Putting a Price Tag on Football Players

Bjørn Skeel-Gjørling, Chritian Lund Sørensen,

Guillaume Slizewicz & Amer Skaljic

26 August 2016

1 Introduction

The purpose of this paper is to predict the transfer value of football players in the five major European leagues.[1] We use a variety of methods from the field of data science to gather, study and lastly predict footballers' transfer value. The paper is structured in three sections.[2] In the first section we briefly present our data scraping processes, our sources, the nature of our data and the ethical challenges we considered. In the second section the dataset is described using visualisations in order to get an overview of the different observations and variables which seem to affect the transfer values. In the final section, we use different prediction approaches including statistical learning models to predict the transfer values of footballers. The emphasis will be on testing different models and comparing their predictive power on a test set. The paper concludes that the random forest model is slightly better at predicting the footballers' transfer value compared to the rest of the presented models in the paper.

2 Data Gathering

Our data scraping process relied heavily on the websites Transfermarkt.co.uk, and to a lesser extent on Google.com and Wikipedia.org. Transfermarkt.co.uk contains information on football transfers, player background and performance statistics. We expect most of this information to be relevant for predicting the transfer value of a player. We used Wikipedia.org

[1] Premier League (England), Bundesliga (Germany), La Liga (Spain), Serie A (Italy) and Ligue 1 (France).

[2] All calculations, graphics and writing was conducted using the programming language R and RStudio.

to find data about the final table ranking for the five leagues in the season before the transfers (season 14/15). To extract the relevant data we built three web scrapers. One for Transfermarkt, another for Google and a third for Wikipedia. The scraper for Transfermarkt both collected all transfer information from the transfer windows “Summer 15” and “Winter 16” in the five leagues and the individual performance statistics of each transferred player. We then used the Google-scraper to gather information on the number of google hits when searching for the name of each transferred player followed by the word “footballer”. Google hits serves as a proxy for a player’s popularity, in addition, it is likely to correlate with attributes that are not measured in the performance statistics. Information on transfer, performance statistics and google hits were then merged into one data frame consisting of players’ data. The scraper for Wikipedia ran through the different webpages with the table ranking for the season 14/15 in the five leagues. The table rankings were then merged into one data frame containing the clubs’ data. The two different datasets were first cleaned and then merged into one combined dataset containing only the variables thought to be relevant in predicting transfer values.

Our conviction is that there is no substantial ethical issues in scraping the data from Google and Wikipedia. This information is publicly available and scraping it does not endanger the firms’ business viability. On the other hand, it can be argued that the performance and transfer data is the core of Transfermarkt’s business model and they therefore have the right to privacy. In that regard, we would like to stress that we only use the data for scientific purposes and not for commercial purposes. An illustration of the data gathering process is provided below.

2.1 Data Cleaning

In order to do visualisations and predictions we had to clean the raw data. Our experience was that Transfermarkt was inconsistent in their way of handling statistics which made the cleaning process challenging.

