# Predicting FIFA International Results

Anna Koumi, Lewis Crowley, Radu Priboi, Alexandru Panin, Boran Shensoy

## 1 Introduction

Predicting the outcome of football matches is a task well suited to machine learning and which benefits a number of stakeholders. To benefit the growth of the sport, it allows fans and enthusiasts to make informed decisions about which teams, players, and fixtures to follow, helping to increase fan engagement and excitement. Beyond this, accurate forecasts can also be utilised by sports betting organisations, broadcasters, and advertisers to make strategic decisions and provide their clients with more enjoyable and lucrative experiences (Horvat and Job, 2020; Baboota and Kaur, 2019). Sports betting forms the most obvious commercial application of this work, with the BBC having estimated that the football betting industry is worth $700B-1$ T (Keogh and Rose, 2013). Additionally, managers can use match performance predictions to alter their strategies and increase their chances of success and, if a team is on the market, these predictions can also inform valuations of a club and influence business decisions.

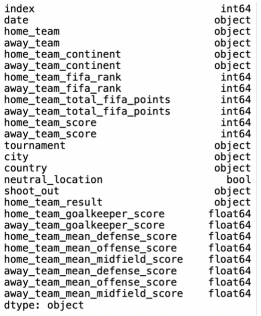
In this paper, we develop a machine learning model to predict the 2022 FIFA games leading up to the 2022 FIFA World Cup. This test dataset contained 247 games, and the outcomes were predicted based on a training data from the past 2 decades of international football (5641 matches). Ultimately, we were able to achieve a model accuracy score of >55%.

## 2 Data transformation and Exploration

In this section, raw data was visualised, creating an overview of the different features before proceeding to any further manipulation. Non-numerical, categorical datatypes were also transformed into numeric datatypes allowing their analysis and incorporation in further steps. Also, any missing values were found and dealt with as appropriate. We only discuss the pre-existing features in the raw data here, with engineered features being discussed in Section 3 – Methodology.

### 2.1 Data Transformation

Data was provided in different forms, some as numeric datatypes and others as objects, as seen in Figure 1. All the categorical features of interest appearing as objects were converted to 'int64' to enable incorporation into our models. For example, features such as 'away team' or 'tournament' were encoded as integers, with one name mapped to one integer, to create usable features. We knew we were already working with cleaned data and thus did not have to worry about whether e.g. the 'away team' data having listed separate instances of 'France' and 'FRA' to mean the same thing. It should be noted that the features discussed here are nonlinear and thus were not applicable when experimenting with linear classification models (e.g. logistic regression).



```
index                              int64
date                              object
home_team                         object
away_team                         object
home_team_continent               object
away_team_continent               object
home_team_fifa_rank                int64
away_team_fifa_rank                int64
home_team_total_fifa_points        int64
away_team_total_fifa_points        int64
home_team_score                    int64
away_team_score                    int64
tournament                        object
city                              object
country                           object
neutral_location                    bool
shoot_out                         object
home_team_result                  object
home_team_goalkeeper_score       float64
away_team_goalkeeper_score       float64
home_team_mean_defense_score     float64
home_team_mean_offense_score     float64
home_team_mean_midfield_score    float64
away_team_mean_defense_score     float64
away_team_mean_offense_score     float64
away_team_mean_midfield_score    float64
dtype: object
```

**Figure 1:** The features and datatypes present in the raw training data.

The other key data transformation we had to undertake was to find any missing data and replace it where appropriate. Pandas *isnull*() allowed us to find and quantify the NaN results quickly. The most relevant NaN results we found were in the team offense / defense / midfield / goalkeeper scores, with 3-7% of the data missing for each of these features. Since a low proportion of the values for this data were missing, we decided to replace these values. An initial approach to this was to replace the value with the mean value for the relevant team when they were playing home / away or, failing this, the mean of all teams playing home / away. We then considered an alternative strategy of instead replacing them with recent averages. However, in doing so, we found that these scores are not updated game to game, and thus replacing them (more simply) with the last known value for the team would be sufficiently accurate. In the few cases where this was not possible (e.g. the same value is missing for all matches played by a given team), a recent value for a similarly ranked team was used instead. A team was deemed to be similarly ranked if its FIFA rank was within 3 of the team in question.

There was also missing data (NaN values) for 'home/away team total Fifa points'. A much larger proportion of this data was missing, with all values before the start of the 2011/12 season missing. We concluded that replacing the data was unreasonable in this instance and would become a source of inaccuracy in our modelling. To make use of some of the categorical data at our disposal such as 'home/away team' (whilst we were still considering using it during feature selection), we partitioned the data, excluding datapoints before 2012. This was because the fact that e.g. the 'home team' being Argentina would have meant a very different thing in 2007 than in 2022 (due to changes in the player roster). Whilst originally seeming like a neat solution the problem of missing data for the 'home/away team total Fifa points', we later found separate issues with using this feature (see Data Exploration). Thus, the data for this feature was omitted entirely from use in our models. N.B. After feature selection, none of the categorical features discussed here were used and so we reverted to using all of the data and not excluding pre-2012 datapoints.

