2022 FIFA World Cup Winner Predictor

Zachary Anderson
zanderson2@nd.edu
University of Notre Dame
Notre Dame, Indiana, USA

Ashley Armelin
aarmelin@nd.edu
University of Notre Dame
Notre Dame, Indiana, USA

Simran Moolchandaney
smoolcha@nd.edu
University of Notre Dame
Notre Dame, Indiana, USA

Jack Schlehr
jschlehr@nd.edu
University of Notre Dame
Notre Dame, Indiana, USA

Ryan Wachter
rwachte2@nd.edu
University of Notre Dame
Notre Dame, Indiana, USA

ABSTRACT

This is a proposal for a Data Science project in accordance with Notre Dame's CSE-40647. The topic includes developing a methodology to predict the outcome of of an international soccer match and in the FIFA World Cup.

ACM Reference Format:
Zachary Anderson, Ashley Armelin, Simran Moolchandaney, Jack Schlehr, and Ryan Wachter. 2022. 2022 FIFA World Cup Winner Predictor. In *Proceedings of ACM Conference (ACM '22).* ACM, New York, NY, USA, 11 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

The FIFA World Cup, most commonly known as the World Cup, is an international association football competition contested by the senior men's national teams of the members of the Fédération Internationale de Football Association (i.e., FIFA, the International Federation of Association Football), the sport's global governing body. Football is said to have the power to bring people together, regardless of their age, race, gender, culture, or nationality, and that is never truer than at the World Cup. The World Cup's format involves a qualification phase, which takes place over the preceding three years, to determine which teams qualify for the tournament phase. In the tournament phase, 32 teams compete for the title at venues within the host nation(s) over about a month. Not only is the World Cup the most prestigious association football tournament in the world, but also it is the most widely viewed and followed single sporting event in the world. [1]

Given its known precedence in the athletic world and the many different factors that can change the final score, coaching staffs have a desire to formulate winning strategies to enable their teams to clinch the World title. In addition, the significant ongoing amounts of monetary betting involved in predicting match outcomes and brackets are what has motivated us to develop the following project: develop a method of predicting the outcome of an international soccer match in the FIFA World Cup using the past statistics of both playing countries. For each match, features include:

- Home team
- Away team
- Current team FIFA rank
- Neutral location
- Mean team FIFA Rank past four years
- Number of goals scored per team in a match
- Mean number of goals scored by the team in a match in the last four years
- Q1 of the number of goals scored by the team in a match in the last four years
- Q2 (i.e., median) of the number of goals scored by the team in a match in the last four years
- Q3 of the number of goals scored by the team in a match in the last four years
- Standard deviation of the number of goals scored by the team in a match in the last four years
- Variance of the number of goals scored by the team in a match in the last four years
- Minimum number of goals scored by the team in a match in the last four years
- Maximum number of goals scored by the team in a match in the last four years
- Interquartile range (i.e., IQR) of the number of goals scored by the team in a match in the last four years
- Total world cup games played
- Winning percentage for the team in the past four years
- Percent of draws for the team in the past four years
- Percent losses for the team in the past four years
- Difference between Home and Away team FIFA ranks

Where the following statistical measures represent:

- Min: The minimum value in a set of values, excluding any outliers
- Max: The maximum value in a set of values, excluding any outliers
- Mean: The mathematical average of a set of two or more numbers; found by adding all numbers in the data set and then dividing by the number of values in the set
- Median: The middle number in a sequence of numbers
- Standard Deviation: Measure of the amount of variation or dispersion of a set of values; where a low standard deviation indicates that the values tend to be close to the mean of the

ACM '22, June 03–05, 2022, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

set, while a high standard deviation indicates that the values are spread out over a wider range
- Variance: Average squared deviations from the mean used to measure the spread between numbers in a data set
- Quartiles and IQR: Q1, Q2, Q3 are the value under which 25%, 50%, and 75% of the data points are found when arranged in increasing order. The interquartile range is the difference between the upper and lower quartiles (i.e., IQR = Q3-Q1)

Our assumption is that these statistics together will serve as features that will produce data objects that contain useful knowledge that can be used to predict the team that will win any given match. That is, we will be able to predict the label of a given match in the tournament which will be Win, Loss, or Draw for both the Home and Away teams. We plan to build a supervised classification machine learning model using the existing data to make the corresponding predictions.