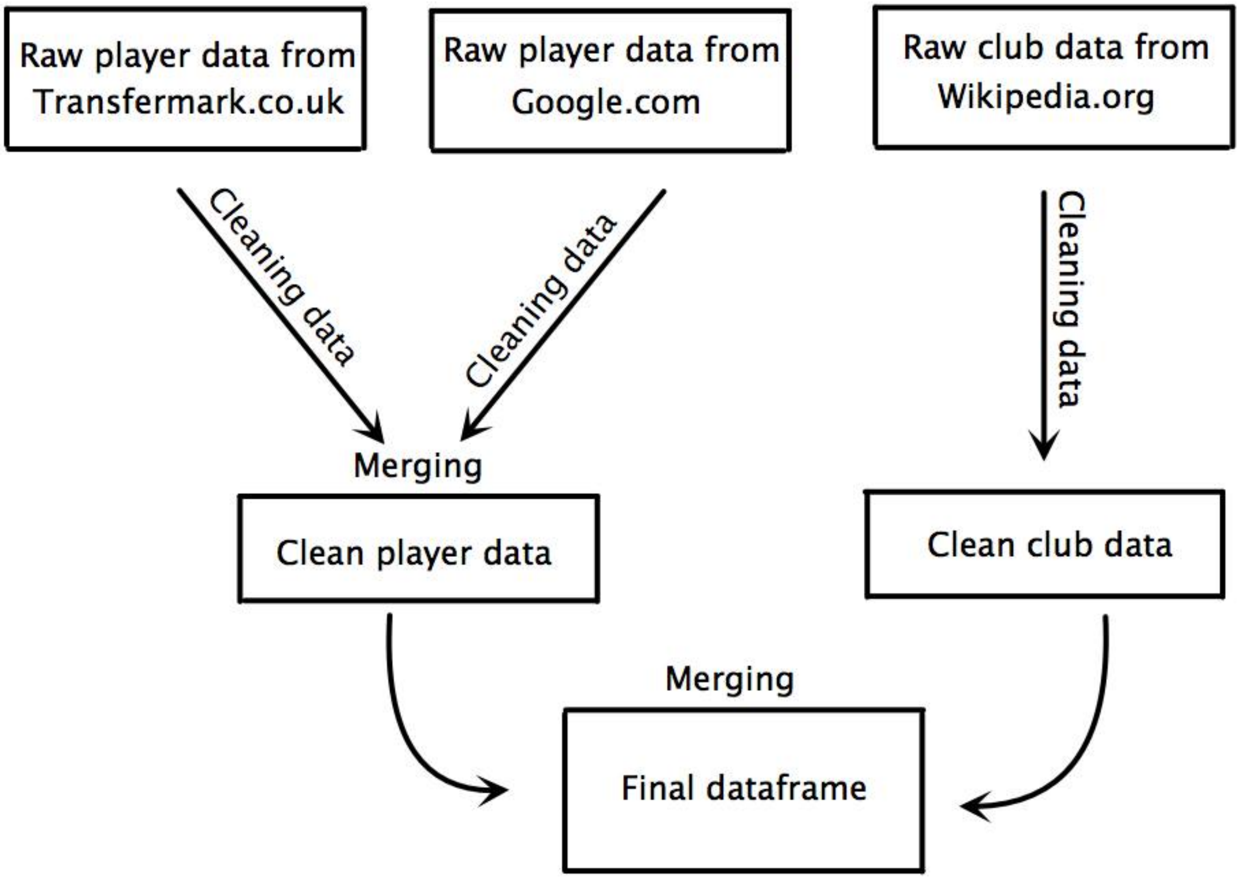

Raw player data from Transfermark.co.uk
Raw player data from Google.com
Raw club data from Wikipedia.org
Cleaning data
Cleaning data
Cleaning data
Merging
Clean player data
Clean club data
Merging
Final dataframe

Figure 1: Illustration of the data gathering process

2.1.1 Cleaning The Player Dataset

For the players' data we first turned all unavailable data into *NA*. Furthermore, we turned the transfer date, end date of contract and birth date into the correct date-format used by R. The main reason for this process was to be able to calculate the precise age (in years) and contract period left (in months) at the day of transfer. Afterwards, we removed all keepers, duplicated observations and observations for which the transfer fee is unknown. The reason for removing the keepers is that our scraped performance data is mostly offensive statistics, which obviously do not describe the performance of goalkeepers and thereby affect their transfer value. At last we simplified the position variable so it only contains three categories: Defender, Midfielder and Attacker.

2.1.2 Cleaning The Club Dataset

For the club dataset we started by grouping the different clubs in 4 categories: Top Club (the five highest ranking clubs in the season 14/15), Middle Club (the following ten clubs, ranking from six to fifteen that season), Bottom Club (the remaining clubs in the league that season) and Promoted Club (clubs that entered the league in season 15/16). Afterwards we renamed the clubs so they matched the player dataset, which enabled us to merge the player data set and the club data set. Lastly, we selected the club variables that we found the most interesting for the prediction (club name, club status and league) and removed all other variables.

2.1.3 The Final Dataset

The final and tidy dataset contains 369 observations and 25 variables. All the variables are listed with a short description in the appendix. As stated, the final data set contains a lot of variables. We will therefore in the following section describe how we used different visualisations to explore and select the different variables that seemed the most suited for predicting transfer values.

3 Data Visualisation

The objective of this section is to highlight possible correlations and important variables via the use of data visualisation. The creation of maps, scatter plots and bar charts enabled us to have a better understanding of our dataset and convey its most important characteristic. After experimenting different scatterplots, we were able to have a better grasp of the relationship between transfer fees and other variables such as age, position, appearance, total goals and time left on contract. Due to space limitations we only include visualisations and detailed descriptions for a handfull of the predictors used in the later models.[3] In order to transform our dataset into graphs and maps we used different digital tools: GGplot for scatterplots and bar charts and Google maps API for maps. The dataset used for these visualisations was cleaned by removing all observations with a transfer fee equal to zero.

3.1 A European Market Dominated by the UK.

Figure 2 below is a mapping of all the clubs buying players. The size of the dots correspond to the amount of money spent on transfers. The map allows us to check the validity of our sample by comparing it to the most well known clubs. Unsurprisingly, the Premier League appears to have more clubs with a higher spending, furthermore, the most reknown clubs such as Real Madrid, Barcelona and PSG clearly appear on the map.

Our second visualisation (figure 3)[4] illustrates the transfer path of each transfered player in the dataset. This highlights the fact that the market for football players is a European market, or even a global one and that country borders do not seem to play a major role in transfering players. As you can see on the map, most of the players are traded between countries and not within them.

[3] The list of the importance of each predictor is provided in the appendix. %IncMSE of the different predictors are listed in the table below.

[4] For simplification reasons, transfers from outside Europe are not represented here.