The final transformation of the training data we undertook was partitioning it to exclude all data relating to 'friendly' games. This was based on the knowledge that, during these matches, managers often opt to use experimental tactics and inexperienced players, meaning that the results of these matches would not be representative of most data. Obviously, this would make our model worse at predicting the outcomes of friendly games. However, most matches are not 'friendlies', and friendlies are certainly not the games of the most consequence. As in any sport, the games part of structured competition attract the most fan engagement, have the most bets placed, and are the ones into which players, managers, and team staff invest the most resources.

## 2.2 Data Exploration

Before creating a model for football match outcome predictions, we needed to understand the training data we were provided with and the relationships between features within it. We the unmodified numerical features against each other to determine correlations and relationships between them and with the model target (match outcome). This gave us a general idea of which features had a primary role in determining match outcomes and which correlated features may be targets for dimensionality reduction.

Figure 1 shows example scatter matrices which were created to illustrate the correlations between the different, raw, numerical features. All features were visualised this way in an attempt to decide initial relations and determine the possible factors affecting the outcome of a game. The features investigated in Figure 2 (below) are the home team's FIFA rank, FIFA points, offence and midfield scores and the match result. Some obvious positive/negative correlations are detected amongst these features (e.g. home team offense and midfield scores). Moreover, the expected classification-like plots were achieved for all the features plotted against the 'home team result' with '1' for a win, '0' for a draw and '-1' for a loss. For example, teams with lesser offense scores clearly lose more often. This simple analysis was used as a basis for further feature engineering and data selection, as will be explained in section 3.

A key outcome of the data exploration achieved in Figure 2 was that the 'home/away team total FIFA points' feature was deemed unusable, even when excluding the missing datapoints occurring before 2012. Figure 2A (which plots training datapoints between 2012 and 2021) shows two strands/clusters of data in the subsections relating to total FIFA points. Investigation into this revealed that this was because the methodology behind calculating this score was altered in 2018 (FIFA, 2021). Figure 2B shows only the datapoints after 2018 and, testament to this, the two stranded subplots are no longer visible. Gaining this understanding ultimately informed our decision to omit total

FIFA points from our modelling entirely. Although the changes referenced also affected how FIFA rank was calculated (since it is determined directly based on FIFA points) the similarities between Figure 2A and 2B show that the relationships between rank and other variables were largely resilient to these changes. The nature of the rank, only showing that teams have greater/lesser points than each other and not how far apart they are, meant that it the effects of the scoring metric change were dramatically reduced with respect to the rankings, and we concluded that they were minimal enough for the rank to be included as a feature in our model.

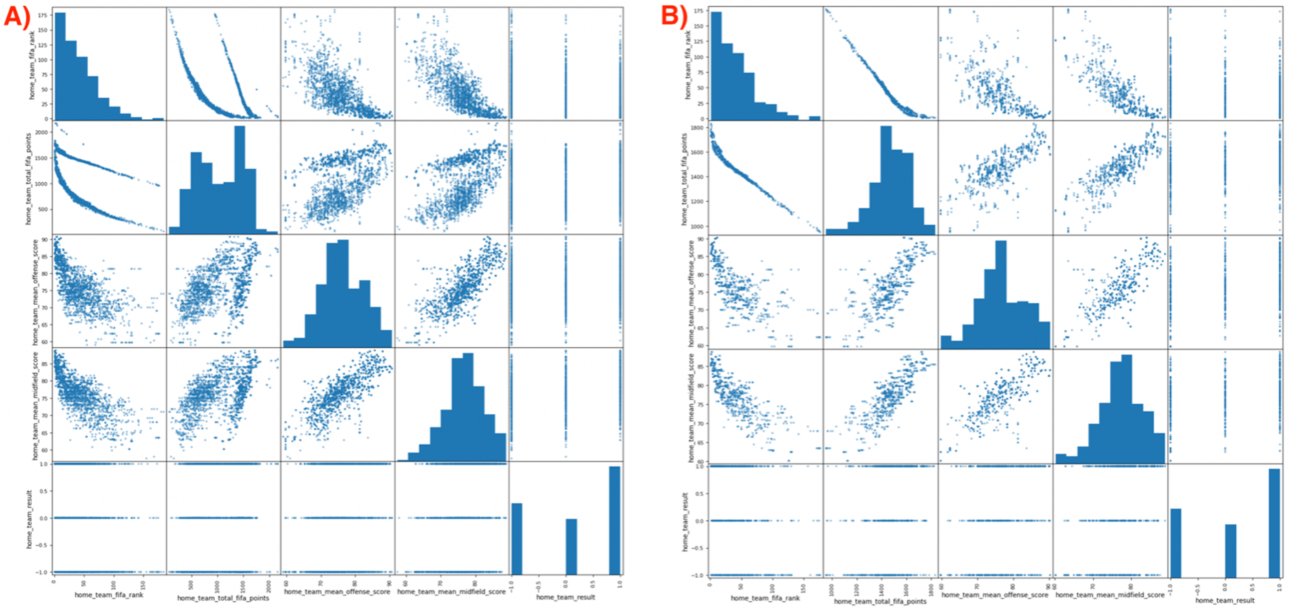


**Figure 2:** Scatter plots used to determine the relationship between the different features. (A) Shows data from 2012-2021 whilst (B) shows data from 2018-2021.