We scraped the match and team statistics from online databases to create our dataset, which was preprocessed for encoding, data cleaning, and integration purposes. Using the preprocessed data, several different modeling algorithms were employed, evaluated, and validated, which led us to our best and final model: KNN with K = 10.

2 RELATED WORK

The problem of predicting the outcome of a sports match is not one that has been unheard of. It's been an area of growing interest given (1) the desire to formulate strategies needed to win matches, and (2) due to the large monetary amounts involved in betting [2]. Hence, prediction models developed for the 2018 FIFA World Cup by Groll et. al. implemented ranking methods that estimated adequate ability parameters that reflect the current strength of the teams best. Furthermore, the authors showed that by combining the Random Forest with the team ability parameters from the ranking methods as an additional covariate can improve the predictive power substantially [3]. Moreover, Cambridge Intelligence has developed a Graph Model that'll make a prediction on who's going to win based on the quality of the teams using only the shape of the network they make with other teams and clubs [4]. Finally, TGM Research has developed an Ensemble combining Logistic Regression, Random Forest, and SVM, with FIFA ranks as the main input, to predict the World Cup results [5]. These models and conducted research reveal the precedence of predicting the outcome of a FIFA match.

3 PROBLEM DEFINITION

Given two countries playing a match in the FIFA World Cup, what will the outcome of the match be (i.e., which country will win/lose, or will there be a draw)? How does the model we create compare to the real-life outcomes of those matches?

4 SOLUTION AND METHODS

4.1 Baseline Approach

This problem is a classification problem, so the baseline non-ML approach would be to perform a random guess on the label for each country involved in a match should be. Since there are three possible labels - Win, Loss, Draw - for the outcome of each match, there is a 1/3rd chance of randomly guessing the correct label. However, given the unbalanced label distribution in the test dataset and knowing that for the Home team a Win or a Loss is more likely to occur than a Draw (as can be visualized in the label breakdown in Figure 1, a randomized approach was taken to calculate the baseline performance of a correct prediction. The chance of each instance being classified as a Win, Loss, or Draw was proportional to the percentage of each class in the dataset. The F1 Micro calculated from this randomized baseline was 0.3853, and the F1 Macro was 0.3328 from the baseline model. These baseline performance metrics were used as our baseline to compare our models to in our analysis. Moreover, as will be further explained in section 6.1, F1 Macro will be the prioritized evaluation metric used when performing such comparison.

4.2 Machine Learning Approaches

The team has implemented several machine learning models in an attempt to find the optimal method of determining the outcome of a FIFA World Cup match, which would perform better than the baseline approach. All of these were implemented in Scikit Learn. These models include:

4.2.1 K-Nearest Neighbors (KNN). Finds the K training instances that are closest to the test instance (where closeness can be defined by any distance metric, which is usually the Euclidean distance), and classifies the test instance as the majority class of the K nearest training instances.

4.2.2 Random Forest. Ensemble classifier method based on bagging that consists of many decision trees and outputs the class that is the mode of the class's output by individual trees, each of which is trained on a random subset of the features in the feature space. Each decision tree is a flowchart-like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label, with the hope that objects in the same node will have similar labels.

4.2.3 Multilayer Perceptron (MLP). Is a supplement of feed-forward neural network. It consists of three types of layers—the input layer, output layer, and hidden layer (which consists of an affine transformation followed by a nonlinear activation function). In simpler terms, the input layer receives the input signal to be processed; the required task is performed by the output layer, and an arbitrary number of hidden layers that are placed in between the input and output layers are the true computational engine of the MLP.