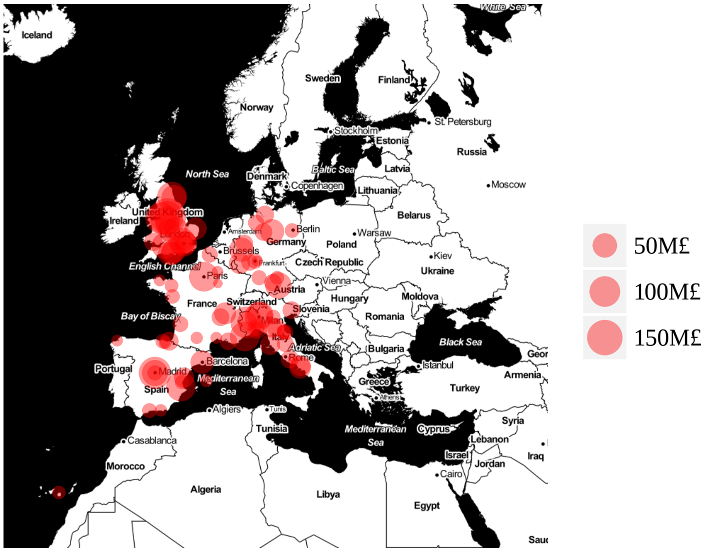


Figure 2: The European transfer market

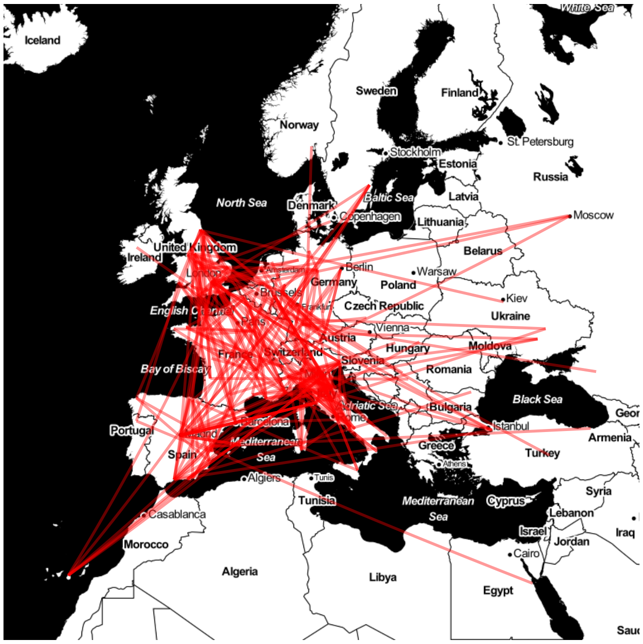


Figure 3: Maps of transfer paths

3.2 Age and Contract Length Seem to Influence Transfer Value

The following two scatterplots (figure 4) indicate that age and period left on contract correlate with a player's transfer value. The scatterplot to the left shows the relationship between age and transfer value. The observations indicate a bell-shaped relationship with a flat top around 25 years. Perhaps forseeably, players in their prime (between 20 and 27 years old) are traded for higher prices. One interpretation might be that the promises of a rising star are more sought after than the celebrity of an older player. Due to bell-shaped relationsship we include age squared as a predictor in the later models.

The scatterplot to the right highlights a correlation between the time left of a player's contract and his transfer value. As you could expect, the more time left on a contract, the higher the price. When the period is less than 20 months, the transfer fees do not exceed 30 million pounds in our sample. When more than 20 months, the transfer fees goes up to 30 million pounds in our sample. Some transfer fees even reach 40 million pounds, when the time left on the contract exceeds 40 months.

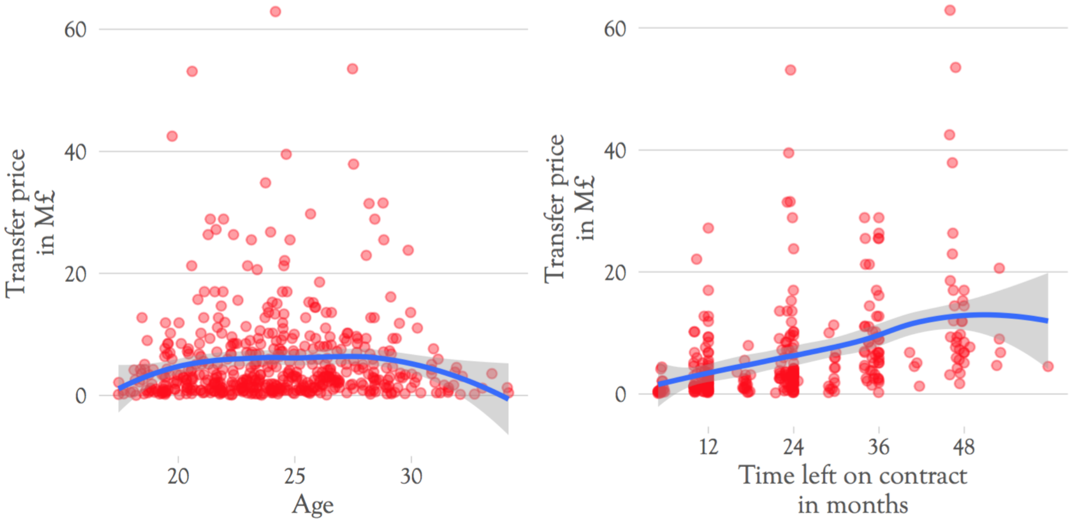
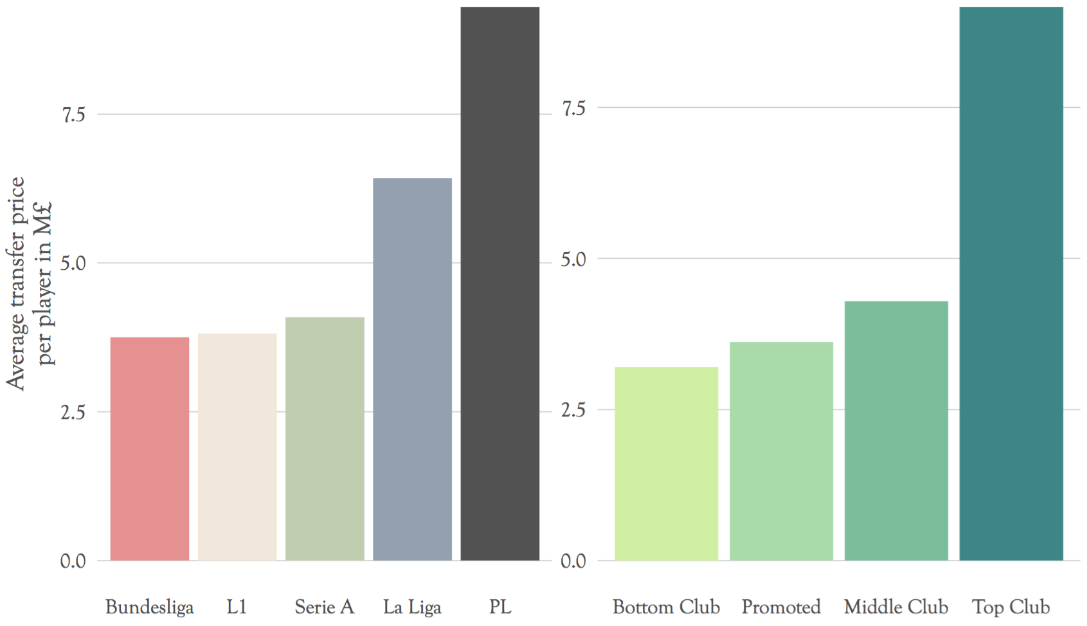


