University of Warsaw
Faculty of Economic Sciences

Working Papers
No. 22/2021 (370)

PREDICTING FOOTBALL OUTCOMES FROM SPANISH LEAGUE USING MACHINE LEARNING MODELS

Michał Lewandowski
Marcin Chlebus

Warsaw 2021

WORKING PAPERS 22/2021 (370)

Predicting football outcomes from Spanish league using machine learning models

Michał Lewandowski*, Marcin Chlebus

University of Warsaw, Faculty of Economic Sciences
**Corresponding author: lewandowski.michal.1995@gmail.com*

Abstract: High-quality football predictive models can be very useful and profitable. Therefore, in this research, we undertook to construct machine learning models to predict football outcomes in games from Spanish LaLiga and then we compared them with historical forecasts extracted from bookmakers, which knowledge is commonly considered to be deep and high-quality. The aim of the paper was to design models with the highest possible predictive performances, get results close to bookmakers or even building better estimators. The work included detailed feature engineering based on previous achievements of this domain and own proposals. A built and selected set of variables was used with four machine learning methods, namely Random Forest, AdaBoost, XGBoost and CatBoost. The algorithms were compared based on: Area Under the Curve (AUC) and Ranked Probability Score (RPS). RPS was used as a benchmark in the comparison of estimated probabilities from trained models and forecasts from bookmakers' odds. For a deeper understanding and explanation of the demonstrated methods, which are considered as black-box approaches, Permutation Feature Importance (PFI) was used to evaluate the impacts of individual variables. Features extracted from bookmakers odds' occurred the most important in terms of PFI. Furthermore, XGBoost achieved the best results on the validation set (RPS equals 0.1989), which obtained similar predictive power to bookmakers' odds (their RPS between 0.1977 and 0.1984). Results of the trained estimators were promising and this article showed that competition with bookmakers is possible using demonstrated techniques.

Keywords: predicting football outcomes, machine learning, betting, adaboost, random forest, xgboost, catboost, ranked probability score, auc, permutation feature importance

JEL codes: C13, C51, C52, C53, C61, L83, Z29

1. Introduction

Amongst all unimportant subjects, football is by far the most important. This beautiful and simple sentence is attributed to Saint John Paul II. In the opinion of the authors of this paper, this is considered the best description of the importance of football in recent decades. Indeed, football may be perceived as not important, even though is crucial to millions of people around the world. Despite the increasing digitization and the dynamically growing interest in video games including esports, television rights to major football competitions are still extremely desirable. Globalization fosters interest in football, which also translates directly into the increasing size of the sports betting market.

The betting market is nowadays one of the most important financial areas of football. Bookmakers companies are becoming more and more general partners of individual football competitions as well as of the teams themselves. For example, in England half of the Premier League's kits was emblazoned with a bookmakers' logo during the 2019-20 season. Combined kits sponsorship deals was broken a record generating a total of £ 315.6 million for the 2018-19 season, and almost £ 350 million for the 2019-2020 season, of which approximately 20% is the financial contribution of bookmakers (source: http://www.sportingintelligence.com/). These values clearly show that the betting industry has enormous capital, which proves that it generates high profits. Bookmakers earn on margins, undoubtedly, however, without the use of appropriate and high-quality tools to predict results in football (as well as in other sports) they would not be able to survive on the market.

Currently, both using betting strategy in long terms or live betting force bookmakers to adapt their offer of hundreds or even thousands of betting odds to the expectations of gambling players, which makes having very high-quality models for forecasting football results extremely important. Hence, a lot of analysts and experts are hired to develop better and more effective methods used to estimate the probabilities of not only football outcomes, but more and more sophisticated events - for example the number of yellow cards in the last quarter of a given game. Football is a sport in which unexpected can occur relatively often and it additionally proves that modelling football events probabilities is challenging. The above arguments encourage authors to deal with bookmakers on the basis of matches from a predetermined range.

In this study, multiple classification machine learning models have been built, the aim was to predict football matches results in the Spanish LaLiga (Spanish 1st league). In particular, the aim of the research was to compare the predictions made using state-of-the art machine

learning algorithms (AdaBoost, Random Forest, XGBoost and CatBoost) with the probabilities estimated by selected bookmakers (William Hill, Bet365, Interwetten, bwin, BetVictor). The models were trained on data extracted for seasons from 2009-2010 to 2017-2018 (training set) and 2018-2019 and 2019-2020 seasons were out-of-time samples (validation set). A deep feature engineering has been carried out, allowing for the creation of an extensive set of variables that can have a significant impact on the results in football matches. The features included were divided into four categories: variables extracted from betting odds, form of teams in the current season, statistics from past several matches of the current season and overall power of teams (namely ratings). Before training the models, a following features selection techniques were used: Mutual Information measure, ANOVA F-value and recursive feature elimination based on different base learners. In order to compare the models between each other the Area Under the Curve (AUC) and Permutation Feature Importance (PFI) were considered. Finally, the results, based on a Ranked Probability Score (RPS) measure, were compared with the bookmakers' odds retrieved from 5 aforementioned leading companies.

This paper is organized as follows: the first section presents literature overview of works devoted to the subject of predicting football results using numerous machine learning techniques; the second section describes models and performance assessment methods used in this research; the third section shows methods including feature engineering and feature selection processes, which lead to the presentation of the final dataset used for the further exploration; the fourth section demonstrates results obtained from the trained models, explanatory analysis and comparing the results to bookmakers' performance. The work ends with a conclusions and summary of the paper.

2. Literature overview

Until now, many researchers have tackled the problem of predicting football outcomes in different approaches. Joseph *et al.* (2006) and Owramipur *et al.* (2013) took up this issue for single football teams (Tottenham Hotspur from England and Barcelona F.C. from Spain, respectively). Nevertheless, very often it was decided not to focus on just one team (due to, among other things, smaller data to be trained), but on individual football competitions, for example the English Premier League (Baboota & Kaur, 2018, Constantinou *et al.*, 2013), Dutch Eredivisie (Buursma, 2011, Tax & Joustra, 2015) or Spanish LaLiga (Zaveri *et al.*, 2018). In some articles, the use of match results from various leagues and competitions could be observed (Berrar *et al.*, 2019, Constantinou, 2019, Hubáček *et al.*, 2019).

The right choice of features implemented to them may be extremely crucial when it comes to performance of trained estimators. Baboota & Kaur (2018) and Hucaljuk & Rakipovic (2011) used numerous statistics extracted from last few games in order to show the current form of teams. Information from a longer period or history of competitions with a given opponent also play their role in the literature (Owramipur *et al.*, 2013, Hubáček *et al.*, 2019). Nevertheless, although football is undoubtedly a team sport, individual players, as well as their skills and forms, can decide about match results. Therefore, the availability of key players and their statistics were also considered as variables to analysis (Owramipur *et al.*, 2013, Joseph *et al.*, 2006). We paid special attention to Tax & Joustra (2015), where a wide range of interesting features was chosen, including managerial change, club budgets or travel distance.

Various methods of artificial intelligence (AI) were considered in the research on predicting football results. Among them, one should mention tree based methods (being particularly explored in this paper), including: Random Forests, XGBoost or Gradient Boosted Trees (Baboota & Kaur, 2018, Hucaljuk & Rakipovic, 2011, Berrar *et al.*, 2019). Other techniques used for the given problem were also: k-Nearest-Neighbour, called KNN (Berrar *et al.*, 2019, Joseph *et al.*, 2006), Support Vector Machine, called SVM (Baboota & Kaur, 2018, Zaveri *et al.*, 2018). In addition, Bayesian Network is one of the most commonly used method for forecasts of football matches, which can be found in Buursma (2011), Constantinou (2019) and Constantinou *et al.* (2013). The problem was also developed using Artificial Neural Networks (ANN), for example in studies by Mccabe & Trevathan (2008), Buursma (2011) and Zaveri *et al.*(2018).

Literature analysis indicates that two types of techniques have been distinguished for performance assessment of trained models in predicting results of football matches. A commonly used measure was accuracy, calculated on the basis of the percentage of correctly classified outcomes. This technique could be found in many articles (Hucaljuk & Rakipovic, 2011, Tax & Joustra, 2015). The second type of technique focused not only on class outcomes, but also on estimated probabilities for individual events. The use of Ranked Probability Score (RPS) has proven successful for this purpose, as reflected in Constantinou (2019), Hubáček *et al.* (2019) and Baboota & Kaur (2018).

3. Materials & Methods

3.1. General approach

General approach in this study was the following. First, individual variables (concerning betting odds, current season's forms of teams, statistics and overall quality of teams) were constructed based on data from websites: http://www.football-data.co.uk/ and https://www.fifaindex.com/. After constructing the variables, feature selection was performed using Mutual Information measure, ANOVA F-value and Recursive Feature Elimination Approach. Then machine learning methods were built and trained: AdaBoost, Random Forest, XGBoost and CatBoost. For this purpose, the data has been divided into train set and validation set. In particular, the 3-fold cross validation approach was used in the training sample. This has been implemented into the hyperparameters tuning process using the Bayesian Optimization method. After selecting the best models and their individual parameters for the selected methods, a comparison of their performance was undertaken using Area Under The Curve (AUC) and Ranked Probability Score (RPS). Feature importance was measured from Permutation Feature Importance. Then, RPS was also used to compare the results with the bookmakers' industry.