4.2.4 Support Vector Machine (SVM). Is a discriminative classifier formally defined by a separating hyperplane, which given labeled training data, outputs an optimal hyperplane that categorizes new examples.

5 DATA AND EXPERIMENT SETTINGS

5.1 Data Collection

The first task in attempting to solve this problem was collecting the raw data that would later be used to make the model. We are using a Kaggle dataset [6] which provides a complete overview of all international soccer matches played since the 90s, including

the Home and Away teams and their respective scores. Moreover, the strength of each team is provided by incorporating actual FIFA rankings as well as player strengths based on the EA Sports FIFA video game. There are 23,992 games in the data set covering games from August 1993 to July 2022. The dataset resulted in a dataframe containing the following information for each soccer match played:

- Game date
- Home team
- Away team
- Home team continent
- Away team continent
- Home team FIFA rank
- Away team FIFA rank
- Home team total FIFA points
- Away team total FIFA points
- Home team score
- Away team score
- Tournament
- City
- Country
- Shoot-out
- Home team result
- Home team goalkeeper score
- Away team goalkeeper score
- Home team mean defense score
- Neutral location
- Home team mean offense score
- Home team mean midfield score
- Away team mean defense score
- Away team mean offense score
- Away team mean midfield score

5.2 Data Objects

Each data object represents an international soccer match between two national soccer teams. Each object consists of the features described in Section 1 which were obtained from those listed above. The label for each object was determined by the result only for the Home team in each match. Since it is not based on both teams, the number of wins does not necessarily equal the number of losses. To determine the label, we subtract the Away team score from the Home team score to determine the result of the match: a positive differential representing a Win, a negative differential representing a Loss, and a 0 differential representing a Draw. When looking at the results from each match, we noticed a class imbalance between the Win, Loss, and Draw labels. Out of the 21,385 total matches used for the models, there were 10,449 wins (roughly 49% of all games), 4,846 draws (23% of all games), and 6,090 losses (28% of all games). This class imbalance most likely occurred because the result of all the matches was based on the Home team, pointing to the potential of a "Home-team advantage" (which was incorporated to our analysis by creating a feature that represented the FIFA Rank difference between the Home and Away teams) that is apparent in our dataset. Figure 1 depicts the label distribution for the Home team and the described class imbalance.

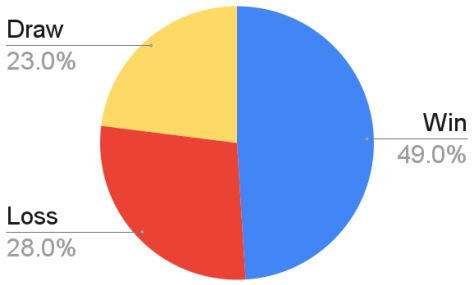

Figure 1: Label distribution for Home team

5.3 Data Preprocessing

5.3.1 Encoding. The team names were encoded using a label encoder such that they could be accepted by the ML models we plan on using. The label encoder assigns a unique integer (which ranges from 0 to the total number of teams - 1) to each team in the dataset which will then be used for reference in the models. We also encoded the labels of Draw, Loss, and Win into 0, 1, and 2 respectively.

5.3.2 Missingness. When scanning the initial dataframe, we encountered missing data, i.e., lacking attribute values, lacking certain attributes of interest, or containing only aggregate data. As we know there are several reasons for which data might be missing: equipment malfunction, inconsistency among other recordings, among others. In our case, some teams simply did not have records for very old matches. Hence, the missing attribute values were automatically filled with the attribute mean of the samples belonging to the category.