Figure 4: Scatter plots

3.3 League and Status of the Buying Club Seem to Affect the Transfer Value

The following two bar charts (figure 5) show two relationships. The left graph shows the relationship between transfer value and league of the buying club. The right graph shows the relationship between the transfer value and club status of the buyer. The first graph clearly shows, that clubs from the Premier League spend much more money on each player compared to the clubs from the other leagues. The league of the buying club could therefore be a good indicator of a players transfer value. A reason for this pattern could be Premier League's popularity all over the world, resulting in higher TV rights income, bigger sales of fan merchandise and a higher attendance during british matches. Both being able to buy expensive players and the ambition to remain at the center of international football push british clubs into buying more expensive players (also shown on the map in figure 2).
The second graph shows that the more successful a club is, the more money it can afford to spend on transfers. One interesting feature of this graph is that the promoted clubs are spending more money on transfers than bottom clubs. This can be explained by their intention to improve their roster in order to be able to compete in the major league. The graph therefore indicates that the ranking of the buying club in the league in the previous season can be used as a signal for the transfer value of a player.



In the appendix we also have a bar chart of the average spending on transfers per club in the different leagues. The visualisations support the inclusion of the mentionned variables in the prediction models.

4 Prediction Models for Transfer Value

4.1 Prediction versus causality

The main purpose of this section is to find the best way of predicting transfer values. We therefore focus on finding the most efficient way to estimate our dependent variable when we know the value of several different independent variables - hereafter called predictors. The issue of prediction stands in contrast to the regular goal in social sciences where we want to estimate the causal effect of one particular independent variable on the dependent variable. The key difference is that the focus shifts from the individual effect of a predictor (effects of causes) to the interplay between several predictors and how this interplay can help us predict the value of the dependent variable. Due to this scientific purpose we use the predictors as signals of a specific transfer value and do not pay attention to whether they come before the dependent variable in time or whether there is an omitted variable bias. A good example of this is our use of google hits as a predictor for the players' transfer value. The amount of hits is recorded in real time and therefore after the transfer was made. Due to the time order, the recorded number of google hits cannot be the reason why the club bought the particular player for a given price. On the other hand a high number of google hits can signal that the player has had a successful career or was part of a news generating transfer. The predictors used in this section are listed in the appendix.

4.2 Evaluating prediction models using RMSE

To find the best way of predicting a football player's transfer value we will make a comparative analysis of the accuracy of five different models. The models we will use to predict the transfer values are 1) simple average, 2) ordinary least squares, 3) lasso model, 4) decision tree and 5) random forest.

