What Contributes to the World Cup Results? Prediction of the FIFA World Cup 2022 with Ensemble Learning Approaches

Adelina Nie, Runyuan Deng

Econ 420 Topics in Machine Learning

Abstract. This paper compares two novel ensemble methods, *Random Forest and Gradient Boosting*, which are used to predict the score of FIFA world cup matches by using the feature-engineered data containing all FIFA international matches covered from 1872 – 2022 and world rankings from 1992-2022. A 2022 world cup simulation is subsequently performed through the optimized best-performing model, namely, gradient boosting. In the paper, the results are also extracted by using three machine learning explainability methods, permutation importance, partial dependency plot, and Shapley value, to better understand the underlying mechanism of the prediction models and the critical factors that contribute to the final result of the game and how each of them leads to the outcome.

Keywords: FIFA World Cup; Machine learning; Data mining; Football prediction

1. Introduction

Machine learning has been becoming an increasingly powerful tool in sports, especially football tournaments. Various professional teams are currently reliant on the use of cutting-edging machine learning techniques for their operations. The optimized results and conclusions based on these innovative methods assist professional clubs in scouting, administration, and performance. The successes also indicate the massive potential of machine learning in football forecasting. As a matter of course, the FIFA World Cup, the most prominent association football tournament in the world, has attracted the attention of mainstream scholars who manage to predict the champion. One classical approach is based on the prospective information contained in the odds of bookmakers using a consensus model (Leitner, Zeileis & Hornik, 2010b). Through reverse simulation, they can compute the underlying strengths of teams from winning probabilities of teams transformed by the winning odds from online bookmakers. Therefore, it is possible to rate teams in a competitive game into a common framework, and the forecasting program of the ultimate winner of one tournament is thus completed. Under this methodology, it is predicted that the highest winning probability at 2018 World Cup is 16.6% for Brazil, followed by a winning probability of 15.8% for the defending champion Germany. (Zeileis, Leitner, and Hornik 2018)

One may suggest that another class of statistical methods that performs well in estimating football tournaments, World Cups for instance, is the class of Poisson regression models used to directly model the number of goals scored by both competing teams in a single match of the tournament. Let X_{ij} and Y_{ij} indicate the goals scored by the two teams, respectively, in a game between teams i and j, where i, j $\in$ {1,...,n} and n represents the total number of teams in the tournament concerned. One assumes that $X_{ij} \sim$ Po(λ_{ij}) and $Y_{ij} \sim$ Po(μ_{ij}) where λ_{ij} and μ_{ij} denote the intensity parameters (i.e. the expected number of goals) of the respective Poisson distributions. For these intensity parameters, there are several modeling strategies co-existing, from conditionally independent possion distribution (see Dyte & Lake 2000), bivariate Poisson distribution (see Karlis & Ntzoufras 2003), to copula-based models (see, for example, McHale & Scarf, 2007), which incorporate the playing abilities or covariates of the competing teams in a progressive way.

With the development of machine learning technique, the so-called *random forest,* a fundamentally different ensemble learning approach has been proposed by Breiman (2001) for classification and regression tasks with satisfactory prediction outcomes achieved. According to the preliminary work of Schauberger and Groll (2018), it can be found out that the predictive performances with random forests in football matches lead to remarkable results, outperforming the models proposed in literature, (i.e., the bookmaker consensus model and the class of Poisson models that are based on the teams' covariate information as mentioned above). These encouraging results prompt the author to use random forests as one of the algorithms in the present study to predict the ongoing matches of FIFA World Cup 2022.

Gradient Boosting, which is another prediction model in the form of an ensemble of weak prediction models used to perform regression and classification, usually outperforms the random forests(Hastie, T. & Tibshirani, R 2009; Piryonesi, S. Madeh & El-Diraby, Tamer 2021). However, the focus of existing literature is placed on how to further optimize the accuracy of Poisson models and random forests in the practice of football tournament prediction, with no research conducted yet by using this algorithm. To fill this research gap, the gradient boosting is used in this paper as the second algorithm to compare the testing results of it with that of random forest. In this way, a better-performing model is identified for the final prediction simulation.

In addition, the paper places emphasis on explainability, which is one factor of machine learning that drives the focus not only on the results but on the data's inherent patterns and the ability of the algorithms to explain them. Thus, in addition to predicting the winning probability of world cup, the paper also aims to identify the features that play a more significant role in contributing to the winning of the match and how each significant feature influences and jointly determines the prediction result of the model through permutation importance, partial dependency plot and Shapley value respectively.

The rest of the manuscript is structured as follows: in Section 2 the paper explains the basic idea of random forest, gradient boosting, and some explainability tools. Next, in Section 3 a general workflow of modeling and forecasting of the World Cup 2022 is introduced. Then, the importance of the features is quantified and how each of them jointly contributes to the outcome explained in Section 4. Finally, everything will be summarized in the conclusion section.

2. Methodology

The paper aims to work on two tasks, predicting the outcomes of the FIFA World Cup 2022 and interpreting the importance of the features contributing to the outcome. This section will briefly explain the approaches we use for outcome prediction and feature interpretation.

2.1 Data Source

To simulate the FIFA 2022 World Cup game, we obtain 2 data sets, *International Football Results from 1872 to 2022* and *FIFA World Ranking 1992-2022* from *Kaggle* (2022), an online community platform for data scientists where users are allowed to publish and acquire data sets. The former data set contains all historical international A-level football matches from 1872-2022, covering the World Cup, Confederations Cup, continental championships, Olympic Games, international friendlies, etc. The later records the continuously updated information of the world ranking of national teams.

