ITM Web of Conferences **70**, 03020 (2025) https://doi.org/10.1051/itmconf/20257003020
DAI 2024

A Hybrid Machine Learning Framework for Soccer Match Outcome Prediction: Incorporating Bivariate Poisson Distribution

Zhong An Chen

BASIS International School Guangzhou, 8 Jiantashan Rd, Huangpu District, Guangzhou, Guangdong, China, 510653

Abstract. The 2022 FIFA World Cup final attracted 1.5 billion viewers, while billions of dollars are wagered on soccer matches every year. The increasing demand for accurate predictions, both for academic research and betting purposes, has driven the development of advanced forecasting models. This study explores the application of mathematical and machine learning models to predict results of soccer matches, with the dual aim of academic advancement and profitable betting. The author utilizes a comprehensive dataset from top European leagues (2014-2022) and employ models including Bivariate Poisson Distribution, Naive Bayes, Neural Networks, Support Vector Machines, Random Forests, and Gradient Boosting. The paper's feature engineering combines historical match statistics, FIFA ratings, and betting odds. While Random Forests achieved the highest accuracy (56.25%), predicting draws remains challenging. The study highlights the potential for improved prediction systems and suggests future research in advanced draw prediction techniques and profitability analysis, the paper provides research directions for researchers in related fields.

1 Introduction

Soccer, football in British, is the most fascinating sport in the world. In the 2022 FIFA Qatar World Cup, 88966 spectators crowded into Lusail Stadium, and about 1.5 billion people watched this game. When it comes to soccer leagues, the English Premier League is the most popular soccer league in the world, with an online audience around 4.7 billion people. The reason for the author to predict soccer game results is not only to progress in academics but also to strive for a large amount of profit in soccer betting. During the whole period of the World Cup, a total of $35 billion dollars is used by people for soccer betting.

Mathematical models and machine learning models are widely used in different areas, such as finance credit prediction [1], basketball result prediction [2], and American football result prediction [3]. All the studies above demonstrated that mathematical models and machine learning are suitable for sports analytics and prediction, specifically in classifying win, draw, and lose.

Corresponding author: zhongan.chen17296-bigz@basischina.com

In terms of soccer, scholars have already pursued predicting the result of this sport. Initially, Stefani [4] used a linear regression to calculate the rating of different teams, and did a simple comparison of teams' ratings to decide the winning team. Yip [5] used a Bivariate Poisson model to predict the expected goals and corners for each game to find the final result. Mwembe [6] applied a bivariate Poisson model to predict the soccer game result in Zimbabwe premier soccer league. Besides mathematical models, Machine Learning (ML) was also employed in developing a prediction system. For example, Baboota [7] used various machine learning models to build a predictive system with a better performance compared to mathematical models. According to Baboota [7], their models have similar accuracy and all of them have a higher accuracy than the predictive system made by bookmakers, Bet365. Although they technically gained a positive expected value in soccer betting, they only focused on English Premier League matches, resulting in a limited amount of both training and testing data. What's more, they were still struggling to correctly predict draws.

The objective of this study is to develop a robust, comprehensive prediction system in terms of soccer game results and obtaining a satisfactory profit in long-term soccer games. This study is divided into two major parts, mathematical modeling, which is Bivariate Poisson distribution, and machine learning model, including Neural Network, Support Vector Machine, and Random Forest, using the result in the previous part as features. After independent training and validation of all the models, the study will conclude with a method to compare and contrast the predictions from all models, striving for a system that makes more accurate prediction than independent models.

2 Feature Engineering

2.1 Data Acquisition and Cleaning

The paper obtains data from a public dataset in Kaggle (Top 5 European Football Leagues (2014-2022)). The paper uses data from 2014 to 2022 from the top 5 leagues in Europe, including English Premier League, La Liga, Bundesliga, Serie A, and Ligue 1.

Some of the matches are deleted because the early soccer statistics were lost for technical reasons.

2.2 Feature Selection

One of the most important processes in the development of the prediction system is selecting a strong correlation feature for the machine learning model, which will have a great impact on the final prediction accuracy, according to Joseph [8]. This paper will use mathematical formulas to thoroughly explain the origin of each feature selected.

Cluster A: This class contains features acquired from historical statistics. For each game, the previous k games played by the specific team were aggregated, and the mean value for the past k games' statistic as the predicted indicator in the current prediction.

$$\alpha_x = \sum_{t=1}^{t\le k} \frac{\alpha_{x-t}}{k}, k \in 1,3,5 \tag{1}$$

αx represents the team's statistics in cluster A recorded in-game x. A special case happened in calculating the overall performance of the team in the previous k game. The average point gained was used to represent this.

$$\text{FTRpk} = \sum_{t=1}^{t\le k} \frac{3W_k + 1D_k + 0L_k}{k} \tag{2}$$

W_k, D_k, and L_k represent the time of the game win, draw, or lose in the previous k games accordingly. This equation uses the points calculation method provided by the English Premier League (EPL) (www.premierleague.com), and most soccer leagues in the world are using this system.