One further notable outcome of data exploration was that teams from either Europe or South America are more likely to win matches, as is shown in Figure 3. We had hypothesised that this was the case given our preexisting knowledge of international football. Given that the data confirmed this hypothesis, we opted to later engineer a feature to represent whether one team in a match was from either of these continents.

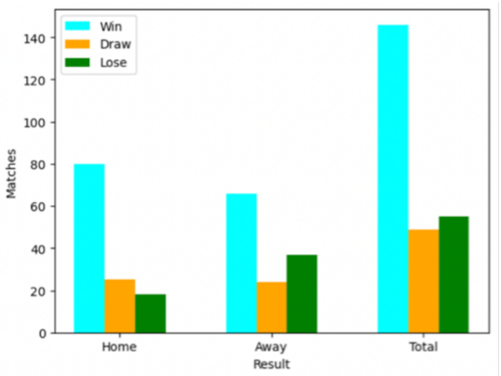


**Figure 3:** European and South American team matches were measured against opponents from different continents. The bar chart above shows that most of the matches involving a team from either of the two continents are more likely to win.

# 3 Methodology

This section details the approaches we took towards engineering features, feature selection, model training, model evaluation, and hyperparameter tuning.

## 3.1 Background Reading

In order to engineer new features that are to be implemented and to better understand approaches which had been successfully applied to similar problems, we researched past machine learning algorithms that predict football matches alongside possible variables that might influence the result of the games.

Football predictions using machine learning is a field which has gained a lot of traction in the past 10 - 15 years. It was used successfully by German national team for scouting and studying opposing teams' playstyles and players, based on the results of the matches (the reverse principle of what we are trying to use). It was also used in different European leagues with a consistent accuracy rate of above 50%. In the Spanish Premier Division, a multi-agent system was adopted, with an accuracy rate of 61%. However, some outliers have indeed appeared, due to the unpredictability of the game, such as the 2015/16 English Premier League season which was won by the underdogs Leicester City. Still, using a regression model, in that season of the English topflight, the accuracy registered was around 69.5%, with the most important variable that influenced the game being the home/away advantage (Rodrigues and Pinto, 2022). In Rahman, 2020, "A deep learning framework for football match prediction", the algorithm uses multiple factors such as team dynamics, player's stats, number of cards and so on. However, the author observes that, once again, the most important feature that determines the playstyle and the result of the game is the home/away advantage. (Rahman, 2020)

Regarding potential features that might be engineered, we observed that psychology such as morale plays a relevant part in the result. Both Lea Boecker, 2021, and Sigmundsson et. al, 2022, mention how the underdog is more likely to be affected by morale issues, especially when playing away. Besides, domination of a team over another (when the odds are completely stacked against the underdog) has an even bigger impact on the performance of the latter (Boecker, 2021), (Sigmundsson et al., 2022).

Another relevant feature is the weather conditions. According to Brocherie et. al, 2015, analyzing international games played by national teams from the Gulf Region, home teams experience an advantage of 3% for every 1- unit increase in temperature difference compared to the away team. It was also observed that humidity plays a role in the outcome of the game, as the Gulf teams playing in highly humid environments performed below the expected average (Brocherie et al., 2015).

## 3.2 Overview of Methodology

After feature engineering (which is discussed independently below) was complete, we decided we could take a brute force approach to feature selection, model selection and validation whilst still accomplishing this in a considered and robust manner. Due to the limited size of our dataset and the number of features present, we knew we could trial numerous classification models and combinations of features with the time and computational resources available to us. As such, we trialed feature combinations across eight commonly used models within the 'Scikit-Learn' library, validating with a manually defined train-test split as well as cross validation. The classification models used were a combination of linear and nonlinear models and, for models of interest, we undertook hyperparameter tuning to ensure optimal outputs. The classification models we used had to be capable of multiclass classification, since our brief was to predict wins, draws and losses.

The eight 'Scikit-Learn' classification models trialed were: Logistic Regression, K-Nearest Neighbours, Adaboost, Quadratic Discriminant Analysis, Decision Tree, Random Forest, Multi-Layer Perceptron, Support Vector Classifier.