Python language and its packages were used to write all scripts including: data pre-processing, machine learning pipelines with training, models selection, validation and analysis of the results. AdaBoost and Random Forest models were taken from Scikit-Learn library (Pedregosa *et al.*, 2012), XGBoost from xgboost package (Chen & Guestrin, 2016) and CatBoost from catboost package (Dorogush *et al.*, 2017). A further part of this section will help to analyse selected materials and methods in depth.

3.2. Feature engineering

One of the most important issue in predicting results in soccer is the selection of variables that can have a significant influence on actual outcomes. Bookmakers take into account a number of factors when modelling betting odds. Each statistic or additional information can have a key impact on the predictions. For example, an injury to one of the best players right before the game or a manager change may be factors that will change the bookmakers' odds and will be more suited to the actual chances of both teams. Variables can be divided into 4 types: features built based on betting odds, team form in the current season, statistics from past k games of the current season and overall team quality (ratings). The original primary dataset has been retrieved from a football-data website (http://www.football-data.co.uk/).

3.2.1. Features based on betting odds

Betting odds can be interpreted as the inverse of probabilities of given events. For example, if the odds for the home win equals 1.66, it can be assumed that according to the bookmaker the probability of this event equals 1 / 1.66, which is around 60%. Due to the margins included in the odds, this method does not ensure that the estimated values will sum up to 100%. Therefore, inverse bookmaker odds have been standardized.

In this work, this procedure was done for the odds of 5 bookmakers: William Hill (WH), Bet365 (B365), Interwetten (IW), bwin - formerly Bet & Win (BW), BetVictor - formerly VC Bet (VC). Estimated probabilities were averaged and this is how three variables depending on bookmakers' odds were constructed: $H\_mean\_proba, D\_mean\_proba$ and A_mean_proba, concerning the average probabilities of the home win, draw and away win, respectively.

3.2.2. Features based on the current season's team form (Form Coefficients)

Each season may differ from the previous one for each team. In one year, the team can defend itself against relegation, in the next - boost with form and fight for the championship, e.g. after a change coaching stuff or excellent transfers into the team. Therefore, it is a good idea to create a variable that reflects the current form of the them. It should be a feature that depends on the team's results. We defined Form Coefficients that were built based on the idea of Baboota & Kaur (2018) and own modifications.

At the beginning of each season we assume that each team's Form Coefficient equals 1. Let A and B be Spanish LaLiga teams and let ϕ_i^A and ϕ_i^B denote their Form Coefficients after the i-th round, respectively. Let us assume that A and B play a match against each other in the j-th round and let A be the home team and B the away team. Then, the formulas for the Form Coefficients after the j-th match looks as follows:

- if team A wins against team B then:

$$\phi_j^A = \phi_{j-1}^A + \gamma_H \phi_{j-1}^B \quad (1)$$

$$\phi_j^B = \phi_{j-1}^B - \gamma_H \phi_{j-1}^B \quad (2)$$

- if team B wins against team A then:

$$\phi_j^A = \phi_{j-1}^A - \gamma \phi_{j-1}^A \quad (3)$$

$$\phi_j^B = \phi_{j-1}^B + \gamma \phi_{j-1}^A \quad (4)$$

- if there is a draw then:

$$\phi_j^A = \phi_{j-1}^A - \gamma\left(\phi_{j-1}^A - \phi_{j-1}^B\right) \quad (5)$$

$$\phi_j^B = \phi_{j-1}^B - \gamma\left(\phi_{j-1}^B - \phi_{j-1}^A\right). \quad (6)$$

The formulas for the Form Coefficients are recursive, depending on the form from previous games, especially the last one. The principle is as follows: if team A defeats team B, team A *steals* some of the form of team B, similarly if team B defeats team A. If there is a draw, the Form Coefficient will drop to the team that was the favourite of this match, i.e. had a higher Form Coefficient and by the same amount will increase the other team. What fraction of the Form Coefficient will be stolen will depend on the parameters: stealing fraction γ ($\gamma \in (0,1)$) and stealing fraction home γ_H. The higher the γ, the Form Coefficient is more sensitive to which team is the opposing team.

γ_H is a fraction of gamma, that is $\gamma_H = \gamma \cdot \alpha$, where $\alpha \in (0,1]$, hence $\gamma_H \leq \gamma$. The aim of γ_H is for the home team to be less or equally sensitive to form factor changes after home victories, as home wins are more common in football. The table below shows an example of how the Form Coefficients can change for teams A and B in different outcome variants.

Table 1. Numerical demonstration of the computation of Form Coefficients updates for $\gamma = 0.33$, $\alpha = 0.6$ (hence $\gamma_H = 0.2$).

		Home win		**Draw**		**Away win**	
	ϕ_{j-1}	ϕ_j	**Update**	ϕ_j	**Update**	ϕ_j	**Update**
A team	6	6.3	0.3	4.5	-1.5	4	-2
B team	1.5	1.2	-0.3	3	1.5	3.5	2

Source: own preparation.

The following parameter values were selected for building features - for γ: 0.25, 0.33, 0.5, for α: 0.6, 0.8, 1. Features Form Coefficients were built separately for the home team and away team. Features that are the differences between Form Coefficients between form team home and form team away have also been constructed. Hence, the final number of form variables is equal to 27 (all combinations of: 3 possible values of gamma times 3 possible values of lambda times 3 options – form home team, form away team or their difference).

3.2.3. Features based on statistics from past k games

Statistics are an inseparable part of soccer. On the one hand, one situation in the match may decide to win, and no numbers or statistics will matter. On the other hand, there is no coincidence that it is now common that analysts and statisticians work in coaching stuff, having

a significant impact on teams, sometimes even a key one. Similarly is with betting and predictions of results, statistics from previous matches can have a big influence on the forecasts (Baboota & Kaur, 2018, Razali *et al.*, 2017). The table below shows selected statistics that were averaged over the last k games. Features were counted separately for home team and away team.

Table 2. Description of features based on statistics from past k games.

Feature abbreviation	Description	Which mean used to aggregate
GD	goal difference - sum of the numbers of goals scored differenced by the sum of the numbers of goals conceded	arithmetic mean
C	corners	
S	shots	
ST	shots on target	
F	fouls	
Y	yellow cards	
R	red cards	
streak	points	
streak_weighted	points	weighted mean

Source: own preparation.

A feature *streak_weigthed* is the only one that uses a weighted average number of points. This is made in order to increase the significance of the results from recent matches in this way. The older match outcome, the less weight will be assigned to it. The *streak_weigthed* is more sensitive to team form than the *streak*. Nevertheless, the *streak* is less prone to one-off form changes from the last game, hence both variables were worth considering.

Another issue was how many games to include in the average – let us denote this parameter as k. To ensure uniformity in Baboota & Kaur (2018) they analysed the results of their models and decided to choose $k = 6$ for all models. In this paper the approach was different: in order to avoid one choice of k, features were built based on statistics from past k games for the following k choices: 1, 2, 3, 5, 6 and 10. This was to ensure that different information was provided to models from different time periods.

The presented method of averaging variables causes missing values for first k games for all teams. In order to fill these initial missing values, k has been set to $j - 1$ for the j-th round (except for the first round) up to k round (i.e. $k = 1$ for the second round, $k = 2$ for the third

round, and so on, up to the k-th round). For the first round of seasons, there are still missing values, which it is logical that before the first game, despite different information about the teams, their form is quite unknown.

The last issue regarding features based on statistics was building variables that take into consideration the statistics of both teams in a given match. For this purpose a set of features containing information about the differences between the selected statistics for the home team and the away team were prepared. For example, if two teams are competing for the championship, they may have a high streak but the difference between them will be small, indicating that there is no clear favourite to win.

Combinations of selected statistics from matches (9 statistics), k parameter (6 values) and team they concern (3 options: home team, away team and the differences for both) made a total of 162 statistics-based features.

3.2.4. Features based on overall team quality (ratings)

The form and results of the teams from the current season have a great impact on the outcomes of the teams, but the overall strength of the team cannot be ignored either. For example, Real Madrid, even if the beginning of the season would not be very successful, it can still be considered as favourite in the next rounds for many reasons. Betting odds and predictive models should take this into account. For this purpose, using the idea of Baboota & Kaur (2018), ratings from https://www.fifaindex.com/ were taken. The ratings are determined by the algorithm used by EA Sports for their widely known and popular video game series FIFA. 4 ratings were used: attack (ATT), defence (DEF), midfield (MID) and overall (OVR) and they have been used as initial values for each season from training set. In order to update ratings after each game, own proposals were implemented. In consequence, ratings referred to the overall power of the teams and power in individual formations (attack, defence, midfield) could also react to the results in subsequent matches. For example, if a team scores a lot of goals, the rating attack should go up. Likewise, if a lot of goals are lost, for example, the defence rating should go down and so on.