Cluster B: Features from Cluster B contain the betting odds (home win, draw, away win) provided my B365 company for each game.

Cluster C: Features from Cluster C contain the constant indicator for two teams, using the Elo system from FIFA game, which is released annually by Electronic Arts. Elo system has been used in several sports match predictions as features, such as Hvattum [9], which proves the practicality of this ranking system. The market value for each team is obtained from the transfer market(https://transfermarkt.com)

The results will be used in the following machine learning model as features, shown in table 1.

Table 1. Selected Features

Feature	Explanation	Cluster
FTRpk	Average Full-time points for last k games	A
HSpk	Home shots for last k games	A
ASpk	Away shots for last k games	A
HSTpk	Home shots on target for last k games	A
ASTpk	Away shot on target for last k games	A
HFpk	Home foul for last k games	A
AFpk	Away foul for last k games	A
HCpk	Home corner for last k games	A
ACpk	Away corner for last k games	A
HYpk	Home yellow cards for last k games	A
AYpk	Away yellow cards for last k games	A
HRpk	Home red cards for last k games	A
AYpk	Away red cards for last k games	A
B365H, B365D, B365A	betting odds for home win, draw, and away win in B365	B
PoissonH, PoissonD, PoissonA	the probability of home win, draw, and away win predicted by Bivariate Poisson Distribution	B
HTOa, ATOa	Overall rating in FIFA game	C
HTAt, ATAt	Attack rating in FIFA game	C
HTMid, ATMid	Midfield rating in FIFA game	C
HTDef, ATDef	Defense rating in FIFA game	
HomeSquad, AwaySquad	Teams number of players	C
HomeMV, AwayMV	Teams Total Market Value	C
Oadiff	Overall rating in FIFA game difference	C
Atdiff	Attack rating in FIFA game difference	C
Middiff	Midfield rating in FIFA difference	C
Defdiff	Defense rating in FIFA difference	C
MVdiff	Teams total market value difference	C

3 Bivariate Poisson Distribution Prediction

3.1 Model Overview

Poisson Distribution is a widely used statistical distribution in order to describe the probability of rare events that occur, in a specific time or space. In the sport analytics of soccer, a goal can be regarded as a rare event. Therefore, bivariate Poisson distribution has been widely applied in soccer match prediction. For example, Mwembe used a bivariate Poisson model to develop a prediction system and a profitable betting strategy of Zimbabwe premier league [6].

Since there is a complex and undefined relationship between the goals scored by two teams, the paper uses a bivariate Poisson Distribution to describe this relationship, which Joesph [8] already proves its capability.

Considering variable Xr, $r \in \{1, 2, 3, 4, 5\}$ follow simple Poisson distribution, with the other variables $\lambda r > 0$, the goals scores from each team following the Bivariate Poisson distribution, which has the probability of:

$$P_{X,Y}(x, y) = \frac{\lambda_1^x \lambda_2^y}{x!y! \cdot e^{\lambda 1+\lambda_2+\lambda_3}} \sum_{k=0}^{\min(x,y)} \binom{x}{k}\binom{y}{k} k! \left(\frac{\lambda_3}{\lambda_1^x \lambda_2^y}\right)^k \tag{3}$$

In this equation, $P_{X,Y}$ (x, y) represents the probability that team X scores x goals and team Y scores y goals. λ_1 and λ_2 represent the average strength of the team. λ_3 is a measure of the correlation between the goals for Team X and Team Y. If $\lambda_3 = 0$, λ_1 and λ_2 will be conditionally independent, and the Bivariate Poisson Distribution degenerates into the product of two independent Poisson distributions.