Initially, we thought that categorical features such as the teams playing, the competition which the match was affiliated with, and the continents which the teams were from would be of particular importance. Not only are these key things which we ourselves would use to predict the outcome of a match, but these variables would be known ahead of time and thus relevant in predicting future matches. As such, we started by trialling nonlinear classification (e.g. using the Random Forest Classifier) and partitioned our training data so that it only included matches from 2012 onwards. This was because the fact that e.g. the 'home team' being Argentina would have meant a very different thing in 2007 than in 2022 due to changes in the player roster. 2012 was selected as the cutoff since a current player could have foreseeably been playing for their national team, and impacting results, a decade ago. However, not only did this approach limit the amount of training data at our disposal, but the categorical features in question were also determined to be poor predictors. Thus, we reverted to using the entire set of training data at our disposal (bar any friendly games) and incorporated linear classifiers as candidate models.

### 3.3 Feature Engineering

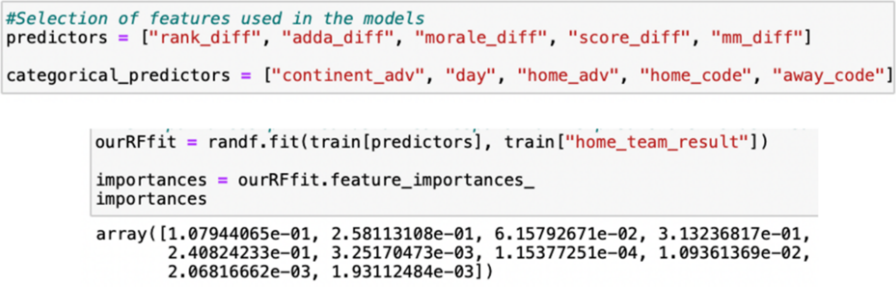
| Feature | Description and Justification | Used in Final Model? |
|---|---|---|
| Mean Score Differences (e.g. 'ad_diff') | These features were the differences between the mean attack / defense / midfield / goalkeeper scores between the home and away teams in the raw data. That is, in the case of 'ad_diff', the difference between the attack score of the home team and defense score of the away team (= home attack score – away defense score). We trialed multiple difference metrics and settled with home attack – away defense (ad_diff),<br>away attack – home defense (da_diff), and home midfield – away midfield (mm_diff). Differences incorporating the goalkeeper scores were selected against. Differences between differences were also experimented with, most notably in 'adda_diff' which was 'ad_diff' – 'da_diff'. We engineered these features for multiple reasons:<br>1. The differences between the teams playing seemed more relevant than their individual quality<br>2. The disparity between e.g. the quality of an attack and a defense would be a strong predictor of goals scored (and thus match outcome)<br>3. This was a straightforward way to reduce dimensionality whilst preserving / improving the richness of the data. | Yes ('ad_diff', 'da_diff', 'adda_diff') |
| Rank Difference | Home team rank – away team rank. Reasoning for taking the difference is similar to above. | Yes |
| Day / Month | Extracted (encoded as integer) from date feature under the hypothesis that teams play differently on different days or in different months. | No |
| Morale and Morale Difference | We wanted to incorporate the recent form of a team. It is well known that recent form is an important predictor of how well a team will play, and clearly recent results are more relevant to a team's future performance and will give them a morale boost. However, we obviously don't know the results of future games, so the approach we used to confirm that this was a relevant feature, using recent 'home team results' in the training data, was unapplicable to the test data. To circumvent this, we decided to sum the rank difference from the three most recent games a team has played. We knew that rank difference was a good predictor of match outcome and so having a summed rank difference for a team's past three games would be a good proxy for saying how many they had won. Again, the difference between the morale values for the teams playing was ultimately used for the reasons described above for other features. | Yes (Morale Difference) |
| Team Codes | The home/away team name, encoded as an integer. Engineered with the hypothesis that teams retain quality over extended periods and also that this might reflect how many managerial / administrative / support staff and training facilities would remain constant over extended periods. | No |
| Tournament | The competition type (e.g. world cup) encoded as an integer, used under the hypothesis that teams play differently in competitions which matter more/less to them. | \No |
| Score and Score Difference (via PCA) | With the 'home/away team total FIFA points' being unusable for reasons described previously (see Data Transformation and Exploration), we wanted a continuous metric to describe the overall quality of a team. Whilst the rank was a good predictor, it lacked a richness of description in that the difference in quality between e.g. ranks 1 and 2 could have been much greater than that between 2 and 3. We used PCA to reduce the home or away team attack/defense/midfield/goalkeeper scores down to a single metric which was representative of team quality. This was done under the hypothesis that, ultimately, the investment of a country in the sport and its culture would determine the quality of players it produced, regardless of position, and was thus a relevant latent variable. Data exploration had illumed the positional scores to be highly correlated, which supported this hypothesis and the approach we ultimately took with this dimensionality reduction. Again, the difference between the morale values for the teams playing was ultimately used for the reasons described above for other features. | Yes (Score Difference) |
| Continent Difference | Data exploration illumed how teams from South America and Europe won more. We turned whether a country was from one of these continents or not into a binary variable. From there, we then created a further variable to indicate whether there was a difference between the two teams playing in this respect (i.e. whether one team was from these either of these two continents whilst the other was not). | Yes |

The features we engineered and trialed, the reasons supporting our decision to engineer them, and whether they were incorporated into our final model or not can be seen summarised in Table 1 above.