The formulas for changing ratings during the season are the following:

- attack rating after n-th game of A team:

$$\mathrm{ATT}_n = \mathrm{ATT}_{n-1} + \mathrm{sign}(\theta_{GF}) \cdot \min(|\theta_{GF}|, 2) \cdot \left(\frac{\mathrm{DEF}_{n-1}^{opp}}{\mathrm{ATT}_{n-1}^{A}}\right)^{\mathrm{sign}(\theta_{GF})} + \frac{\overline{\mathrm{GF}}_d^A - \overline{\mathrm{G}}_d}{1+\overline{\mathrm{G}}_d} \quad (7)$$

- defence rating after n-th game of A team:

$$\mathrm{DEF}_n = \mathrm{DEF}_{n-1} + \mathrm{sign}(\theta_{GA}) \cdot \min(|\theta_{GA}|, 2) \cdot \left(\frac{\mathrm{ATT}_{n-1}^{opp}}{\mathrm{DEF}_{n-1}}\right)^{\mathrm{sign}(\theta_{GA})} + \frac{\overline{\mathrm{G}}_d - \overline{\mathrm{GA}}_d^A}{\overline{\mathrm{G}}_d + 1} \quad (8)$$

- midfield rating after n-th game of A team:

$$\mathrm{MID}_n = \mathrm{MID}_{n-1} + \frac{\theta_{GF} + \theta_{GA}}{2} \cdot \left(\frac{\mathrm{MID}_{n-1}^{opp}}{\mathrm{MID}_{n-1}}\right)^{\mathrm{sign}(\theta_{GF} + \theta_{GA})} + \frac{\frac{\overline{\mathrm{GF}}_d^A - \overline{\mathrm{G}}_d}{1 + \overline{\mathrm{G}}_d} + \frac{\overline{\mathrm{G}}_d - \overline{\mathrm{GA}}_d^A}{\overline{\mathrm{G}}_d + 1}}{2} \quad (9)$$

- overall rating after n-th game of A team:

$$\mathrm{OVR}_n = \max\left(m, \min\left(M, \mathrm{OVR}_{n-1} + \theta_{\mathrm{OVR}} \cdot \left(\frac{\mathrm{OVR}_{n-1}^{opp}}{\mathrm{OVR}_{n-1}}\right)^{\mathrm{sign}(\theta_{\mathrm{OVR}})}\right)\right) \quad (10)$$

where:

- *opp* index stands for ratings of the opponents in n-th game of A

- GF_n^A, GA_n^A, - number of goals for A (scored) and against A (conceded) in n-th game, respectively

- $\overline{\mathrm{GF}}_d^A$, $\overline{\mathrm{GA}}_d^A$ - average number of goals scored and conceded by A over all matches in the last d days, respectively

- $\theta_{GF} = \mathrm{GF}_n^A - \overline{\mathrm{GF}}_d^A$

- $\theta_{GA} = \overline{\mathrm{GA}}_d^A - \mathrm{GA}_n^A$

- $\overline{\mathrm{G}}_d$ - average number of goals (scored/conceded) per team from all games in the last d days; it is equivalently to the average number of goals divided by 2

- $m = \min(\mathrm{DEF}_n, \mathrm{MID}_n, \mathrm{ATT}_n)$

- $M = \max(\mathrm{DEF}_n, \mathrm{MID}_n, \mathrm{ATT}_n)$

- $\theta_{\mathrm{OVR}} = (\mathrm{DEF}_n - \mathrm{DEF}_{n-1} + \mathrm{MID}_n - \mathrm{MID}_{n-1} + \mathrm{ATT}_n - \mathrm{ATT}_{n-1})/3$.

Although formulas may seem complex, the idea behind them is simple. All formulas are recursive, before the first game ratings are taken from the FIFA Index as mentioned above. Part of the rating change is supposed to be due to the current match where teams may have scored or lost a lot of goals, which would correctly increase or decrease ratings. For example, if Eibar scores a lot of goals against Barcelona F.C. attack rating of Eibar is going to be higher, defence rating of Barcelona is going to be much lower after this game. However, if Barcelona scores a lot of goals, the rating attack is going to be increased, but this impact would be less, because in this case we can assume that Eibar has weaker defence and is easier to score more goals.

The last element of formulas in attack, defence and midfield ratings is to consider the general form of individual formations in the last d days. For example, if the average number of goals conceded by a given team would be less than the average number of goals conceded for all teams, then it could raise the ratting defence of this team, etc. The parameter d was set to

15, 30 and 60 days. Ratings were calculated for home teams and away teams, and - similarly to other features – differences. Differences were calculated between midfield and overall ratings, however for attack and defence it was more logical to define the attack power of home team (ATT_d_power) as the difference between the attack rating of a home team and defence rating of away team, because strikers of home team are competing with the defence of away team. Similarly, the home team defence power (DEF_d_power) feature was built as the difference between defence of home team and attack of away team ratings. Summing up, 4 rating variants, 3 d parameter values and 3 measures (for home team, away team and a difference) give a total of 36 features based on overall team quality.

3.3. Feature selection

A total of 228 features were built. This is a fairly large number of variables to predicting soccer results. Hence, before the models were built, it was decided to limit the set of features to leave behind those that would potentially have a much less impact or introduce unnecessary noise during training models. Eventually, this procedure should be either to improve estimators' scores used to comparisons with the bookmakers. In order to select variables, 4 feature selection methods were chosen: Mutual Information, ANOVA F-value, recursive feature elimination based on Random Forest and AdaBoost models:

1. Mutual Information is a measure of the mutual dependence between the two variables. It is more general than correlation coefficient and, unlike the correlation coefficient, it examines non-linear relationships (Smith, 2015).
2. Analysis of Variance (ANOVA) is a statistical method used to test differences between two or more means. The purpose of using this method is to find features that are the best separators for classes of target variables (Kumar *et al.*, 2015).
3. The main principle of the recursive feature elimination is to use external estimator to asses and assign weights to features and based on that select final set of features recursively eliminating the less important ones (Guyon *et al.*, 2002). In this paper Random Forest and AdaBoost were used as the estimators, which assign weights based on feature importance. First model is train on all features, then after evaluation, one variable lowest in the importance ranking is pruned from the current set of the features, this approach is called Leave One Covariate Out – LOCO (Lei *et al.*, 2016). In each step of the estimation, 3-fold cross validation was used to reduce the variance of the results. The choice of 2 predictive models as the estimator was caused to avoid

making decisions about the quality of variables based on only one estimator and to have a broader view of how recursive feature elimination works on different predictive models. The order based on Random Forest turned out to be much more correlated with the results of feature selection based on Mutual Information and ANOVA F-value.

Ultimately, for feature selection, the rankings of the four above-mentioned methods, were put together. The approach can be interpreted as anti-selection, i.e. the variables that were not included in the top 70 features for each of the methods were removed from the set of all features.

3.4. Machine Learning Algorithms

The forecast of outcomes in football is multiple classification problem. Hence, all the algorithms described below are used for multiclassification case, although it should be emphasized that each of them also has very common and practical applications in regression predictions. All of them are ensemble methods. The main idea behind ensemble approach is that a set of *weak learners* can combine and merge together to a *strong learner* which is a finally classifier (Zhou, 2009). They are used for supervised learning problems, where the objects from training data are used to predict desired target variables. One of the biggest advantages of tree-based models is good handling of heterogenous and correlated data (Tuv *et al.*, 2009, Rabinowicz & Rosset, 2021). It is important in the context of this study, since there were relatively many such features due to their design. On the other hand, a disadvantage of chosen models is being *black-box* algorithm, which means that they are not easily interpretable. This characteristic is a logical consequence caused by high randomness both in training sets and predictor variables in each of learners. However, in this paper techniques increasing explainability of machine learning were used – for example Permutation Feature Importance – which are presented later in this section.

The first algorithm used in the study is an Adaptive Boosting (called AdaBoost) which is a boosting method. This algorithm uses the approach to improve and correct its precursor iteratively. It draws more attention to training instances, which were incorrectly predicted by the prior tree. Each next estimator is focused more on the samples which were more difficult in a given classification problem more than on the others.

The second method is Random Forest which is one of the method of bagging that builds and train a large set of decision trees. Bagging approach reduces the variance of a single tree's prediction and it can enhance performance of the estimator as a whole. The final decision about

a given instance from Random Forest are made by aggregating (averaging or majority voting) the predictions of all estimators.

The third one is a XGBoost that is an efficient implementation of the gradient boosting framework. It can solve real world scale problems using a minimal amount of resources (Chen & Guestrin, 2016). Wide range of hyperparameters gives control over the training procedure including avoiding overfitting or accelerating of learning procedure.

The fourth is a CatBoost which is an developed by Yandex. The main motivation for the CatBoost is so called the target leakage occurring in other gradient boosting approaches (Dorogush *et al.*, 2017). CatBoost provides a gradient boosting framework which attempts to solve for categorical features using a permutation driven alternative compared to the classical algorithms (Prokhorenkova *et al.*, 2018).

3.5. Performance assessment

The key issue in evaluating trained models is choosing the proper measure for that. It should be emphasized that on the one hand, predicting results in football is a multiple classification problem, where the classes are: home win, draw and away win (H, D and A class, respectively). Therefore, Area Under the Curve (AUC) measure has been selected to compare the outputs of models to give an overview of models' performance. In this study, due to the multiple classification case, AUC based on *one vs. all* approach has been used (Aly, 2005). In order to define ROC (Receiver Operating Characteristics) curve of class c, $c \in \mathrm{C} = \{H, D, A\}$, positive class was assigned to c class, negative class to the rest of classes and this *one vs. all* approach was repeated for each $c \in \mathrm{C}$. It means that in order to plot ROC should be considered one of the classes (suppose H label) as positive labels, while the other classes together as negative labels (D and A labels in the example).

However, this measure would be insufficient to compare the results with bookmakers. For this purpose, a special measure Ranked Probability Score (RPS) was used, which allows to evaluate the performance of outputs' probabilities. In addition, the Permutation Feature Importance was used to compare the models, which allows to assess the quality of features in predictive process.