3.2 Calculating the Parameters

This model introduces attack strength (att) and defensive strength (def) metrics to quantify team offensive and defensive capabilities, respectively. Attack strength is calculated as the ratio of a team's goal scored (GS) to the average goals scored in the league (GS avg), while defensive strength is computed as the ratio of goals conceded (GC) to the average goals conceded in the league (GCavg).

$$\lambda_1 = \frac{att_h \cdot def_a}{GS_{avg}} \tag{4}$$

$$\lambda_2 = \frac{att_a \cdot def_h}{GS_{avg}} \tag{5}$$

$$att_x = \frac{GS_x}{GS_{avg}} \tag{6}$$

$$def_x = \frac{GC_x}{Gc_{avg}} \tag{7}$$

The author directly uses a straightforward approach to estimate λ_3 using the method of moments. The method of moments is computationally efficient and simple, making it suitable for large datasets, but its accuracy relies on the stability of the underlying distribution compared to popular methods such as Maximum likelihood estimation. However, as the author mentioned in previous chapters, which the sample size is large. Therefore, the dataset is able to support the method of moments to produce a moderately accurate estimation of the parameters in a short time and save computational resources simultaneously. The moments method uses covariance of the historical scoring data from two teams:

$$\lambda_3 = \frac{\mathrm{cov}_{x,y}}{\lambda_1 \lambda_2} \tag{8}$$

After Calculating all the parameters, each row of data will be substituted +n equation 3 to get the probability of scoring situations. To calculate the probability of home team winning, the model adds up all the probabilities where x > y, home scored more goals than away. Similar calculations are done to obtain the probability of away wins and draws.

4 Machine Learning Models

The paper tries various models in order to give a more robust prediction than the mathematical model and combine them together.

4.1 Baseline

The paper believes that bookmakers, such as B365, have already developed a comprehensive and accurate prediction system for soccer matches, so their betting odds may be a suitable indicator for the game result. While it is one of the features in my machine learning model since the motivation of this paper is to beat bookmakers, the author uses its indicator, the betting odds comparison, as the baseline model. The model is simple:

(i) In each row of the dataset, comparing the betting odds for home win, draw, and away wins, the smallest betting odd is chosen as the predicted result of that game.

(ii) Compared with the real results FTR, and obtain the accuracy.

This method obtained an accuracy of 0.50.

4.2 Naïve Bayesian

Naive Bayesian is a relatively simple algorithm compared to other machine learning models, since only a small number of hyperparameter needs to be adjusted. This paper uses the Gaussian Naive Bayesian method because the values of the features are continuous. Gaussian Naive Bayesian assume the prior probability of all features follows a Gaussian Distribution:

$$P\left(X_j = x_j \mid Y = C_k \right) = \frac{1}{\sqrt{2\pi\sigma_k^2}} \exp\left(-\frac{\left(x_j - \mu_k\right)^2}{2\sigma_k^2} \right) \tag{9}$$

In this equation, Ck is the k category of y, and μk and σ_2 is the parameters that are calculated from the data in training set. Naive Bayesian have technically the best accuracy compare to other classification models. However, that does not always happen, because Naive Bayesian model assume the property of independence, which usually not the true situation in a real-life soccer game. When there are a large number of features, or there are strong relations among features, the accuracy of classification is not good. In this paper, the features are moderate related. For example, statistics such as shots and shots on target are positively related, resulting in a lower accuracy score compare to other machine learning model.

4.3 Neural Network

Neural Network is a strong machine learning model that can process complex data with a relatively large noise. This paper used a Back Propagation Feed-forward Neural Network (BPNN). This algorithm iteratively adjusts the network weights connecting neurons, minimizing the error between the final output and the expected result. BPNN is a typical type of neural network widely used in various classification systems. It makes up of two stages: training and applying. The training stage is the foundation and prerequisite for BPNN to be put into use, while the usage stage is a very simple process. Given an input, BPNN will perform calculations based on the already trained parameters to obtain the output result. Figure 1 shows the flowchart of the neural network.

Model Architecture:

- **Input Layer**: This layer takes the preprocessed feature vectors as input. The dimensionality of the input data determines how many neurons are present in this layer.
- **Hidden Layers**: The model employs three hidden layers with 256, 128, 64, and 32 neurons, respectively. These layers extract relevant features from the input data using the ReLU activation function, which allows the model to become non-linear.
- **Output Layer**: The final layer has 3 neurons, representing the three possible out- comes of a soccer match: win, draw, or loss. The Softmax activation function is used to normalize the output probabilities, ensuring that they sum up to 1.

