption of such decision-making models. In order to bridge this gap and ensure that our research is accessible to the football community, we chose to present our findings in a manner consistent with existing football literature and terminology. This allowed us to integrate the technical aspects of machine learning with the practical requirements of football coaching.

In addition, we considered the limitations of data accessibility. Rarely are all player characteristics readily recorded, making it difficult to apply exhaustive models. To address this, the second step of our research included a phase of feature selection. This phase aimed to identify and emphasize the most critical features associated with each player's position, making it easier for coaches to understand how the models arrive at their decisions. This streamlined approach not only improves the interpretability of our models but also provides coaches with actionable insights that correspond to their existing knowledge and expertise.

We employ the recursive feature elimination algorithm for this purpose. This feature selection algorithm is a greedy and wrapper methods. This method selects features in a recursive manner, considering progressively smaller sets of features at each step. In this method, features are ranked based on the order in which they are eliminated from the feature space. This procedure continues until the desired number of features has been reached. Various machine learning algorithms may be utilized to implement the recursive feature elimination method. In this study, the random forest algorithm is utilized. Here's how the random forest algorithm works:

1. First, a set of decision trees is created by randomly selecting subsets of the training data and features. This is done to reduce overfitting and increase the diversity of the trees in the forest.

2. Each decision tree is trained on its respective subset of data using a random subset of features. During the training process, the tree splits the data into smaller subsets based on the values of the features.

3. Once all the decision trees have been trained, they are used to make predictions on new data. The final prediction is determined by taking the majority vote (for classification) or the average (for regression) of the predictions made by each individual tree in the forest. Random forest has several advantages over other machine learning algorithms. It can handle high-dimensional data and is resistant to overfitting. It is also relatively fast to train and can handle missing values in the data. Additionally, random forest provides estimates of feature importance, which can be useful for feature selection.

For the purpose of identifying the most vital and effective characteristics for each football position, we use the overall score as our response variable. The overall score is calculated using a completely linear relationship between the variables of the players in the dataset[‖]. However, the weights of this linear relationship are known to vary between positions, and these positional weights have never been formally announced. It is important to note that the overall score calculation also takes into account variables other than the physical and technical abilities of the players, such as their international reputation[**]. However, to focus solely on the physical and technical abilities of players, we exclude these variables from our analysis and by using only these characteristics, we obtain the features that are most effective for determining each position's overall score.

To identify the most important features for each position, we select all players at each position whose overall scores are higher than the average overall score of players at that position. This ensures that we focus on the physical and technical characteristics of elite players. We randomly split the selected players into 80% training data and 20% testing data, and Table 3 provides the details of this split.

For each random forest model, we consider 500 decision trees. The random forest algorithm evaluates each decision trees by using the mean squared error (MSE) loss function, which measures the difference between the predicted values and the actual values of the response variable. The lower the MSE, the better the model's fit to the data.

To prevent overfitting and to evaluate robustness of the procedure, we use 10-fold cross-validation during the training phase. In this technique, the training data is divided into 10 equal portions, and the model is trained on 9 parts and validated on the remaining part. This process is repeated 10 times, with each part used for validation once. The performance of the model is then evaluated by taking the average of the validation errors.

After performing 10-fold cross-validation, we calculated the mean validation error and standard deviation for each model. These metrics were then used to assess the robustness of the models. If the validation error was low and consistent across all folds, it indicated that the model was likely to generalize well to new, unseen data. On the other hand, if the validation error was high or had a large standard deviation, it suggested that the model may have overfit to the training data and may not perform well on new data.

[‖]Sohns, J. (2021, August 22). FIFA ratings explained: How is the overall rating created?. EarlyGame.[https://earlygame.com/fifa/fifa-ratings-explained-overall-rating]

[**]Sohns, J. (2021, August 22). FIFA ratings explained: How is the overall rating created?. EarlyGame.[https://earlygame.com/fifa/fifa-ratings-explained-overall-rating]

Table 3. The number of players and the number of players with an overall score above the average of the players of that position, across different positions.

RWB	LWB	CB	RB	LB	RCB	LCB	Position
208	203	356	1456	1435	1730	1803	**No. Players**
93	85	161	689	650	852	881	**No. above Average**
RDM	**LDM**	**CDM**	**CAM**	**RCM**	**LCM**	**CM**	**Position**
672	633	431	834	1141	1148	216	**No. Players**
320	294	211	410	523	560	121	**No. above Average**
ST	**RS**	**LS**	**RW**	**LW**	**RM**	**LM**	**Position**
1294	622	612	456	444	1165	1175	**No. Players**
629	304	276	200	189	581	585	**No. above Average**

Table 4. Performance of logistic regression models on test data across different positions.