We measure the models' prediction accuracy by finding the *out of sample error* (the prediction error when applying the models on a dataset on which they were not trained). The first step of our comparative analysis is therefore to randomly split our data into a *training sample* and a *test sample*. We chose to put 70 percent of the observations in the training sample because we want a sufficiently large amount of observations to train our models.
We evaluate and compare the models by using the root mean squared error (RMSE) which is stated in the equation below. The RMSE measures how well the model predicts the observations in the test sample. We created a function in R to calculate the RMSE. We use the RMSE instead of the mean square error (MSE) because RMSE is more robust to outliers.

$$RMSE = \sqrt{\frac{\sum\limits_{i=1}^{n} (y_i - \hat{y}_i)^2}{n}}$$

The *bias-variance trade-off* is important to keep in mind when dealing with prediction. The reason is that the prediction errors are the sum of errors due to bias and errors due to variance between the training and test sample. When we build our prediction models, we do it on the available data in the training sample which also include standard noise. If we on the one hand build a model with only a few variables the model will not be able to comprehend the complexity of the real world. Due to this *underfitting* we end up with a biased model with estimates quite far from the real world values. If we on the other hand increase the number of variables and thereby the model's complexity, then the model will adjust too much to the training sample. The result of this *overfitting* will be prediction errors due to the variance between the training sample and the real world. The take-away point is that the most accurate models have found a balance between over- and underfitting so that the increase in bias is equivalent to the decrease in variance. The trade-off is illustrated in the appendix.

4.3 Simple Average

We use a simple average as the first prediction model. This is a good baseline model because it is the simplest way of estimating a player's transfer value without using any other variables as predictors. When we take the mean transfer fee of the training sample and apply it on the test sample we get a RMSE at 9.06.

4.4 Ordinary least squares

As a second prediction model, we use the linear regression model - ordinary least squares (OLS). OLS works by minimizing the *sum of squared residuals* (SSR) and is thereby also minimizing the bias. Unfortunately, the goal of minimizing the in-sample errors lead to a risk of overfitting and thereby increases the errors due to variance. Therefore, OLS is not considered among the best prediction methods. On the other hand, OLS is a very good method when inference is the goal because the model's significance can be tested. When applying the OLS model on the test sample we receive a RMSE of 6.34.

4.5 The Lasso model

One way to deal with the problem of overfitting is using the Lasso model. The Lasso model punishes complexity by adding a loss function to the OLS-equation:

$$\underset{\beta_j}{\text{minimize}} : \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{i=1}^{p} \beta_j x_{ij}^2) + \lambda \sum_{j=1}^{p} |\beta_j|$$

From the equation above we see a clear trade-off between minimizing the SSR and the penalty term. The penalty is given by the sum of the absolute β coefficients. For a given λ the model returns a corner solution of the most significant variables. The λ parameter weights the penalty according to the complexity of the model. The larger λ, the heavier a penalty which will lead to exclusion of more variables. The Lasso is performing a so-called *variable selection.* The optimal size of λ is decided by running the regression on our training data for different values of λ. For each of the estimated models we estimate the expected transfer values in the test data. We then calculate the RMSE for all the different models and find the λ which minimizes the RMSE in our test data. The model with the optimal λ ($\lambda = 0.0009$) has a RMSE of 6.34. Due to the low optimal size of the penalty term overfitting does not seem to be an issue for our regular OLS model.

4.6 Regression Trees

Decision trees is a machine learning method which can be applied on both categorical and continuous variables. When used for predicting continuous outcomes (such as transfer value) the method is called *Regression Trees.* The decision tree is grown by using our training data in the following way: First all observations are divided into J different regions depending on patterns of the predicting variables. For all observations in one region the prediction is given by the mean of the observed transfer values. The regions are constructed in a way that minimizes the SSR:

$$\sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_m})^2$$

It is computationally infeasible to consider all possible splits into J regions. Therefore, an approach called *recursive binary splitting* (you do not take into account how a split affects the following splits) is used (Friedman et. al, 2001). In our case the first split is searchresults>50300 which means that this division of the observations provides the greatest reduction of the SSR of all possible first splits across all predictors and all values.

There is a risk that this procedure will lead to overfitting. To avoid this, we can use a procedure called pruning which punish the complexity of the model. To prune the tree, we use the following equation:

$$\sum_{m=1}^{|T|} \sum_{x_i \in R_j} (y_i - \hat{y}_{R_j})^2 + \alpha|T|$$

$|T|$ is the number of terminal nodes in the subtree T, R_m is the subspace of the region M, and y_{R_m} is the predicted response associated with R_m. When pruning the tree, we obtain different versions of the tree as a function of α. We evaluate the prediction power of each version of the tree using *K-fold cross-validation* (in our case we divide the training data into 10 folds) to choose the right α value. A new tree is grown on all but the 10^{th} fold of the training data. The process is repeated for each value of α. All the grown trees are evaluated by calculating the MSE on the 10^{th} fold. The α value with the lowest average MSE across the folds are chosen and then used on the original tree. As with the Lasso model overfitting wasn't an issue and the original tree produced a lower RMSE than the pruned one. The RMSE is 6.60.

The figure illustrates the regression tree made to predict the transfer value. The first split assigns observations with less than 50300 search results to the left branch. The next split of the left branch assigns players with less than 31.885 months of the contract left to the left branch and so on. In our model the tree divides the players into 11 terminal nodes and predicts a transfer fee for each of these subgroups. The variables at the top of the tree are the most important ones and the length of a branch shows the relative importance of the split. Hence, in our model the variable google search results is the most important predictor of a player's transfer value.