There are many other engineered features which were conceived during feature engineering. For example, after our background research (see above), we were keen to create a feature which specified whether the locations teams were playing in climates different to that of their own countries, quantifying differences in temperature and humidity. However, this would have required more data to be obtained and was ultimately beyond the scope of the project.

### 3.4 Feature Selection

Much of feature selection was done with our primary validation train-test split (see Validation below). We chose to do this for several reasons. Firstly, running the models on a simple train-test was faster than using 10-fold cross-validation, which took some time to compute when using many features with e.g. a multilayer perceptron model operating at a high number of iterations. Also, confusion matrices could be more simply created when running models once on a single train-test split, allowing us to see how the introduction of different features affected the bias of the model towards predicting a win / loss / draw. However, perhaps the most important reason for using a simple train-test split was to make efficient use of the 'feature_importances_' attribute of the Scikit Learn random forest classifier (Scikit Learn, 2023). This provided a simple metric to tell us which features were the most influential in our classification task. Figure 4 shows an experimental feature set being used and the outputs when the 'feature_importances_' attribute was utilised. Higher importance scores are assigned to the better predictors and in this case it is clear that the categorical predictors used were relatively uninfluential. As the most influential categorical feature we found, in this example and others, continent advantage was used in the final model. Cross-validation results (see Validation) were also used to inform feature selection.



```
#Selection of features used in the models
predictors = ["rank_diff", "adda_diff", "morale_diff", "score_diff", "mm_diff"]

categorical_predictors = ["continent_adv", "day", "home_adv", "home_code", "away_code"]
```

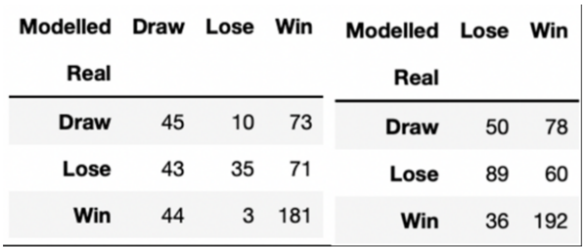
```
ourRFfit = randf.fit(train[predictors], train["home_team_result"])

importances = ourRFfit.feature_importances_
importances
```

```
array([1.07944065e-01, 2.58113108e-01, 6.15792671e-02, 3.13236817e-01,
       2.40824233e-01, 3.25170473e-03, 1.15377251e-04, 1.09361369e-02,
       2.06816662e-03, 1.93112484e-03])
```

**Figure 4:** An experimental set of predictors being defined as inputs to the random forest classifier (above) and the 'feature_importances_' attribute being used to generate importance scores (below). Importance scores are given in an order corresponding to that in which the features are listed.

As aforementioned, due to the limited size of the dataset and number of features we had to select between, a 'brute force' approach trialling many different combinations of features could be used. An interesting observation during this work was that, with experimental feature sets containing many of the less 'important' categorical features, many models were more likely to predict draws (albeit with a lower overall accuracy). As an example, Figure 5 shows the confusion matrix of a multilayer perceptron model with one such feature set alongside the corresponding matrix using our final feature set (which only used the five features which were determined to be the best predictors. It's possible that some of the less 'important' features contained the information necessary to make the more nuanced prediction of a draw. However, without having undertaken further investigation, we are inclined to suggest much of this effect can be attributed to them introducing noise into the model.



| Modelled | Draw | Lose | Win |
|---|---|---|---|
| Real | | | |
| Draw | 45 | 10 | 73 |
| Lose | 43 | 35 | 71 |
| Win | 44 | 3 | 181 |

| Modelled | Lose | Win |
|---|---|---|
| Real | | |
| Draw | 50 | 78 |
| Lose | 89 | 60 |
| Win | 36 | 192 |

**Figure 5:** Confusion matrices generated by a multilayer perceptron classification model using (left) an experimental set of predictors being defined, including many categorical features which were ultimately omitted from our model and (right) the final five predictors which were ultimately selected.

It's worth noting that, despite our use of principal component analysis to reduce dimensionality and combine the four attack/defence/midfield/goalkeeper scores into a single feature, we did still utilise the individual component scores through the 'adda_diff' feature (see Feature Engineering). Although the dimensionality reduction resulted in a strong predictor representing the overall quality of a team, it couldn't represent the more nuanced differences between the attacks and defences of the home and away sides, which were found to be important predictors themselves.

Several of the notable features present in the raw data, such as 'neutral_location' or 'shoot_out' were selected against, ultimately performing poorly as predictors (in terms of ‘importance scores’) and decreasing overall model accuracy. The neutral location feature did not have a significant impact on the bias towards home team wins shown throughout the project in the presence of other features which were more strongly related to match outcome.

