Predicting Transfer Value of Professional Football Players Based on Player Skills and Characteristics Using Multiple Linear Regression, Support Vector Regression, and Random Forest Regression

Student details

Gijs (G.P.K.) Laros

Student number: 2005189

g.p.k.laros@tilburguniversity.edu

Word count: 8700

THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE IN DATA SCIENCE & SOCIETY
DEPARTMENT OF COGNITIVE SCIENCE & ARTIFICIAL INTELLIGENCE
SCHOOL OF HUMANITIES AND DIGITAL SCIENCES
TILBURG UNIVERSITY

Thesis committee

Supervisor: dr. B. Čule

Second reader: dr. G.A. Chrupała

Tilburg University

School of Humanities & Digital Sciences

Department of Cognitive Science & Artificial Intelligence

Tilburg, The Netherlands

June 24, 2022

Abstract

Football clubs tend to buy new players to try to increase their performance, which leads to more revenue. These clubs need to know the transfer value of a football player before buying them, because otherwise they might pay too much for a player. We examine this in this research. Our research question is: *To what extent is the transfer value of professional football players predictable, based on player skills and characteristics?* This research focuses on predicting the transfer fee a club is willing to pay to buy a player, whereas in earlier research the main focus was on predicting the hypothetical market value of a player. We use data from the football video game FIFA to make our predictions. The applied models are multiple linear regression, support vector regression, and random forest regression. We find that the transfer value is predictable to some extent, but our models perform worse than models used in earlier research where the predictability of the market value is examined. Out of the models we used, the random forest regression model is the best model to predict the transfer value (R^2 = 0.673). We find out what the most important features are in predicting the transfer value, and we learn that there is a difference in the most important features for predicting the transfer value for players in different positions.

Data Source, Code & Ethics Statement

Work on this thesis did not involve collecting data from human participants or animals. The author of this thesis acknowledges that they do not have any legal claim to this data or code. The data used in this thesis is derived from https://github.com/ewenme/transfers and https://www.kaggle.com/datasets/stefanoleone992/fifa-21-complete-player-dataset. The code used in this thesis is not publicly available. All figures in this thesis have been created or adapted by the author.

1. Introduction

Football is one of the most popular sports in the world (Parrish & Nauright, 2014). Many people from all over the world watch football. For example, the UEFA Euros final, the European Championship for countries, of 2021 was watched by a live average audience of 328 million people (UEFA, 2021b). Football clubs need to perform as well as possible. For instance, when a football club performs well in the UEFA Champions League and continues to the knock-out stage, they earn approximately 9.6 million euros and reaching the final of the tournament earns a club an additional 15.5 million euros (UEFA, 2021a).

To be able to perform as well as possible, clubs tend to search for new players who are likely to increase the clubs' performance. There are three different options to sign a new player, namely on loan, by signing a free agent or buying a player. This research focuses on the option of buying a player, which means that one football club pays compensation to another club to be able to have that player play for them. This compensation is called the transfer fee (BBC, 2017b). The selling and buying clubs must agree on a transfer fee. The right balance of the actually paid transfer fee and the alleged value of a player is essential, as otherwise the divestment in case a player would not perform as expected, might be large. Besides the relevance of knowing the transfer value of players for football clubs, the transfer value is also interesting for football fans and analysts. When a football club buys a new player, fans and analysts all have an opinion on whether the paid transfer fee is in line with the player's qualities (BBC, 2017a; Walker, 2021). In this research, we want to find out whether we can predict the transfer value of a player, which could be used by clubs during negotiations and by fans and analysts during discussions about whether a player is worth the transfer fee.

The goal of this research is to find out to what extent it is possible to predict the transfer value of a player, based on their skills and characteristics. The main research question that will be answered in this research is:

RQ: To what extent is the transfer value of professional football players predictable, based on player skills and characteristics?

In section 2.4, we formulate two additional sub-questions that will help answer the research question.

Next to the societal relevance of this problem, this research is also relevant from a scientific point of view. This subject has not been researched before with multiple machine learning techniques and with data that contains the actually paid transfer fee of players to predict the transfer value of a player. In most previous papers, to be discussed in section 2, the

market value of players has been used, but this is a hypothetical value and cannot be compared to the actual transfer value. One paper used the actually paid transfer fee to predict the player value, but they only used one machine learning algorithm.

In this research, we find that the transfer value is predictable to some extent, but our models perform worse than models from earlier research where the predictability of the market value is examined. This could be due to the fact that the market value is a computed value, while the transfer value is the actual paid fee to buy a player. There could be other factors that contribute to the actual transfer fees in comparison to the market value, which we did not consider in this research. This might cause the worse performance. The best model, out of the models we used, to predict the transfer value is the random forest regression model, with an R^2 of 0.673. We find out what the most important features are in predicting the transfer value, and we learn that there is a difference in the most important features for predicting the transfer value for players in different positions.

2. Related Work

This section discusses the background information and related work. In the subsections that follow related work on the prediction of football player values, the use of football video game data, and data-driven methods to predict (player) values are considered. Next to that, we discuss the gaps in earlier research.

2.1 Prediction of Football Player Values

Previous research already researched the value of professional football players. Müller et al. (2017) made use of player performance and characteristics to estimate the market value of players with multi-level regression methods. Additionally, Müller et al. included the popularity of a player as a feature. Barbuscak (2018) researched the market value of football players using data from transfermarkt.com, a website that keeps track of football statistics. Barbuscak ran a linear regression, which showed a significant effect of multiple variables on the market value, such as years on the contract left. The influence of years on the contract left on the market value is also reported by other research (Carmichael et al., 1999; Frick, 2007). Research by Felipe et al. (2020) shows that age is an important factor when predicting the market value of players as well. In their research, Inan and Cavas (2021) predicted the market value of Turkish Super League players with an artificial neural network. Inan and Cavas found that the estimations they made were better than the market value estimated by StatsPerform, which is a well-known sports technology company.

The problem with predicting the market value is that it is a hypothetical value. It says something about the value of a player but does not necessarily have to be paid by the football clubs when they want to buy a certain player, and therefore it is not the actual value of a player. To better predict the real value of a player, we use the actual transfer fee paid for a player in this research.

This is something that was researched in a paper by Poli et al. (2022). Poli et al. try to predict player transfer fees with a multiple linear regression model. They use transfermarkt.com and respected data sources from within the world of football to validate the transfer fees paid for players. Their model performs well, but Poli et al. only apply one model. Furthermore, in their research they do not use data about player skills, like dribbling, but only player statistics such as goals scored. Other research, which will be discussed in section 2.2, points out that football game data with information about player skills can be used to create appropriate models as well.

2.2 Use of Football Video Game Data

In some research done to predict player market values, data from football video games was used. Kirschstein and Liebscher (2019) studied the possibility to predict football player values with the help of machine learning techniques. Kirschstein and Liebscher developed a model to estimate the market value of players based on skill variables from FIFA 16. FIFA is a popular football game, named after the world football association Fédération Internationale de Football Association (FIFA). In this game, every player is rated based on their skills. Kirschstein and Liebscher (2019) matched this with the market values of players from the German First Division and Second Division from transfermarkt.com. Their analysis showed an impact of the reputation of a club on the market value of players. In the paper by Yiğit et al. (2020), models were created based on the skills of players in the football-related video game Football Manager. In their paper, random forests, gradient boosting, and ridge and lasso regressions are used. The paper concludes that using data from the Football Manager video game in combination with market values, found on transfermarkt.com, made it possible to give a better prediction of the market value. Yiğit et al. suggest building a model considering the competition and player positions in future work as this might improve results. Behravan and Razavi (2021) used the FIFA 20 dataset to predict the market value of football players and found that it is indeed necessary to take into consideration the different positions of players on the field. Another important variable to consider, according to their research, is the overall rating. This rating tells something about the current skill level of a player. According to Al-Asadi and Tasdemir (2022) the potential rating, which indicates what the possible future overall rating of a player is, is an important factor as well.

There is a gap in the literature concerning the prediction of the transfer value of professional football players with the use of football video game data and multiple machine learning techniques. Therefore, in this research we predict the transfer value of football players, based on their real-life characteristics and their in-game skills from the football video game FIFA, using multiple machine learning methods to see which of these methods can make the most accurate predictions.

2.3 Methods

Many algorithms can be used to make predictions. In section 2.1, we discussed some research predicting the values of football players with linear regression models, which showed significant results (Barbuscak, 2018; Müller et al., 2017; Poli et al., 2022). Another research that used a linear regression model, is the research of Emioma and Edeki (2021). Emioma and Edeki wrote a paper about stock predictions, which is a comparable problem to the one in this

research as it takes multiple attributes and outputs a prediction of a numeric value. Dash et al. (2021) also wrote a paper about stock prediction. Dash et al. proposed using a support vector regression model with optimized hyperparameters to make better predictions of the close price of a stock. This method was compared to a support vector regression model which was not optimized and a hybrid neural network. Dash et al. concluded that the optimized support vector regression model performs better and takes less long to run. Support vector regression is used in multiple papers issuing value prediction problems (Henrique et al., 2018; Parbat & Chakraborty, 2020).

Stock price prediction models were also researched by Zhang and Lou (2021). In their research, a backpropagation neural network is used and achieved adequate results. Inan and Cavas (2021) used an artificial neural network to predict football player market values. Both these papers had satisfactory results. However, neural networks need a large dataset to be trained (Y. Zhang et al., 2021). Also, according to Jiang et al. (2021), support vector machines give better results on smaller datasets in comparison to neural networks. A method that was used in previous research to predict the market value of football players is random forest regression. This method is used in the research of Al-Asadi and Tasdemir (2022), where they researched the possibility to predict the market value of football players in Europe, based on data from FIFA. The implemented models were linear regression, multiple linear regression, decision trees, and random forest regression. Their research concludes that random forest regression is the best method to predict the market value. Some of the methods applied by Al-Asadi and Tasdemir were used in the paper of Huang et al. (2020) as well. They examined to what extent blood pressure can be predicted. Huang et al. (2020) use different machine learning techniques, including linear regression and random forest regression. The random forest regression model performs best in their research and appears to be one of the best models for predicting values with smaller datasets.

In this research, we will use the multiple linear regression, support vector regression, and random forest regression models to predict the transfer value of football players, based on the fact that these models gave adequate results according to other papers. Also, these models are suitable for our problem.

2.4 Additional Research Questions

To find out to what extent it is possible to predict the transfer value, we need to know what the most important features are. In previous research, some attributes had the biggest influence on the market value of football players, such as contract duration, overall, and potential. However, these results were not unambiguous as they differed in the different

papers. In this research, we will look at the effect of multiple skills and attributes on the transfer value of players. The first sub-question that is formulated is:

SQ1: What effect do different player skills and characteristics have on the transfer value of professional football players?

One of the attributes that has influence according to multiple papers, is the position of players. It was often reported (Behravan & Razavi, 2021; Yiğit et al., 2020) that players are valued based on different skills and characteristics that are more important for their specific position. The second sub-question that is formulated is:

SQ2: What is the difference in the effect of different player skills and characteristics on the transfer value of professional football players for different positions on the field?

