Predicting FIFA World Cup Results

Group I

January 27, 2023

1 Introduction

This project aims to apply supervised machine learning models (namely classification) to predict international FIFA game outcomes accurately. Such predictive models are widespread in football, with Maher's (1982) work on a bivariate Poisson model predicting match results based on the attack and defence capabilities from goals scored and conceded. [1] Since then, Joseph et al. (2006) demonstrated that incorporating expert judgments with data into a Bayesian network model led to more accurate predictions than numerous other data-driven machine-learning techniques. [2] Although there is scepticism regarding the efficiency of football predictive models (e.g. Leicester's odds of winning the 2015/2016 PL season was 5000/1), we can underline that the popularity of football models is driven by the prospect of identifying market inefficiencies for betting. [3]

With this in mind, we aim to rectify the inefficiencies of previous statistical models, such as those published by Peel and Pope (1989); Forrest et al. (2005); Graham Stott (2008); Vecer et al (2009), through revised featurisation and optimisation processes. [4][5][6] This includes training a suitable algorithm by feeding through labelled inputs and attributes alongside specified outputs and matching results. The model can then make predictions on test data with quantifiable predictive accuracy.

2 Data Transformation and Exploration

2.1 The Data

The dataset includes various games between international teams, with features such as **FIFA rankings of the two competing teams prior to the game, number of goals scored, the location (city) of the game, goalkeeper, defence, midfield, and offence scores**. The dataset comprises categorical data (e.g., continent) and numerical data, which indicates the FIFA ranking of a team. Featurisation is a technique to change data (e.g., text, graphs, categorical, and time-series data) into a numerical vector. The process differs from feature engineering, which focuses on transforming the numerical features to improve model efficiency (the features are already in numerical form). Featurisation techniques depend on the data type - text data techniques may include Bag of Words (BoW) and Weighted Word2Vec, and categorical data may include one-hot encoding and mean response rate.

The total dataset includes games between the years 2004 and 2021. Given that FIFA ranks teams based on the last four years of their performance and that the test set is for 2022, we used the data from 2017 onwards, [7] [8]. The reasons for dropping data before 2017 include: (1) player changes - this may include nationality changes, such as Aymeric Laporte playing for Spain (previously France) or teams bringing in new players; (2) coaching staff changes; (3) style of play changes - a team may have been offence dominated in the past years, but not in 2004; (4) team success - a football team's success can change over the years, such as Italy not winning since the world cup since 2006.

2.2 Cleaning and Organisation of Data

To deal with missing values, we impute them by either (1) replacing them with zeros as showcased by van Buuren and Groothuis-Oudshoorn (2011); (2) using forward fill to copy previous data as used by Nguyen et al. (2019), (3) using the mean or median as done in Herbinet's (2018) paper on "Predicting Football Results Using Machine Learning Techniques" or; (4) applying statistical approaches that take into account covariance structures such as temporal dependence in time series data as showcased in Suresh et al. (2020) paper [9][10][11][12].

This report uses the mean method as a way of dealing with missing values, as it is most suited for football-scoring predictive models and is the most widespread imputation method across related research. The mean method is time-effective and computationally efficient that can only be used with numeric data. Despite efficiencies, the mean method does not consider the underlying distribution of the data, and it is sensitive to outliers and skewed distributions; hence, it is advisable to avoid the technique on encoded categorical features [13][14].

The mean method can be implemented using the *scikit learn* library, class *sklearn.impute.SimpleImputer* with the method, *SimpleImputer(strategy='mean')*.[15]
The dataset includes both numerical and categorical data. Categorical data can include information such as continents, cities, and game results. There are two types of categorical data:

- Ordinal: Has a particular order such as Disagree, Neutral, Agree
- Nominal: Has no particular order, such as continents

The majority of the dataset is nominal data. Only, the result of the game, "win, draw, lose" are ordinal and are represented by "1, 0, -1" respectively.

2.3 Data Visualisation

In this model, we aim to predict the outcome of the home team's result (win, loss or draw). We can use various methods to visualise this relationship, such as heat mapping our categorical and numerical variables. Using a heatmap also supports our feature engineering process. However, the heatmap method uses Pearson correlation, which assumes a linear relationship between variables. Hence, we will also use the mutual information method, which is an alternative to linear correlation. These methods quantify the relationship between two variables (i.e., calculates the statistical dependence). The two visualisation techniques support the comparison of the relationship between features and domain knowledge; a necessary process for effective feature engineering. As the heatmap method uses Pearson correlation, each feature has a correlation coefficient between -1 and 1. Zero means no correlation.