Ultimately, the five features selected for use were rank difference, score difference, ‘adda_diff’, morale difference, and continent advantage.

### 3.5 Model Training and Evaluation

We employed a standard workflow to train our models. Firstly, we defined the predictors (features which we wished to use) and then regularised data relating to them using Sickit Learn’s ‘standard scaler’. Regularisation is required for some of our models (e.g. support vector classifiers) to be used and does not affect the others functionally. Non-categorical features were the only features regularized, since the regularization of categorical features is problematic. Though, this was only truly relevant during feature selection when we were experimenting with using multiple categorical features. When using the test set, the standard scaler was fitted to our training data and then used on the test data.

After regularization, a train-test split was defined (when relevant) and classification models were defined. Where relevant, hyperparameters were included in the model definition and a ‘random state’ was set to ensure consistent and comparable results between runs of models incorporating randomness. Once this was complete, models were then trained on the training data and used to predict results for the test set.

Besides the use of confusion matrices, our models were evaluated using ‘accuracy scores’ to indicate the proportion of the predictions they made which were correct. There are several methods of evaluating predictive machine learning models. These can include accuracy, precision, recall and F1-scores. Yet, for the purposes of our models the accuracy was identified as a sufficiently reliable scoring mechanism aloe. This was due to the fact that a large balanced dataset with a lot of attributes was used.

### 3.6 Hyperparameter Tuning

To tune hyperparameters, we iteratively trained and tested models of interest whilst varying individual hyperparameters. To train and test the models, we used a train-test split and produced mean accuracy scores on both training and validation sets. The results were then plotted to determine the optimal hyperparameter values. We plotted both train and test scores together to understand whether any over- or under-fitting was occurring. Varying hyperparameters may result in a better fit to the training data due to overfitting, producing a misleadingly high accuracy score, which we wanted to avoid. An example plot can be seen in Figure 6 below, where the optimal number of ‘nearest neighbours’ for the K-Nearest Neighbours model is determined to be 1000.

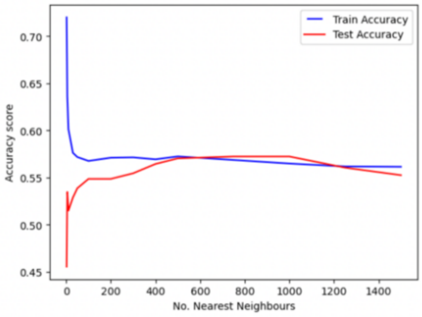


**Figure 6:** A plot used for the hyperparameter tuning of the K-Nearest Neighbours model wherein the ‘number of nearest neighbours’ hyperparameter is varied and accuracy scores for each value are plotted.

Tuning hyperparameters for the multilayer perceptron model was a more nuanced and complex task than for other models. For this, we used code from Panjeh (2020) which made use of the ‘GridSearchCV’ method.

## 4 Validation

### 4.1 Train-Test Split Validation

Train-test splits were not our primary approach for validation, being more useful in feature selection and hyperparameter tuning as previously discussed (see Feature Selection). However, the results of our train-test split modelling were not ignored, and the confusion matrices generated (which were not easily produced in cross-validation) were particularly useful in determining model bias and seeing which predictions (of wins / losses / draws) were correct. Our train test split involved defining the training set as all matches played before 2020 and the validation set as the matches played afterwards. This might seem an unreasonable partitioning (the earliest data is from 2004), and it is if it was to be the only validation approach used. However, it did allegory our ultimate task of using a large amount of data to make a relatively small number of predictions, all in the future, and we knew that most of our validation would be done using cross validation.

### 4.2 Cross Validation

For cross validation, we used a 10-fold cross validation strategy implemented via Scikit Learn's 'cross_val_score' function. This made ten unique, random test-train splits of our training data, fitting the relevant model and testing it each time to produce arrays of accuracy scores. We took the mean and standard deviation of these scores to then evaluate the performance of the models in this cross validation scenario. The mean accuracy score was the main metric used to assess model efficacy, and the standard deviation showed how much variation there was in performance between different test train splits. If the standard deviation was high, then we knew that the model could less reliably perform to the given accuracy score, and we should be cautious in interpreting the results of the model when used on the test set as an indicator of how generally the model was applicable. Similarly, we would know to be wary of any accuracy scores in testing which deviated significantly from those found in cross-validation.