3.5.1. Ranked Probability Score

Choosing an appropriate measure to assess forecasts of football outcomes is not a trivial question. A large number of scoring rules have been defined so far and still there is an open

debate which are the most appropriate (Wheatcroft, 2019). One of possibility is Ranked Probability Score (RPS).

Let r be a number of possible outcomes, p_j - the probabilistic forecast for the event to happen in j-th outcome and o_j - actual outcomes at position j. RPS for a single problem instance is defined as:

$$\text{RPS} = \frac{1}{r-1}\sum_{i=1}^{r-1}\left(\sum_{j=1}^{i} p_j - \sum_{j=1}^{i} o_j\right)^2. \tag{11}$$

RPS measures how good predictions, expressed in terms of probability distributions, are in matching actual outcomes. This scoring rule is sensitive to distance, which means that RPS value increases the more the cumulative distribution forecasted distinguishes from the observed labels (Wilks, 2005).

Applying the specified scoring rules, Constantinou & Fenton (2012) show different examples and desired expectations in terms of forecasts assessment. In their analysis, RPS values are considered the most expected, which is also confirmed by the choice of this measure to assess predictions of football outcomes in articles (Baboota & Kaur, 2018, Hubáček *et al.*, 2019) discussed in the literature overview. Although Constantinou & Fenton (2012) highlight that RPS does not have to be the only valid score rule to measure quality of football outcomes and a discussion amongst researchers is still open (Wheatcroft, 2019), nevertheless in this paper RPS was found to be appropriate and an intuitive measure for comparison between estimated predictions and betting odds.

3.5.2. Permutation Feature Importance

Permutation Feature Importance is defined as deterioration in a chosen model score when a single feature value is randomly shuffled (Breiman, 2001). This technique allows for comparing impact of variables for a given model and between models. It increases explainability und understating of used methods.

As machine learning becomes a crucial component of an ever-growing number of user-facing applications, interpretable machine learning has become an increasingly important area of research for a number of reasons. Understanding why machine learning models behave the way they do empowers both system designers and end-users in many ways: in model selection, feature engineering, in order to trust and act upon the predictions, and in more intuitive user

interfaces (Ribeiro, et al., 2016). Permutation Features Importance helps to better capture and understand the relationship between features and the target outputs.

PFI can measure the importance of variables using a variety of measures, RPS was selected in this paper to PFI calculation. In particular, the definition PFI calculated for the model called M and feature called x is the following:

$$\mathrm{PFI}_x^M = \mathrm{RPS}_{\mathrm{PFI}_x}^M - \mathrm{RPS}_{base,}^M \tag{12}$$

where RPS_{base}^M stands for RPS value from estimated model without any permutation values of feature x, $\mathrm{RPS}_{\mathrm{PFI}_x}^M$ is the average of RPS values calculated for 20 different permutations of values of feature x. The correctness of the definition is proved by the fact that the higher PFI_x^M, the better feature importance of x feature is.

3.6. Model training method

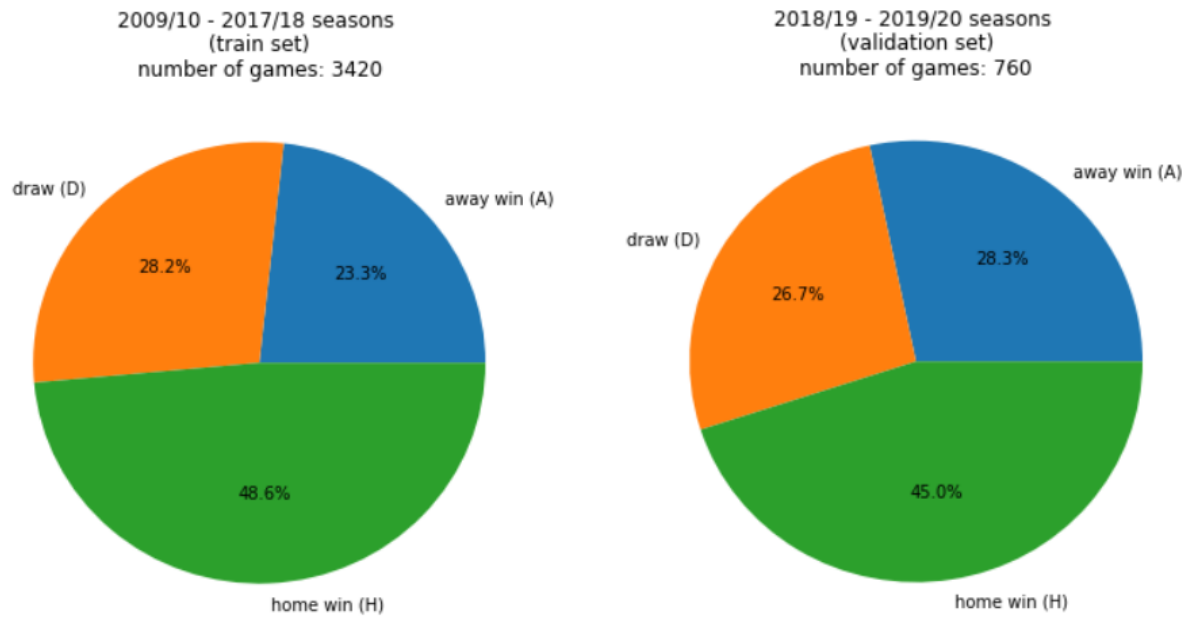
For all models Bayesian optimization method was performed during hyperparameters tuning process (Frazier, 2018). In this paper all estimated models have own architecture and essential is to find a set of parameters that will be the best for each one in terms of minimizing RPS which were set as an objective function. The Bayesian optimization approach focuses on a probability model for $P(score|configuration)$ that is obtained by updating a prior from a history H of $(configuration, score)$ pairs (Bergstra *et al.*, 2012). Among others, this is what makes this method more efficient than other methods - grid search or random search (Putatunda & Kiran, 2018). Bayesian optimization, in contrary to the others, uses the results from the prior steps to choose the next hyperparameter value candidates and therefore is more effective in finding the best set of hyperparameters.

In this study, 3-fold cross-validation method was used to train models in the pipeline in each iterations of the Bayesian optimization, all splits had approximately equal size. The score that was used to search and choose the best hyperparameters set after tuning was averaged RPS value calculated from 3 tested datasets from each fold. K-fold cross-validation is a statistical method, one of the aims of which is to avoid overfitting. In this approach given dataset X is randomly split into K mutually exclusive subsets (called folds) $X_1, \dots, X_K$ and then model is trained and tested K times, i.e. for all $i \in \{1, \dots, K\}$ it is trained on $X \backslash X_i$ and tested on X_i (Kohavi, 2001).

4. Results

The data that was used came from the https://www.football-data.co.uk/ and https://www.fifaindex.com/ websites. All extracted and constructed features were numerical. The whole dataset was split into train and validation sets. The training sample contained all matches from Spanish LaLiga starting from season 2009-2010 up to season 2017-2018 (3420 results), testing sample were out-of-time sample and consisted of seasons' 2018-2019 and 2019-2020 results (760 games).

Figure 1. Distributions of target variables (football outcomes) for train and validation sets.



Source: own preparation.

A higher percentage of home win outcomes is visible. Nevertheless, it should be noted that class sample sizes are not unbalanced, since minority classes are about 23-28%. 228 variables were constructed during the feature engineering procedure. Then, 88 variables were dropped and after feature selection the set of features was built for further analysis and predictive models estimation. The total number of selected variables equals 140.

Table 3. Feature selection summary with number of features divided into categories.

		Feature category				
		Betting odds	**Form coefficients**	**Ratings**	**Statistics**	**All**
Selected	**No**	0	2	0	86	88
	Yes	3	25	36	76	140
	All	3	27	36	162	228

Source: own preparation.

As shown in the Table 3., the variables from statistics category were mainly removed after the feature selection procedure, especially features that aggregates information about number of fouls, yellow cards and red cards – 37 of 88 dropped from dropped features. Also, as expected, there were variables that could have a significant impact on the prediction of football results, i.e. betting odds, the current form of the teams and the overall rating of the teams' power.

4.1. Performance results for all models

After feature engineering and feature selection, the process of training chosen machine learning models (AdaBoost, Random Forest, XGBoost, CatBoost) and searching for the best sets of hyperparameters using Bayesian optimization supported by 3-fold cross validation approach have been performed. The table below shows selected optimal parameters for each methods. All of them preferred shallow trees, i.e. *max_depth* or *depth* were equal to 2 or 3.

Table 4. The best sets of hyperparameters for chosen and trained machine learning models.