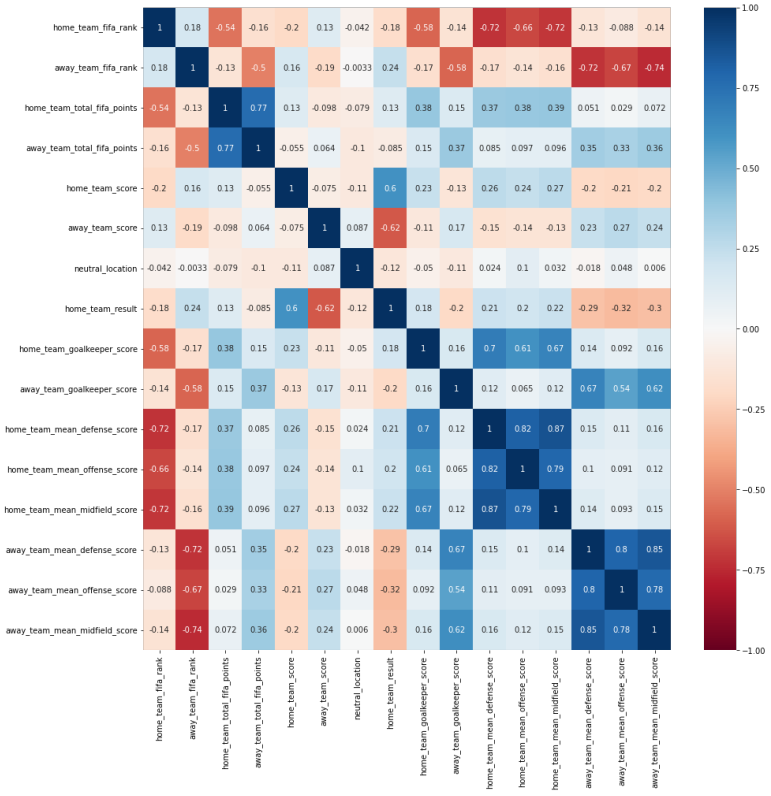

Figure 1:
Heatmap displaying Pearson correlations between training data features

We are interested in the row of hometeam results, as we take home team winning as our reference and seek the features correlations to these, see Figure 2 (b).

As can be seen from the heatmap, the correlation coefficient is for pairs of data. For a set of pairs $\{(x_1, y_1), ...(x_n, y_n)\}$ we can express the coefficient r_{xy} as [16]:

$$r_{xy} = \frac{\sum_i^n x_i y_i - n\bar{x}\bar{y}}{\sqrt{\sum_i^n x_i^2 - n\bar{x}^2}\sqrt{\sum_i^n y_i^2 - n\bar{y}^2}} \tag{1}$$

Which can be expressed in terms of variances s_x and s_y [17]:

$$r_{xy} = \frac{1}{n-1}\sum_{i=1}^{n}(\frac{x_i - \bar{x}}{s_x})(\frac{y_i - \bar{y}}{s_y}) \tag{2}$$

An alternative is the Mutual Information method, which is computationally efficient, resistant to overfitting, and can detect any relationship between two variables. Additionally, it is easy to implement and describes the relationship in terms of uncertainty. In statistics, the uncertainty of a single random variable, X, is Entropy. Conditional Entropy is when this uncertainty is dependent on another variable; Entropy of X given Y can be expressed as $H(X|Y)$ [8]. For a joint probability distribution between X, Y, the reduced uncertainty of X given random variable Y represents the mutual information, $I(X;Y)$, between the two random variables. Such that [18]:

$$I(X;Y) = H(X) - H(X|Y) = \sum_{x,y} p(x,y) log \frac{p(x,y)}{p(x)p(y)} \tag{3}$$

Representing a quantified relationship between two random variables that varies logarithmically in the range $[0, \infty]$. We can then calculate a Mutual Information score for the hometeam winning the game. This asks the question, what is the reduction of uncertainty in hometeam winning, given another feature? The mutual information scores are used to "double check" the most effective features for home team to win the game.

We can then compare these results, with the linear correlation results using data visualisation methods for a better understanding of features, see figure 2.