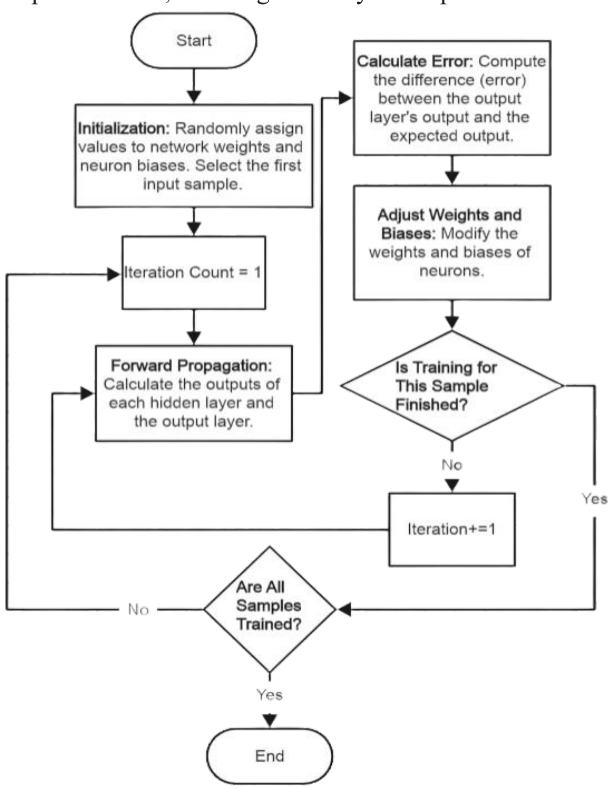

Figure 1. A flow chart showing the process of a Neural Network

4.4 Random Forest

Random Forest is a strong machine learning model, proposed by Leo Breiman in 2001 [10]. In order to solve classification and regression issues, this model builds a huge number of decision trees. In each decision tree, this model uses bootstrap sampling to choose a proportion of the training dataset, and in every split point, the Random Forest does not consider all possible features. The combination of different random decision trees enables this model to obtain robust accuracy when facing problems with complexity and randomness,

According to Jie and his colleagues [11], proper hyperparameters for the Random Forest model can significantly affect the overall performance.

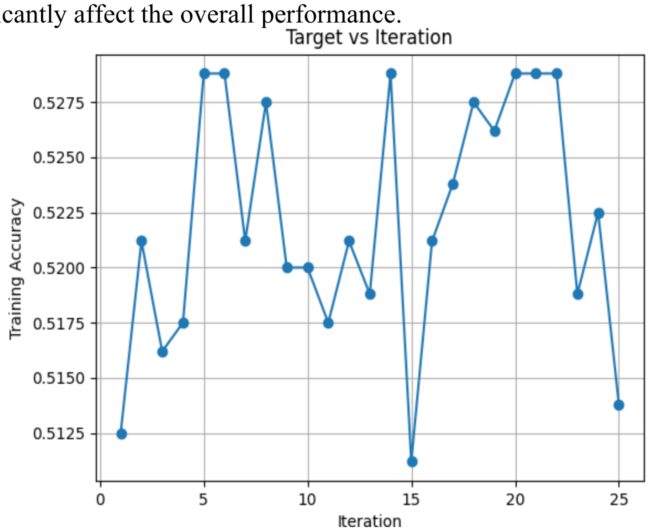

Figure 2. The graph of training accuracy at each iteration of Bayesian Optimization

Figure 2 illustrates the progression of a Bayesian optimization process applied to a Random Forest classification model. The x-axis represents the iterations of Bayesian optimization, while the y-axis represents the target value. The target value is measured as the accuracy score.

Table 2. Bayesian Optimization for Hyperparameters in Random Forest

Hyperparameters	Definition	Boundaries	Optimal Value
n_estimator	The number of decision trees in the random forest model.	(50, 500)	212
max_depth	The maximum depth of each tree in the random forest model.	(5, 10)	8
min_samples_split	The least number of samples needed to divide an internal node.	(2, 20)	16
min_samples_leaf	The least number of samples needed to be at a leaf node.	(2, 20)	5

The table 2 shows the optimal hyperparameter values that were found through Bayesian optimization for the given Random Forest model.

4.5 Gradient Boosting

The fundamental idea behind Gradient Boosting is to use the loss function of the existing model's negative gradient information to train a newly added weak classifier. Next, an additive integration of the learned weak classifier is made with the current model. Same as Random Forest, Gradient Boosting is also an algorithm based on decision trees, but gradient boosting use the method of gradient descent as a reference.

Same as above, the paper uses Bayesian Optimization to tune the hyperparameters in the gradient boosting model. Considering the limited amount of computational resource, and the increased number of hyperparameters compared to Random Forest, the paper made the range of each hyperparameter relatively small.