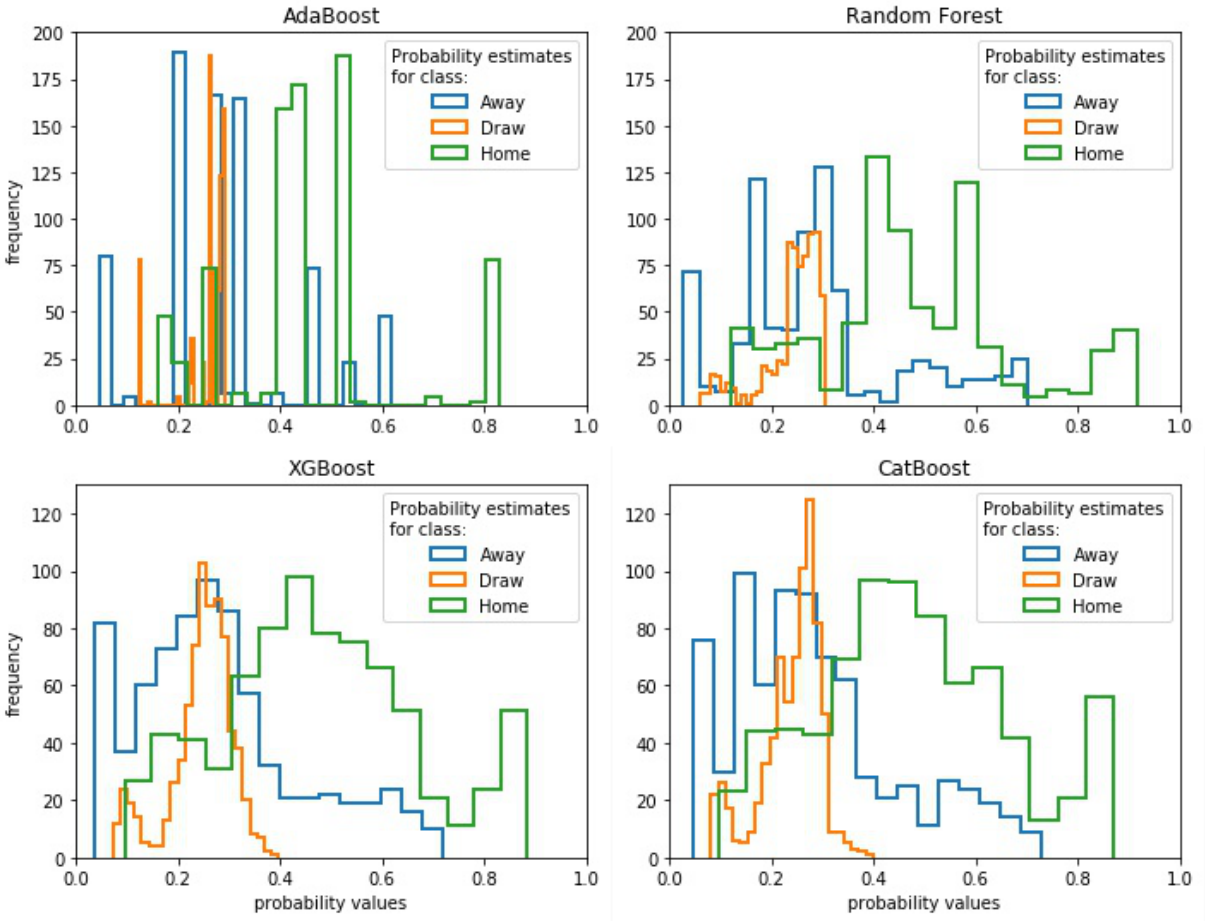
AdaBoost	**Random Forest**
'n_estimators': 20 'learning_rate': 0.019368 'base_estimator': DecisionTreeClassifier(max_depth=2)	'n_estimators': 525 'max_depth': 3 'min_samples_split': 30 'min_samples_leaf': 120 'max_features': 0.66 'max_samples': 0.81
XGBoost	**CatBoost**
'learning_rate': 0.05 'max_depth': 2 'n_estimators': 150 'min_child_weight': 60	'learning_rate': 0.046 'depth': 3, 'iterations': 140 'l2_leaf_reg': 2.663

'gamma': 3.4	'random_strength': 0.03
'subsample': 0.77	'bagging_temperature': 0.56
'colsample_bytree': 0.81	'rsm': 0.76
'colsample_bylevel': 0.81	
'colsample_bynode': 0.63	
'alpha': 0.944897	
'lambda': 0.081059	

Source: own preparation.

Further part of this section is about comparing the results between models with selected parameters. In particular, we start the analysis with the probability distributions estimated on the basis of the data from validation set.

Figure 2. Histograms of estimated probabilities of football outcomes for validation set.

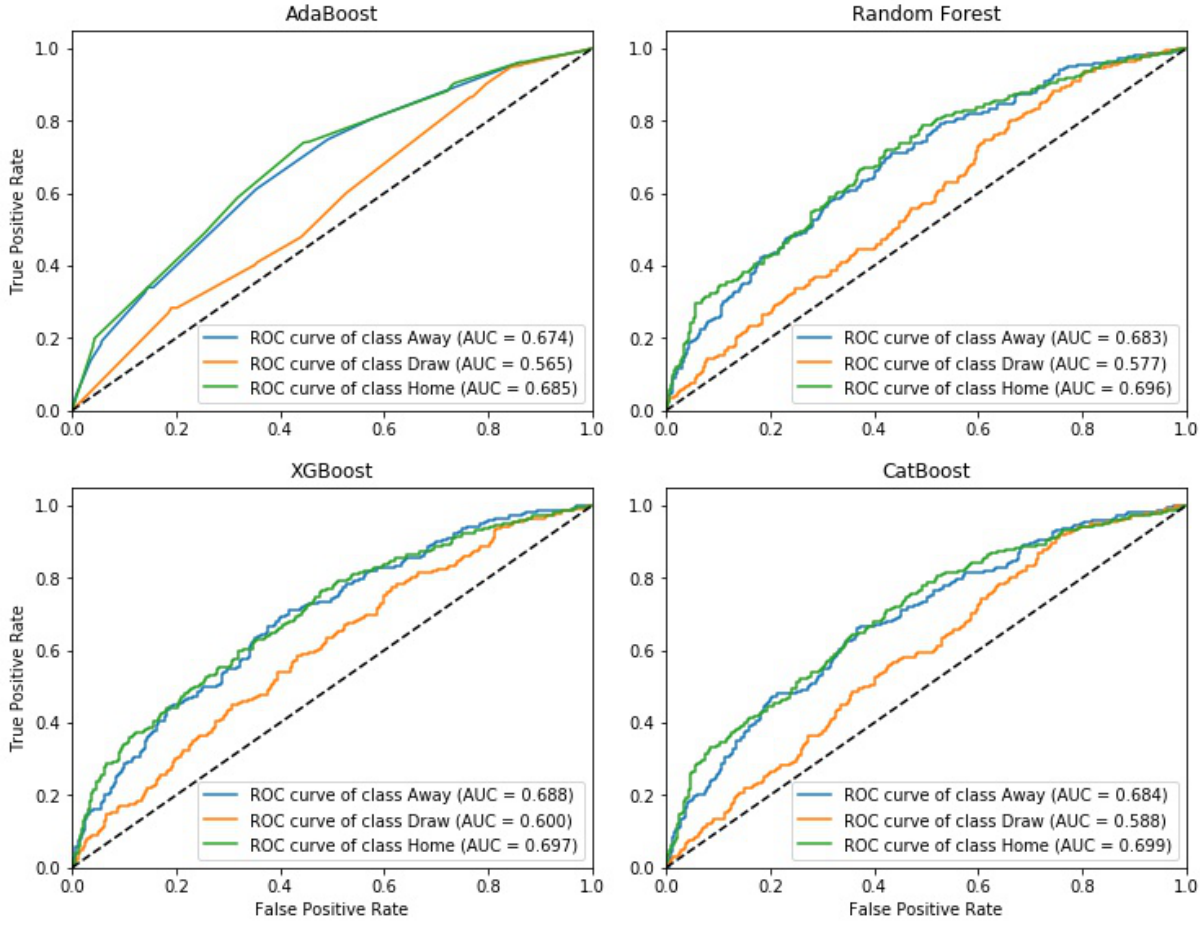


Source: own preparation.

Figure 2. presents histograms of estimated probabilities distributions per each class. AdaBoost's distributions differs from the other models. Empty spaces between parts of this

histogram suggest that is the simplest model in terms of probabilities' estimator. Indeed, trained AdaBoost classifier has got 20 estimators being decision trees, all of them with depth of 2 which means that AdaBoost finally does not have many splits determining the division of estimated odds. It is expected that distributions should have more continuous structure that shows distributions from the other classifiers. All of the histograms confirm that home wins are the most likely football outcomes. Moreover, for all classifier at least 90% observations have less ten 30% probabilities of draw that explain statement that these outcomes are the most difficult to capture. This fact can also be seen in Figure 3., which demonstrates ROC curves and AUC scores.

Figure 3. ROC curves on validation set per each class for all models.



Source: own preparation.

We start our analysis with AUC for draw outcomes. Score is between 0.57 (AdaBoost, Random Forest) and 0.6 (XGBoost). When AUC is about 0.5, the estimated classifier has no discrimination capacity to find differences between positive and negative classes. This can be equated with making decisions about class selection based on a coin toss, which is in fact an

almost completely random decision. An area under dashed diagonal line equals exactly 0.5. AUC values just over 0.5 suggest that finding draws in each of the models turned out the most difficult challenge. XGBoost was the best in terms of searching draws.

For all models AUC scores for home wins (between 0.69 and 0.7) are slightly higher than for away wins (between 0.67 and 0.69). Taking into consideration all AUC values per model, it can be concluded that XGBoost and CatBoost have the best and comparable results. While all scores are inconsiderably lower than XGBoost and CatBoost in Random Forest, the scores appear to be comparable as a whole, suggesting further exploration steps by analysing results in another way. We are going to analyse calculated predictions, and not only from the built models, but also together with estimated odds from bookmakers, in which the RPS will help.