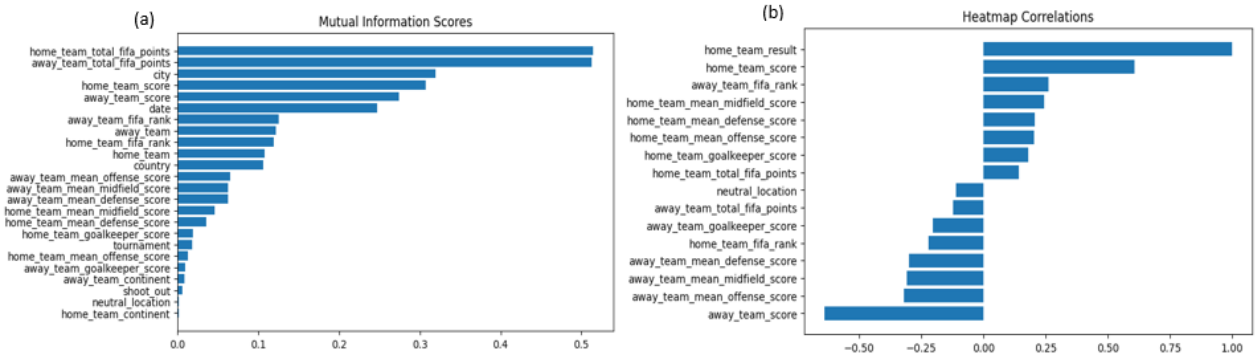

Figure 2:
(a) Mutual Information Scores Bar Chart, (b) Pearson Correlation Coefficients Bar Chart

2.4 Feature Engineering

The initial features used in our predictive model may not always yield optimal results. Hence, we select and create new features through a process called feature engineering, enhancing the predictive power of the model and potentially overcoming overfitting. During feature engineering, we aim to (1) maintain a similar number of features, which can help to avoid overfitting or to underfit the model; (2) transform non-numeric data (explained above); (3) use explanatory methods to ensure that the created and transformed features remain relevant; (4) use high-quality features (e.g., features with a high number of missing values can negatively impact model performance). These aspects are paramount to maintaining critical balances, such as the balance between the model complexity and the predictive performance.
For example, we are given the number of goals scored by each team in the data set. Let us propose a hypothesis: the more goals a team scores, the more likely it will win. In the last six years, the champion of the English Premier League has come either first or second in terms of goals scored per game [19]. Defensive strategies cannot be ignored in football; however, it is evident that given a team having the probability of scoring more, there is less uncertainty in the probability of that team winning a game. This process requires industry knowledge and contextualising the initial feature analysis. We can then use the average goals scored per game in the past "n=5" years as a feature in our analysis. Below we represent the engineered features with the most effective Pearson Correlation Coefficients. The analysis in section 2.2 is used to engineer these features.

- home (or away) team past offense ability
  - Motivation: Feature assesses a team's ability to score, where we take opponents strength into account.

- Method: Used the features "home team score", "Away Team Midfield/Defense Score", such that (Number of goals Scored by Home team + 1)/ (Away Teams Defense + Offense Ability) where +1 term is the bias term.
- Pearson Correlation Coefficient = 0.61, when similar method is used for away teams past offense ability we achieve PCC = -0.64.

- home field advantage
  - Motivation: Feature quantifies home teams advantage due to playing at home.
  - Method: Used features "home team result", "away team result". This feature calculates the difference of ratios of a teams winning rate at home and away, such that, (home wins / games played at home) - (away wins / games played away)
  - Pearson Correlation Coefficient = 0.52

- Home team (or away) result per rank
  - Motivation: Teams often play against teams of varying strengths. Therefore, we can engineer a feature which accounts for this.
  - Method: Used "home team result", and "away team fifa rank" such that, (home team result) / (away team fifa rank), then the mean of the values are taken for each team.
  - Pearson Correlation Coefficient = 0.47. When the similar process is applied the engineered feature "away team result per rank" yields a PCC = -0.44

3 Methodology Overview

3.1 Background Reading

Predicting the outcome of football matches has been a subject of scholarly inquiry since the mid-20th century. However, the methods employed have undergone significant evolution over time. The first modelling approach (1956) employed the Poisson distribution to model the number of goals scored by each team in a match. [20] Subsequently, in 1971, Reep advanced this approach by using a negative binomial distribution to model the same outcome. [21] Despite these early attempts, there persisted a high degree of scepticism regarding the ability to predict the outcomes of football matches, given their perceived dependence on chance. However, in 1974, Hill presented a seminal study which demonstrated that the results of football matches are not solely determined by chance and can be modelled and predicted. [23] Building upon this foundation, Maher (1982) proposed a method that utilised the Poisson distribution to model both home and away team attacking and defensive capabilities to predict the mean number of goals scored by each team. [24] Subsequently, Dixon and Cole's model, which was able to output probabilities for match results and scores, represented a further advancement in the field. [25]

At the beginning of the 21st century, there was a pivotal shift in the focus of research, moving towards modelling match results (win/draw/loss) directly. [20] This trend towards direct prediction of match results has persisted, with recent research primarily utilising classification models, ranging from logistic regression to Bayesian networks. The scope of this research has also been broadened, encompassing a wide range of competitions, from domestic leagues such as the English Premier League to international competitions such as the World Cup.