3. Methodology

This section discusses the models used in this research. In section 3.1, multiple linear regression is explained. After that, we describe support vector regression and then random forest regression is discussed.

3.1 Multiple Linear Regression

The first algorithm is multiple linear regression. Multiple linear regression is a well-known machine learning technique. It takes multiple independent variables to explain one dependent variable. The relationship between these independent and dependent variables is analyzed by this algorithm (Uyanık & Güler, 2013). The formula for multiple linear regression is:

$$y_i = \beta_0 + \beta_1 X_{1i} + \cdots + \beta_n X_{ni} + \varepsilon_i,$$

where y_i is the predicted value of the dependent variable, $\beta_0, \beta_1, \dots, \beta_n$ are the regression coefficients, $X_{1i}, \dots, X_{ni}$ are values of the independent variables, and ε_i is the error term. The coefficients give an understanding of the relationship between the dependent and independent variables.

This method is used because in the dataset there are multiple independent variables that we can use to predict the transfer value of players. Earlier research (Barbuscak, 2018; Müller et al., 2017; Poli et al., 2022) proved that this method gave adequate results on similar data, and therefore we will use this method to make our predictions.

3.2 Support Vector Regression

Next to that, support vector regression will be used. While classic regression models estimate coefficients by reducing the square error, support vector regression is a machine learning prediction model built for continuous values that relies on the structural risk minimization concept (Vapnik, 1999). The goal of support vector regression is to discover the function that fits the data with an error of less than or equal to ε as closely as possible while keeping the data as flat as possible (Smola & Schölkopf, 2004). This can be written down as follows:

$$\begin{aligned} &minimize && \frac{1}{2}||w||^2 \\ &subject\ to && \begin{cases} y_i - \langle w, x_i \rangle - b \leq \varepsilon \\ \langle w, x_i \rangle + b - y_i \leq \varepsilon \end{cases}, \end{aligned}$$

where $||w||^2$ is the dot product of w, which we want to minimize to ensure having as flat as possible data, y_i is the value of the dependent variable in i, $\langle w, x_i \rangle$ is the dot product of i, b is the error in i, and ε is the predefined acceptable error (Smola & Schölkopf, 2004).

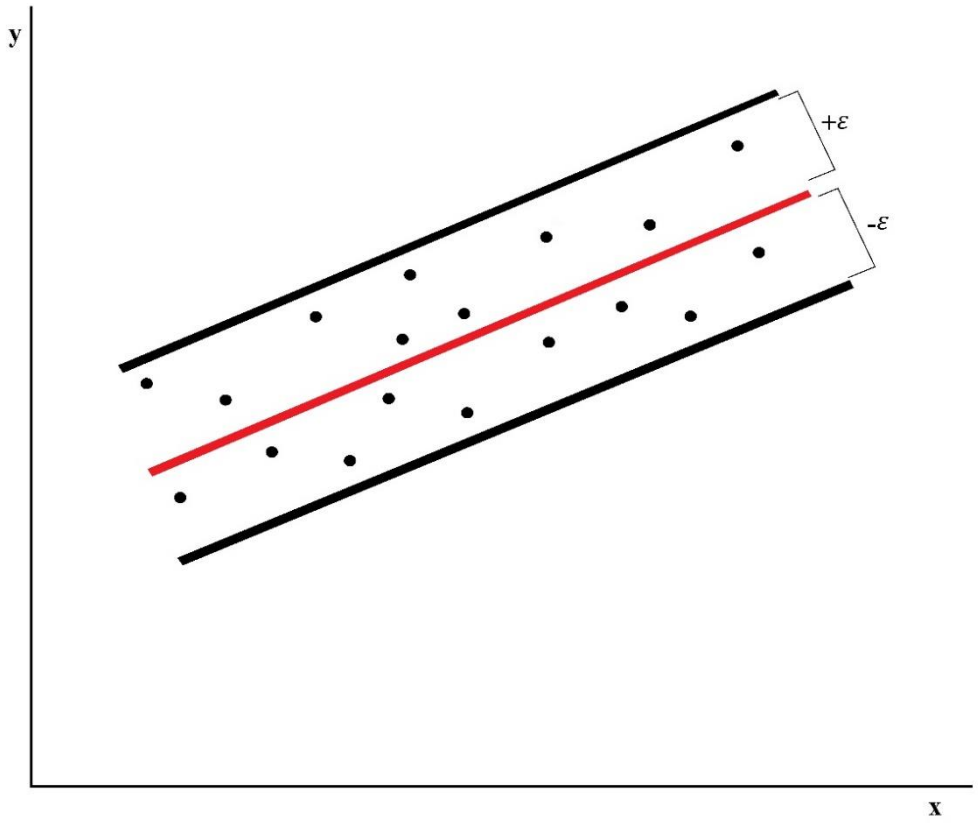
The support vector regression model only analyzes data points that fall within the range of the decision boundaries, and then fits the best fit line, also called the hyperplane. The hyperplane is then used to predict outcomes. In Figure 1, support vector regression is visualized.

Besides epsilon, a support vector regression model has multiple hyperparameters. The most important hyperparameters are the kernel, regularization parameter C, and gamma. The kernel transforms the data from linearly inseparable to linearly separable data and usually corresponds to the dot product in a high-dimensional feature space (Hofmann et al., 2008). Pontil and Verri (1998) refer to the regularization parameter C as the 'trade-off between the largest margin and the lowest number of errors.' Gamma determines the width of the kernel function. When the gamma value is low, the curve of the decision boundary becomes low, resulting in a relatively large decision zone (Al-Mejibli et al., 2020).

The support vector regression model has several advantages. It has high accuracy and generalizes well (Awad & Khanna, 2015). Next to that, this algorithm does not have many critical parameters, making it a useful algorithm to get excellent results (Smola & Schölkopf, 2004). Also, previous research showed that this model gives great results when making predictions for numeric values (Dash et al., 2021).

Figure 1

Visualization of support vector regression. The red line is the hyperplane. The black lines are the decision boundaries. The black dots are the data points. Adapted by permission from Springer Nature Customer Service Centre GmbH: Springer Nature, Statistic and Computing, "A tutorial on support vector regression" by Smola & Schölkopf (2004).



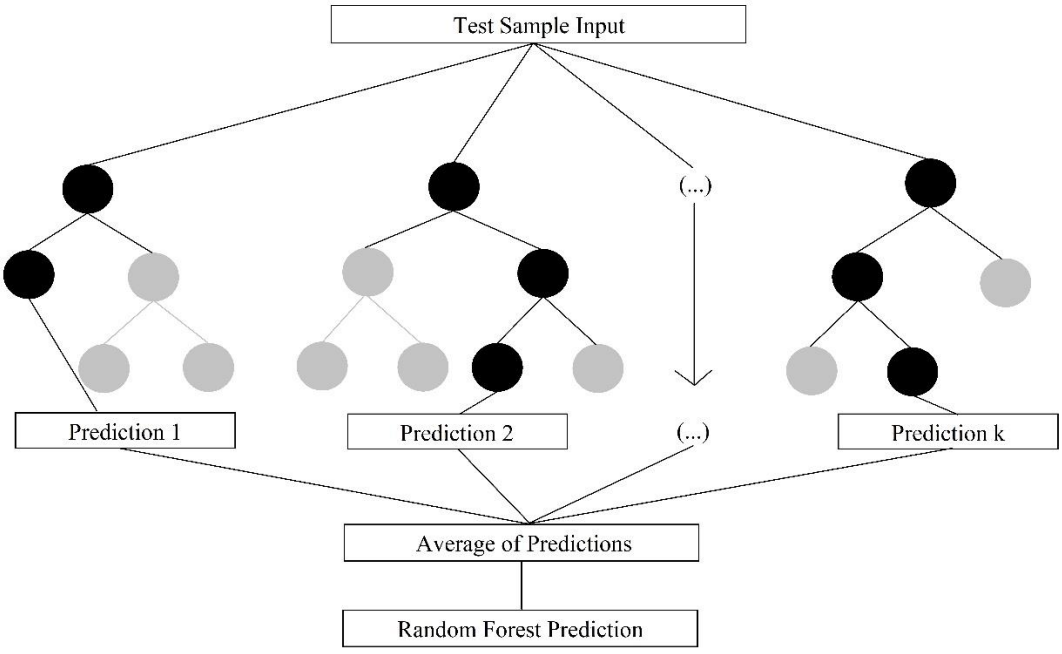
3.3 Random Forest Regression

Lastly, the random forest regression model is implemented. The random forest technique operates by building numerous decision trees and averaging the results of those trees (Navyashree et al., 2019). In a random forest, the decision trees are built with a subset of the variables in the dataset. This way, trees will differ from each other even if the selected subset of data is the same (Altman & Krzywinski, 2017). Random forests can, just like decision trees, be used for both regression and classification. This depends on the type of the target variable (Chan & Paelinckx, 2008). A visualization of what a random forest looks like can be found in Figure 2.

The random forest method is known for its exceptional performance (Segal, 2003). Next to that, some of the papers (Al-Asadi & Tasdemir, 2022; Huang et al., 2020) reported that random forest regression performed best on similar types of problems, making it interesting to see if this is the case in this research as well.

Figure 2

Visualization of random forest regression. k stands for the number of trees the random forest consists of. The prediction for each tree is used to calculate the average prediction. Adapted by permission from IEEE, 10th International Conference on Cloud Computing, Data Science & Engineering, "A Comparative Analysis of Classifiers for Image Classification" by Chugh et al. (2020).



4. Experimental Setup

In this section, we will first explain how the data is gathered, cleaned, and merged. After that, the preprocessing, splitting, and transformation of the data are described. Then, the evaluation metrics that will be used are listed and we will explain how we build and tune our models. The experiment is performed on a HP Pavilion Laptop 13-an0xxx, with the Intel(R) Core i5-8265U processor which has 1.60GHz clock speed. This laptop has 7,89 GB RAM. We use the programming language Python 3.8.8 in JupyterLab 3.3.2.

4.1 Data Description

The dataset that will be used for this research is a combination of two datasets. These two datasets are described in this section. First, the dataset with all information about the transfers is described. After that, we specify what the FIFA dataset consists of.

4.1.1 Transfer Data

The first dataset contains all transfers between 2015 and 2021 where the buying club plays in the English Premier League, the English Championship, the Spanish La Liga, the Italian Serie A, the German Bundesliga, the French Ligue 1, the Portuguese Primeira Liga, or the Dutch Eredivisie. These are the top-tier leagues from the countries with the highest country coefficient according to the European Football Association UEFA (2022). The second tier of England, the Championship, is included as well because England has the highest country coefficient. This dataset is a merger of smaller datasets. For these eight competitions, there are separate datasets per year with all outgoing and incoming transfers. These separate datasets are merged into one dataset with all transfers between 2015 and 2021. The variables that are part of this dataset can be found in Table A1, in Appendix A**.** In total, this dataset consists of 45,245 observations.