Table 5. RPS values for estimated models and calculated on train set, whole validation set (All) and with the division validation set into two seasons.

	AdaBoost	**Random Forest**	**XGBoost**	**CatBoost**
Train set	0.1876	0.1849	0.1799	0.1786
Validation set (All)	0.2020	0.2004	0.1989	0.1995
Validation set (2018-2019)	0.2051	0.2037	0.2018	0.2025
Validation set (2019-2020)	0.1989	0.1970	0.1960	0.1964

Source: own preparation.

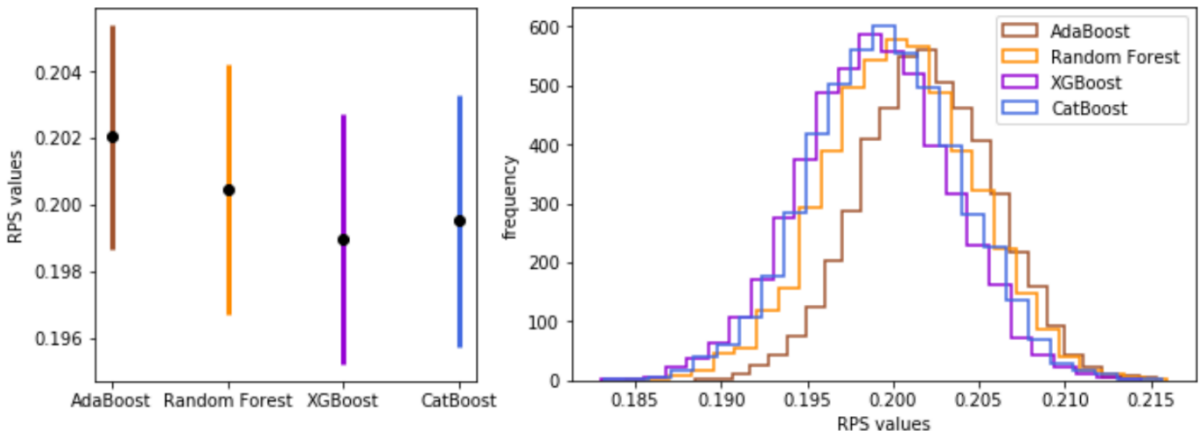
Let us remind that one of the main goals when determining the prediction is finally minimizing the RPS calculated on the validation set, which was not involved in either the learning process or the hyperparameters tuning process. It should be noted that both the results counted for the entire validation set as well as the division into individual seasons do not change the order in which we would rank the models from the best to the least predictive of football outcomes.

Table 5. confirms the earlier observations regarding AdaBoost that achieves the highest RPS values (0.2020 for whole validation set), which seems to be the weakest estimator. From Table 5., it can also be concluded that Random Forest is a third choice. It has to be noted that the difference between Random Forest RPS and values from XGBoost and CatBoost is less than the distance from AdaBoost results. Especially this observation is visible for scores for 2019-2020 season. XGBoost obtained the best RPS regardless of the set under consideration - 0.1989 for whole validation set, which is better than CatBoost and Random Forest by 0.0006

and 0.0015, respectively. The highest advantage of XGBoost over the other models is in the results of RPS based on predictions from 2018-2019, which may suggest that XGBoost would be better at predicting potentially more difficult football games in terms of forecasts. When it comes to CatBoost model, it came closest to the XGBoost results in 2018-2019 (0.1964 vs. 0.1960), although the overall discrepancies are stable for each analysed set. The order of RPS values from train set is quite correlated with the results from validation set, performances of AdaBoost and Random Forest were lower than XGBoost and CatBoost. It suggests that the model optimization processes were carried out correctly and overfitting was avoided.

The above analysis on train set and different subsets of validation set indicates a certain order of trained models in terms of RPS values: best results from XGBoost, then CatBoost, Random Forest and the last AdaBoost. However, being careful with the final verdict of which of them is the most appropriate, it was decided to consider the predictive power of the models in more depth using the bootstrap approach. It was used for further inference: 60% of observations were drawn from the validation set without replacement and it was repeated 5000 times. In order to compare further calculations not only as a whole, but also for individual samples, the sampling was performed once for all the models jointly. For each bootstrapped sample and for all trained models, the RPS was calculated separately. Then the results were aggregated and presented in the graphs below.

Figure 4. Results from trained model based on bootstrap approach – left graph: averaged RPS with 95% confidence interval, right graph: empirical distributions of RPS.



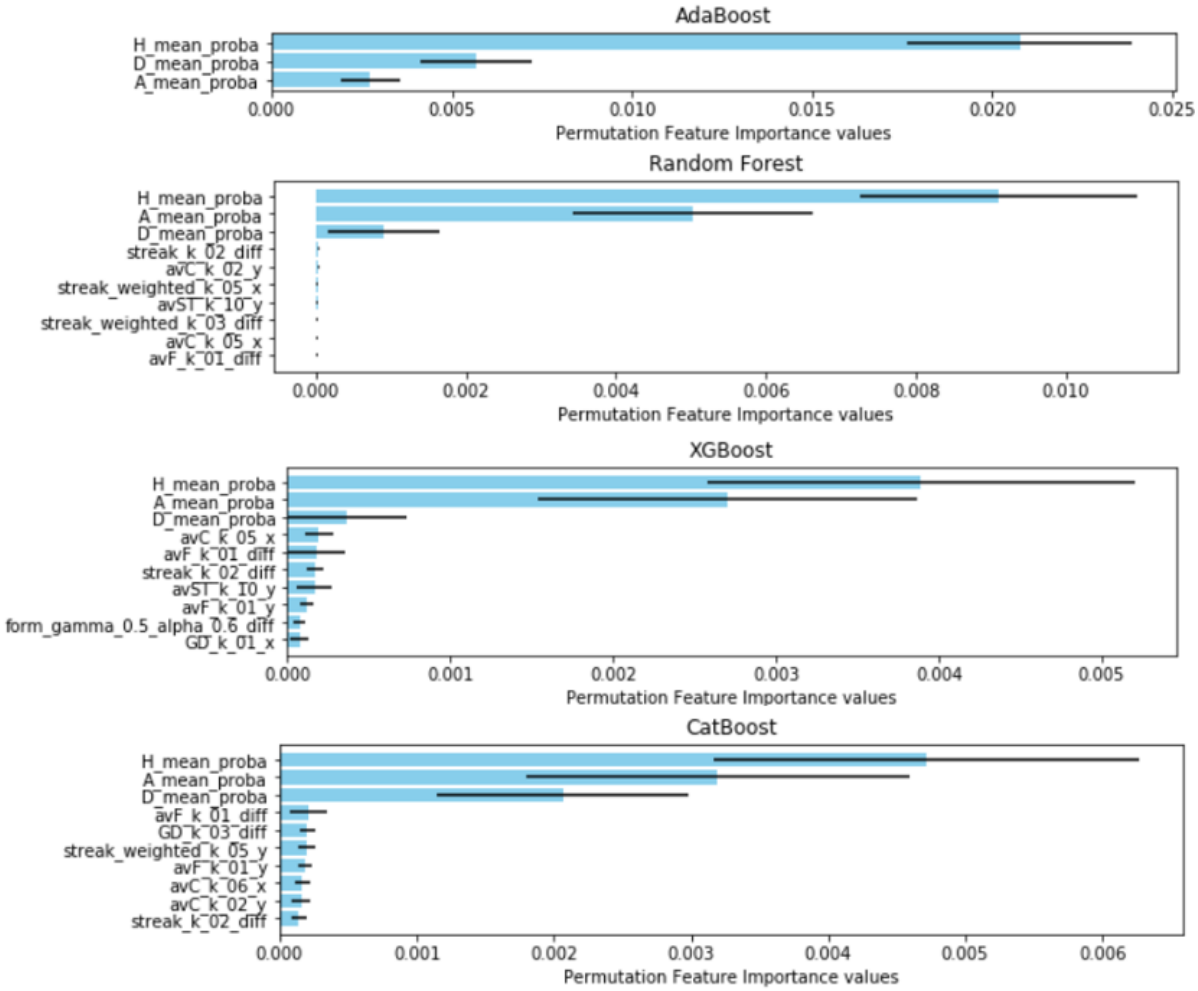
Source: own preparation.

The graph representing the average results (RPS^{boot}_{mean}) still indicates an advantage of XGBoost (0.1988) over the other estimated models (0.1998, 0.2003, 0.2020 for CatBoost,

Random Forest and AdaBoost, respectively). A 95% confidence interval was selected for the analysis, that is, the standard deviation (let denote us $RPS^{boot}_{\sigma(0.95)}$) was calculated after truncating the values at the 2.5 and 97.5 percentiles. This reduced the impact of outliers and allowed for the conclusion that 95% likelihood of classification score is between $RPS^{boot}_{mean} - RPS^{boot}_{\sigma(0.95)}$ and $RPS^{boot}_{mean} + RPS^{boot}_{\sigma(0.95)}$. The widths of confidence intervals are very similar, CatBoost is slightly narrower than the others. However, the mean RPS for CatBoost is approximately equal to the third quartile of XGBoost, which can also be seen when looking at the discrepancies between the histograms on the right graph of the Figure 4. The histogram of AdaBoost is clearly shifted to the right in relation to the other models, and let us remind that the more the empirical distribution is shifted to the right, the more it indicates the weaker predictive power of the trained model. In the histograms we can also observe a slightly worse Random Forest performance compared to CatBoost and XGBoost, especially when we look at the parts of distributions to the left of the medians. In general, it should be noted that all empirical distributions can be considered symmetrical, which confirms the validity of the inference also based on confidence intervals.