2.2 Prediction Algorithms

We will use two machine learning models, random forest, and gradient boosting, to model the features and the outcome, from which we choose the more accurate one to make final predictions for the FIFA World Cup 2022.

2.2.1 Random forest

Random forest has become one of the most used supervised machine learning models for non-linear regression and classification problems by improving on a single decision tree to reduce bias and variances. It employs the idea of *bootstrap aggregating* (or *bagging* in short) to generate different training data samples from the original single data set, where a decision tree is trained for each of the bootstrapped training data. The algorithm creates an ensemble of trees instead of just getting a single tree, from which we average out all the predictions. To make sure that each tree is uncorrelated from the other, random feature selection is also performed by only choosing a subset of m predictors out of a total of p predictors at each split. This is called the "decorrelation" of the trees. (James et.al, 2021) For the classification problem of our study, the prediction assigns a class label to an unlabeled instance which is achieved by majority voting. Each decision tree votes for its predicted class label, and the instance will eventually be classified under the class label with the most votes. (Fawagreh, Gaber & Elyan, 2014)

In our work, we will use random forests with classification trees to predict whether each team will win or lose the game. For this purpose, we will use the predictors introduced in section 3 and import *RandomForestClassifier* with *GridSearchCV* from the *scikit-learn* library in Python.

2.2.2 Gradient boosting

While the random forest creates a collection of uncorrelated trees based on different subsets of observations and predictors, the boosting method trains the trees on a sequential basis where the construction of the latter trees is dependent on the fitted values of the previous trees (James et.al, 2021). To illustrate how this works, suppose we have trained the first decision tree $f_1(x)$ using the original data. We construct the second tree $f_2(x)$ so that the errors from the first tree are corrected. Regressing on the residual values $y - \lambda f_1(x)$ from the first tree, we obtain the second tree $f_2(x)$ and get new residuals $y - \lambda f_1(x) - \lambda f_2(x)$ where λ is the learning rate of boosting. The same step is repeated for all the later trees until the errors are reduced satisfactorily. Now we incorporate gradients into this boosting process. Let $L(y_i, \gamma)$ be the square loss function which we would like to minimize. For m = 1 to M (the number of features), the gradient boosting is completed with the following steps (Friedman, 2001):

1) Initialize the model with constant value γ:

$$F_0(x) = argmin_{\gamma} \sum_{i=1}^{N} L(y_i, \gamma)$$

2) Compute the gradient direction of residuals $r_i =- [\frac{\partial L(y_i,(fx_i))}{\partial f(x_i)}]_{F(x)=F_{m-1}(x)}$ for all *i*

3) Get the parameter a_m according to least square approach and fit the model $h(x_i, a)$ using the residuals from last step.

$$a_m = argmin_{a,\gamma} \sum_{i=1}^{N} [(r_i - \gamma h(x_i, a)]^2$$

4) Calculate the weight of the current model:

$$\gamma_m = argmin_{a,\gamma} \sum_{i=1}^{N} L(y_i, F_{m-1}(x_i) + \gamma h(x_i, a))$$

Like the implementation of random forest, we construct gradient boosting decision trees by importing the *GradientBoosting Classifier* from *sklearn.ensemble* in Python, combined with the optimization tool *GridSearchCV*.

2.2.3 Grid search cross-validation

We emphasized in the section 2.2.1 and 2.2.2 that the random forest model and gradient boosting decision trees are implemented together with *GridSearchCV*. It chooses the optimal model by performing hyperparameter optimization guided by k-fold cross validation which divides the data set into k groups with k-1 training sets and 1 validation set. In our models, the hyperparameters are the learning rate in gradient boosting, or the number of variables in each split, the node size, and the number of trees etc. in random forest (Probst, Wright & Boulesteix, 2019). Grid search is one of the most common approaches for hyperparameter tuning which starts with a large search space step size and gradually narrows them based on the previous results (Yang & Shami, 2020). The algorithm evaluates all the possible combinations of hyperparameters and selects the one with the highest average performance score, which is indicated by R^2 (Liu, Liu & Feng, 2022).

2.3 Evaluation of Model Performance

We use two measures, the confusion matrix, and the AUC score, to assess the model's performance and the accuracy of the predicted results. We produce the confusion matrix via the *confusion_matrix()* function from the *sklearn* library in Python for our binary classification problem. The confusion matrix is composed of 4 cases: True Negative (TN), False Positive (FP), False Negative (FN), and True Positive (TP). We can calculate the accuracy from the confusion matrix by the following formula (Kulkarni1, Chong, & Batarseh, 2020):

$$Accuracy = \frac{TN + TP}{TN + FP + FN + TP}$$

AUC score is another indicator usually used in machine learning model comparison. This score indicates the area under the ROC curve, which plots the True Positive rate against the

False Positive rate. Ranging from 0 to 1, we can interpret the AUC score as the probability that the model is able to classify observations correctly into the corresponding classes. (Carrington et.al, 2021) If the AUC score is much higher on training data than on test data, then the model is overfitting; otherwise, the model is underfitting. Thus, a model with reliable predictions should have AUC scores relatively consistent on both training and test data.

2.4 Feature Interpretation

There is always a trade-off between the predictive performance and interpretability of the machine learning models. While the models introduced in sections 2.2.1 and 2.2.2 usually perform very well in prediction, their complexity makes it challenging for humans to interpret the models' internal logic and understand how the decision is made (DV, EM & JS., 2019). Therefore, a significant part of our work is determining which features contribute the most to the outcome and explaining how they affect the model's prediction. We will use three approaches of model interpretation: Permutation Importance, Partial Dependence Plot, and Shapley Values.