4.1.2 FIFA Data

The other dataset contains all data that is available of players in the popular football video games of FIFA. Every year, there is a new game released with updated characteristics and skills for all players from all the biggest competitions in the world. This means that there are up-to-date characteristics and skills of the players that were transferred between 2015 and 2021. The player characteristics can be found in the variables: player name, age, height in centimeters, weight in kilogram, nationality, player position, preferred foot, international reputation, and contract duration. The field player skills are divided into six subcategories, namely passing, defending, physical, dribbling, shooting, and pace. For goalkeepers, the subcategories are kicking, handling, shooting, reflexes, diving, and speed. The subcategories

all have subcategories themselves as well. Also, every player has an overall rating and a potential rating in the game. All ratings are in the range of 1 and 99, where the lowest is 1 and where 99 is the highest possible score. An overview of all variables in the FIFA dataset can be found in Table A2, in Appendix A. The information in the datasets is derived from sofifa.com, which is a well-known website with an overview of all FIFA data per year.

4.2 Data Cleaning

From the dataset, free transfers and loan transfers are excluded as we focus only on players for whom a transfer fee has been paid. Both incoming and outgoing transfers are part of the dataset. This causes some transfers to occur twice in the dataset. Therefore, only incoming transfers are considered in this research. After removing the free agents, loan transfers, and outgoing transfers from the dataset, 4,570 observations remain in the dataset.

4.3 Data Merging

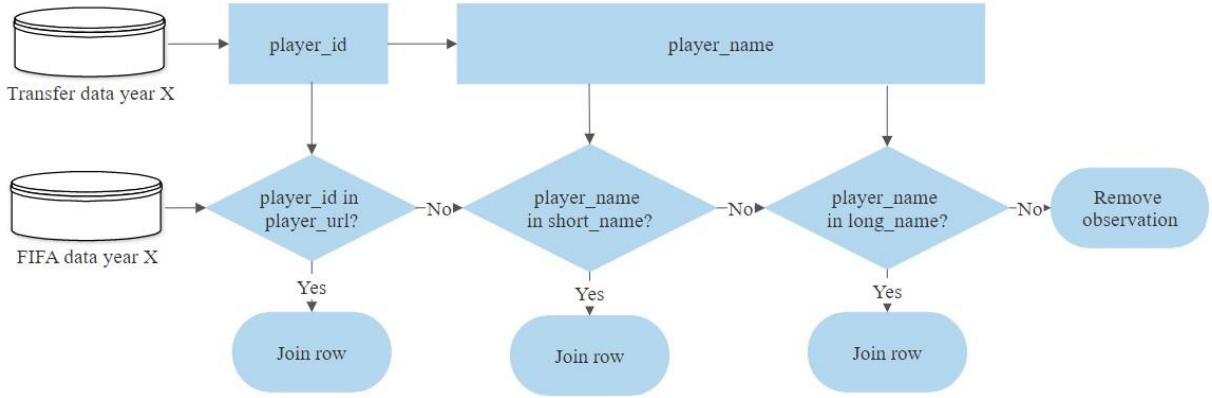
The two datasets are merged based on the player names and the year of a transfer. If a transfer took place in 2015, we use the data of that player in 2015. In both datasets, the names are included, but some of them are written differently and special characters are not processed in the same way. To deal with this problem, we try to match the variable ‘player_name’ from the transfer dataset with the variables ‘player_url’, ‘short_name’, and ‘long_name’ from the FIFA dataset. To find a match between the ‘player_name’ and ‘player_url’, we transform the ‘player_name’ to a new variable, namely the ‘id_name’ which is a merger of the names of a player in ‘player_name’ with a hyphen between them. For example, for the player Frenkie de Jong, the ‘id_name’ frenkie-de-jong is created. This makes it possible to match the ‘player_name’ with the ‘player_url’, as the ‘player_url’ consists of an URL containing the player names written with a hyphen between the names.

However, a problem occurred after this process. After doing some exploratory data analysis, we found that some players with a very low score on the variable ‘potential’ were sold for a high transfer fee. As this is not in line with our expectations, we look at the data and find that some players with a single name (such as ‘Rodri’ or ‘Pedro’) have a chance of being matched with the wrong player from the FIFA dataset. For example, ‘Rodri’ is matched with a player named ‘Rodrigo de Paul’, as ‘Rodri’ is in ‘Rodrigo de Paul’. To prevent this from happening, we decide to match the players with a single name manually.

The observations that could not be matched with the FIFA dataset are disregarded. Eventually, 3,916 observations were matched. The process of the data merging is illustrated in Figure 3.

Figure 3

The data merging process. In this process, only players with more than one name are considered. Players with one name are matched and merged manually.



4.4 Data Preprocessing

4.4.1 Feature Selection

From the merged dataset, some variables are removed because they are not valuable to this research. The variables ‘player_name’, ‘player_url’, ‘short_name’, and ‘long_name’ are removed because these are too specific to an individual player and therefore are not useful to make generalizable predictions. The variables ‘buying_club’ and ‘selling_club’ are also removed. An important factor to consider in removing variables is the runtime (Gupta et al., 2017). By including ‘buying_club’ and ‘selling_club’, the runtime of some of the models increases from 10 minutes to over two hours. We want it to be possible to run the models using basic hardware and software, such that anybody can reproduce the experiment. Therefore, we decide to remove the variables ‘buying_club’ and ‘selling_club’ from our dataset.

In the FIFA dataset, there are two variables called ‘league_rank’ and ‘league_level’. The variable ‘league_rank’ is used in the FIFA games that were released between 2015 and 2020 to describe the level of the league and has a scale of 0-4, while in 2021 the variable is renamed to ’league_level’ and uses a scale of 1-3. Although these variables have a numeric value, they are categoric. Therefore, it is hard to scale these two variables in such a way that they can be compared, and thus, they are not used in this research.

The ‘value_eur’ represents the value of a player, and ‘wage_eur’ shows the wage of a player according to FIFA. As it is not clear how the values for these variables are computed and whether they are realistic or not, they are excluded.

Some of the variables that are derived from the FIFA games do not give information about a players’ skills or characteristics. For that reason, these variables are removed. In Table A2, in Appendix A, an overview of the used and left out variables from the FIFA dataset can be found.

4.4.2 Feature Engineering

To be able to answer SQ2, *what is the difference in the effect of different player skills and characteristics on the transfer value of professional football players for different positions on the field?*, we need to divide the dataset by position. As the dataset is quite small, we do not use specific positions, such as left-back and striker. We group the players by part of the field they play in, and put this into a new variable, ‘position_group’. The created groups are goalkeeper, defender, midfielder, and attacker.

In the football video game FIFA, goalkeepers are evaluated on different criteria in comparison to field players. This makes it hard to compare goalkeepers to field players, and therefore we will exclude goalkeepers from the main models of this research. However, the goalkeeper data is used separately to see what the most important factors are in predicting the transfer value of a goalkeeper. After removing the goalkeepers from the main dataset, 3,650 observations remain in the dataset.

The variables ‘position_group’, ‘buying_league’, ‘nationality’, ‘preferred_foot’, and ‘transfer_period’ are categoric. These variables are changed into dummies to be able to use them in the models. For each dummy variable, one category of the dummy is removed from the dataset, to avoid the dummy variable trap (Pal & Bharati, 2019). The categories that are removed are now the base category. Table 1 shows the base category for each dummy variable.

Table 1

Dummy variables with their base category.

Dummy variable	Base category
position_group	Attacker
buying_league	1 Bundesliga
nationality	Afghanistan
preferred_foot	Left
transfer_period	Summer

4.4.3 Missing Data

In the merged dataset, there are a total of 14 rows with missing values. The total number of rows was 3,650, meaning that a small percentage of 0.384% had a missing value. When a small percentage of the data is missing, compared to the dataset, it is possible to use a deletion technique, according to Mockus (2008). Therefore, the 14 rows with missing values are deleted, leaving a total of 3,636 observations.

4.5 Data Splitting

The dataset is randomly split into a train and a test set. This is done to make sure that we have an independent set of data to evaluate our fitted models. In this research, the dataset is split into 80% training data and 20% test data. This means that in total 2,908 observations are part of the train set, and 728 observations are in the test set. The data is split with the use of Scikit-learns' train_test_split function. An overview of all packages used in this research can be found in Appendix B.

The test set should only be used to test how well the model performs on unseen data. In our research, we want to tune the hyperparameters in our models and we want to find the best way to transform our data per model, which will be explained in section 4.6. To tune our hyperparameters, without the risk of overfitting, we need a way to take a separate set to test this. In this research, cross-validation is used. K-Fold cross-validation is a method where the train data is split into k parts. Then, the model is trained using k-1 parts, and the performance of that model is measured using the left-out part. This process is repeated until all parts have been left out. The performance is the average of the performances of all trained models. In this paper, we use a k of 10, meaning that we split the train data in 10 when performing cross-validation.

After using cross-validation to tune the hyperparameters of our models, the test set is used to see to what extent our final models are generalizable. As the test set data is not used to fit the models, we can use this data to see if our models perform well on unseen data and to

find out whether our models have a similar performance on unseen data compared to the cross-validated performance.

4.6 Data Transformation

To increase the performance of models, it is possible to transform the data. One form of data transformation is standardization. With standardization, the independent variables are transformed by first subtracting the mean of the variable and then dividing that by the standard deviation. This way, the values of each variable are standardized and will have a mean of 0 and a standard deviation of 1. The formula for standardization is as follows:

$$z = \frac{X - \mu}{\sigma},$$

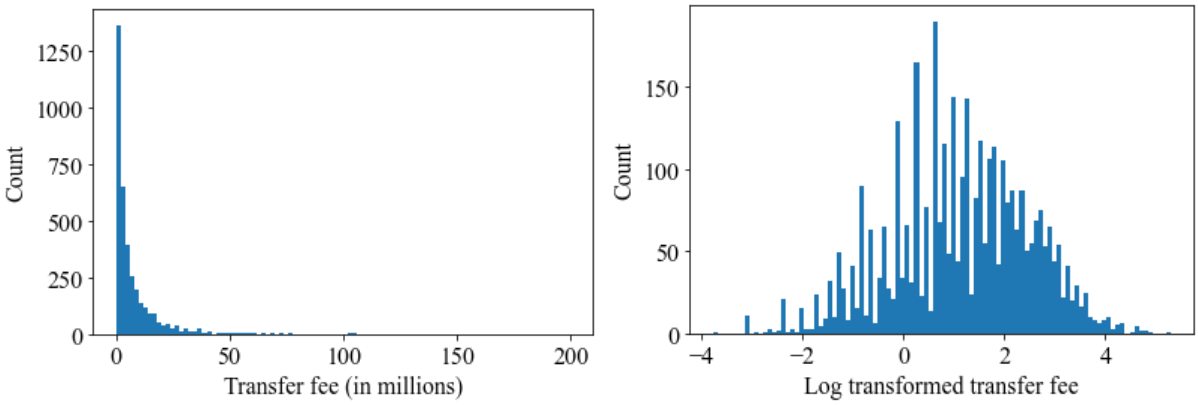
where z is the standardized value, X is the initial value, μ is the mean of a variable, and σ is the standard deviation of a variable.

To standardize the data, we use Scikit-learns' StandardScaler on our independent variables to transform them. All Python packages that are used in this research can be found in Appendix B. The standardization has to be done after the train-test split, as the transformed variables are used to train the model. Including the test data in conducting the mean and standard deviation would mean that the independent variables of the test data are used in the transformation process and therefore in the model training, which is not what we want. The independent variables in the train and test data are standardized using the mean and standard deviation of each of the independent variables in the train data.