## 5 Validation Results

| Model | Mean (Acc.) | Standard Deviation |
|---|---|---|
| Random Forest Classifier | 0.568 | 0.020 |
| Decision Tree Classifier | 0.436 | 0.017 |
| AdaBoost Classifier | 0.548 | 0.016 |
| Quadratic Discriminant Analysis | 0.566 | 0.022 |
| MLP Classifier | 0.573 | 0.018 |
| SVC (Kernel) | 0.572 | 0.020 |
| KNN Classifier | 0.570 | 0.020 |
| Logistic Regression | 0.571 | 0.016 |

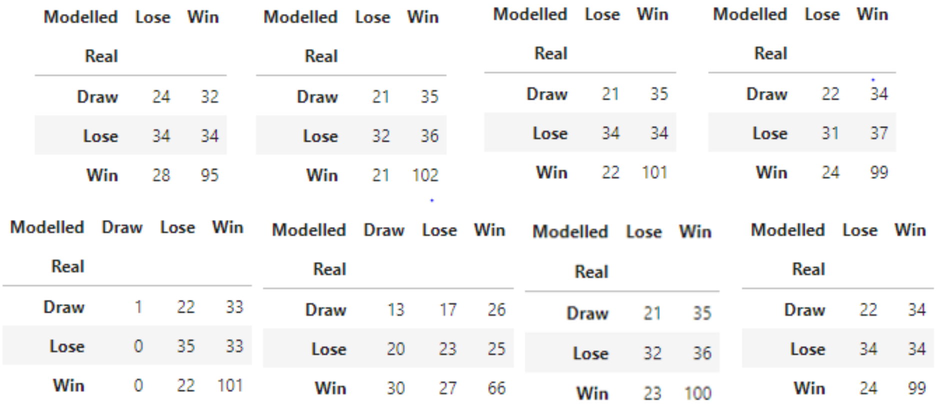
The results from cross validation (shown in Table 2) were very positive, with all models except the decision tree classification producing mean accuracy scores significantly above 50%, indicating their viability for use in predicting match outcomes. In a non-commercial scenario, even the decision tree model may be considered useful as this would likely outperform many individuals attempting to predict match outcomes. The standard deviation of the accuracy scores was low in all cases, giving us confidence in the mean accuracy scores and the future testing scores.

In train-test split validation, confusion matrices showed how these most successful iterations of feature selections and models employed tended to predict almost entirely, if not entirely, wins and losses. We expected this to be the case since the decision to play for a draw is more nuanced tactically, reliant on the present context of where a team stands in each competition or match. For example, if a team is in the group stages of a tournament and would be guaranteed to progress to the knockout stages with the single point afforded by a draw, they may choose to play defensively (for a draw) rather than playing more aggressively (for a win) and risk losing the game to a counter-attack. We did not expect our models to be able to (effectively) predict these outcomes with the data they were trained on. Another, more obvious, explanation is a team of greater/lesser quality than their opponent (which is what most of our selected features ultimately looked to describe) was overall more likely to win/lose than draw in the training data. This doesn't wholly explain why our models didn't predict more draws for relatively even matchups though. A possible explanation is that international football is focused around tournament play, meaning that wins and losses are the most common outcomes of international games (with full-time draws in knockout matches resulting in penalty shootouts to determine a winner and loser).

There is not much to separate the top five performing models in cross validation, with all of them producing mean accuracy scores of 57 ± 0.2%. Given that, among them, logistic regression is a simple, computationally efficient classification model, we might be inclined to suggest that it is the best performing and most suitable model.

## 6 Testing

As explained in Section 3, eight models were applied to the test data, to predict the outcome of each game in the test set. Figure 7 is a summary of the results from each predictor. The accuracy of each can be determined by looking at the main diagonals, which show the number of correct, modelled outcomes. AdaBoost (V) appears to have the best results. With this model, 110 false outcomes were predicted, and 137 correct predictions were made, with 55.47% success rate. Moreover, unlike other more accurate models, it does not predict solely wins and losses (albeit only predicting a single draw, though that was a successful prediction. Models (I), (II), (III), (IV), (VI), (VII), (VIII) have a success rate of 53.44%, 54.25%, 54.66%, 52.63%, 41.30%, 53.44% and 53.85% respectively. As expected, most models do not reach the level of accuracy achieved in validation. However, the drop in performance is only a few percentage points, thus deemed acceptable and not a cause for suspicion. Again, logistic regression is among the best performing classifiers. Since this is a simple, computationally efficient model, it could be considered the best candidate in many scenarios.



| Modelled | Lose | Win |
|---|---|---|
| Real | | |
| Draw | 24 | 32 |
| Lose | 34 | 34 |
| Win | 28 | 95 |

| Modelled | Lose | Win |
|---|---|---|
| Real | | |
| Draw | 21 | 35 |
| Lose | 32 | 36 |
| Win | 21 | 102 |

| Modelled | Lose | Win |
|---|---|---|
| Real | | |
| Draw | 21 | 35 |
| Lose | 34 | 34 |
| Win | 22 | 101 |

| Modelled | Lose | Win |
|---|---|---|
| Real | | |
| Draw | 22 | 34 |
| Lose | 31 | 37 |
| Win | 24 | 99 |

| Modelled | Draw | Lose | Win |
|---|---|---|---|
| Real | | | |
| Draw | 1 | 22 | 33 |
| Lose | 0 | 35 | 33 |
| Win | 0 | 22 | 101 |

| Modelled | Draw | Lose | Win |
|---|---|---|---|
| Real | | | |
| Draw | 13 | 17 | 26 |
| Lose | 20 | 23 | 25 |
| Win | 30 | 27 | 66 |

| Modelled | Lose | Win |
|---|---|---|
| Real | | |
| Draw | 21 | 35 |
| Lose | 32 | 36 |
| Win | 23 | 100 |

| Modelled | Lose | Win |
|---|---|---|
| Real | | |
| Draw | 22 | 34 |
| Lose | 34 | 34 |
| Win | 24 | 99 |