2.4.1 Permutation importance

Permutation importance (PI) is the first measurement we use for model interpretability, which aims to calculate the importance of each feature to the outcome by shuffling the observations of this feature while keeping the others unchanged. According to Huang, Lu and Xu (2016), the PI value is defined as the difference between the errors of the new model after rearranging the values of one feature and the original model. If the feature significantly impacts the outcome variable, then the feature's PI value should be high in absolute value; otherwise, the feature tends to be relatively insignificant to the model. We compute the PI values for both random forest and gradient boosting. Since both models are ensembles of trees, the PI value of each feature will be computed for every tree respectively, and the final PI value of the feature will just be the average PI value of each tree.

2.4.2 Partial dependence plot

While determining predictor importance only constitutes part of the machine learning problems, it is also crucial to assess how the changes in each feature contribute specifically to the outcome prediction. For this purpose, we utilize partial dependence plot (*PDP*), a model interpretation tool which visualizes the marginal effect of each feature on the response variable while keeping the other features unchanged. A PDP is constructed by the following simple algorithm (Greenwell, 2017):

1) Let $x_i = \{x_{i1}, x_{i2}, x_{i3}, x_{i4}, \ldots, x_{ij}\}$ be one of the features in the model.
2) We make j copies of training data by replacing the whole feature column x_i with each of its value from x_{i1} to x_{ij}.
3) Obtain the predicted values from each copy of the training data described above and compute the average prediction $f(x_{ij})$ for each value of this feature.
4) Plot the average prediction $f(x_{ij})$ against the corresponding feature value x_{ij}.

Based on the features selected by permutation importance mentioned in the previous sub-section, we create the partial dependence plots on those features via the display function *PartialDependenceDisplay.from_estimator()* from *sklearn.inspection*.

2.4.3 Shapley value

Introduced by Lloyd Shapley in 1953, the Shapley Value is an economic concept solution in cooperative game theory which computes individual's average marginal contributions in all possible coalitions of players. To illustrate how this works, suppose there are 3 players, *a, b,* and *c* in the game where each of them make contribution to the total payoff. The possible cases of coalitions are as follows (Benchekroun, 2022):

1) No coalition: {a}, {b}, {c}
2) Coalitions with 2 players: {a, b}, {b, c}, {a, c}
3) Grand coalition with all players cooperating: {a, b, c}

Let *v ()* denote the payoff function, and consider the marginal contribution (MC) of *player a* in the following cases:

1) Player a is the first one entering the game with orders {a, b, c} and {a, c, b}, the marginal contribution is just *v({a})*.
2) Player a is the second individual entering the game with possible orders {b, a, c} and {c, a, b}, so *MC = v ({b, a}) – v({b}) & v ({c, a}) – v({c})* respectively.
3) Player a is the last one entering the game with possible orders {b, c, a} and {c, b, a}, where *MC = v ({b, c, a}) – v ({b, c}) since v ({b, c, a}) = v({c, b ,a}).*

To get the Shapley Value for player a, we simply take the average of the player's marginal contribution in all cases. Merrick and Taly (2020) point out that this concept is applied to machine learning model interpretation by setting up a cooperative game where the players are the input features and payoff is the model prediction. The Shapley value of each feature quantifies how much individual features contribute to the model's performance on a set of data points. Suppose there are M features to predict the outcome, the predicted value, $y = f(x)$ on the input value $x = (x_1, x_2, x_3, \ldots, x_M)$ can be decomposed as $f(x) = \varphi_0 + \sum_{j=1}^{M} \varphi_j$, where φ_0 is the average of the predicted values data points, φ_j is the amount of contribution, or Shapley Value of the *j*th input feature.

3. Modelling and Forecasting

3.1 Data Preparation

We begin with creating a new database that will be applied to the two machine learning models introduced in the last section. First, to ensure that the match data reflect the recent level of competition between teams, the abridged FIFA match results are obtained from the dataset *International Football Results from 1872 to 2022* by filtering the dates from the 2018 World Cup to the last games before the 2022 World Cup. Further, to restrict home advantage, which is the benefit that the home team is said to gain over the visiting team, the paper chooses to not only predict the match results based on the original home and away information, but also the results after switching the away and home information of each match teams (because there is no home advantage in World Cup). The ultimate winning probability of each team is computed as the mean of the two predictions with the opposite home and away information. Finally, the merge is made to obtain a complete dataset including both abridged FIFA game results and teams' rankings.

3.2 Feature Engineering

The core idea is to create as many features as possible (other than features already contained in processed dataset) that can exert influence on the results of football games prediction by manipulating on the original features in the dataset.[1] For instance, teams might be unwilling to play to their full competitive strengths in a match if the match is not considered significant, such as a friendly match. For this reason, we create a new column to quantify the importance of a game: *is_friendly*. It is also supposed that FIFA rank points and FIFA ranking of the same team are negative correlated. In this case, only one of them should be used to create new features. This supposition is checked in Figure 1:

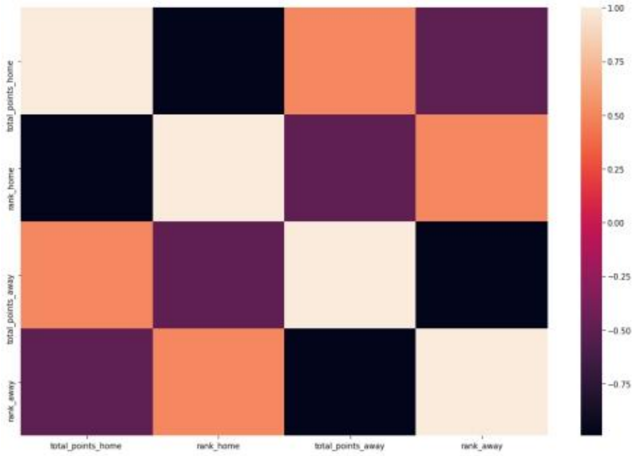


Fig 1. Correlational Matrix for FIFA rank points and FIFA ranking

After manipulation, the author chooses only the features that are conducive to Features' analysis and carries on to the next part.