Next to transforming the independent variables, there is also a possibility to transform the dependent variable. In our data, most of the observations have a small value for the transfer fee. This is illustrated on the left side of Figure 4. As this distribution might be problematic for some algorithms, we logarithmically transform the dependent variable, the transfer fee. We take the natural logarithm of each value, instead of the initial value. This is done before the train-test split, as transforming the dependent variable does not affect the model training. On the right side of Figure 4, a histogram of the logarithmically transformed transfer fee can be found.

Figure 4

Distribution of the transfer fee before and after logarithmic transformation. On the left, the initial distribution of the transfer fee is visualized. On the right, the logarithmically transformed transfer fee can be found. In both graphs, the values are binned in one hundred bins.



We test for each model whether it performs better without transformation, with logarithmically transformed transfer fees, with standardized independent variables, or with both transformations. The best performing (transformed) data for each model will be used as final data for that model.

4.7 Evaluation Metrics

To judge the difference in the performance of the different models, the R^2 will be used as the main evaluation method. The R^2 is an evaluation metric, which explains to what extent a model can correctly predict outcomes. The higher the R^2, the more accurate the predictions of the model are. The R^2 is calculated as follows:

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y}_i)^2},$$

where $\hat{y}_i$ is the predicted outcome, y_i is the true outcome in the test set and $\bar{y}_i$ is the mean of all values of y.

Another evaluation method that will be used is the Root Mean Square Error ($RMSE$). The $RMSE$ is an often-used evaluation metric that measures the performance of models, according to Chai and Draxler (2014). A low $RMSE$ means that the average error of a model is low. The formula to calculate the $RMSE$ is:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (y_i - \hat{y}_i)^2}{N}}$$ (Wang & Lu, 2018),

where $\hat{y}_i$ is the predicted outcome, y_i is the true outcome in the test set and N is the number of predicted outcomes.

A problem with the $RMSE$ is that models with a logarithmically transformed dependent variable are not comparable with models where this variable is not transformed. For the models where the dependent variable is logarithmically transformed, we transform the logarithmically transformed values and the predicted transfer fees to non-logarithmically transformed values when we calculate the $RMSE$. This is done by taking the exponential with the help of NumPy's exp function (Harris et al., 2020). This way, the calculated $RMSE$ on the test set for the final models with logarithmically transformed transfer fees can be compared to the $RMSE$ of models without logarithmically transformed transfer fees.

The earlier described models, multiple linear regression, support vector regression, and random forest regression will be compared using these two metrics.

4.8 Model Building and Tuning

4.8.1 Baseline Model

To compare the results of the models with a basic model, a baseline model is created. A baseline model is a basic model that serves as a reference. In this research, the baseline model is a simple linear regression, where the independent variable is the league the buying club plays in and the dependent variable is, as in the rest of the models, the transfer fee of a player. We choose the competition, as previous research suggests that this is an influential variable (Yiğit et al., 2020). For the baseline model, it is best to use the logarithmically transformed transfer fee, but not the standardized data. An overview of which transformation provides the best results per model can be found in Appendix C.

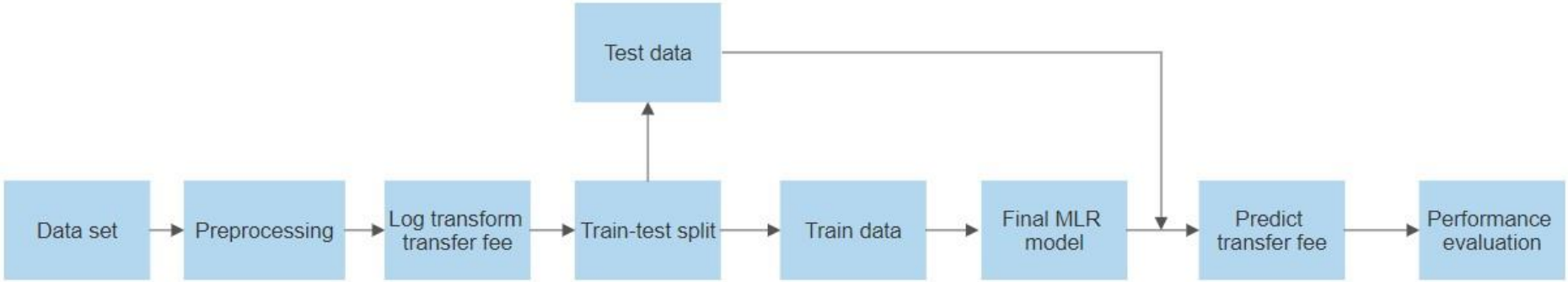
4.8.2 Multiple Linear Regression

The first model is the multiple linear regression model. For this model, Scikit-learns' LinearRegression is used. The model is fitted on the train data, and the model is used to predict the transfer fees for the test data. These predicted outcomes are compared to the actual transfer fees in the test data to evaluate the model.

After performing cross-validation to find out whether we should use logarithmically transformed, standardized, or initial data, we find that it is best to logarithmically transform the transfer fee, but not to use the standardized data. The cross-validated scores for each transformation can be found in Appendix C. The model building process is visualized in Figure 5.

Figure 5

Flowchart of multiple linear regression. MLR stands for multiple linear regression. The transfer fee is predicted by using the test data in the final multiple linear regression model.



4.8.3 Support Vector Regression

Our second model is support vector regression. The hyperparameters of this model are discussed in section 3.2**.** We use Scikit-learns' SVR for this model. For support vector regression, we find that we should use standardized data and that we should logarithmically transform the target variable, as can be seen in Appendix C. When including the kernel as a hyperparameter for this model, the search for the best hyperparameter grid becomes computationally too expensive. Therefore, we try four different kernels of support vector regression ('rbf', 'linear', 'sigmoid', 'poly') to see which kernel performs best. The default settings for the other hyperparameters are used. The 'linear' kernel performs best when using the default settings for the other hyperparameters but when tuning the C for the 'linear' kernel, we decided to shut down the process after it did not finish in six hours. Tuning the hyperparameter of this kernel proves to be computationally too expensive and we decide to use the second-best kernel, namely 'rbf'. Tuning the hyperparameters of the model when using the 'rbf' kernel proves to be computationally less expensive in this research. The scores per kernel are listed in Table 2.

To tune the hyperparameters Scikit-learn's RandomizedSearchCV is first used because it takes a set of random hyperparameter values within a pre-selected range and randomly chooses a predefined number of combinations of hyperparameters. These combinations are evaluated with cross-validation, and the best set of hyperparameters is given from the ones that are inputted. The values that are used in the RandomizedSearchCV and the best parameters according to this search are listed in Table D1, in Appendix D. Based on the best set of hyperparameters from the RandomizedSearchCV, we create a new grid with the best hyperparameters, and values close to the best results from the RandomizedSearchCV. This

Table 2

Support vector regression cross-validated performance scores per kernel without tuning other hyperparameters. The cross-validated $RMSE$ is computed with logarithmically transformed target values.

	Cross-validation score	
Kernel	R^2	$RMSE$
rbf	0.583	0.884
linear	0.600	0.864
sigmoid	0.135	1.270
poly	0.406	1.054

new grid is passed through Scikit-learn's GridSearchCV, which does the same as RandomizedSearchCV but takes all possible combinations of the hyperparameters and outputs the set of hyperparameters that performs best. Grid search has a high precision (Shunjie et al., 2012), but takes more time than the RandomizedSearchCV. We repeat this process a few times until we do not get a new set of best hyperparameters. All values used in the grid searches are listed in Table D2, in Appendix D. The different best sets of hyperparameters and their cross-validated performance can be found in Table 3.

The hyperparameters that will be used to predict the transfer fees of the observations in the test set with the support vector regression model are kernel = 'rbf', C = 38, epsilon = 0.3, and gamma = 0.001. The complete process of the model building of support vector regression is visualized in Figure 6.

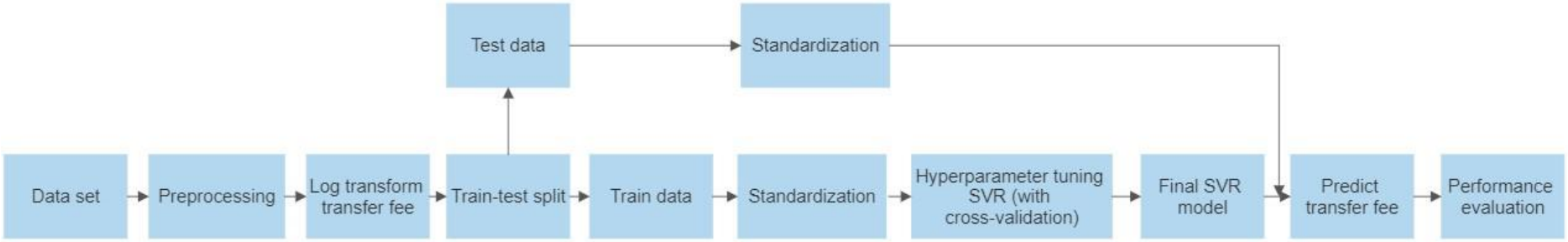
Table 3

The best hyperparameters and the cross-validated R^2 and $RMSE$ per search for support vector regression. GridSearchCV (1) stands for the first round of grid search, and so on. The cross-validated $RMSE$ is computed with logarithmically transformed target values.

	Best hyperparameters			Cross-validation score	
Search	C	Epsilon	Gamma	R^2	$RMSE$
Default parameters	1	0.1	'scale'	0.583	0.884
RandomizedSearchCV	55	0.613	0.001	0.617	0.847
GridSearchCV (1)	40	0.4	0.001	0.620	0.844
GridSearchCV (2)	40	0.3	0.001	0.621	0.843
GridSearchCV (3)	38	0.3	0.001	0.621	0.843

Figure 6

Flowchart of support vector regression. SVR stands for support vector regression. The transfer fee is predicted by using the test data in the final support vector regression model.



4.8.4 Random Forest Regression

The process of the model building of the random forest regression is similar to the support vector regression. For this model, we use Scikit-learns' RandomForestRegressor. The data is standardized after the split, but the transfer fee is not transformed logarithmically (see Appendix C). We try to find the best hyperparameters for this model. The hyperparameters tuned for this model are listed in Table 4, with a short explanation (Pedregosa et al., 2011).

RandomizedSearchCV is used to find the first set of best hyperparameters. This set is, together with approximate values, used to create a new grid with multiple values for each hyperparameter. The input values and the best parameter outputted by the RandomizedSearchCV can be found in Table E1, in Appendix E. This grid is used for a cross-validation grid search, with GridSearchCV, which outputs the best hyperparameter settings. This process is repeated a few times until no new best hyperparameter settings are outputted by the grid search. The values per hyperparameter used in each grid search, and the outputted, best hyperparameters in that search, can be found in Table E2, in Appendix E. The cross-validated performances of each grid, with the best hyperparameters, are listed in Table 5. The settings that give the highest cross-validated score are used to predict the outcomes of the test set.

Table 4

The tuned hyperparameters of the random forest regression model with an explanation from Pedregosa et al. (2011).