Parallel to this research, data availability has grown exponentially over the past decade, with many high-level clubs, such as Arsenal, hiring data scientists to gain a competitive edge. [27] Some analytics companies now track over 5,000 data points per game, providing access to a far more comprehensive range of metrics that can be used to model predictions. [29] Capitalising on this data, Herbinet proposed an innovative approach, utilising expected goals models and ELO ratings to predict match outcomes with over 50% accuracy. [20]

3.2 Data Pre-Processing and Partitioning

The dataset contains information regarding the outcomes of international football matches dating back to the 1990s, initially consisting of 25 attributes and 5,641 samples. A traditional data partitioning method puts 80% of the data in the training set, 10% in the validation, and 10% in the test set [30]; however, this project used an 88:12 train-test split. The rationale was that data would later be partitioned using cross-validation, so more data needed to be retained in the training set. Finally, the data was scaled using StandardScaler, considered one of the optimal scaling techniques [31]; this process normalises the data, ensuring that all variables contribute equally to the model fitting. [32]

3.3 Feature Selection

In machine learning problems, many candidate features are often irrelevant or redundant to the learning task [33]; thus, their inclusion will deteriorate the model's performance and lead to overfitting [34]. In this project, three methods of feature selection were employed: LASSO, XGBoost feature selection, and SelectKBest.

3.4 Scoring

The scoring metrics used were accuracy, the area under the curve (AUC) and weighted F1-Score. While accuracy represents the generic approach, it is ineffective for unbalanced datasets [11]. This problem is inherently unbalanced, given football's historic home advantage driven by home fans' influence [36]. The F1 score computes the harmonic mean of precision and recall [37], and its weighted variant considers each outcome proportion, accounting for the potential class imbalances discussed prior [38]. The AUC computes the area under the ROC, which is a graph that depicts the performance of a classification model at all classification thresholds [39]. For this project, an accuracy function was created to select the most suitable scoring function, which provided the best representation, for each model.

3.5 Optimisation

A vast range of different machine learning models can be applied to many problems; nevertheless, the one commonality is that the quality of predictions is crucial [40]. Ideally, the model should avoid overfitting and underfitting and achieve a low generalisation error, meaning that it performs well on out-of-sample data [41]. In this project, the optimisation process involved hyperparameter tuning and ensemble modelling, intending to attain the most accurate model.

3.5.1 Hyperparameter Tuning

In machine learning models, two types of parameters exist: model parameters and hyperparameters [42]. Unlike model parameters, hyperparameters require initialisation before model training since they represent the model architecture [43]. Ghawi and Pfeffer define hyperparameter tuning as "the problem of choosing a set of optimal hyperparameters for a learning algorithm" [44]. It is a crucial part of optimisation since selecting the best hyperparameter configuration directly impacts the model's performance [42]. In this project, we used grid search, considered a brute force method of hyperparamter tuning [45]. Grid search evaluates the model using every possible combination of hyperparameter values [44]. Since we tested ten models, it would have been computationally expensive to do this for every hyperparameter; thus, we selected the most influential, using prior knowledge and supplementary research, only tuning these. Ultimately, we selected the hyperparameter values that maximised model performance.

3.5.2 Ensemble Modelling

Ensemble modelling combines several models to improve generalisation performance [46]. As long as these models are diverse and independent, the generalisation error of the predictions will decrease [47]. In this project, we evaluated the individual models, and ensemble modelling was used to amalgamate the three optimal models: gradient boosting, decision tree, and random forest.

3.6 Approach Amendments

In the end, XGBoost feature selection was not included in model evaluation since it returned only one feature.

4 Model Training and Validation

Ten models were used: Logistic Regression, K-Nearest Neighbours, Multi-Layer Perceptron, Gradient Boosting Classifier, Random Forest Classifier, Decision Tree Classifier, AdaBoost Classifier, SVM, LDA, and QDA.

In this project, we evaluated all ten models using 5-fold cross-validation, selecting this approach due to its practicality and flexibility [48]. Given that we used such a diverse range of models, its flexibility was essential, meaning it could be applied to any model, regardless of technical details [41]. Moreover, we selected 5-fold cross-validation since it is preferable computationally [49]. The process involves splitting the dataset into five groups. Then, in each iteration, one group represented the validation data, and the rest constituted the training data. Each iteration was evaluated, and an average of all iterations' cross-validation accuracy was calculated to estimate the model's performance [41].