All conclusions from the analysis of RPS led to further exploration of XGBoost and CatBoost classifiers compared to bookmakers. Nevertheless, before we compare the estimated predictions of football outcomes from chosen trained models and forecasts of bookmakers, we will conduct a study of the impact of individual features that have been used in built classifiers using the Permutation Feature Importance (PFI).

Figure 5. Permutation Feature Importance for features from trained models.



Source: own preparation.

PFI calculation method is presented in the second section. By repeating the shuffle procedure 20 times for each feature, 95% confidence intervals were also obtained, the widths of which were marked on Figure 5. with black horizontal lines. For AdaBoost only 3 variables are presented because the others were not selected during learning, for the rest models top 10 features are visible. All results of PFI indicate that the most important features during training process were variables on betting odds. Especially 1st places were assigned to estimated probabilities of home win events (H_mean_proba). The difference between PFI values for top 3 features and the others was noticeable. For XGBoost and CatBoost, the variables based on the statistics from k games also turned out to be important. Overall, variables form betting odds showed the greatest significance, but the AdaBoost example shows that using only these variables did not necessarily determine the training of a better classifier. Other observation is

that for Random Forest, XGBoost and CatBoost there are visible features based on statistics for different *k* values.

4.2. Compare results with bookmakers

Last but not least - this part will be devoted to comparing the results with predictions from selected bookmakers. The selected betting bets were William Hill (WH), Bet365 (B365), Interwetten (IW), bwin (BW) and BetVictor (VC). Besides, 2 of the 4 best estimated models - XGBoost and CatBoost were selected for the comparative analysis. The table below shows the RPS values for the different subsets of the setting validation set.

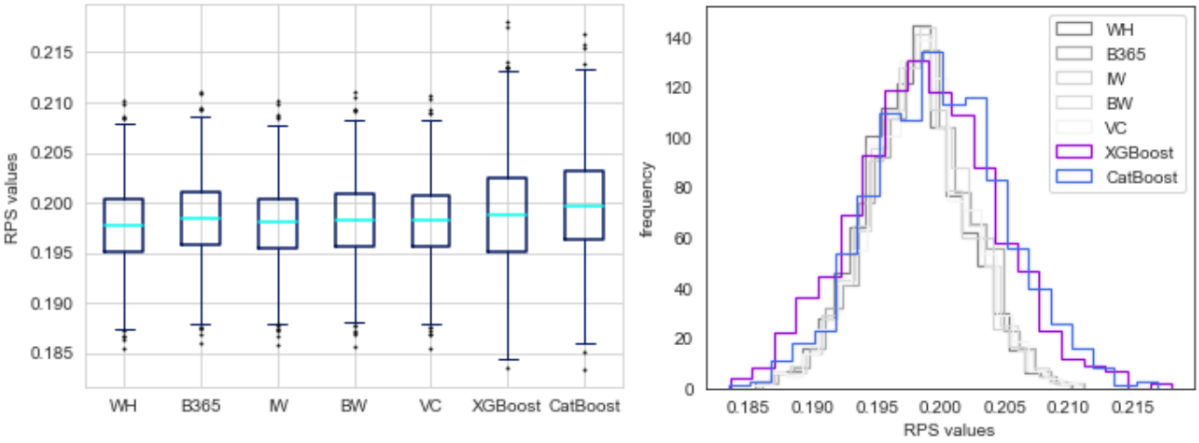
Table 6. RPS for bookmakers and estimated chosen models calculated on whole validation set (All) and with the division validation set into two seasons.

	Bookmakers					**Estimated models**	
	B365	**BW**	**IW**	**VC**	**WH**	**XGBoost**	**CatBoost**
All	0.1984	0.1982	0.1979	0.1981	0.1977	0.1989	0.1995
2018-2019	0.2008	0.2004	0.2000	0.2006	0.2001	0.2018	0.2025
2019-2020	0.1959	0.1960	0.1957	0.1957	0.1952	0.1960	0.1964

Source: own preparation.

William Hill achieved the best RPS on all observations from validation set and amounts to 0.1977. If we look at different seasons, Interwetten and BetVictor have comparable results, followed by Bet365 and bwin. We can see that RPS for XGBoost and CatBoost are slightly worse than selected bookmakers. However, it should be noted that the discrepancy between the results is relatively small, especially when comparing XGBoost with RPS from bookmakers. Moreover, it is optimistic that for the 2019-2020 season XGBoost obtained the same result as one of the bookmakers (0.1960) and CatBoost had a slightly worse result - 0.1964. In general, the discrepancies between the periods in which we analyse RPS indicate that, as in the section comparing the results for all trained models, use the bootstrap technique for a deeper comparative analysis of the estimated predictions of football outcomes. The figure below presents graphical representation of results of bootstrapped approaches to compare performances between bookmakers and trained models.

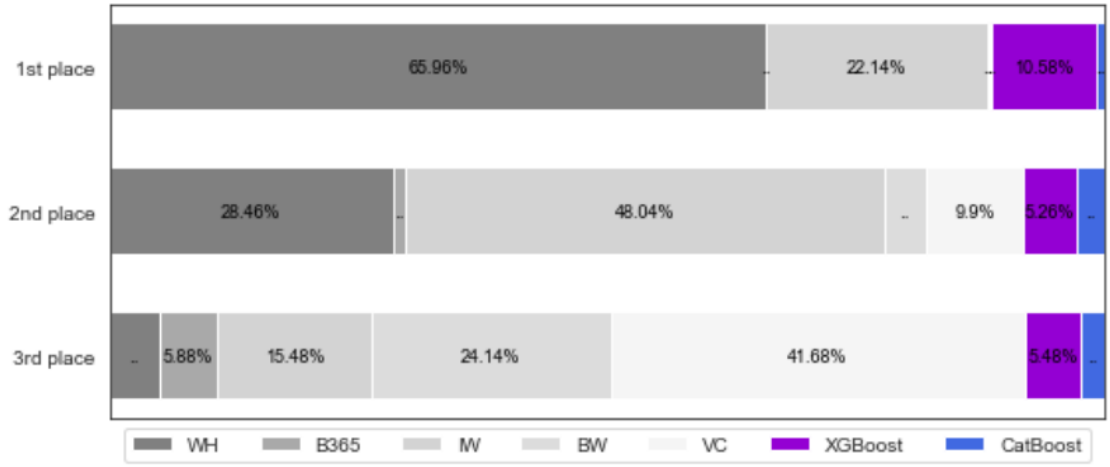
Figure 6. Results from chosen trained model and predictions from bookmakers based on bootstrap approach – left graph: boxplots, right graph: empirical distributions of RPS.



Source: own preparation.

When observing both boxplots and histograms, it can be seen that the distributions from trained models have slightly heavier tails than the distributions based on betting odds. This is especially visible in the boxplots, since the distances between the extremes are clearly greater than for the others. Interestingly, this may indicate a greater sensitivity of the estimated classifiers depending on the bootstrapped samples. While the variance of these results is nominally the best result for XGBoost and CatBoost for certain samples, one should not draw far-reaching conclusions and analyse boxplots also by looking at its other components. There is a higher RPS median for CatBoost compared to other distributions, which could also be assumed by looking at the histograms. Overall, the distributions based on bookmakers' forecasts are slightly shifted to the left and more stable in terms of variances. All distributions can be considered symmetric, which is due to both the position of the quartiles in the boxplots and the histogram. To see how the trained model performed in comparison with bookmakers, we also analysed podium places for individual bootstrapped samples.

Figure 7. Distributions of individual places after ordering RPS for bootstrapped samples for trained models and bookmakers.



Source: own preparation.

For clarity of Figure 7., if the percentage was less than 5%, it was masked. After arranging the RPS values for individual bootstrapped samples, it is necessary to confirm the presumption that the odds from William Hill had the best approach to the observed football outcomes, wins in almost two-thirds of the cases (66%). The next place goes to Interwetten, which had the lowest RPS in about one in five samples (22%), but also second place in almost half of the cases (approximately 48%). It may be satisfying that XGBoost was third in the order with the lowest RPS (for one in ten samples), and also appeared on the second and third place of the podium much more often than CatBoost. Although BetVictor appeared in the top 3 more often than XGBoost (52% vs. 21%), it was definitely on the top step of the podium less times (less than 1% vs. 11%). It confirms higher stability predictions from bookmakers, however, it also makes that XGBoost can compete with them in certain cases.

5. Summary and conclusions

In this paper, a very interesting, practical and applicable case concerning forecasting results in football were considered.

At the beginning, the main focus was on feature engineering. Efforts have been made to ensure that the proposed variables simultaneously reflect the actual and overall strength of individual teams before the game, and can capture various details that may affect football outcomes. The ideas were based on concepts from the literature overview (for example, Form Coefficients or ratings extracted from FIFA Index) and additional novel modifications were

made, which allowed for obtaining variables dynamically adjusting to the diverse form of football teams from the Spanish league.

A total of 228 variables were considered, 140 of which, based on various statistical techniques of feature selection (Mutual Information, ANOVA F-value and recursive feature elimination), were chosen for modelling. Feature selection methods allowed to effectively reduce the number of variables and remove potentially correlated ones. Available dataset from LaLiga was split into train and validation sets, 9 seasons from 2009-2010 to 2017-2018 and 2 seasons from 2018-2019 to 2019-2020, respectively. The algorithms were trained with 3-folds cross validation method with Bayesian optimization method for hyperparameters tuning.