Hyperparameter	Explanation
n_estimators	'The number of trees in the forest'
max_features	'The number of features to consider when looking for the best split'
max_depth	'The maximum depth of the tree'
min_samples_split	'The minimum number of samples required to split an internal node'
min_samples_leaf	'The minimum number of samples required to be at a leaf node'
bootstrap	'Whether bootstrap samples are used when building trees'

Table 5

The best hyperparameters and the cross-validated R^2 and $RMSE$ per search for random forest regression. GridSearchCV (1) stands for the first round of grid search, and so on.

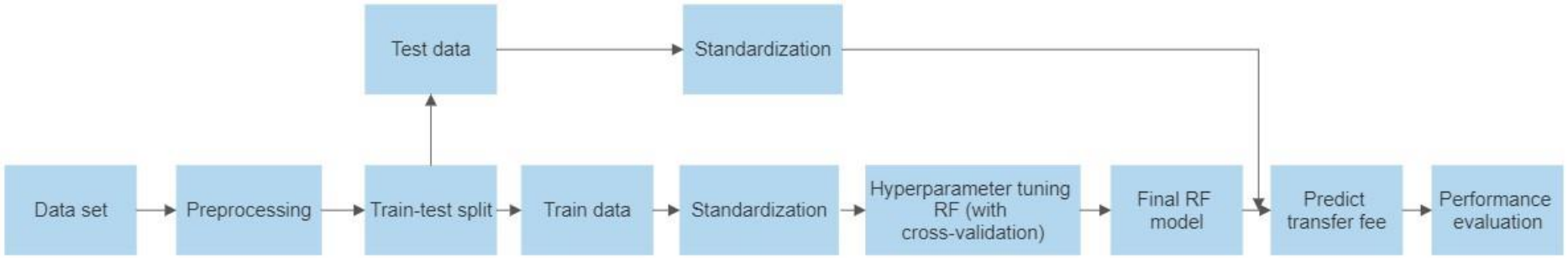
	Best hyperparameters						Cross-validation score	
Search	n_estimators	max_features	max_depth	min_samples_split	min_samples_leaf	bootstrap	R^2	$RMSE$
Default parameters	100	'auto'	None	2	1	True	0.617	6.905
RandomizedSearchCV	176	'auto'	126	2	2	True	0.625	6.803
GridSearchCV (1)	150	'auto'	60	2	2	True	0.625	6.803
GridSearchCV (2)	150	'auto'	60	2	2	True	0.625	6.803

In the end, the best performing hyperparameters are n_estimators = 176, max_features = ‘auto’, max_depth = 60, min_samples_split = 2, min_samples_leaf = 2, and bootstrap = True. This process of the random forest regression model building, and tuning is visualized in Figure 7.

To measure the quality of a split in the random forest regression algorithm, a built-in feature of the random forest regression algorithm of Scikit-learn, named feature_importances_, is used. This function calculates the Gini impurity per feature, which gives insight into the importance of that feature. The higher the value, the more important a feature is for the model (Pedregosa et al., 2011). This will be used to find out what the most important features of our model are, which we need to answer both sub-questions.

Figure 7

Flowchart of random forest regression. RF stands for random forest regression. The transfer fee is predicted by using the test data in the final random forest regression model.



5. Results

In this research, we compare three models to see how well each model predicts the transfer value of professional football players. To do this, a model is built for each algorithm. The performance of the models is compared so that we can find out what the best performing model is. For some models, the transfer fee is logarithmically transformed as this increases the performance. However, to be able to compare the performance of different models with each other, these values are transformed back before calculating the evaluation scores of the test data as described in section 4.7. These evaluation scores are used to compare the final models with each other. The performance of each model is described in Table 6. Note that the cross-validated $RMSE$ scores are not comparable with each other due to the logarithmic transformation of the target values in some of the models.

The R^2 on the test set is 0.673 for the random forest regression, while the other models perform less with 0.615 for the multiple linear regression and 0.633 for the support vector regression. All models outperform the baseline model. The random forest regression gives the best results for the $RMSE$ as well. The results of the research show that the random forest regression can give the best prediction of the transfer value of football players.

Table 6

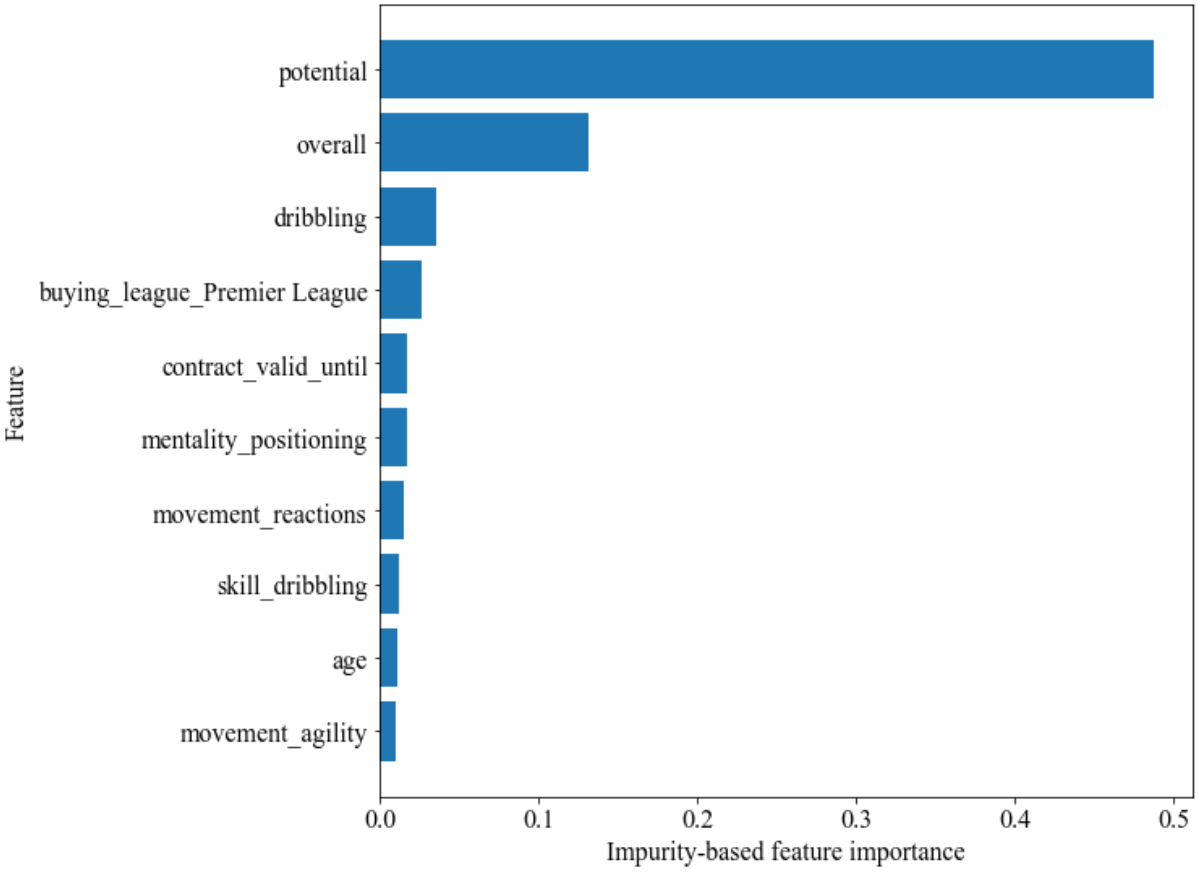
The model performance based on R^2 and $RMSE$. The cross-validated $RMSE$ scores are not comparable with each other as the target values of the baseline, multiple linear regression and support vector regression models are logarithmically transformed, while for the test set the $RMSE$ scores are comparable, as explained in section 4.7.

	Cross-validation score		Test score	
Model	R^2	$RMSE$	R^2	$RMSE$
Baseline	0.204	1.222	0.189	10.439
Multiple Linear Regression	0.590	0.864	0.615	6.322
Support Vector Regression	0.621	0.843	0.633	6.391
Random Forest Regression	0.625	6.837	0.673	6.034

To answer the research question and be able to say to what extent it is possible to predict the transfer value of professional football players, we need to know what features are most important when predicting this value. Previous research suggests that some variables are more important than others. As the random forest regression model gives the best results, we will look at the most important features in this model. The most important variable is 'potential', which indicates how much a player can develop in his career according to FIFA. The second most important feature is 'overall'. This feature shows the current rating of a football player. Among the other most important features are 'dribbling', the buying club playing in the Premier League, the duration of the current contract of a player, 'age', and some specific skills such as reactions. In Figure 8, the 10 most important features in the random forest regression model are visualized.

Figure 8

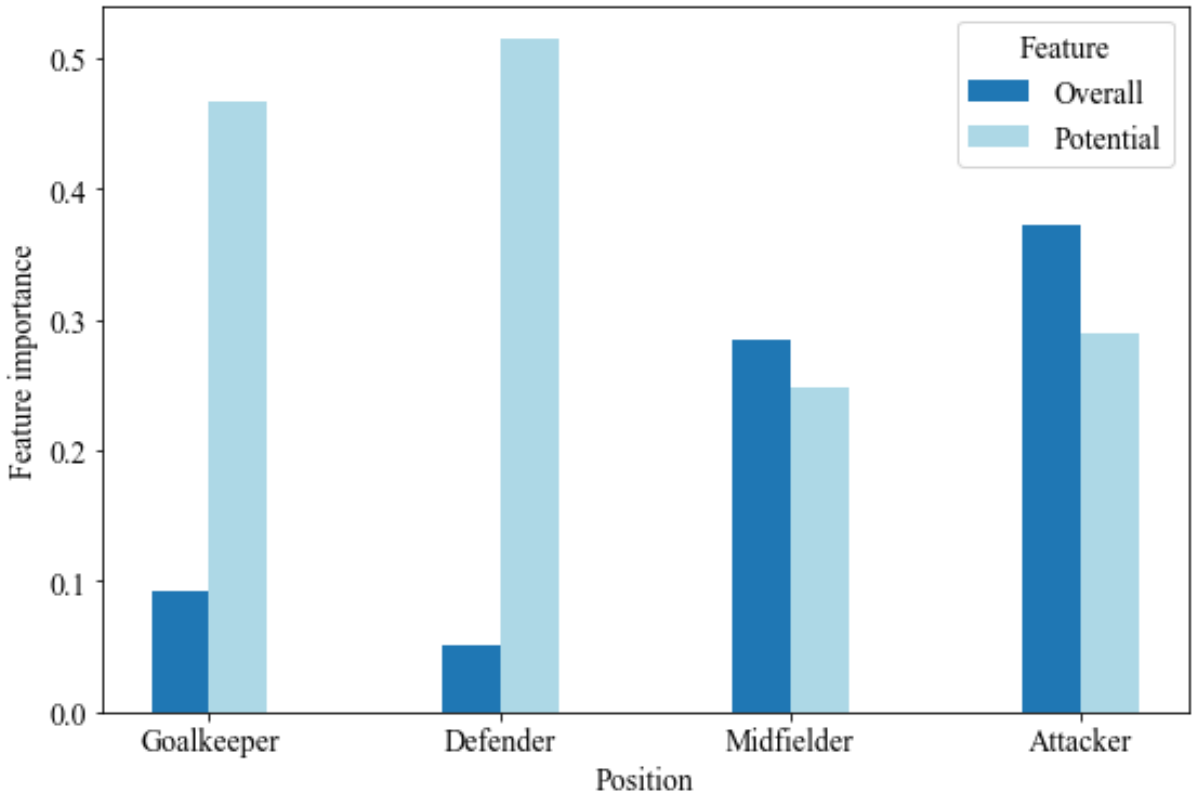
The 10 most important features in the random forest regression model. The higher the impurity-based feature importance, the more important a feature is to the model.