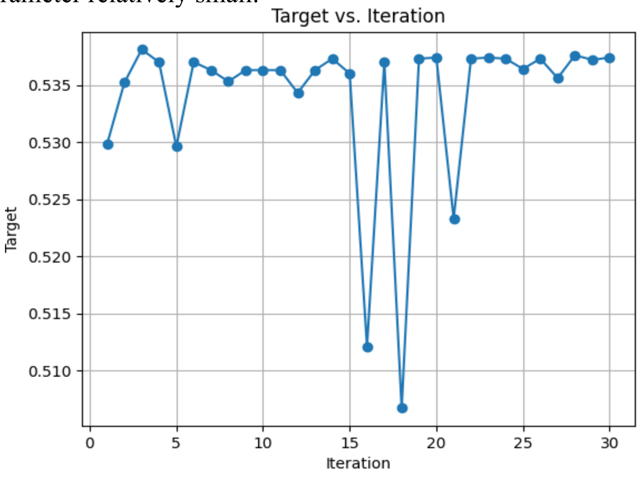

Figure 3. Feature importances in Random Forest soccer match prediction

Figure 3 illustrates the progression of a Bayesian optimization process applied to a Gradient Boosting classification model. The x-axis represents the iterations Bayesian optimization, while the y-axis represents the target value. The target value is measured as the accuracy score.

Table 3. Bayesian Optimization for Hyperparameters in Gradient Boosting

Hyperparameters	Definition	Boundaries	Optimal Value
n_estimator	The number of boosting iterations (or stages). Each iteration adds a new decision tree to the ensemble.	(100, 200)	118
max_depth	The maximum depth of each decision tree added to the ensemble.	(3, 20)	4
min_samples_split	The minimum number of samples required to split an internal node in a newly added tree.	(2, 15)	5
min_samples_leaf	The minimum number of samples required to be at a leaf node in a newly added tree.	(2, 15)	4
learning_rate	A hyperparameter that controls the step size at which the model is updated in each iteration. A lower learning rate often leads to a more stable but potentially slower convergence.	(0.01, 0.5)	0.0160

Table 3 shows the optimal hyperparameter values that were found through Bayesian optimization for the given Gradient Boosting model.

4.6 Support Vector Machine

Support Vector Machine (SVM) is a model for binary classification. The linear classifier with the biggest margin in the feature space, while utilizing a linear kernel, is the definition of its basic model. Put otherwise, the SVM learning technique involves maximizing the margin, which can be converted into a solution for a convex quadratic programming problem. Because this study focuses on the outcomes of soccer matches, which involve a certain degree of randomness, a linear SVM can actually reduce the risk of overfitting and introduce more randomness into the predictions. In the sample space, the hyperplane can be described by the following linear equation:

$$w^T x + b = 0 \tag{10}$$

The mathematical solution to this model can be obtained using the Lagrangian function and the Sequential Minimal Optimization (SMO) algorithm, posed by Platt [12]. To simplify this process, this paper employs the SVC function in Python's scikit-learn library for solving. Similarly, Bayesian optimization was used to tune the hyperparameters of this model.

4.7 Enhance Prediction on Draws

Since the initial models this paper used didn't predict draws very often, or even neglected to report them at all, this paper added a method to enhance draw predictions. The process begins by making predictions on the test set. Then, this paper analyzes the predicted probabilities to identify cases where the model is uncertain about the outcome. This uncertainty is measured by the difference between the probabilities of a home win and an away win. A threshold was introduced to filter out predictions where this probability difference is below the threshold. These predictions are then considered draws.

$$P_{diff} = \left|P_{home} - P_{away}\right| \tag{11}$$

In this context, P_{home} represents the model's predicted probability of a home team victory, while P_{away} denotes the predicted probability of an away team victory. The P_{diff} metric, calculated as the absolute difference between P_{home} and P_{away} serves as a measure of the model's confidence in either outcome. If P_{diff} falls below a predefined threshold, the match is classified as a draw, indicating that the model is essentially equivocal regarding the winner. To optimize the model's sensitivity to draw outcomes, a hyperparameter tuning approach was employed. By iteratively adjusting the threshold and evaluating the model's performance on a held-out test set, the optimal threshold value was determined.

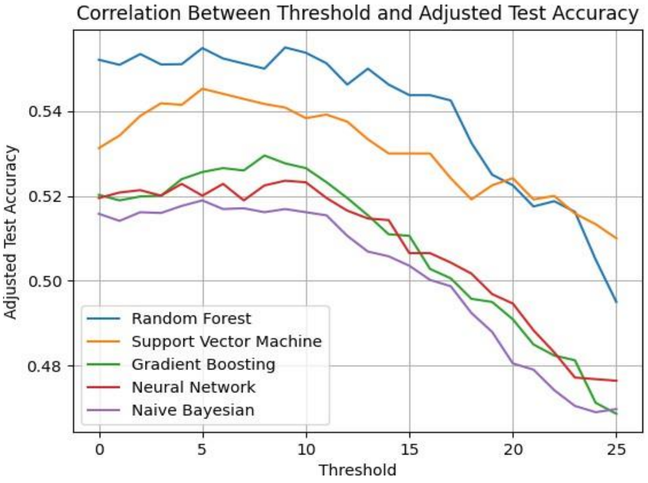

Figure 4. The correlation between threshold and the adjusted accuracy in models

Figure 4 shows the correlation between threshold and adjustment accuracy in the model Across various model iterations, the optimal threshold consistently fell within the range of 0.07 to 0.09. Consequently, a mean value of 0.08 was adopted as the final threshold for subsequent predictive modeling.