RWB	LWB	CB	RB	LB	RCB	LCB	Position
100%	100%	100%	100%	100%	100%	100%	**Accuracy**
RDM	**LDM**	**CDM**	**CAM**	**RCM**	**LCM**	**CM**	**Position**
100%	100%	100%	100%	100%	100%	100%	**Accuracy**
ST	**RS**	**LS**	**RW**	**LW**	**RM**	**LM**	**Position**
100%	96/78%	100%	100%	100%	100%	100%	**Accuracy**

By using cross-validation, we were able to select the best performing features and ensure that it was not overfitted to the training data. This helps to increase the reliability and generalizability of our results, making them more useful for practical applications.

The goal of this step is to identify the five most important characteristics for each position in determining the overall score. During the training phase, we evaluate the model's performance by measuring the average error rate, which represents the difference between the predicted and actual overall scores.

The average R squared of the models on 21 positions during the training phase was 0.873, and a standard deviation of 0.058. These results are comparable to the errors obtained on the test data and suggest that the model's performance is consistent across different samples of the data. The use of 10-fold cross-validation also ensures that variations in the random partitioning of the data are taken into account in the evaluation of the model's performance.

In the results section, we will delve deeper into the performance of the models and discuss the top five features for each position that we identified using the algorithm.

3. Results

In this section, the outcomes of each methodological step are analyzed in detail.

3.1. Results of the first step

The logistic regression algorithm was found to be the best-performing algorithm among those used in this study. Table 4 presents the accuracy results of the logistic regression models on the test dataset, which show that 20 out of 21 models achieved a 100% accuracy rate, indicating that the recorded player features are highly effective in distinguishing players for different positions. Logistic regression is thus a suitable algorithm for this task. However, the accuracy for the RS position was slightly lower at 96.78%.

Table 5 illustrates the performance of four algorithms, namely logistic regression, random forest, support vector machine, and deep neural network, in determining the position of players for LB, CB, RCM, ST, and LW positions. Table 5 indicates that all four algorithms performed well in solving this problem, with logistic regression achieving the highest accuracy. The outcomes for the remaining positions were similar to those presented in Table 4.

According to Table 5, the logistic regression algorithm achieved 100% accuracy for all five positions, demonstrating its superior performance in distinguishing players for these positions based on their physical and technical characteristics. The random forest algorithm also performed well, achieving an accuracy rate of over 94%. The support vector machine algorithm, and the deep neural network algorithm achieved an accuracy rate of over 97% and 90% respectively. In summary, Table 5 demonstrates that all four algorithms are effective in determining the most important features for each position, with the logistic regression algorithm achieving the highest accuracy rate.

To further illustrate the effectiveness of the logistic regression model, we provide an example using Cristiano Ronaldo, who is known for his versatility on the field and ability to play in multiple positions. Using the logistic regression models developed

Table 5. Comparison of accuracy of logistic regression, random forest, support vector machine, and deep neural network algorithms on LB, CB, RCM, ST, and LW positions.

ST	LW	RCM	CB	LB	algorithm
100%	100%	100%	100%	100%	**Logistic regression**
95/95%	94/94%	96/72%	94/41%	97/04%	**Random forest**
97/10%	98/87%	97/59%	100%	98/78%	**Support vector machine**
95/37%	96/07%	95/62%	90/21%	98/61%	**Deep neural network**

Table 6. Results of logistic regression models of different positions for Cristiano Ronaldo. Cells filled with 1 are the positions that Cristiano Ronaldo has been declared fit to play.

RWB	LWB	CB	RB	LB	RCB	LCB	Position
0	0	0	0	0	0	0	**Prediction**
RDM	**LDM**	**CDM**	**CAM**	**RCM**	**LCM**	**CM**	**Position**
0	0	0	0	0	0	0	**Prediction**
ST	**RS**	**LS**	**RW**	LW	**RM**	**LM**	**Position**
1	1	0	1	0	1	0	**Prediction**

Table 7. The value of the loss function (MSE) on test data for various positions using five features selected for each position.

RWB	LWB	CB	RB	LB	RCB	LCB	Position
1/984	1/166	2/140	1/045	0/750	0/530	0/508	**MSE**
RDM	**LDM**	**CDM**	**CAM**	**RCM**	**LCM**	**CM**	**Position**
0/785	1/141	1/304	0/886	2/197	1/286	1/174	**MSE**
ST	**RS**	**LS**	**RW**	**LW**	**RM**	**LM**	**Position**
0/707	1/116	1/220	1/607	1/346	0/639	0/956	**MSE**

for different positions, we input Ronaldo's characteristics to predict which positions he would be fit to play. Table 6 shows the results of logistic regression models for different positions regarding Cristiano Ronaldo. As the table 6 shows, Ronaldo has been declared fit to play in several positions, including right midfielder, right winger, right striker, and striker. The cells filled with 1 in the corresponding columns indicate the positions that the logistic regression model has predicted Ronaldo to be suitable to play. The model predicts that Ronaldo is not fit to play in any of the left-side positions, including LWB, LW, and LCM. The model's output is in line with Ronaldo's real-world performance, as he is known for his versatility and ability to play in various positions.