Outcome of Hyperparameter Tuning	
Model	Optimal Hyperparameter Values
Logistic Regression	Tolerance = 0.01
K-Nearest Neighbour	Number of Neighbours = 17 Weights = Uniform
Multi-Layer Perceptron	
Gradient Boosting Classifier	Learning Rate = 0.1 Maximum Depth = 1 Number of Estimators = 4
Random Forest Classifier	Maximum Depth = 2
Decision Tree Classifier	Maximum Depth = 2
AdaBoost Classifier	Number of Estimators = 10
SVM	Degree = 2 Kernel = Linear Tolerance = 0.01
LDA	Number of Components = None Solver = SVD Tolerance = 0.001
QDA	

Table 1: The optimal hyperparamter values for each model

5 Results and Discussion

Table 2 shows that some models, like Decision Tree and AdaBoost, predicted the training outcomes with great accuracy. Furthermore, it shows that, as a whole, feature selection had a net positive impact on the model's performance.

Cross Validation Accuracy of Each Model on Training Data			
Model	No Feature Selection	LASSO	Select-K-Best
Logistic Regression	0.990	0.991	0.990
K-Nearest Neighbour	0.767	0.951	0.981
Multi-Layer Perceptron	0.988	0.991	0.989
Gradient Boosting Classifier	0.465	0.465	0.465
Random Forest Classifier	0.754	0.766	0.753
Decision Tree Classifier	1.0	1.0	1.0
AdaBoost Classifier	1.0	1.0	1.0
SVM	0.993	0.993	0.993
LDA	0.910	0.914	0.903
QDA	0.934	0.808	0.931

Table 2: Cross validation of each model using varying degrees/methods of feature selection on the train data

Figure 3 shows that hyperparameter tuning had a net positive impact. Either the cross-validation accuracy increased significantly (e.g. QDA), or the model performance was unchanged (e.g. Gradient Boosting), but no model's performance worsened. Further, many models whose performance was unaffected had already performed

exceptionally well, with some exhibiting 100% accuracy; thus, there was little scope to improve performance. Nevertheless, grid search may not have been the most effective since the manual selection of what hyperparameters to tune was required. A method like Bayesian optimisation may see an even more significant impact.

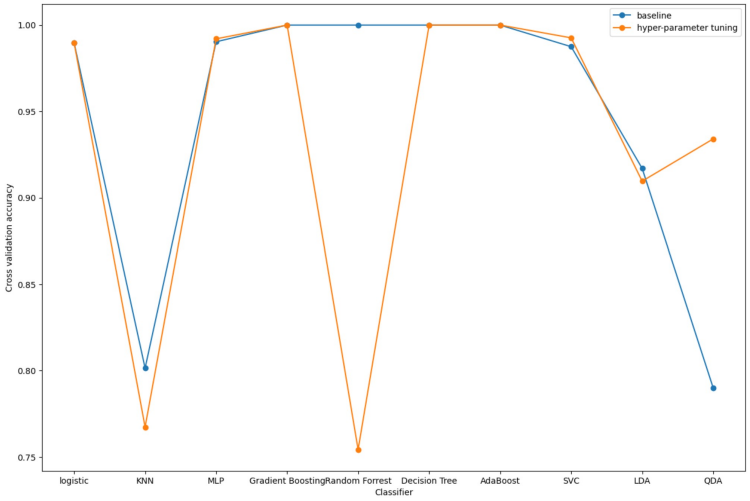

Figure 3: Effect of hyperparameter tuning on each model's performance

Figure 4 shows that the impact of feature engineering varies; for models like KNN, model performance deteriorates with feature engineering, while model performance improves for models like Random Forest. This outcome suggests that some of the features created may have been irrelevant and added unnecessary noise to the model; however, it could also result from the curse of dimensionality. For these comparisons, feature selection still needed to be implemented, so adding new features may have caused the curse of dimensionality to arise, whereby more features caused the model to struggle to effectively map the inputs to the outputs, negatively impacting its accuracy. However, after feature selection, the positive impact of feature engineering may be more noticeable.