5 Results and Discussion

5.1 Comparing Classification Reports

Table 4. Classification Report of Naive Bayesian

	precision	recall	f1-score	support
A	0.52	0.51	0.52	820
D	0.44	0.01	0.01	683
H	0.53	0.84	0.65	1190

Table 5. Classification Report of Random Forest

	precision	recall	f1-score	support
A	0.48	0.52	0.50	798
D	**0.47**	0.05	0.09	596
H	**0.61**	0.82	**0.70**	1299

Table 6. Classification Report of Neural Network

	precision	recall	f1-score	support
A	0.55	0.47	0.51	820
D	0.00	0.00	0.00	683
H	0.52	**0.88**	0.66	1190

Table 7. Classification Report of Support Vector Machine

	precision	recall	f1-score	support
A	**0.62**	0.46	**0.53**	859
D	0.38	**0.09**	**0.15**	627

H	0.54	0.85	0.66	1207

Table 8. Classification Report of Gradient Boosting

	precision	recall	f1-Score	support
A	0.53	**0.52**	0.52	820
D	0.33	0.02	0.04	683
H	0.54	0.83	0.65	1190

The offered classification reports provide insight into the effectiveness of various machine learning techniques, including Naive Bayes, Random Forest, Neural Network, Support Vector Machine, and Gradient Boosting, when applied to the given dataset. Each report includes support for three classes (Away win, Draw, and Home win), precision, recall, and F1 score.

In all the models, class H yields the highest recall and F1-score, meaning that it has a better classification performance than classes A and D. For example, the Random Forest model has a recall of 0. 82 for class H, while the best recall for classes A and D is only 0. 52 and 0. 05, respectively.

Random forest and SVM are fairly accurate, with the precision and recall of all classes being almost equal. For instance, the F1-score of class D in Random Forest is 0. 09, while in SVM it is 0. 15.

The following is an analysis of Tables 4-8:

• Class A: The precision of each classifier is also presented in the table below, and it can be observed that Random Forest and Support Vector Machine have the highest precision for class A, which means that they are more effective in identifying true positives for this class. The highest precision score for class A is 0. 62 as predicted by Support Vector Machine.

• Class D: All models have low precision, recall and F1-scores for class D. For instance, the recall for class D in Gradient Boosting is only 0.02. This could be because of the fact that it is quite challenging to differentiate between class D and the other classes. However, even when applying the method of enhancing the prediction of draws, the models' performance is not satisfactory.

• Class H: As it can be observed in the results all models have high recall and F1-scores for class H. For instance, the F1-score of class H in Neural Network and Support Vector Machine is 0. 66, while in Random Forest it is 0. 70. This implies that the models are capable of identifying most of the cases belonging to class H.

Figure 5 shows the ROC curves of different methods:

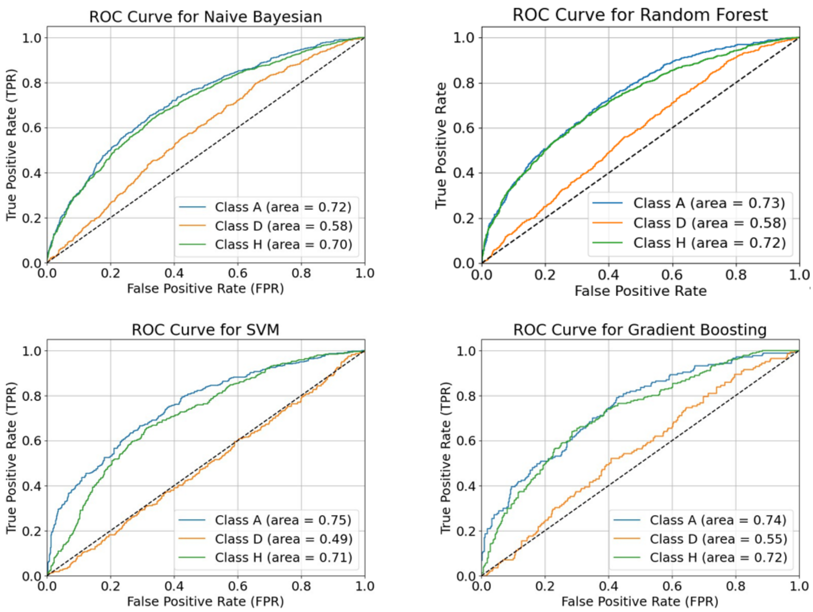

Figure 5. ROC curves of different methods

5.2 Accuracy Score

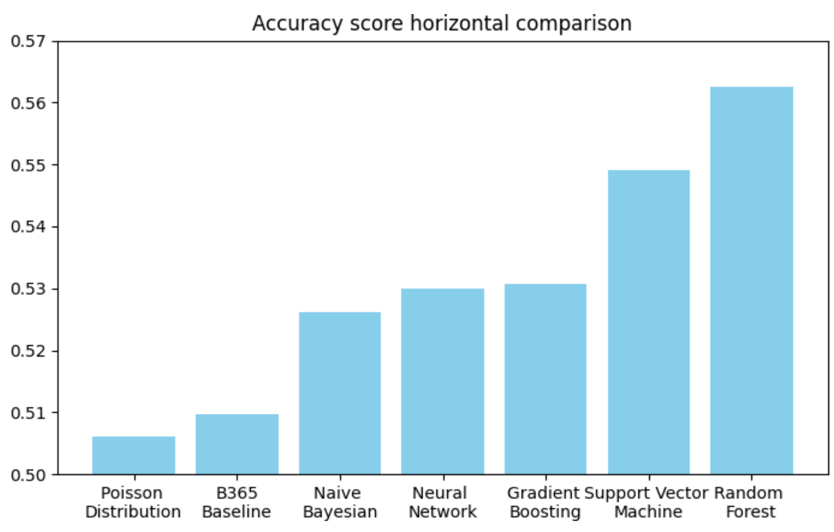

Figure 6. The graph comparing the accuracy scores produced by different models

According to figure 6, Random Forest was found to have the highest accuracy score of 0. 5625 and therefore was deemed to have the ability to capture the intricate relations and interdependencies in soccer game data. However, it is crucial to mention that all the models showed fairly reasonable accuracies, which implies the difficulty of soccer result prediction. Compared to the baseline model, which is based on the betting odds provided by B365 Company, although the bivariate Poisson Distribution does not perform better, all machine learning models have a better accuracy score compared to the Baseline. The best model-Random Forest, has a 10.38% better accuracy compared to baseline.

5.3 The Influence of Poisson Distribution Results on Machine Learning

One advantage of random forest is that it can find the strength of the relationship between each feature and the result. The relationship of this model is shown in the figure 7:

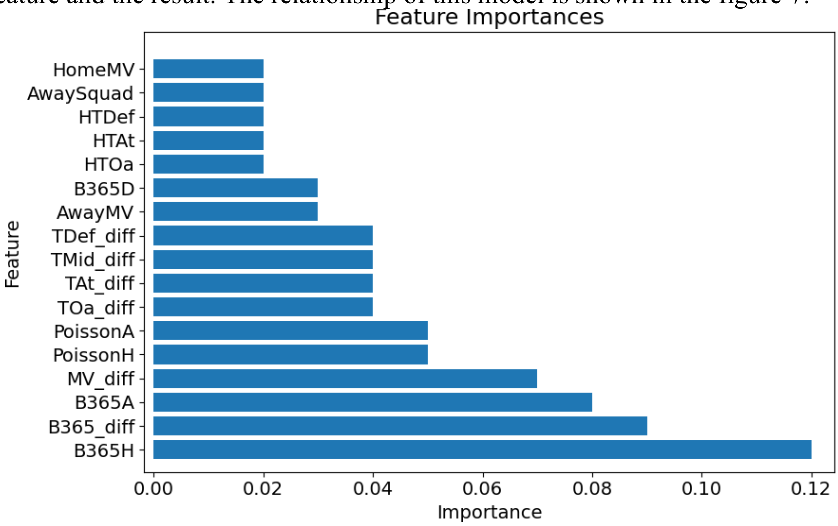

Figure 7. Feature Importance for Soccer Match Prediction

The bar graph indicates the significance of various attributes with respect to the prediction of the outcome of a soccer match. The x-axis is used for feature importance and the y-axis for listing the features. The longer the bar, the more essential the feature is for ascertaining the outcome of the match.

B365H is the feature that has the greatest influence which indicates that there is a high likelihood of the actual event going by home win odds.

MV_{diff} and B365A also play crucial roles with respect to the sequences which indicates that the market value surplus to the opportunities between individuals is also up there along the sequences.