3.3 Data Anlaysis

In this section, all the created features are analyzed to checked if they have any predictive power. Also, if they lack such power, it is essential to create with that power, like the differences of home and away teams. To analyze the predictive power, the author considers the games that end up with a draw as a loss of the home team and thus defines match results as the target with 1 as win and 0 as lose / draw, which creates a binary problem. An analysis will be conducted:

(i) Applying violin plot and boxplot to analyze if the features have different distributions according to the target (i.e. the match results)

(ii) Applying scatter plots to analyze correlation

[1] All features that are not differences should be created for both teams (away and home). The games with NA are the ones who mean could not be calculated (games from the beginning of the dataset). Those will be dropped.

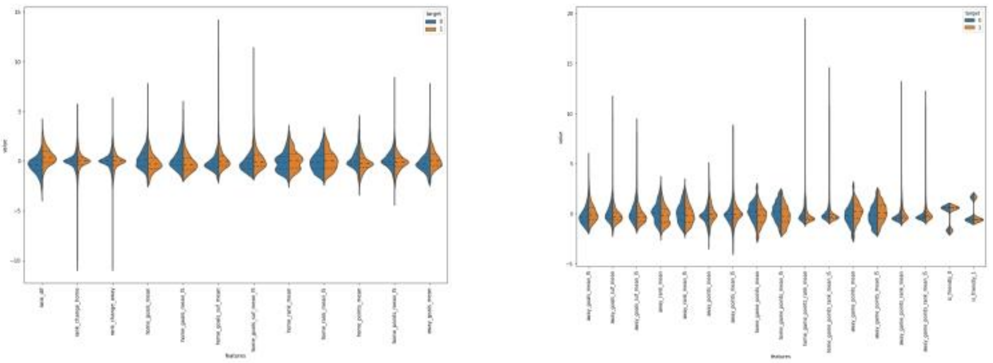


Fig 2. Violin Plot for engineered features

Figure 2 displays two violin plots, revealing that the difference of rank is the only effective separator of the data. We, however, can create other features used to analyze whether the differences between home and away team are effective in separating the data. Besides, it is clear to see that, in Figure 3, goal differences and goals suffered difference are indeed effective separator, but the differences between the goals scored and the goals conceded by the teams are not very effective separators.

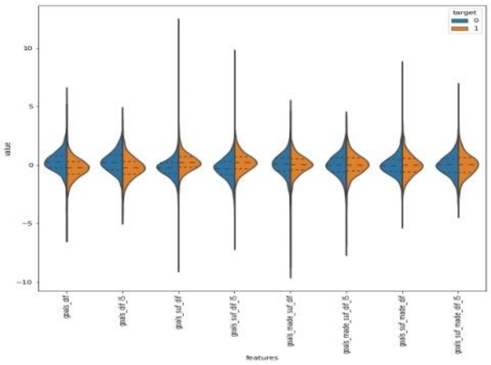


Fig 3. Violin Plot for goals/goals suffered difference

Now, there are 5 features which pass the test: rank_dif, goals_dif, goals_dif_l5, goals_suf_dif, and goals_suf_dif_l5. Also. We can still create other features, i.e the differences of points attained, the differences of points brought by rank faced and the differences of rank faced. Due to the low values, however, the violin plot is not a good choice to analyze if the features are effective in separating the data in this case. Thus, these features are displayed in the boxplot instead:

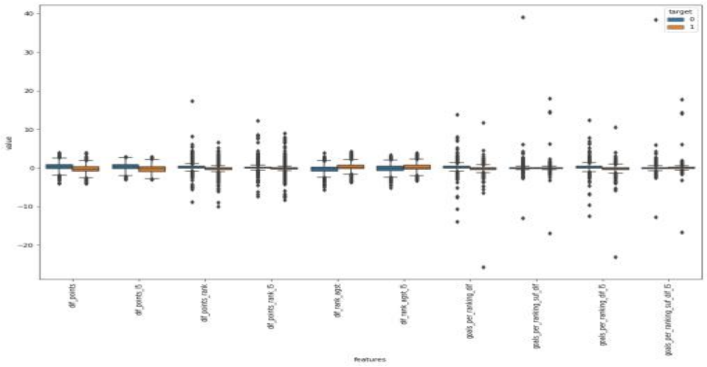


Fig.4 Boxplot for a new series of difference features

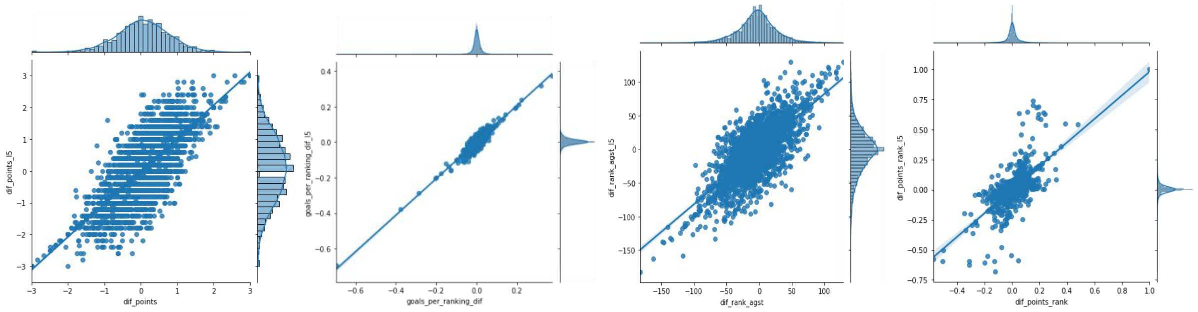


Fig.5 Scatterplot for a series of difference features

The difference of points (full and only last 5 matches), the difference of points by ranking faced (full and last 5 matches) and the difference of rank faced (full and last 5 matches) are all effective features (see Figure 4). Also, some of the generated features show highly similar distributions which will be analyzed using scatterplots (see Figure 5). Then, it turns out that for the differences of rank faced, the game points by rank faced and mean game points by rank faced, the two versions of feature distribution (full games and only last 5 matches) are dissimilar. Hence, both will be used. On this basis, the final features are:

(1) rank_dif; (2) goals_dif; (3) goals_dif_l5; (4) goals_suf_dif; (5) goals_suf_dif_l5; (6) dif_rank_agst; (7) dif_rank_agst_l5; (8) goals_per_ranking_dif; (9) dif_points_rank; (10) is_friendly

3.4 Modelling and Optimization

Now, we have prepared a database of features with predictive power, which is sufficient for the modeling. The two different algorithms introduced in Sections 2.2.1 and 2.2.2, *Random Forest* and *Gradient Boosting*, are now compared in terms of their predictive performance after being optimized by *GridSearchCV*. Finally, the one with better performance will be selected and used for the 2022 World Cup simulation. For this purpose, the following general process is conducted on the feature-engineered data:

1) Split the dataset into training set (matches from 2018-2021) and testing set (matches of 2022)
2) Fit each of the methods to the training data.
3) Use GridSearchCV to search through the best hyperparameter values
4) Make predictions on 2022 testing data
5) Find best-performing model by confusion matrix and ROC plots

Confusion matrix and ROC plots in Figure 6 present the results of those performance measures. With an AUC score of 0.75 on test while 0.96 on training, the random forest is apparently overfitting. Thus, we choose Gradient Boosting in the following World Cup 2022 simulation.

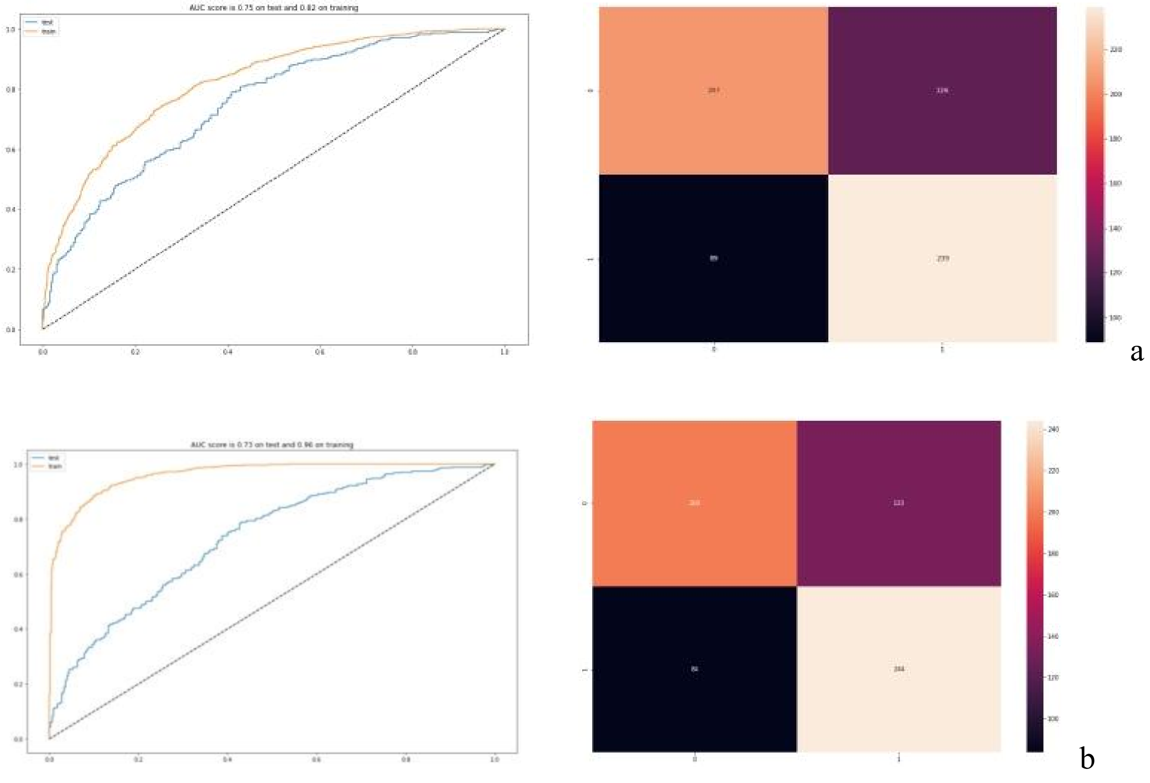


Fig.6 Confusion Matrix and ROC curves for random forest and gradient boosting
(a) Gradient Boosting (b) Random Forest

3.5 World Cup Simulation

To start with, we collected the World Cup grouping information from Wikipedia and then created 48 matches against each other based on this information. By using the gradient boosting model selected in sections 3.4, the victory of different matches was predicted. Meanwhile, the prediction was made multiple times. The model was applied after eliminating the impact caused by the home advantage via switching the home team and the away team, with the winning probability of each team calculated as the average of multiple prediction results. Then, the qualified teams were taken to predict the last 16, the last 8 and eventually, the champion of the tournament. Figure 7 presents the predicted World Cup journey of each team with corresponding winning probabilities.