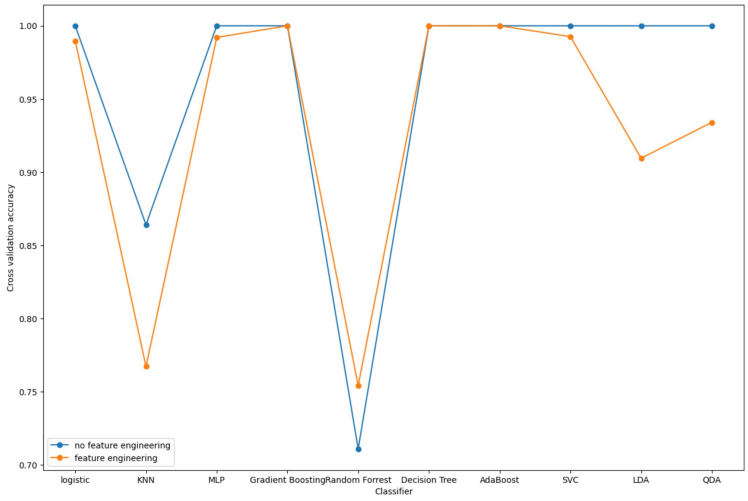

Figure 4: Effect of feature engineering on each model's performance

6 Final Predictions

Table 3 shows that the best-performing model was Random Forest with either no feature selection or LASSO, boasting 52.5% cross-validation accuracy. While this model outperformed the ensemble model (using LASSO), the latter was selected given its increased robustness - its predictions will be more stable on new data - giving a final cross-validation accuracy of 51.6%.

This model performed significantly worse than the models performed on the training data; however, this is expected, given that football matches exhibit randomnesses, and many factors, which cannot be modelled, can

impact the outcome, including dubious referee decisions. The seemingly unpredictable can happen, like Saudi Arabia beating Argentina (the eventual champions) at the 2022 FIFA World Cup.

Cross Validation Accuracy of Each Model on Test Data			
Model	No Feature Selection	LASSO	Select-K-Best
Logistic Regression	0.369	0.418	0.500
K-Nearest Neighbour	0.225	0.389	0.348
Multi-Layer Perceptron	0.287	0.418	0.496
Gradient Boosting Classifier	0.496	0.496	0.496
Random Forest Classifier	0.525	0.525	0.492
Decision Tree Classifier	0.410	0.516	0.504
AdaBoost Classifier	0.402	0.402	0.402
SVM	0.348	0.422	0.492
LDA	0.282	0.393	0.496
QDA	0.225	0.320	0.225
Ensemble Model	0.410	0.516	0.504

Table 3: Cross validation of each model using varying degrees/methods of feature selection on the test data

Moreover, Figure 5 shows that the test data had a much different distribution to the training data, seemingly less linearly separable than the latter, explaining the lack of generalisability.

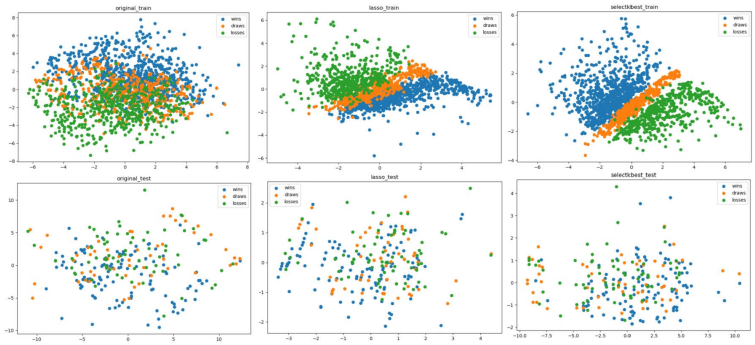

Figure 5: Comparing distribution of training and test data

Table 4 shows that the ensemble model predicted home wins well, identifying 93% of these outcomes and predicting them with 53% precision. Nevertheless, it only identified 19% of draws and 0% of away wins, indicating that it over-predicted home wins. This finding is a severe limitation; since home wins constituted a large proportion of the test data, the model's accuracy is likely overstated. This limitation results from the data imbalances, with a far more significant proportion of home wins in the training data. Nevertheless, in the future, we could rectify this issue through resampling methods.

Classification Report				
Outcome	Precision	Recall	F1-Score	No. of occurences
Home Win	0.53	0.93	0.68	121
Draw	0.46	0.19	0.27	68
Away Win	0.00	0.00	0.00	55

Table 4: Classification report of ensemble model

7 Conclusions

In conclusion, the final model performed acceptably, performing at the level of previous studies like Herbinet [20], attaining an accuracy of 51.6%. Nevertheless, there remains extensive scope for improvement. In the future, the featurisation process could be enhanced by combining features to give a higher final correlation. Furthermore, the study could capitalise on the extensive available data and utilise more advanced statistics; for instance, passes allowed per defence action (PPDA), a great indicator of a team's pressing intensity. In addition, kernel methods could be employed in the optimisation process to enable the classes to be more easily classified. All of these potential improvements would enhance the predictive capabilities of our model; however, predicting football matches remains an intensely difficult task, for which researchers will continue to incessantly search for an improved approach.

References

[1] Maher, M. J. (1982). Modelling association football scores. Statistica Neerlandica, 36(3), 109–11.