Overall, the results of our study indicate that logistic regression is a suitable algorithm for predicting soccer player positions based on their recorded characteristics. The selected features were found to be highly effective in distinguishing players for different positions, and the logistic regression algorithm produced the most accurate results. Moreover, the model's predictions for Cristiano Ronaldo were consistent with his real-world performance, demonstrating the model's practical applicability. These findings could be useful for soccer teams and coaches in player selection and team formation, as well as for player management and development.

3.2. Results of the second step

Table 7 provides the value of the mean squared error (MSE) for each position when the random forest algorithm is used the five selected features to predict the players' overall scores. The MSE is a measure of the average squared difference between the predicted and actual values. The lower the value of the MSE, the better the performance of the model.

As we can see from Table 7, the average value of the MSE for all positions is 1.166, indicating that the model has performed well on test data. This value is lower than the average MSE observed during the training phase, which was 1.215, indicating that the model is not overfitted and can generalize well to new data. Table 7 shows the MSE values for each position separately. The values range from 0.508 for the LCB position to 2.197 for the RCM position. Overall, the results of Table 7 indicate that the random forest algorithm performs well in predicting players' scores using the five selected features for each position.

Table 8 shows the most important features for each position, ranked in order of importance. For example, for the position of RWB, the most important feature is "movement_reactions", followed by "mentality_interceptions" and "defending". Similarly, for the position of ST, the most important feature is "potential," followed by "shooting" and "attacking_finishing".

Interestingly, when comparing LS, ST, and RS positions, LS and RS have the same five features, while ST has only one feature, namely "mentality_positioning", that is different from the other two. This suggests that positioning is particularly important for the ST position, while LS and RS rely more on other skills such as shooting and dribbling.

Table 8. The most important features of each position in order of importance.

Feature	LB	RCB	LCB
First Feature	potential	defending	defending
Second Feature	defending	potential	potential
Third Feature	movement_reactions	defending_standing_tackle	mentality_aggression
Fourth Feature	attacking_short_passing	physic	defending_standing_tackle
Fifth Feature	defending_sliding_tackle	mentality_aggression	movement_reactions
Feature	**LWB**	**CB**	**RB**
First Feature	skill_ball_control	defending	potential
Second Feature	dribbling	defending_standing_tackle	defending
Third Feature	movement_reactions	potential	movement_reactions
Fourth Feature	defending	defending_sliding_tackle	defending_standing_tackle
Fifth Feature	skill_dribbling	mentality_interceptions	defending_sliding_tackle
Feature	**LCM**	**CM**	**RWB**
First Feature	potential	movement_reactions	movement_reactions
Second Feature	skill_ball_control	attacking_short_passing	mentality_interceptions
Third Feature	attacking_short_passing	potential	defending
Fourth Feature	movement_reactions	skill_long_passing	defending_sliding_tackle
Fifth Feature	defending	defending_standing_tackle	potential
Feature	**CDM**	**CAM**	**RCM**
First Feature	defending	potential	potential
Second Feature	potential	attacking_short_passing	movement_reactions
Third Feature	attacking_short_passing	dribbling	skill_ball_control
Fourth Feature	mentality_interceptions	skill_ball_control	attacking_short_passing
Fifth Feature	defending_standing_tackle	movement_reactions	defending
Feature	**LM**	**RDM**	**LDM**
First Feature	potential	potential	potential
Second Feature	skill_ball_control	defending	defending
Third Feature	movement_reactions	attacking_short_passing	attacking_short_passing
Fourth Feature	dribbling	movement_reactions	movement_reactions
Fifth Feature	attacking_crossing	defending_standing_tackle	mentality_interceptions
Feature	**RW**	**LW**	**RM**
First Feature	skill_ball_control	skill_ball_control	potential
Second Feature	dribbling	potential	movement_reactions
Third Feature	shooting	dribbling	skill_ball_control
Fourth Feature	potential	movement_reactions	dribbling
Fifth Feature	skill_dribbling	skill_dribbling	mentality_positioning
Feature	**ST**	**RS**	**LS**
First Feature	potential	potential	potential
Second Feature	shooting	shooting	shooting
Third Feature	attacking_finishing	movement_reactions	skill_ball_control
Fourth Feature	mentality_positioning	skill_ball_control	attacking_finishing
Fifth Feature	movement_reactions	attacking_finishing	movement_reactions

4. Discussion

The primary objective of this research is to address a real-world problem in football analysis, and our focus has always been on achieving the highest accuracy possible in player position prediction. Instead of introducing complex, sophisticated models, our

objective is to provide readily applicable solutions for football coaches and analysts.