**Figure 7:** Test results with: (I) Random Forest Classifier, (II) Logistic Regression, (III) Multi-Layer Perceptron, (IV) K-nearest Neighbours, (V) AdaBoost, (VI) Decision tree, (VII) Quadratic Discriminant Analysis, (VIII) Support Vector Classifier. The main diagonal in each table shows the number of modelled outcomes that were agreed with reality.

Many of the well performing models are clearly biased towards predicting wins for the home team, in both validation and testing, which was an obviously explainable bias resulting from the phenomenon of 'home team advantage' being evident in the training set. Since our attempts to utilise the 'neutral_location' feature did not help to remedy this, a new strategy should be considered in future work. Creating separate models to predict watches with or without a home team advantage, and partitioning training data as such, would be one way to combat this. However, this would reduce the amount of training data available to each model significantly.

## 7 Conclusion

With this project, we have demonstrated the applicability of multiple machine learning algorithms to classification tasks in sports outcome predictions. With data on a relatively limited number of metrics, compared to that which is collected and used in professional sports modelling, we have achieved a predictive accuracy significantly greater than 50%.

One way to expand our validation and testing would be to swap all of the home / away results in the training/test sets and re-test the models. Most of the better performing models appear biased towards predicting home team wins, which would be a poor bias to have in this new scenario. Any results which remained consistent between the testing we have already performed and testing on this new set could be suggested to be the ones which a model is the most confident of and can most reliably predict, giving us a greater understanding of where the model's strengths and weaknesses lie.

Other than this, the most obviously useful ways to build upon this work would be to consider strategies to eliminate the bias in our models towards predicting home team wins (see Testing) and to consider the incorporation of additional useful data and feature engineering. The training dataset here does not have any information on the players present in each squad or match and, as our background research suggested, incorporating climate data for match locations would also likely be a fruitful addition.

## References

1. Baboota, R. and Kaur, H., 2019. Predictive analysis and modelling football results using machine learning approach for English Premier League. International Journal of Forecasting, 35(2), pp.741-755.
2. Baboota, R. and Kaur, H., 2019. Predictive analysis and modelling football results using machine learning approach for English Premier League. International Journal of Forecasting, 35(2), pp.741-755.
3. Boecker, L. (2021). One group's pain is another group's pleasure: Examining schadenfreude in response to failures of football teams during the World Cup 2018. Psychology of Sport and Exercise, 56, p.101992. doi:10.1016/j.psychsport.2021.101992.
4. Brocherie, F., Girard, O., Farooq, A., and Millet, G.P. (2015). Influence of Weather, Rank, and Home Advantage on Football Outcomes in the Gulf Region. Medicine & Science in Sports & Exercise, 47(2), pp.401–410. doi:10.1249/mss.0000000000000408.
5. FIFA, 2021. Men's Ranking Procedures, FIFA. Available at: https://www.fifa.com/fifa-world-ranking/procedure-men (Accessed: January 21, 2023).
6. Horvat, T. and Job, J., 2020. The use of machine learning in sport outcome prediction: A review. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 10(5), p.e1380.
7. Keogh, F. and Rose, G, 2013. Football betting - the global gambling industry worth billions - BBC Sport, BBC News. BBC. Available at: https://www.bbc.co.uk/sport/football/24354124 (Accessed: January 21, 2023).
8. Panjeh, 2020. Scikit learn hyperparameter optimization for MLPClassifier, Medium. Available at: https://panjeh.medium.com/scikit-learn-hyperparameter-optimization-for-mlpclassifier-4d670413042b (Accessed: January 20, 2023).
9. Rodrigues, F. and Pinto, Â. (2022). Prediction of football match results with Machine Learning. Procedia Computer Science, 204, pp.463–470. doi:10.1016/j.procs.2022.08.057.
10. Rahman, Md. A. (2020). A deep learning framework for football match prediction. SN Applied Sciences, 2(2). doi:10.1007/s42452-019-1821-5.
11. Scikit Learn, 2023. Feature importances with a forest of trees, scikit. Available at: https://scikit-learn.org/stable/auto_examples/ensemble/plot_forest_importances.html (Accessed: January 20, 2023).
12. Sigmundsson, H., Dybendal, B.H., Loftesnes, J.M., Ólafsson, B. and Grassini, S. (2022). Passion a key for success: Exploring motivational factors in football players. New Ideas in Psychology, 65, p.100932. doi:10.1016/j.newideapsych.2022.100932