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

[3] R. Tanner, 5000-1 - the leicester city story: How we beat the odds to become Premier League Champions. London: Icon Books, 2017.

[4] Pope, P., Peel, D. (1989). Information, prices and efficiency in a fixed-odds betting market. Economica, 56, 323-341.

[5] Forrest, D., Goddard, J., Simmons, R. (2005). Odds-setters as forecasters: The case of English football International Journal of Forecasting, 21, 551-564.

[6] Vecer, J., Kopriva, F., Ichiba, T. (2009). Estimating the Effect of the Red Card in Soccer: When to Commit an Offense in Exchange for Preventing a Goal Opportunity. Journal of Quantitative Analysis in Sports, 5: Iss. 1, Article 8

[7] Men's Ranking https://www.fifa.com/fifa-world-ranking/men?dateId=id13869 Accessed: 2022-12-26

[8] FIFA world ranking: How it is calculated and what it is used for Goal.com UK https://www.goal.com/en-gb/news/fifa-world-ranking-how-it-is-calculated-what-it-is-used-for/16w60sntgv7x61a6q08b7ooi0p Accessed: 2022-12-26

[9] van Buuren, S., Groothuis-Oudshoorn, K. (2011). mice: Multivariate Imputation by Chained Equations. R. Journal of Statistical Software, 45(3), 1–67.

[10] M. Nguyen, T. He, L. An, D. C. Alexander, and J. Feng, "Predicting Alzheimer's disease progression using deep recurrent neural networks," ResearchGate, Sep-2019.

[11] C. Herbinet, "Predicting Football Results Using Machine Learning Techniques," Imperial College London, 20-Jun-2018. Available: https://www.imperial.ac.uk/media/imperial-college/faculty-of-engineering/computing/public/1718-ug-projects/Corentin-Herbinet-Using-Machine-Learning-techniques-to-predict-the-outcome-of-profressional-football-matches.pdf.

[12] Aneesha K Suresh, James M Goodman, Elizaveta V Okorokova, Matthew Kaufman, Nicholas G Hatsopoulos, Sliman J Bensmaia (2020) Neural population dynamics in motor cortex are different for reach and grasp eLife 9:e58848 https://doi.org/10.7554/eLife.58848

[13] Sharpening the BLADE: Missing Data Imputation Using Supervised Machine Learning Suresh MTaib RZhao YJin WSee fewer Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), (2019), 215-227, 11919 LNAI

[14] "Dealing with missing data: Key assumptions and methods for applied analysis" M. Soley-Bori, Boston University, Boston, tech., 2013.

[15] sklearn.impute.SimpleImputer — scikit-learn 1.2.0 documentation https://scikit-learn.org/stable/modules/generated/sklearn.impute.SimpleImputer.html Accessed: 2022-12-27

[16] Pearson, K., "Note on Regression and Inheritance in the Case of Two Parents", Proceedings of the Royal Society of London Series I, vol. 58, pp. 240–242, 1895

[17] A similarity algorithm based on the generality and individuality of wordsPDF Yinfeng Zou, Chunping Ouyang et al. Lecture Notes in Computer Science https://link-springer-com.libproxy.ucl.ac.uk/chapter/10.1007/978-3-319-50496-4$_4$8

[18] Cover, Thomas M, and Joy A Thomas. Elements of Information Theory. John Wiley amp; Sons, Incorporated, 2006.

[19] "Premier League 2020/2021 Table amp; Standings - football rankings," Eurosport. [Online]. Available: https://www.eurosport.com/football/premier-league/2020-2021/standings.shtml. [Accessed: 05-Jan-2023].

[20] Herbinet, C., 2018. Predicting football results using machine learning techniques. MEng thesis, Imperial College London.

[21] M. J. Moroney. Facts from figures, 3rd edn. Penguin: London, 1956. pages 16

[22] C. Reep. Skill and chance in ball games. Journal of the Royal Statistical Society Series A 131: 581-585, 1971. pages 16

[23] I.D. Hill. Association football and statistical inference. Applied Statistics 23: 203- 208, 1974. pages 16

[24] M. J. Maher. Modelling association football scores. Statistica Neerlandica, 1982. pages 16

[25] M.J. Dixon, S.C. Coles. Modelling association football scores and inefficiencies in the football betting market. Applied Statistics, 1997. pages 17

[26] D. Forrest, R. Simmons. Forecasting sport: The behaviour and performance of football tipsters. International Journal of Forecasting, 2000. pages 18

[27] Carey, M. (2022) The Genie is well and truly out of the bottle when it comes to data in football, The Athletic. The Athletic. Available at: https://theathletic.com/3209373/2022/03/26/the-genie-is-well-and-truly-out-of-the-bottl e-when-it-comes-to-data-in-football/ (Accessed: January 25, 2023).