For each position on the field, it is expected that some different skills and characteristics play an important role in the valuation of a football player. To examine this, we split the dataset into four subsets, one for each position on the field (goalkeeper, defender, midfielder, and attacker). The results show that, to some extent, there is a difference in the most important features. For all positions, 'potential' and 'overall' are the most important features to make a good prediction of the transfer value of a player. However, for goalkeepers and defenders, 'potential' is far more important than the 'overall' rating, while these attributes are much closer to each other for midfielders and attackers, as is visualized in Figure 9.

Figure 9

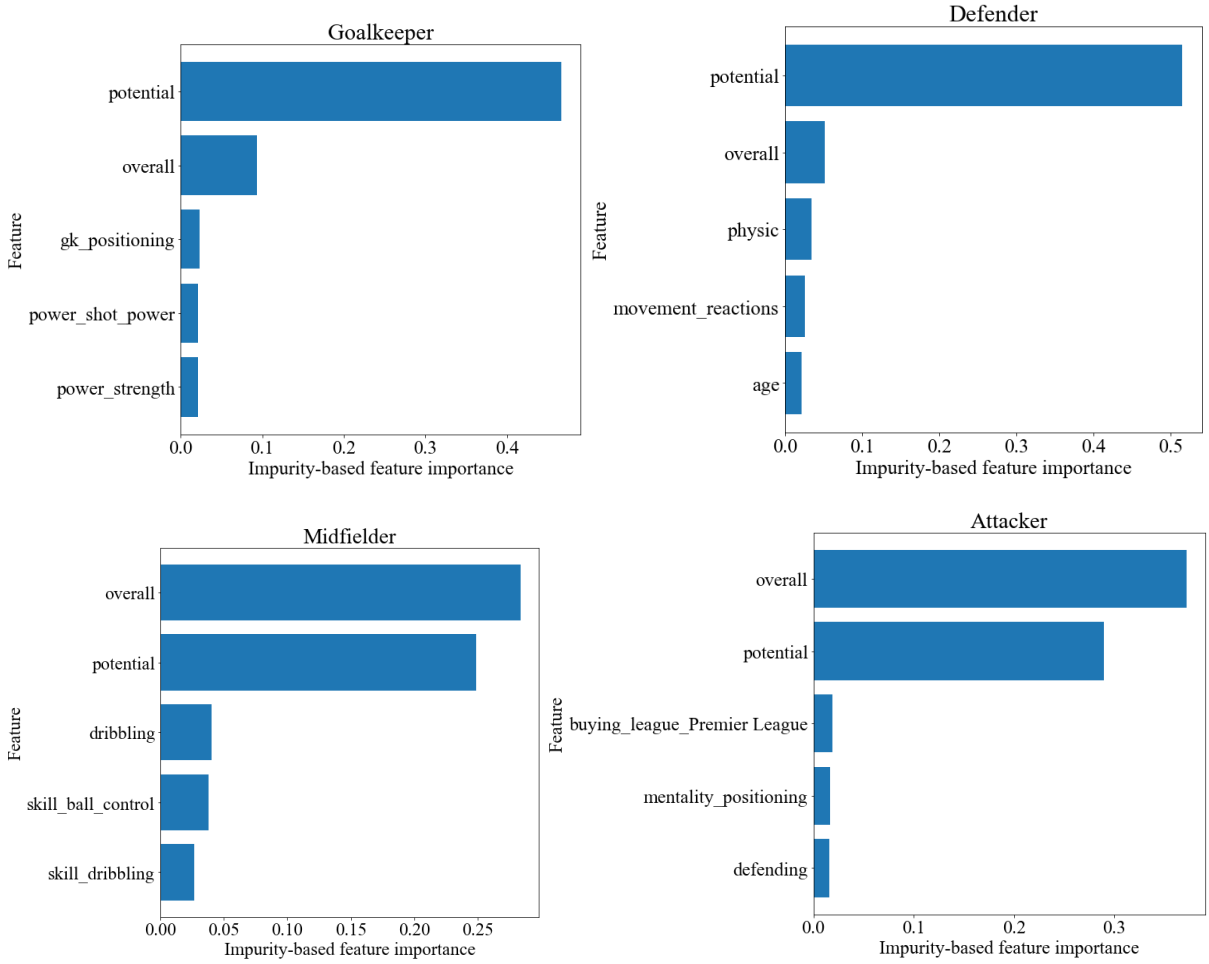
The difference in feature importance of overall and potential per position. The variables 'overall' and 'potential' per position. For goalkeepers and defenders 'potential' is more important, while the difference in importance is smaller for midfielders and attackers.



Besides ‘potential’ and ‘overall’, each position has some specific attributes that are more important for that position in comparison to other positions. For goalkeepers, positioning is more important, but also ‘power_shot_power’. For other positions, there are some other important features. Examples of the most important features per position are ‘physic’ for defenders, ‘dribbling’ for midfielders, and the buying club playing in the Premier League for attackers. The five most important features per position are visualized in Figure 10.

Figure 10

The five most important features per position. The upper left plot illustrates the five most important features for goalkeepers, the upper right does the same for defenders. The bottom left describes the important features for midfielders, and the bottom right visualizes the most important features for attackers. The higher the impurity-based feature importance, the more important a feature is for that position.



6. Discussion

In this section, the results of this research are compared to other papers. Next to that, we discuss the limitations of this research and give recommendations for future research.

6.1 Discussion of Results

The results show that the random forest regression model performs best when it comes to predicting the transfer value of professional football players (R^2 = 0.673). This is in line with earlier research (Al-Asadi & Tasdemir, 2022; Huang et al., 2020), where the best performing model was the random forest regression. However, the results show that our models perform worse in comparison to the best performing models of other studies where the value of football players is predicted (Barbuscak, 2018; Behravan & Razavi, 2021; Inan & Cavas, 2021; Kirschstein & Liebscher, 2019; Müller et al., 2017). Other studies reported a R^2 in the range of 0.74 (Behravan & Razavi, 2021) to 0.99 (Inan & Cavas, 2021).

An explanation for the lower performance of our model could be that, whereas we examined to what extent the transfer value is predictable, other research focused on the predictability of the market value. The transfer value is a real, monetary value, and that might mean that we should consider other variables when predicting it compared to research where the market value is predicted. Some variables that might be interesting to add are the transfer budget of the buying club, as clubs with a higher transfer budget might pay more for a player, or the transfer budget of the selling club because a club with a smaller transfer budget might sell a player for less. These variables are not expected to influence the market value, but they might affect the performance of models predicting the transfer value.

We also discovered what the most important features to predict the transfer value are. Some of these were in line with earlier research and our expectations, such as potential rating (Al-Asadi & Tasdemir, 2022), overall rating (Behravan & Razavi, 2021), and contract duration (Barbuscak, 2018; Carmichael et al., 1999; Frick, 2007). Other features appear to be less important than previous research suggests. Age does not play an important role in our model, while Felipe et al. (2020) found that this is one of the most important features in predicting the market value. According to their research, an older player has in general a lower market value. The difference between our research and the research of Felipe et al. (2020) might be that we used FIFA data in our research, with the feature ‘potential’. Age plays a role in the market value if a young player has the potential to become better. This is covered in our data with the feature ‘potential’ and might make the age variable less important.

Next to that, we found that for different positions, different features are more important, which is in line with our expectations and the research of Behravan and Razavi (2021). Although the majority of the most important features seem logical, some important features differ from our expectations. For example, for defenders, one of the expected main attributes is the skill of defending. Not one defending skill, such as 'defending_sliding_tackle' or 'defending_standing_tackle', made it to the top five of most important features for predicting the transfer value of defenders. Also, for attackers, there are some features rated less important unexpectedly. The most notable is 'attacking_finishing', which relates to the ability of a player to score. An explanation for this could be that in this research the positions have been clustered into four groups. The important features of a center-back may be different compared to a full-back, and the same holds for a striker and a winger.

6.2 Limitations and Recommendations

One of the limitations of this research is that we use video game data to predict the transfer value. Even though much research has shown that well-performing models can be created with the use of football video game data (Behravan & Razavi, 2021; Kirschstein & Liebscher, 2019; Yiğit et al., 2020), some of the values remain subjective and not verifiable. To partly tackle this limitation, we would suggest adding real-life statistics, such as goals scored and duels won, during the season before a transfer to the data. Other studies (Majewski, 2016; Poli et al., 2022) proved that using statistics for value prediction works. Adding this might be interesting for other researchers.

Another limitation of using football video game data is that it could be possible that some features of a player in the games of FIFA are partially based on the transfer fee that has been paid by a football club for a player. For example, if a player is sold for 100 million euros, the creators of the game may choose to increase the overall rating as the player apparently is worth 100 million euros to that club and therefore has to be a top player. We did not take this into account in this research, but for future research it could be interesting to investigate this potential relationship.

Furthermore, we used random grid search and normal grid search to come to the best hyperparameters within the time we had, but it was not possible to run all sensible combinations of parameters. For future research, it could be useful to optimize the hyperparameter grid to see if this affects the model performance.

Next to that, a constraint of this research is that there was no dataset available of FIFA before 2015. Therefore, our dataset was quite small, which limited us in the types of algorithms we could use. In addition, it was not possible to find the most important features

per specific position, as we had to group the specific positions into goalkeeper, defender, midfielder, and attacker. For further research, it would be interesting to see if there is more data that can be used.

Another limitation of the dataset is that not all competitions are in FIFA, meaning that not all players are in FIFA as well. Therefore, not the whole set of transfers between 2015 and 2021 could be used. For future work, we recommend using another football-related video game with all competitions, like Football Manager, instead of FIFA to be able to get a larger dataset.

7. Conclusion

In this research, we wanted to find out to what extent the transfer value of professional football players is predictable based on skills and characteristics. To examine this, we built a model to predict the transfer value of professional football players. Our best-performing model, the random forest regression model (R^2 = 0.673), performs worse than market value prediction models from related work. This could be due to the fact that the market value is a computed value, while the transfer value is the actual paid fee to buy a player. There could be other factors that contribute to the actual transfer fees in comparison to the market value, which we did not consider in this research. This might cause the worse performance. The prediction of the transfer values of professional football players instead of the market value, in combination with the use of multiple machine learning algorithms, is the main novelty of this research as this has not been done before to our knowledge.

This research provides insights into what the most important features are in predicting the transfer value. Some features are characteristics, and others are skill ratings from the video game FIFA, which proved to be a useful source for predicting the transfer value, as nine of the ten most important features in our random forest regression model came from the video game data. The most important features are a players' potential and current overall rating. Previous research suggested taking the different positions in the field into consideration when looking at the most important features, as there is a difference per position. In this research, we found that there is a difference in the most important features per position.