As demonstrated in the results section, the straightforward logistic regression model outperformed the more complex models we evaluated in terms of accuracy. This outcome is consistent with Occam's razor, which favors simple solutions when equally effective alternatives exist. By selecting simpler models that achieve exceptional precision, we prioritize utility and usability for football professionals.

It's crucial to highlight that the dataset we employed for this study indeed contains a wealth of informative features. This extensive dataset was instrumental in enabling our models to achieve remarkable accuracies. The abundance of relevant and informative data within the dataset provides a distinct advantage and magnifies the significance of the models' performance in addressing the practical challenges of football player position determination.

Here we would like to clarify the reasoning behind not using a multi-class classification model, which employs a collection of binary models for football positions. As stated in introduction section, we recognize that adjacent positions on the football pitch frequently require comparable skills. In light of this, we firmly believe that a player's ability to play in multiple positions is a realistic scenario in football, reflecting the complexity of the actual game.

By employing a binary model for each position, we avoid the assumption that a player can only fulfil one predefined role, thereby enabling a more flexible and nuanced evaluation. Converting the problem into a multi-class classification problem would require setting a threshold on the softmax output, determining a player's position based on this threshold, or choosing a fixed number of positions with the highest probabilities. Both of these methods introduce subjectivity into the process, which we aim to eliminate with our methodology.

In addition, we investigated the multi-class classification approach using a fully connected multilayer neural network, which yielded an approximate 42% accuracy in determining players' best position. While this is a useful benchmark in terms of accuracy, it performs far less than the results obtained by our introduced binary models.

Our decision to employ a set of binary models in our player position prediction system stems from our desire to maintain flexibility, objectivity, and practicability.

5. Conclusions

We developed a machine learning-based method to determine the suitable positions for football players based on their physical and technical characteristics. Our approach involved training models to find these positions for each player and then identifying the five most important features for each position.

In this study, we have selected five most important features for each position using the recursive feature elimination method. The number five was chosen intuitively for the feature selection procedure. However, in practical applications, this number may increase or decrease depending on the sensitivity of decisions and the need for more specific information for some or all positions. Therefore, the number of selected features is not necessarily fixed and can be adjusted according to the requirements of the application. This process allows coaches and training staff to design position-specific exercises and gain a deeper understanding of the reasoning behind the models' decisions.

While our feature selection process was effective, further work could explore the impact of other variables, such as age or playing style, on player performance. Additionally, alternative feature selection methods, such as mutual information-based feature selection methods, could be considered.

Overall, our approach offers a data-driven solution to a critical problem in football, improving both individual and team performance on the field. Our study demonstrates the effectiveness of a machine learning-based approach for determining the optimal positions for football players.

Appendix

The dataset analysed during the current study is available in the [Football_analytics] repository.[https://github.com/eddwebster/football_analytics/tree/master/data/fifa/raw].

ORCID

Changiz Eslahchi: https://orcid.org/0000-0002-8913-3904

References

1. D. Abidin. A case study on player selection and team formation in football with machinelearning. *Turkish Journal of Electrical Engineering and Computer Sciences*, 29(3):1672–1691, 2021.

2. K. Apostolou and C. Tjortjis. Sports analytics algorithms for performance prediction. In *2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA)*, pages 1–4. IEEE, 2019.
3. P. Enyindah et al. A machine learning application for football players' selection. *International Journal of Engineering Research & Technology*, 2015.
4. A. E. Evwiekpaefe, E. Bitrus, and F. Ajakaiye. Selecting forward players in a football team using artificial neural networks. *International Journal of Computer Applications*, 176(28):8–13, 2020.
5. M. Frey, E. Murina, J. Rohrbach, M. Walser, P. Haas, and M. Dettling. Machine learning for position detection in football. In *2019 6th Swiss Conference on Data Science (SDS)*, pages 111–112. IEEE, 2019.
6. A. García-Aliaga, M. Marquina, J. Coterón, A. Rodríguez-González, and S. Luengo-Sánchez. In-game behaviour analysis of football players using machine learning techniques based on player statistics. *International Journal of Sports Science & Coaching*, 16(1):148–157, 2021.
7. M. Tavana, F. Azizi, F. Azizi, and M. Behzadian. A fuzzy inference system with application to player selection and team formation in multi-player sports. *Sport Management Review*, 16(1):97–110, 2013.