PoissonA and PoissonH, representing the probabilities of the away and home teams winning predicted by the bivariate Poisson Distribution, respectively, consistently rank among the top features. This means that the expected goals obtained from the bivariate Poisson distribution are informative of the possible match outcomes. By including these probabilities within the machine learning model, this paper is able to improve the model's accuracy in betting or analysis.

6 Conclusion

The study constructed a complete framework for the probabilistic modeling of soccer match results based on mathematical and Machine Learning approaches. Using a large number of records and including the necessary features, the system demonstrated satisfied accuracy rates, which were higher than those of a conventional bookmaker. The results were promising, though there remains room for further improvements in the models and a need to investigate additional features that might enhance their accuracy. It would also be worthwhile to explore the possibility of creating profitable betting strategies using the system's predictions, which could serve as a basis for future research.

In terms of strengths, the study offered comprehensive feature engineering, combining historical data, expert knowledge (FIFA ratings), and market information (betting odds) with factors like the Poisson Distribution prediction score. This provided a holistic view of match dynamics. Additionally, the use of multiple models allowed for a more thorough evaluation of prediction abilities, highlighting the strengths and weaknesses of different approaches. A particular focus on draw prediction proved beneficial, leading to improved performance compared to the initial models.

However, certain weaknesses persist. Although the method implemented to improve draw prediction showed promise, predicting draws remains a considerable challenge, as seen in related studies. Moreover, the study's dependence on data availability posed a limitation, where missing or incomplete data could negatively impact model performance.

Looking ahead, future research could delve into more advanced techniques for predicting draws, incorporating additional factors like team playing styles and in-game dynamics. Another key area for future investigation is the profitability analysis of the developed models, specifically assessing their viability in real-world betting scenarios by incorporating strategies like Kelly Criterion for risk management.

References

1. Y. Jiang, Application and Comparison of Multiple Machine Learning Models in Finance, Sci. Program. 2022, 9613554 (2022). https://doi.org/10.1155/2022/9613554
2. K. Kandhway, Dynamic Outcome Prediction of an NBA Match, 2024 18th International Conference on Ubiquitous Information Management and Communication (IMCOM), Kuala Lumpur, Malaysia, 2024, pp. 1-4. doi: 10.1109/IMCOM60618.2024.10418437
3. L. P. Teklenburg, AI-based Classification of American Football Plays Combining Computer Vision and Historical Play-by-play Data, Diss. Technische Hochschule Ingolstadt, 2024. http://nbn-resolving.de/urn:nbn:de:bvb:573-47451
4. R. Stefani, Football and Basketball Predictions Using Least Squares, IEEE Trans. Syst. Man Cybern. 7, 117–121 (1977).
5. S. Yip, Y. Zou, R. T. H. Hung, K. F. C. Yiu, Forecasting Number of Corner Kicks Taken in Association Football Using Compound Poisson Distribution, J. Oper. Res. Soc. 1–11 (2024). https://doi.org/10.1080/01605682.2024.2306170
6. D. Mwembe, L. Sibanda, N. C. Mupondo, Application of a Bivariate Pois- son Model in Devising a Profitable Betting Strategy of the Zimbabwe Premier Soccer League Match Results, Am. J. Theor. Appl. Stat. 4, 99–111 (2015). https://d1wqtxts1xzle7.cloudfront.net/66519749/10.11648.j.ajtas.20150403.15-libre.pdf
7. R. Baboota, H. Kaur, Predictive Analysis and Modelling Football Results Using Machine Learning Approach for English Premier League, Int. J. Forecast. 35 (2018). https://doi.org/10.1016/j.ijforecast.2018.01.003
8. A. Joseph, N. Fenton, M. Neil, Predicting Football Results Using Bayesian Nets and Other Machine Learning Techniques, Knowl-Based Syst. 19, 544–553 (2006). https://doi.org/10.1016/j.knosys.2006.04.011
9. L. M. Hvattum, H. Arntzen, Using ELO Ratings for Match Result Pre- diction in Association Football, Int. J. Forecast. 26, 460–470 (2010). https://doi.org/10.48550/arXiv.2109.15046

10. L. Breiman, Random Forests, Mach. Learn. 45, 5–32 (2001). https://doi.org/10.1023/A:1010933404324
11. J. Cai, J. Luo, S. Wang, S. Yang, Feature Selection in Machine Learning: A New Perspective, Neurocomputing 300, 70–79 (2018). https://doi.org/10.1016/j.neucom.2017.11.077
12. J. Platt, Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines, Advances in Kernel Methods-Support Vector Learning. (1998) 208. https://www.microsoft.com/en-us/research/publication/sequential-minimal-optimization-a-fast-algorithm-for-training-support-vector-machines/