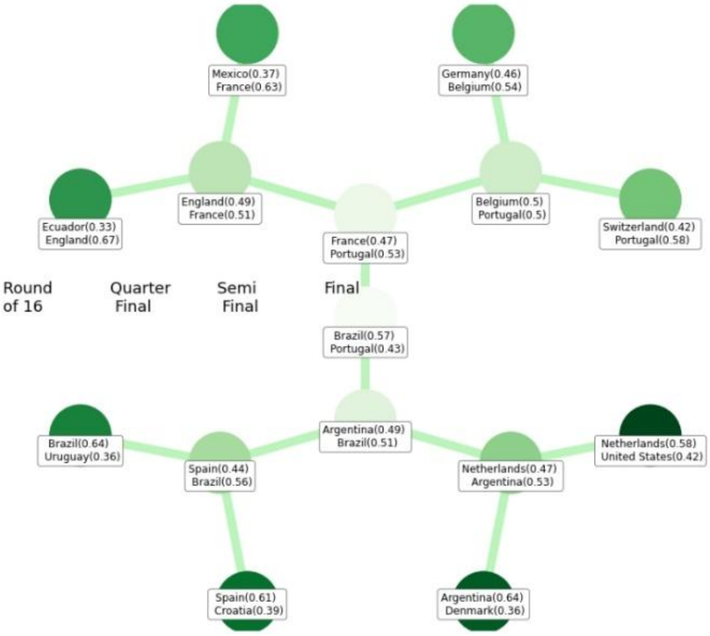
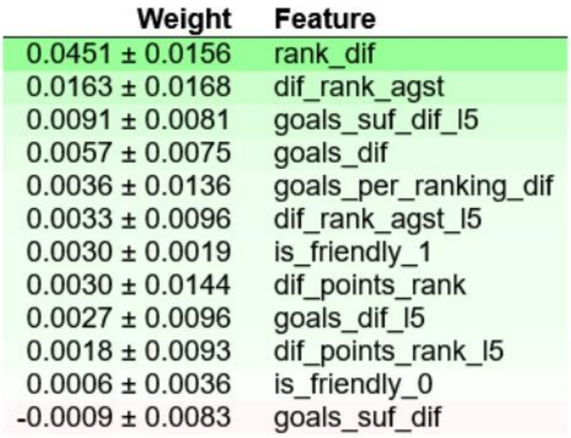


Fig.7 Predicted journey of WC knockout stages

4. Explainability

4.1 Global Explainability

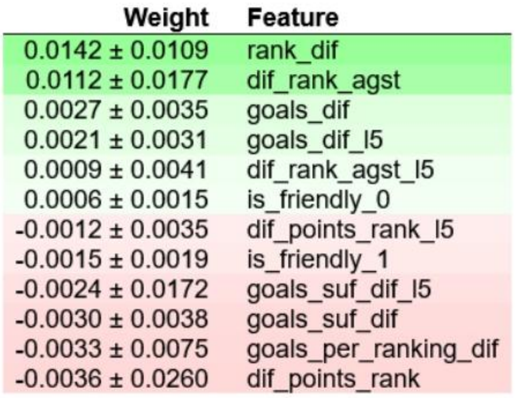
As for the interpretation of a model, it is usually recommended to start with the importance of features. Based on permutation importance, we analyze which features are most important and make the most significant difference to the outcome of the model decisions (i.e. the prediction results of all 2022 matches).



Weight	Feature
0.0451 ± 0.0156	rank_dif
0.0163 ± 0.0168	dif_rank_agst
0.0091 ± 0.0081	goals_suf_dif_l5
0.0057 ± 0.0075	goals_dif
0.0036 ± 0.0136	goals_per_ranking_dif
0.0033 ± 0.0096	dif_rank_agst_l5
0.0030 ± 0.0019	is_friendly_1
0.0030 ± 0.0144	dif_points_rank
0.0027 ± 0.0096	goals_dif_l5
0.0018 ± 0.0093	dif_points_rank_l5
0.0006 ± 0.0036	is_friendly_0
-0.0009 ± 0.0083	goals_suf_dif

Fig.8 Permutation Importance of Random Forest

The figure 8 illustrates the feature importance of the random forest model. As shown in the figure, the most significant feature contributing to the winning of matches is rank_dif, followed by dif_rank_agst_l5. The first one represents the difference of ranking between the home and away teams.



Weight	Feature
0.0142 ± 0.0109	rank_dif
0.0112 ± 0.0177	dif_rank_agst
0.0027 ± 0.0035	goals_dif
0.0021 ± 0.0031	goals_dif_l5
0.0009 ± 0.0041	dif_rank_agst_l5
0.0006 ± 0.0015	is_friendly_0
-0.0012 ± 0.0035	dif_points_rank_l5
-0.0015 ± 0.0019	is_friendly_1
-0.0024 ± 0.0172	goals_suf_dif_l5
-0.0030 ± 0.0038	goals_suf_dif
-0.0033 ± 0.0075	goals_per_ranking_dif
-0.0036 ± 0.0260	dif_points_rank

Fig.9 Permutation Importance of Gradient Boosting

The figure 9 displays the feature importance ranking of the Gradient Boosting model. The most important feature is *rank_dif*, and the second most important feature is *dif_rank_agst*. Apparently, Gradient Boosting and Random Forest are generally consistent in terms of general feature importance. However, gradient boosting considers the overall goals difference as a more important predictor than the goals suffered difference whereas the random forest ranks them oppositely.

4.2 Local Explainability

After obtaining the importance of the features, Partial Dependence Plot is used in this paper to analyze how the decision made by the model is specifically influenced by each feature.