[28] Prasetio, D., 2016, August. Predicting football match results with logistic regression. n 2016 International Conference On Advanced Informatics: Concepts, Theory And Application (ICAICTA) (pp. 1-5).

[29] Franklin-Wallis, O. (2020) There's a big fight brewing over the Premier League's player data, WIRED UK. Available at: https://www.wired.co.uk/article/project-red-card-football-data (Accessed: January 25, 2023).

[30] Baheti, P. (2023) Train test validation split: How to best practices [2023], V7. Available at: https://www.v7labs.com/blog/train-validation-test-set (Accessed: January 25, 2023).

[31] Thara, D.K., PremaSudha, B.G. and Xiong, F., 2019. Auto-detection of epileptic seizure events using deep neural network with different feature scaling techniques. Pattern Recognition Letters, 128, pp.544-550.

[32] Loukas, S. (2021) How scikit-learn's standardscaler works, Medium. Towards Data Science. Available at: https://towardsdatascience.com/how-and-why-to-standardize-your-data-99 6926c2c832 (Accessed: January 25, 2023).

[33] Kotsiantis, S., 2011. Feature selection for machine learning classification problems: a recent overview. Artificial intelligence review, 42(1), pp.157-176.

[34] Bhattacharyya, S. (2023) Ridge and lasso regression: L1 and L2 regularization, Medium. Towards Data Science. Available at: https://towardsdatascience.com/ridge-and-lasso-regression-a-complete-guide-with-python-scikit-learn-e20 e34bcbf0b (Accessed: January 25, 2023).

[35] Tracyrenee (2021) How to select features using SelectKBest in python, Medium. MLearning.ai. Available at: https://medium.com/mlearning-ai/how-to-select-features-using-selectkbest-in-python-c5a5239969f0 (Accessed: January 25, 2023).

[36] Courneya KS, Carron AV (1992) The home advantage in sport competitions: a literature review. J Sport Exercise Psy 14(1):13–27. https://doi.org/10.1123/jsep.14.1.13

[37] Korstanje, J. (2021) The F1 score, Medium. Towards Data Science. Available at: https://towardsdatascience.com/the-f1-score-bec2bbc38aa6 (Accessed: January 18, 2023).

[38] Leung, K. (2022) Micro, Macro weighted averages of F1 score, clearly explained, Medium. Towards Data Science. Available at: https://towardsdatascience.com/micro-macro-weighted-averages-of-f1-score-clearly-expl ained-b603420b292f (Accessed: January 18, 2023).

[39] Google (2023) Classification: Roc curve and AUC | machine learning | google developers, Google. Google. Available at: https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc (Accessed: January 25, 2023).

[40] Raschka, S., 2018. Model evaluation, model selection, and algorithm selection in machine learning. arXiv preprint arXiv:1811.12808.

[41] Emmert-Streib, F. and Dehmer, M., 2019. Evaluation of regression models: Model assessment, model selection and generalization error. Machine learning and knowledge extraction, 1(1), pp.521-551.

[42] Yang, L. and Shami, A., 2020. On hyperparameter optimization of machine learning algorithms: Theory and practice. Neurocomputing, 415, pp.295-316.

[43] Abreu, S., 2019. Automated architecture design for deep neural networks. arXiv preprint arXiv:1908.10714

[44] Ghawi, R. and Pfeffer, J., 2019. Efficient hyperparameter tuning with grid search for text categorization using kNN approach with BM25 similarity. Open Computer Science, 9(1), pp.160-180.

[45] Claesen, M., Simm, J., Popovic, D., Moreau, Y. and De Moor, B., 2014. Easy hyperparameter search using optunity. arXiv preprint arXiv:1412.1114.

[46] Di Napoli, M., Carotenuto, F., Cevasco, A., Confuorto, P., Di Martire, D., Firpo, M., Pepe, G., Raso, E. and Calcaterra, D., 2020. Machine learning ensemble modelling as a tool to improve landslide susceptibility mapping reliability. Landslides, 17(8), pp.1897-1914.

[47] Kotu, V. and Deshpande, B., 2014. Predictive analytics and data mining: concepts and practice with rapidminer. Morgan Kaufmann.

[48] Arlot, S. and Celisse, A., 2010. A survey of cross-validation procedures for model selection. Statistics surveys, 4, pp.40-79.

[49] Fushiki, T., 2011. Estimation of prediction error by using K-fold cross-validation. Statistics and Computing, 21(2), pp.137-146.