XGBoost turned out to be the best of the learned estimators with the lowest RPS value on validation set - 0.1989. In addition, this performance made it possible to beat the bookmakers for around 11% of the bootstrapped samples, which certainly also shows the quite high quality of the estimated predictions. In further research, XGBoost was certainly calibrated even more precisely for new dataset and, above all, based on new variable proposals, it could be a real competition for bookmakers. An interesting idea could also be to use XGBoost for iterative forecasting of results for a given football division more reflecting the actual operation of the tools used by bookmakers. That is, in this approach the model could be trained for historical data and then round by round properly calibrated and updated, so that it could even better detect e.g. difficult to capture draws.

CatBoost presented the results slightly different from the best XGBoost - RPS on validation set equals 0.1995. CatBoost - as it is worth emphasizing being the newest method in terms of this lifetime among the selected methods - has demonstrated its potential during the presentation of the results. Moreover, CatBoost was also created as a competition to XGBoost for cases where categorical variables were used. Hence, CatBoost would certainly be used for further research, especially when these categorical types of variables were applied in subsequent iterations of work dealing with prediction of outcomes. Random Forest and AdaBoost turned out worse than XGBoost and CatBoost in terms of RPS values for both train and validation set.

The results also include a greater explainability of models based on the Permutation Feature Importance, allowing to measure the impact of individual variables on RPS, as well as explaining which variables played the most significant role in learning processes. It occurred that for all approaches the features based on betting odds turned out to be crucial, which only confirmed the high quality of estimated by bookmakers' predictions. In addition, each model had slightly different preferences for other important variables. An interesting observation was

the appearance of statistics calculated for a different past number of games, which indicates a good direction of work, not limited to one selected historical period.

Beating bookmakers would undoubtedly be a great achievement. Their detail-oriented and deep knowledge about the teams before each game and rounds makes comparing with predictions updated by analysts during the season turned out to be more challenging than initially expected. Nevertheless, the results of the best trained XGBoost models were promising and demonstrated that competition with bookmakers is possible.

Several ideas for future works have already been mentioned in this section and others are presented in this paragraph. Long series of wins of losses are relatively rare in football. Perhaps it would be worthwhile to build a parameter that would consider the fact that the longer the series of matches with identical outcomes, the less chance that this result will repeat. It is also a valuable idea to include other competitions that are taking place during the season. Many Spanish teams play in other national competitions (Copa Del Rey) or in the European arena (UEFA Champions League, UEFA Europa League). The high importance of bookmaker odds means that in subsequent studies one should also consider another method of aggregating bookmakers' knowledge, e.g. by dividing it into different subgroups or using odds modelled for individual events in the match, such as the number of goals. Overall, predicting football results can be considered an inexhaustible topic suitable for further and relentless exploration, both for football enthusiasts and researchers.

6. References

Aly, M., 2005. *Survey on multiclass classification methods.* https://csaikku.files.wordpress.com/2012/01/aly05multiclass.pdf

Aulia, D. & Murfi, H., 2020. *XGBoost in handling missing values for life insurance risk prediction*. SN Applied Sciences, 2(8).

Baboota, R. & Kaur, H., 2018. *Predictive analysis and modelling football results using machine learning approach for English Premier League*. International Journal of Forecasting, 35(2).

Bergstra, J., Yamins, D. & Cox, D., 2012. *Making a Science of Model Search*. Proceedings of the 30-th International Conference on Machine Learning. JMLR: W&CP volume 28.

Berrar, D., Lopes, P. & Dubitzky, W., 2019. *Incorporating domain knowledge in machine learning for soccer outcome prediction*. Machine Learning. 108(7), pp. 97-126.

Breiman, L., 2001. *Random Forests.* Machine Learning, 45(1), pp. 5-32.

Buursma, D., 2011. *Predicting sports events from past results "Towards effective betting on football matches".* 14th Twente Student Conference on IT, Twente, Holland.

Chen, T. & Guestrin, C., 2016. *XGBoost: A Scalable Tree Boosting System*. San Francisco, ACM.

Constantinou, A., 2019. *Dolores: a model that predicts football match outcomes from all over the world*. Machine Learning, 108(3), pp. 49–75.

Constantinou, A. & Fenton, N., 2012. *Solving the Problem of Inadequate Scoring Rules for Assessing Probabilistic Football Forecast Models.* Journal of Quantitative Analysis in Sports, 8(1).

Constantinou, A., Fenton, N. & Neil, M., 2013. *Profiting from an inefficient Association Football gambling market: Prediction, Risk and Uncertainty using Bayesian networks.* Knowledge-Based Systems, 50, pp. 60–86.

Dorogush, A., Ershov V. & Gulin A., 2017. *CatBoost: gradient boosting with categorical features support.* Workshop on ML Systems at NIPS 2017.

Frazier, P., 2018. *A Tutorial on Bayesian Optimization*. https://arxiv.org/pdf/1807.02811.pdf

Guyon, I., Weston, J., Barnhill, S. & Vapnik, V., 2002. *Gene Selection for Cancer Classification Using Support Vector Machines*. Machine Learning. 46(1-3), pp. 389-422.

Hubáček, O., Sourek, G. & Železný, F., 2019. *Learning to predict soccer results from relational data with gradient boosted trees.* Machine Learning, 108(2).

Hucaljuk, J. & Rakipovic, A., 2011. *Predicting football scores using machine learning techniques*. Proceedings of the 34th International Convention MIPRO, pp. 1623-1627, IEEE.

Joseph, A., Fenton, N. & Neil, M., 2006. *Predicting football results using Bayesian nets and other machine learning techniques*. Knowledge-Based Systems, 19(7), pp. 544-553.

Kohavi, R., 2001. *A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection*. https://www.ijcai.org/Proceedings/95-2/Papers/016.pdf

Kumar, M., Rath, N., Swain, A. & Rath, S., 2015. *Feature Selection and Classification of Microarray Data using MapReduce based ANOVA and K-Nearest Neighbor*. Procedia Computer Science, 54, pp. 301-310.

Lei, J., G'Sell, M., Rinaldo, A., Tibshirani, R. & Wasserman, L., 2016. *Distribution-Free Predictive Inference For Regression*. Journal of the American Statistical Association, 113(523).

Mccabe, A. & Trevathan, J., 2008. *Artificial Intelligence in Sports Prediction.* Fifth International Conference on Information Technology: New Generations, Las Vegas, Nevada, USA, pp. 1194-1197.

Owramipur, F., Eskandarian, P. & Mozneb, F., 2013. *Football Result Prediction with Bayesian Network in Spanish League-Barcelona Team*. International Journal of Computer Theory and Engineering, 5(5), pp. 812-815.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E. & Louppe, G., 2012. *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, 12.

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. & Gulin, A., 2018. *CatBoost: unbiased boosting with categorical features*. NeurIPS.

Putatunda, S. & Kiran R., 2018. *A Comparative Analysis of Hyperopt as Against Other Approaches for Hyper-Parameter Optimization of XGBoost*. SPML '18: Proceedings of the 2018 International Conference on Signal Processing and Machine Learning, pp. 6-10.

Rabinowicz, A. & Rosset, S., 2021. *Trees-Based Models for Correlated Data.* https://www.researchgate.net/publication/349363567_Trees-Based_Models_for_Correlated_Data

Razali, N., Mustapha, A., Yatim, F. & Aziz, R., 2017. *Predicting Football Matches Results using Bayesian Networks for English Premier League (EPL)*. IOP Conference Series: Materials Science and Engineering, 226(1).

Ribeiro, M., Singh, S. & Guestrin, C., 2016. *Model-Agnostic Interpretability of Machine Learning*. ICML Workshop on Human Interpretability in Machine Learning, New York, NY, USA.

Smith, R., 2015. *A Mutual Information Approach to Calculating Nonlinearity*. Stat 2015; 4(1), pp. 291–303.

Tax, N. & Joustra, Y., 2015. *Predicting The Dutch Football Competition Using Public Data: A Machine Learning Approach*. IEEE Transactions on Knowledge and Data Engineering, vol. 10(10).

Tuv, E., Borisov, A., Runger, G. & Torkkola, K., 2009. *Feature Selection with Ensembles, Artificial Variables, and Redundancy Elimination*. Journal of Machine Learning Research, 10, pp. 1341-1366.

Wheatcroft, E., 2019. *Evaluating probabilistic forecasts of football matches: The case against the Ranked Probability Score*. https://www.researchgate.net/publication/335420083_Evaluating_probabilistic_forecasts_of_football_matches_The_case_against_the_Ranked_Probability_Score

Wilks, D., 2005. *Statistical Methods in the Atmospheric Sciences*. Second Edition, Academic Press.

Zaveri, N., Tiwari, S., Shinde, P., Shah, U. & Teli, L., 2018. *Prediction of Football Match Score and Decision Making Process*. International Journal on Recent and Innovation Trends in Computing and Communication, 6(2), pp. 162-165.

Zhou, Z., 2009. *Ensemble Learning*. Encyclopedia of Biometrics, Springer US, pp. 270-273.

7. List of tables

8. List of figures

University of Warsaw
Faculty of Economic Sciences
44/50 Długa St.
00-241 Warsaw
www.wne.uw.edu.pl