In Figure 10, the four most important features obtained in sections 4.1 are used to analyze the impact on the prediction results. Among them, the impact of difference between home and away rankings reach the most significant extent, with a dramatic increasing trend maintained. According to the random forest model, it is more likely to for the team with a greater difference between home and away rankings to win. The influence of other features, showing either a mild growth or a declining trend, is rather insignificant to the model.

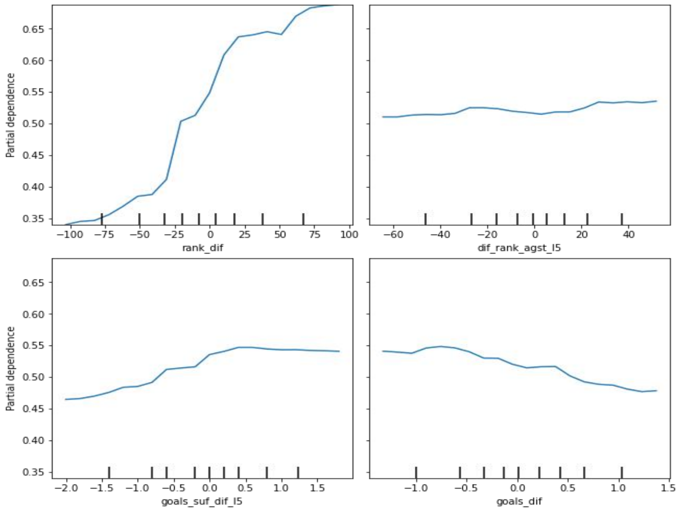


Fig.10 PDP of Random Forest

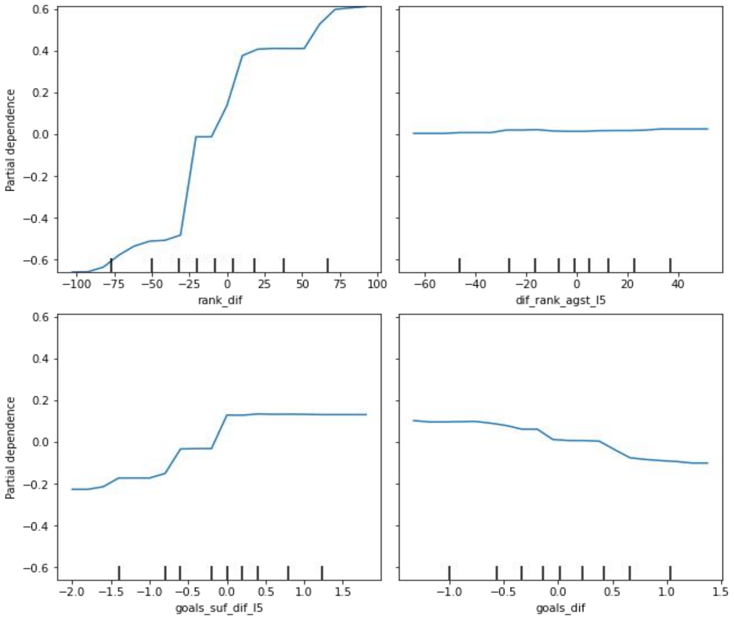


Fig.11 PDP of Gradient Boosting

It can be seen from Figure 11 that the two models, Gradient Boosting and Random Forest, show the same trend in general in the influence of these four features on the prediction results. One piece of matching data is then randomly selected, and Shapley value is used to determine how each feature jointly contributes to the prediction result of this match in two models respectively.

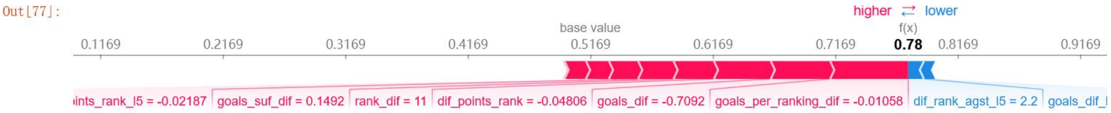

Out[77]:

Fig.12 Shapley Explainer for Random Forest

In the figure 12, the red part indicates the positive effect while the blue part is the negative effect, with a base value of 0.5169. Features *dif_points_rank, goals_dif*, and *goals_per_ranking_dif* provide positive effects, while *dif_rank_agst_l5* and *goals_dif_l5* provide negative effects, contributing together to the final value of 0.78.

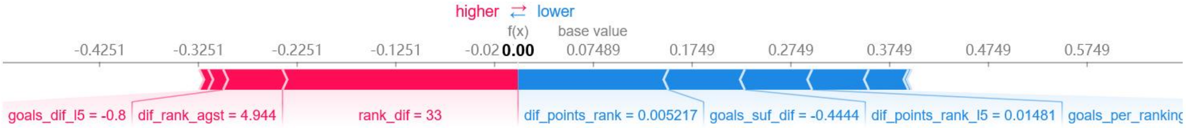


Fig.13 Shapley Explainer for Gradient Boosting

In the figure 13, the red part indicates the positive effect, and the blue part indicates the negative effect. In the Gradient Boosting model, the base value is 0.07489, and the final output value of the different features combined is 0. It can be found that the number of positive influences generated by *rank_dif* feature is large, indicating that this feature is considered by this model to be the most important part of all features. It also validates our conclusion made in Section 4.1.

5. Conclusion

In this paper, we first employ random forest and gradient boosting to model the classification problem which predicts whether each team would win or lose the game. The grid search cross validation is also implemented to achieve model optimization. The gradient boosting model outperforms random forest based on the model evaluation measures, confusion matrix and AUC score. Therefore, we use this model to simulate World Cup 2022 game and make predictions for the winning probabilities of each team.