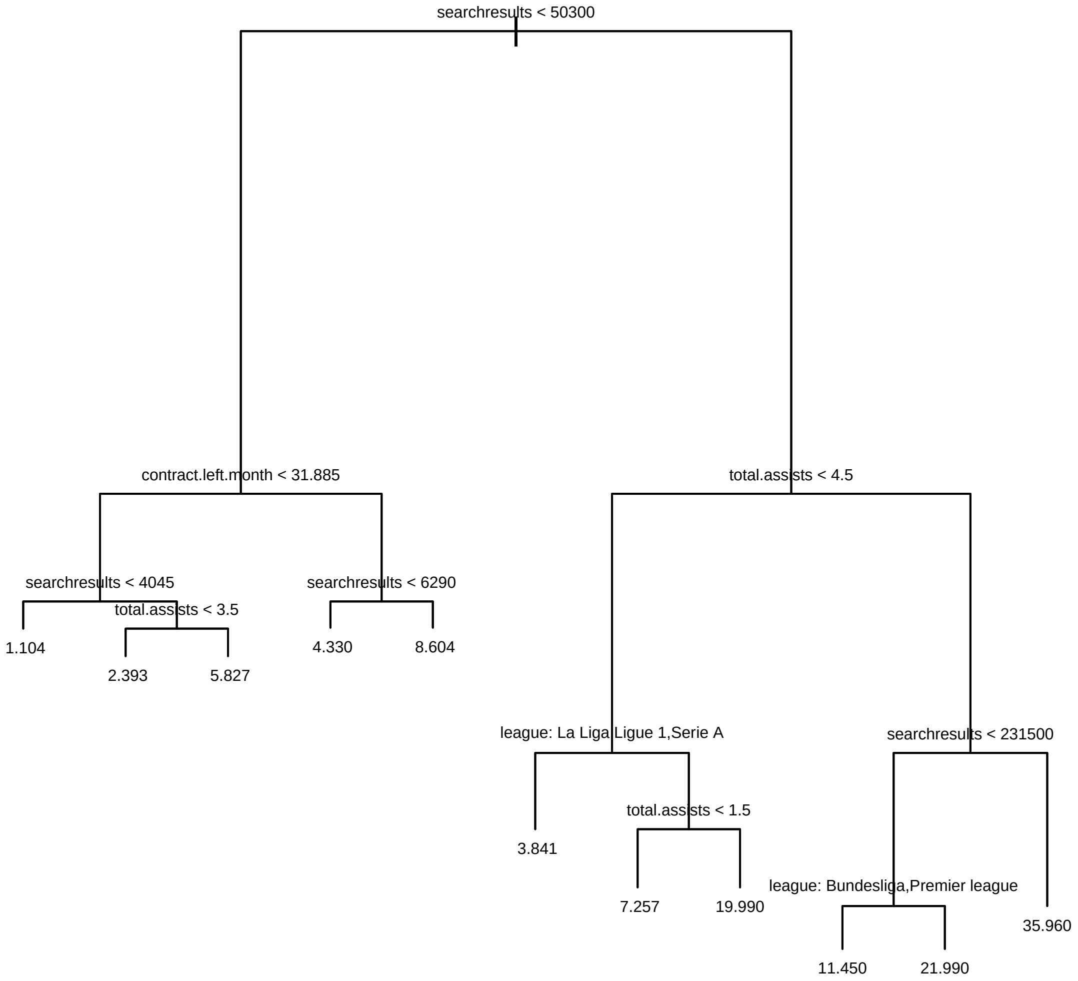


Figure 5: Decision Tree

4.7 Random Forest

As described, the decision tree often has issues with overfitting which creates errors due to variance. The issues can be reduced by another prediction method called *random forest* (RF). The RF is a so-called ensemble model which means that the model consists of several smaller decision trees (in this case regression trees). RF has great similarities with the concept of *bagging* or *bootstrap aggregating* (Breiman, 1996) where the core idea is to reduce the error due to variance from one model by building several models and use their average prediction. A RF prediction can be divided into 4 steps:

1) First, each regression tree is constructed by randomly drawing and replacing 63.2 percent of the observations in the training sample(this is called bootstrap sampling). In our project, this is done 500 times so we end up with a forest of 500 different trees with random selected observations from the training sample.

2) In the next step each tree randomly select *m* number of the predictor variables in the data set. For regression trees like these, m is equal to the total number of all predictors divided by 3. All 500 trees use the same number of variables. The purpose of randomly choosing predictors is to prevent very important variables from overshadowing the effect of weaker variables. This often happens because the underlying algorithm searches for the split with the largest decrease in the loss function.

3) Thirdly, we calculate the out-of-bag error rate for each tree using the remaining 36.8 percent of the data.

4) In the last step the average prediction is calculated out of all predictions from each individual tree.

In theory, RF does not need a separate test sample to examine the validity of the results. The validity is measured internally by the out-of-bag error rate. In this project, we have used a separate test sample. The reason is that we wanted to calculate the RMSE so we can compare the RF model's predictive power to the previous prediction models. The RMSE of the RF on the test sample is 6.26.

We can also use the RF to calculate and rank the importance of each predictor in the data set. This is done by calculating *%IncMSE*. The %IncMSE states the percentage increase of the mean squared error (estimated with the out-of-bag error rate) if the particular variable is permuted (which means that the values are randomly shuffled). The higher the number, the more important is the variable. As with the decision tree google search results is shown to be the most important variable[5].