We can conclude that the transfer value is predictable to some extent, with some features being more important than others. The most important features differ per position. The prediction of the transfer value is not as accurate as the market value predictions of previous research. This may be because we used the same variables as previous research predicting the market value. For future research, it might be interesting to include new variables in the model to see if this influences the performance of the model, such as the transfer budget of clubs.

References

Al-Asadi, M. A., & Tasdemir, S. (2022). Predict the Value of Football Players Using FIFA Video Game Data and Machine Learning Techniques. *IEEE Access*, *10*, 22631–22645. https://doi.org/10.1109/ACCESS.2022.3154767

Al-Mejibli, I. S., Alwan, J. K., & Abd, D. H. (2020). The effect of gamma value on support vector machine performance with different kernels. *International Journal of Electrical and Computer Engineering*, *10*(5), 5497–5506. https://doi.org/10.11591/IJECE.V10I5.PP5497-5506

Altman, N., & Krzywinski, M. (2017). Points of Significance: Ensemble methods: Bagging and random forests. In *Nature Methods* (Vol. 14, Issue 10, pp. 933–934). Nature Publishing Group. https://doi.org/10.1038/nmeth.4438

Awad, M., & Khanna, R. (2015). Support Vector Regression. In M. Awad & R. Khanna (Eds.), *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers* (pp. 67–80). Apress. https://doi.org/10.1007/978-1-4302-5990-9_4

Barbuscak, L. (2018). What Makes a Soccer Player Expensive? Analyzing the Transfer Activity of the Richest Soccer. In *Augsburg Honors Review* (Vol. 11). https://idun.augsburg.edu/honors_reviewAvailableat:https://idun.augsburg.edu/honors_review/vol11/iss1/5

BBC. (2017a, August 3). Neymar: Paris St-Germain sign Barcelona forward for world record 222m euros. *BBC*.

BBC. (2017b, August 29). How does a football transfer work? *BBC*.

Behravan, I., & Razavi, S. M. (2021). A novel machine learning method for estimating football players' value in the transfer market. *Soft Computing*, *25*(3), 2499–2511. https://doi.org/10.1007/s00500-020-05319-3

Carmichael, F., Forrest, D., & Simmons, R. (1999). The Labour Market in Association Football: Who Gets Transferred and for How Much? *Bulletin of Economic Research*, *51*(2), 125–150. https://doi.org/https://doi.org/10.1111/1467-8586.00075

Chai, T., & Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? -Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, *7*(3), 1247–1250. https://doi.org/10.5194/gmd-7-1247-2014

Chan, J. C. W., & Paelinckx, D. (2008). Evaluation of Random Forest and Adaboost tree-based ensemble classification and spectral band selection for ecotope mapping using airborne hyperspectral imagery. *Remote Sensing of Environment*, *112*(6), 2999–3011. https://doi.org/10.1016/j.rse.2008.02.011

Chugh, R. S., Bhatia, V., Khanna, K., & Bhatia, V. (2020). A Comparative Analysis of Classifiers for Image Classification. *2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, 248–253. https://doi.org/10.1109/Confluence47617.2020.9058042

Dash, R. K., Nguyen, T. N., Cengiz, K., & Sharma, A. (2021). Fine-tuned support vector regression model for stock predictions. *Neural Computing and Applications*. https://doi.org/10.1007/s00521-021-05842-w

Emioma, C. C., & Edeki, S. O. (2021). Stock price prediction using machine learning on least-squares linear regression basis. *Journal of Physics: Conference Series*, *1734*(1). https://doi.org/10.1088/1742-6596/1734/1/012058

Felipe, J. L., Fernandez-Luna, A., Burillo, P., de la Riva, L. E., Sanchez-Sanchez, J., & Garcia-Unanue, J. (2020). Money talks: Team variables and player positions that most influence the market value of professional male footballers in Europe. *Sustainability (Switzerland)*, *12*(9). https://doi.org/10.3390/su12093709

Frick, B. (2007). THE FOOTBALL PLAYERS' LABOR MARKET: EMPIRICAL EVIDENCE FROM THE MAJOR EUROPEAN LEAGUES. *Scottish Journal of Political Economy*, *54*(3), 422–446. https://EconPapers.repec.org/RePEc:bla:scotjp:v:54:y:2007:i:3:p:422-446

Gupta, S., Zhang, W., & Wang, F. (2017). Model accuracy and runtime tradeoff in distributed deep learning: A systematic study. *Proceedings - IEEE International Conference on Data Mining, ICDM*, 171–180. https://doi.org/10.1109/ICDM.2016.122

Harris, C. R., Millman, K. J., der Walt, S. J. van, Gommers, R., Virtanen, P., David Cournapeau, Wieser, E., Taylor, J., Sebastian Berg, Smith, N. J., Kern, R., Hoyer, M. P. and S., van Kerkwijk, M. H., Matthew Brett, Haldane, A., del Río, J. F.,

Wiebe, M., Peterson, P., Pierre Gérard-Marchant, … Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, *585*(7825), 357–362. https://doi.org/10.1038/s41586-020-2649-2

Henrique, B. M., Sobreiro, V. A., & Kimura, H. (2018). Stock price prediction using support vector regression on daily and up to the minute prices. *Journal of Finance and Data Science*, *4*(3), 183–201. https://doi.org/10.1016/j.jfds.2018.04.003

Hofmann, T., Schölkopf, B., & Smola, A. J. (2008). Kernel methods in machine learning. In *Annals of Statistics* (Vol. 36, Issue 3, pp. 1171–1220). https://doi.org/10.1214/009053607000000677

Huang, J. C., Tsai, Y. C., Wu, P. Y., Lien, Y. H., Chien, C. Y., Kuo, C. F., Hung, J. F., Chen, S. C., & Kuo, C. H. (2020). Predictive modeling of blood pressure during hemodialysis: a comparison of linear model, random forest, support vector regression, XGBoost, LASSO regression and ensemble method. *Computer Methods and Programs in Biomedicine*, *195*. https://doi.org/10.1016/j.cmpb.2020.105536

Inan, T., & Cavas, L. (2021). Estimation of Market Values of Football Players through Artificial Neural Network: A Model Study from the Turkish Super League. *Applied Artificial Intelligence*, *35*(13), 1022–1042. https://doi.org/10.1080/08839514.2021.1966884

Jiang, D., Wu, Z., Hsieh, C.-Y., Chen, G., Liao, B., Wang, Z., Shen, C., Cao, D., Wu, J., & Hou, T. (2021). Could graph neural networks learn better molecular representation for drug discovery? A comparison study of descriptor-based and graph-based models. *Journal of Cheminformatics*, *13*(1), 12. https://doi.org/10.1186/s13321-020-00479-8

Kirschstein, T., & Liebscher, S. (2019). Assessing the market values of soccer players–a robust analysis of data from German 1. and 2. Bundesliga. *Journal of Applied Statistics*, *46*(7), 1336–1349. https://doi.org/10.1080/02664763.2018.1540689

Majewski, S. (2016). Identification of factors determining market value of the most valuable football players. In *Journal of Management and Business Administration. Central Europe* (Vol. 24, Issue 3, pp. 91–104). Sciendo. https://doi.org/10.7206/jmba.ce.2450-7814.177

Mockus, A. (2008). Missing Data in Software Engineering. In J. and S. D. I. K. Shull Forrest and Singer (Ed.), *Guide to Advanced Empirical Software Engineering* (pp. 185–200). Springer London. https://doi.org/10.1007/978-1-84800-044-5_7

Müller, O., Simons, A., & Weinmann, M. (2017). Beyond crowd judgments: Data-driven estimation of market value in association football. *European Journal of Operational Research*, *263*(2), 611–624. https://doi.org/10.1016/j.ejor.2017.05.005

Navyashree, M., Navyashree, M. K., Pooja, G. R., & Biradar, A. (2019). *Salary Prediction in It Job Market*. https://doi.org/10.26438/ijcse/v7si15.7884

Pal, M., & Bharati, P. (2019). Applications of Regression Techniques. In *Applications of Regression Techniques*. Springer Singapore. https://doi.org/10.1007/978-981-13-9314-3

Parbat, D., & Chakraborty, M. (2020). A python based support vector regression model for prediction of COVID19 cases in India. *Chaos, Solitons and Fractals*, *138*. https://doi.org/10.1016/j.chaos.2020.109942

Parrish, C., & Naurright, J. (2014). *Soccer around the World: A Cultural Guide to the World's Favorite Sport*. ABC-CLIO. https://books.google.nl/books?id=N6qSAwAAQBAJ

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel V. and Thirion, B., Grisel, O., Blondel, M., Prettenhofer P. and Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Poli, R., Besson, R., & Ravenel, L. (2022). Econometric Approach to Assessing the Transfer Fees and Values of Professional Football Players. *Economies*, *10*(1). https://doi.org/10.3390/economies10010004

Pontil, M., & Verri, A. (1998). Properties of Support Vector Machines. *Neural Computation*, *10*(4), 955–974. https://doi.org/10.1162/089976698300017575

Segal, M. R. (2003). *UCSF Recent Work Title Machine Learning Benchmarks and Random Forest Regression Publication Date Machine Learning Benchmarks and Random Forest Regression*.

Shunjie, H., Qubo, C., & Meng, H. (2012). Parameter selection in SVM with RBF kernel function. In *World Automation Congress 2012*.

Smola, A. J., & Schölkopf, B. (2004). A tutorial on support vector regression *. In *Statistics and Computing* (Vol. 14). Kluwer Academic Publishers.

UEFA. (2021a). *Distribution to clubs from the 2021/22 UEFA Champions League, UEFA Europa League and UEFA Europa Conference League and the 2021 UEFA Super Cup Payments for the qualifying phases Solidarity payments for non-participating clubs*.

UEFA. (2021b, September 2). *UEFA EURO 2020 impresses with 5.2 billion cumulative global live audience*. https://www.uefa.com/insideuefa/about-uefa/news/026d-132519672495-56a014558e80-1000--uefa-euro-2020-impresses-with-5-2-billion-cumulative-global-liv/

UEFA. (2022, February 16). *Country coefficients*. Https://Www.Uefa.Com/Nationalassociations/Uefarankings/Country/#/Yr/2022.

Uyanık, G. K., & Güler, N. (2013). A Study on Multiple Linear Regression Analysis. *Procedia - Social and Behavioral Sciences*, *106*, 234–240. https://doi.org/10.1016/j.sbspro.2013.12.027

Vapnik, V. N. (1999). An overview of statistical learning theory. In *IEEE Transactions on Neural Networks* (Vol. 10, Issue 5, pp. 988–999). https://doi.org/10.1109/72.788640

Walker, C. (2021, August 31). Manchester United will recover the costs of the sensational signing of Cristiano Ronaldo... with analysts expecting a windfall of £30 million within 12 months as sponsors line up to cash in on "a match made in heaven." *Daily Mail*.