The study then explores the interpretation of the models using 3 approaches: Permutation Importance, Partial Density Plot and Shapley Value. Ranking *rank_dif* and *dif_rank_agst* as top 2 important features, both models are generally consistent in terms of determining predictor importance. Based on the results from permutation importance, we draw the partial density plots for the most significant four features, which displays that the feature *rank_dif* has the largest impact on the prediction with a significant growing trend. Lastly, we compute the Shapley Values, or the average marginal contributions of these features, and compare whether they have positive or negative power on the predictions.

References

Benchekroun, H. (2022, November 24). *A cooperative approach to public good games* [PowerPoint]. https://mycourses2.mcgill.ca/d2l/le/lessons/595988/topics/6735336

Breiman, L. (2001). Random Forests. *Machine Learning 45*, 5-32. https://doi.org/10.1023/A:1010933404324

Carrington, A.M. et al. (2021). Deep ROC Analysis and AUC as Balanced Average Accuracy to Improve Model Selection, Understanding and Interpretation. *IEEE Transactions on Pattern Analysis and Machine Intelligence 2022.* https://doi.org/10.48550/arXiv.2103.11357

Carvalho, D.V., Pereira, E.M. & Cardoso, J.S. (2019). Machine Learning Interpretability: A Survey on Methods and Metrics. *Electronics,* 8(8), 832. https://doi.org/10.3390/electronics8080832 doi:10.1061/(ASCE)IS.1943-555X.0000602.

Dyte, D. & Clarke, S.R. (2000). A ratings-based Poisson model for World Cup soccer simulation. *Journal of the Operational Research Society,* 51(8), 993-998. https://doi.org/10.1057/palgrave.jors.2600997

Fawagreh, K., Gaber, M.M. & Elyan, E. (2014). Random forests: from early developments to recent advancements. *Systems Science & Control Engineering*, 2(1), 602-609. DOI: 10.1080/21642583.2014.956265

Friedman, J. H. (2001). Greedy Function Approximation: A Gradient Boosting Machine. T*he Annals of Statistics*, 29(5), 1189–1232. http://www.jstor.org/stable/2699986

Greenwell, B. M. (2017). pdp: An R package for constructing partial dependence plots. *R J.,* 9(1), 421.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). Boosting and additive trees. *In The elements of statistical learning (*pp. 337-387*). Springer*

Huang, N., Lu, G. & Xu, D. (2016). A Permutation Importance-Based Feature Selection Method for Short-Term Electricity Load Forecasting Using Random Forest. *Energies,* 9(10), 767. https://doi.org/10.3390/en9100767

James, G., Witten, D., Hastie, T., Tibshirani, R. (2013). *An Introduction to Statistical Learning: with Applications in R* (Second Edition). Springer Texts in Statistics.

Kaggle. (2022). *FIFA World Ranking 1992-2022* [Data set]. https://www.kaggle.com/datasets/cashncarry/fifaworldranking

Kaggle. (2022). *International football results from 1872 to 2022* [Data set]. https://www.kaggle.com/datasets/martj42/international-football-results-from-1872-to-2017

Karlis, D. & Ntzoufras, I. (2003). Analysis of sports data by using bivariate Poisson models. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 52(3), 381-393. https://doi.org/10.1111/1467-9884.00366

Kulkarni1, A., Chong, D. & Batarseh, F.A. (2020). Foundations of data imbalance and solutions for a data democracy. *Data Democracy*. 83-106, https://doi.org/10.1016/B978-0-12- 818366-3.00005-8

Leitner, C., Zeileis, A. & Hornik K. (2010). Forecasting sports tournaments by ratings of (prob)abilities: A comparison for the EURO 2008. *International Journal of Forecasting,* 471-81. https://doi.org/10.1016/j.ijforecast.2009.10.001

Liu, X., Liu, T.Q. & Feng P., (2022). Long-term performance prediction framework based on XGBoost decision tree for pultruded FRP composites exposed to water, humidity and alkaline solution. *Composite Structures 284.* DOI: 10.1016/j.compstruct.2022.115184

McHale, I.G. & Scarf, P. (2007). Modelling soccer matches using bivariate discrete distributions with general dependence structure. *Statistica Neerlandica,* 61(4), 432-445, http://dx.doi.org/10.1111/j.1467-9574.2007.00368.x

Merrick, L. & Taly, A. (2020). The Explanation Game: Explaining Machine Learning Models Using Shapley Values. *Machine Learning and Knowledge Extraction*, 17-38. https://doi.org/10.1007/978-3-030-57321-8_2

Piryonesi, S. Madeh; El-Diraby, Tamer E. (2021). "Using Machine Learning to Examine Impact of Type of Performance Indicator on Flexible Pavement Deterioration Modeling". *Journal of Infrastructure Systems.* 27 (2)

Probst, P., Wright, M.N. & Boulesteix, A. (2019). Hyperparameters and tuning strategies for random forest. *Data Mining and Knowledge Discovery,* 9(3). https://doi.org/10.1002/widm.1301

Schauberger, G. & Groll, A. (2018). Predicting matches in international football tournaments with random forests. *Statistical Modelling,* 18(5-6), 460-482. doi:10.1177/1471082X18799934

Shapley, L. (1953). A Value for n-Person Games. *Contributions to the Theory of Games II,* 307- 317. https://doi.org/10.1515/9781400881970-018

Yang, L. & Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing 415*, 296-316. https://doi.org/10.1016/j.neucom.2020.07.061

Zeileis, A., Leitner, C., & Hornik, K. (2018). *Probabilistic forecasts for the 2018 FIFA World Cup based on the bookmaker consensus model* (No. 2018-09). working papers in economics and statistics.