5 Conclusion

In this paper we try to predict the transfer value of football players. To answer the initial question, we first created several web scraping functions to collect a dataset containing information about transfers made in the season 15/16 in the five major European leagues, the players performance statistics in the season prior to the transfer and information about the buying club's latest results. We scraped the data from Transfermarkt, Google and Wikipedia. After the data gathering and cleaning we ended up with a tidy data set of 369 observations and 25 variables.

To select the best predictors out of all 25 variables we used several visualisations to search for correlation patterns. Furthermore, we used the Google Maps API to visualize our data sample. We then selected 11 predictors which all seemed to be linked to the players' transfer value.

In the third and last section we built five different prediction models using a training sample. The accuracy of each prediction model was evaluated by calculating the RMSE on a test sample. The result of the comparative analysis shows that the OLS, Lasso, regression tree and random forest have quite similar prediction accuracy. The random forest model was the best performing model in predicting the transfer value of football players. If we take the average transfer value of 5.6M into consideration, then a RMSE of 6.5M is very high. This raises the question of whether we are missing variables that could help us differentiate the players and their performances. Otherwise this means that the market for football players is highly irrational due to the fact that players with similar attributes end up with very different price tags.

[5] The list of the importance of each predictor is provided in the appendix. %IncMSE of the different predictors are listed in the table below.

Table 1: Summary RMSE

Model	RMSE
Average	9.06
OLS	6.34
Lasso	6.34
Regression Tree	6.60
Random Forest	6.26

i

6 Litterature

Fortmann-Roe, Scott. 2016. “Understanding the Bias-Variance Tradeoff”. Located at: http://scott.fortmann-roe.com/docs/BiasVariance.html.

Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. 2001. “Introduction to statistical learning”. Vol. 1. Springer, Berlin: Springer series in statistics. Located at: http://www-bcf.usc.edu/~gareth/ISL/ISLR%20Fourth%20Printing.pdf

7 Appendix

Table 2: Variables in the data set

Variables	Description
name	Name of player
nationality	Nationality of player
birth_place	Birth place of player
birth.date	The date at which the player was born
transferage	Age of player when transfer occurred
positions	Position on field of player
total.goals	Total amount of goals scored by player
penaltygoals	Total amount of penalty goals scored by player
total.assists	Total amount of assists made by player
substitutions_in	Total amount of matches where player gets substituted in
substitutions_out	Total amount of matches where player gets substituted out
total.minutes.played	Total amount of minutes played by player
minutes.pr.goal	Amount of minutes played per goal scored by player
yellow cards	Total amount of yellow cards the player got
secondyellow	Total amount of second yellow cards the player got
redcards	Total amount of red cards the player got
contract.left.month	Months left of the transfered players' contract at transfer
club.to	Which club player is transferred to
club.from	Which club player is transferred from
league	The league that buying club is playing in
Status	Status of the buying club
transfer.data	Date of transfer
transfer.fee	The transfer fee measured in million £
searchresults	Number of search results when you google the player name and 'footballer'

Find graph here: http://scott.fortmann-roe.com/docs/BiasVariance.html

Table 3: Predictors

positions
appearances
total.goals
total.assists
total.minuts.played
contract.left.month
transferage
league
Status
searchresults
transferage_sq