Wang, W., & Lu, Y. (2018). Analysis of the Mean Absolute Error (MAE) and the Root Mean Square Error (RMSE) in Assessing Rounding Model. *IOP Conference Series: Materials Science and Engineering*, *324*(1). https://doi.org/10.1088/1757-899X/324/1/012049

Yiğit, A. T., Samak, B., & Kaya, T. (2020). Football Player Value Assessment Using Machine Learning Techniques. In S. and C. O. S. and O. B. and T. A. C. and S. I. U. Kahraman Cengiz and Cebi (Ed.), *Intelligent and Fuzzy Techniques in Big Data Analytics and Decision Making* (pp. 289–297). Springer International Publishing. https://link-springer-com.tilburguniversity.idm.oclc.org/chapter/10.1007/978-3-030-23756-1_36

Zhang, D., & Lou, S. (2021). The application research of neural network and BP algorithm in stock price pattern classification and prediction. *Future Generation Computer Systems, 115*, 872–879. https://doi.org/10.1016/j.future.2020.10.009

Zhang, Y., Tang, J., Liao, R., Zhang, M., Zhang, Y., Wang, X., & Su, Z. (2021). Application of an enhanced BP neural network model with water cycle algorithm on landslide prediction. *Stochastic Environmental Research and Risk Assessment*, *35*(6), 1273–1291. https://doi.org/10.1007/s00477-020-01920-y

Appendix A

Variables in the Datasets

Table A1

Variables in the transfer dataset. The 'Used in final dataset?' column gives information about whether the variable is in the dataset, which is used for the modelling, where an 'x' means that the variable is used in the final dataset.

Variable	Short explanation	Used in final dataset?
club_name	Club of the player	
player_name	Name of the player	
age	Age of the player at the time of the transfer	x
position	Most commonly played position for the player	x
club_involved	Club on the other side of the transfer	
fee	Transfer fee paid for the player with special characters	
transfer_movement	Incoming or outgoing transfer (from the view of club_name)	
transfer_period	Period of the transfer	x
fee_cleaned	Transfer fee in millions of pounds	x
league_name	Name of the league the club_name plays in	x
year	Year of the transfer	x
season	Season of the transfer	

Table A2

Variables in the FIFA dataset. The 'Used in final dataset?' column gives information about whether the variable is in the dataset, which is used for the modelling, where an 'x' means that the variable is used in the final dataset.

Variable	Short explanation	Used in final dataset?
sofifa_id	ID on sofifa.com	
player_url	URL to player profile on sofifa.com	
short_name	Short name of the player	
long_name	Full name of the player	
age	Age of the player	x
dob	Date of birth of the player	
height_cm	Height of the player in centimeters	x
weight_kg	Weight of the player in kilograms	x
nationality	Nationality of the player	x
club_name	Name of the club of the player	
league_name	League the club plays in	x
league_rank	Rank of the league the club plays in	
overall	Overall rating of the player	x
potential	Potential, future rating of the player	x
value_eur	Value of the player according to FIFA	
wage_eur	Wage of the player according to FIFA	
player_positions	Positions of the player	
preferred_foot	Preferred foot of the player	x
international_reputation	International reputation of the player	x
weak_foot	Level of the weak foot of the player (1-5)	x
skill_moves	Level of the skill moves of the player (1-5)	x
work_rate	Work rate of the player	
body_type	Body type of the player	
real_face	Whether the player has a real face on his character in FIFA	
release_clause_eur	The release clause in euros of the player according to FIFA	
player_tags	Player tags in FIFA	
team_position	Position with their team in FIFA	
team_jersey_number	Jersey number with their team in FIFA	
loaned_from	Only applies when a player is loaned	
joined	When a player joined their club	
contract_valid_until	Until when the contract of a player is valid	x
nation_position	Position with their nation in FIFA	
nation_jersey_number	Jersey number with their nation in FIFA	
pace	Rating for pace (only field players)	x
shooting	Rating for shooting (only field players)	x
passing	Rating for passing (only field players)	x
dribbling	Rating for dribbling (only field players)	x
defending	Rating for defending (only field players)	x

Table 2 Continued

Variable	Short explanation	Used in final dataset?
physic	Rating for physical (only field players)	x
gk_diving	Rating for diving (only for goalkeepers)	x
gk_handling	Rating for handling (only for goalkeepers)	x
gk_kicking	Rating for kicking (only for goalkeepers)	x
gk_reflexes	Rating for reflexes (only for goalkeepers)	x
gk_speed	Rating for speed (only for goalkeepers)	x
gk_positioning	Rating for positioning (only for goalkeepers)	x
player_traits	Special traits applicable to a player	
mentality_positioning	Subcategory of shooting	x
attacking_finishing	Subcategory of shooting	x
power_shot_power	Subcategory of shooting	x
power_long_shots	Subcategory of shooting	x
attacking_volleys	Subcategory of shooting	x
mentality_penalties	Subcategory of shooting	x
mentality_vision	Subcategory of passing	x
attacking_crossing	Subcategory of passing	x
skill_fk_accuracy	Subcategory of passing	x
attacking_short_passing	Subcategory of passing	x
skill_long_passing	Subcategory of passing	x
skill_curve	Subcategory of passing	x
movement_acceleration	Subcategory of pace	x
movement_sprint_speed	Subcategory of pace	x
movement_agility	Subcategory of dribbling	x
movement_balance	Subcategory of dribbling	x
movement_reactions	Subcategory of dribbling	x
skill_ball_control	Subcategory of dribbling	x
skill_dribbling	Subcategory of dribbling	x
mentality_composure	Subcategory of dribbling	x
mentality_interceptions	Subcategory of defending	x
defending_standing_tackle	Subcategory of defending	x
defending_sliding_tackle	Subcategory of defending	x
defending_marking	Subcategory of defending	x
power_jumping	Subcategory of physical	x
power_stamina	Subcategory of physical	x
power_strength	Subcategory of physical	x
mentality_aggression	Subcategory of physical	x
ls	The potential of the player on the position ‘left striker’	
st	The potential of the player on the position ‘central striker’	
rs	The potential of the player on the position ‘right striker’	

lw	The potential of the player on the position ‘left winger’	

Table A2 Continued

Variable	Short explanation	Used in final dataset?
lf	The potential of the player on the position ‘left forward’	
cf	The potential of the player on the position ‘central forward’	
rf	The potential of the player on the position ‘right forward’	
rw	The potential of the player on the position ‘right winger’	
lam	The potential of the player on the position ‘left attacking midfielder’	
cam	The potential of the player on the position ‘central attacking midfielder’	
ram	The potential of the player on the position ‘right attacking midfielder’	
lm	The potential of the player on the position ‘left midfielder’	
lcm	The potential of the player on the position ‘left central midfielder’	
cm	The potential of the player on the position ‘central midfielder’	
rcm	The potential of the player on the position ‘right central midfielder’	
rm	The potential of the player on the position ‘right midfielder’	
lwb	The potential of the player on the position ‘left wing-back’	
ldm	The potential of the player on the position ‘left defensive midfielder’	
cdm	The potential of the player on the position ‘central defensive midfielder’	
rdm	The potential of the player on the position ‘right defensive midfielder’	
rwb	The potential of the player on the position ‘right wing-back’	
lb	The potential of the player on the position ‘left-back’	
lcb	The potential of the player on the position ‘left center back’	
cb	The potential of the player on the position ‘center back’	
rcb	The potential of the player on the position ‘right center back’	

rb	The potential of the player on the position ‘right-back’

Appendix B

Python Packages Used in Each Research Step

Table B1

The python packages used in each research step.

Research step	Python package
Load dataset	Pandas
Clean dataset	Pandas
Merge dataset	Pandas
Rename variables	Pandas
Transform categoric variables to dummies	Pandas
Split data per position	Pandas
Transform data	NumPy, Scikit-learn
Split data in train-test set	Scikit-learn
Cross-validation	Scikit-learn
Linear Regression	Scikit-learn
Support Vector Regression	Scikit-learn
Random Forest Regression	Scikit-learn
RandomizedSearchCV and GridSearchCV	Scikit-learn
Feature importance of Random Forest Regression	Scikit-learn
Evaluation metrics	Scikit-learn
Visualizations	Matplotlib

Appendix C

Model Performance with Different Transformations

Table C1

The model performance with different transformations. The R^2 is calculated with cross-validation. For each of the models, the default model, without hyperparameter tuning, is used to see which transformation is best. The bold values are the highest value for a model, and therefore that transformation will be used for that particular model.

	Transformation			
	No transform	Logarithmic	Standardization	Both
Model	R^2	R^2	R^2	R^2
Baseline	0.125	**0.204**	0.125	0.204
Multiple Linear Regression	0.457	**0.600**	-0.000	-0.000
Support Vector Regression	-0.110	0.116	0.321	**0.583**
Random Forest Regression	0.616	0.614	**0.617**	0.614

Appendix D

Hyperparameter Tuning of Support Vector Regression

Table D1

Input values and best hyperparameters with RandomizedSearchCV for support vector regression. For every hyperparameter, random numbers that are within a logical range for that variable are used in the grid. Best parameter is the best parameter according to the RandomizedSearchCV.

Hyperparameters	Input values	Best parameter
C	50 random integers between 1 and 100	55
Epsilon	50 random floats between 0.001 and 1	0.613
Gamma	50 random floats between 0.001 and 1, and 'scale'	0.001

Table D2

Input values and best hyperparameters per GridSearchCV for support vector regression. GridSearchCV (1) stands for the first round of grid search, and so on. Input contains the inputted values for each hyperparameter per grid search, where the bold values where the best hyperparameters in that grid.

	GridSearchCV (1)	GridSearchCV (2)	GridSearchCV (3)
Hyperparameter	Input	Input	Input
C	**40**, 47, 55, 63, 70	15, 25, 30, 35, **40**	**38**, 40, 42
Epsilon	0.1, **0.4**, 0.613, 1	0.1, **0.3**, 0.4, 0.5, 1	0.2, **0.3**, 0.4
Gamma	'scale', **0.001**	**0.001**	**0.001**

Appendix E

Hyperparameter Tuning of Random Forest Regression

Table E1

Input values and best hyperparameters with RandomizedSearchCV for random forest regression. For every hyperparameter, random numbers that are within a logical range for that variable are used in the grid. Best parameter is the best parameter according to the RandomizedSearchCV.

Hyperparameters	Input values	Best parameter
n_estimators	50 random integers between 10 and 200	176
max_features	'auto' and 'sqrt'	'auto'
max_depth	50 random integers between 10 and 200	126
min_samples_split	18 random integers between 2 and 20	2
min_samples_leaf	50 random integers between 1 and 50	2
bootstrap	True and False	True

Table E2

Input values and best hyperparameters per GridSearchCV for random forest regression. GridSearchCV (1) stands for the first round of grid search, and GridSearchCV (2) for the second round and last round. Input contains the inputted values for each hyperparameter per grid search, where the bold values where the best hyperparameters in that grid.

	GridSearchCV (1)	GridSearchCV (2)
Hyperparameter	Input	Input
n_estimators	120, **150**, 176	**150**, 176
max_features	**'auto'**	**'auto'**
max_depth	**60**, 80, 126	**60**, 126
min_samples_split	**2**, 6, 10	**2**
min_samples_leaf	**2**, 4, 8	**2**
bootstrap	**True**	**True**