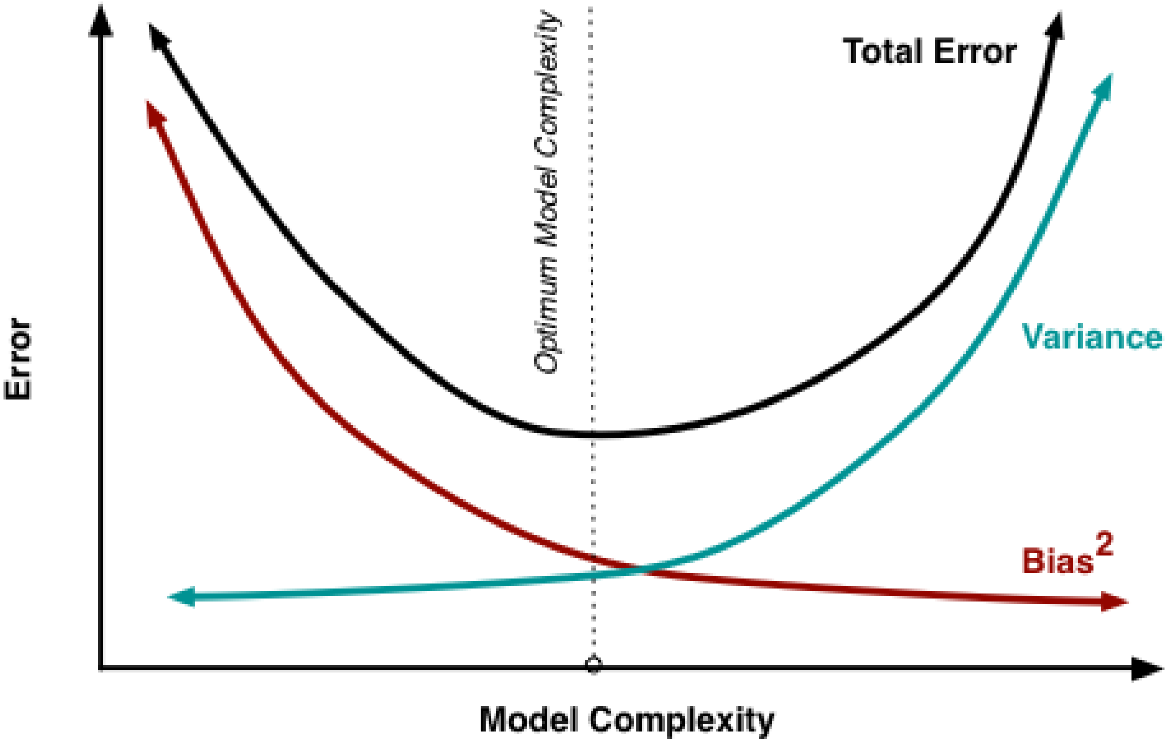


Figure 6: Image is downloaded from Fortmann-Roe (2016)

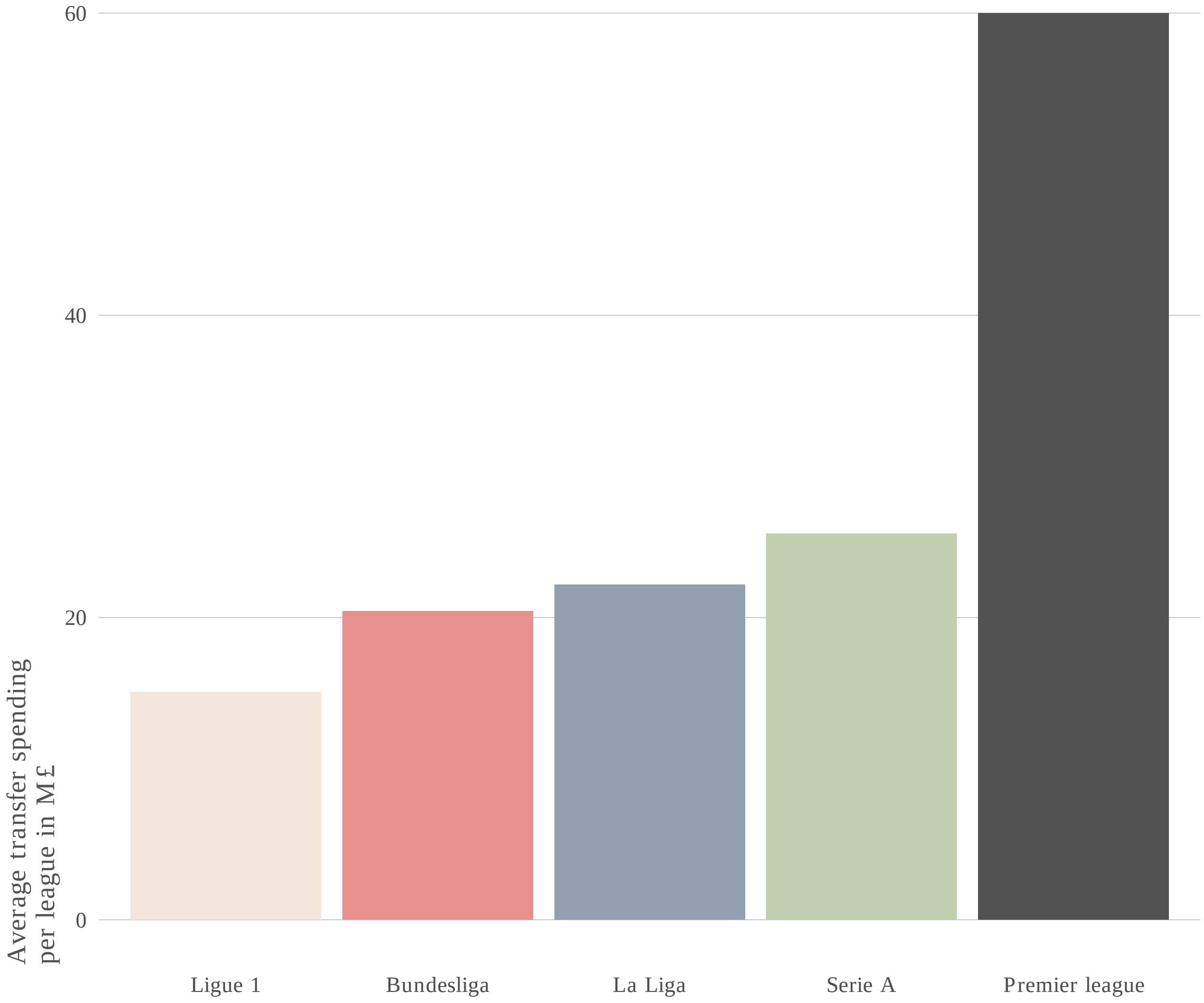


Figure 7: Average Club Spending Between Leagues

Table 4: Variable importance

Variables	%IncMSE
posistions	0.35
appearances	8.00
total.goals	5.44
total.assists	2.67
contract.left.month	6.32
transferage	4.26
league	2.52
Status	5.67
searchresults	16.21
transferage_sq	4.79