5.3.3 Reduction. In performing dimensionality reduction, we removed features that provided no useful information to solving our problem such as the date of the game, the continent of each team, and the city in which it was played. We also removed some sparse features such as the EA FIFA score ratings, as this feature was only available for higher-ranked teams in games from recent years. This left us with a total of 33 features for each data object. Since some of our calculated features relied on data of the performance of the team for the previous 4 years, we also removed the first four years of matches in the dataset, leaving us with a total of 21,385 matches to run our models on. From there, given the array of features we have, chi-square was used to determine the correlation between the features and the label. Figure 2 displays the min-max normalized correlation between the features and the labels for the Home and Away teams. This correlation analysis will then serve us to reduce the dataset to see if training and testing the model with th emost correlated features only yields more interesting results.

5.4 Experiment Settings

Prior to running the developed ML models, K-fold cross-validation was run on the dataset. K-fold cross-validation randomly partitions the dataset into K mutually exclusive subsets. One of the groups is used as the test set and the rest (K-1) are used as the training set. The model is trained on the training set and scored on the test set. Then the process is repeated until each unique group has been

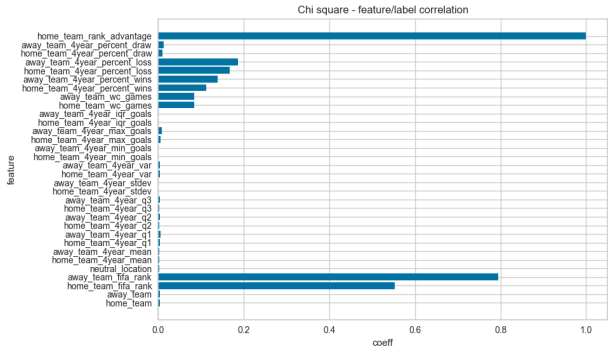

Figure 2: Correlation features and their classification label

used as the test set. This process grants models the opportunity to train on multiple train-test splits, and thus give a better indication of the model's performance when used against unseen data, as well as allows us to retieve average and standard deviation evaluation metrics. Moreover, through K-fold cross-validation, the hyperparameters of the ML models used (i.e., the number of neighbors K in the case of KNN; number of estimators in the case of Random Forest; number of hidden layers, neurons per layer, and actiavtion function in the case of MLP; and the C value in the case of SVM) are tuned in order to determine which ones provide the most accuracy based on the F1 Macro score which will be further explained in Section 6.2. Therefore, 10-fold cross-validation was conducted.

Furthermore, in order to train and test the models the dataset was split into training and test sets, where 80% of the data was used for training and the remaining 20% for testing purposes.

Lastly, a subset of the entire dataset comprised of the 10 most correlated features was also used to run the below models following the same experimental settings just described, in order to study whether the narrowing of features enabled more accurate and/or precise predictions.

The 10 most correlated features correspond to the following:

- home_team_rank_advantage
- away_team_fifa_rank
- home_team_fifa_rank
- away_team_4year_percent_loss
- home_team_4year_percent_loss
- away_team_4year_percent_wins
- home_team_4year_percent_wins
- home_team_wc_games
- away_team_wc_games
- away_team_4year_percent_draw

6 EXPERIMENTAL RESULTS

Now that we had a fully preprocessed dataset, the final step was to actually apply different machine learning models and gauge performance. As mentioned in Section 4.2, all the models we used were Scikit Learn implementations of popular classification machine learning algorithms: KNN, Random Forest, MLP, and SVM.

6.1 Evaluation Methods

Given the multi-label classification problem at hand the following metrics were used when evaluating each model:

- F1 Micro score: Calculate the F1 metric (that is, the harmonic mean of the precision and recall) globally by counting the total true positives, false negatives, and false positives. The average value accounts for the average accuracy, whereas the standard deviation will represent how robust the model is (where smaller standard deviations entail higher robustness).
- F1 Macro score: Calculate the F1 metric (that is, the harmonic mean of the precision and recall) for each label, and find their unweighted mean. This does not take label imbalance into account. The average value accounts for the average accuracy, whereas the standard deviation will represent how robust the model is (where smaller standard deviations entail higher robustness).

F1 Micro often doesn't return an objective measure of model performance when the classes are imbalanced, whilst F1 Macro is able to do so. Therefore, given the class imbalance present in the labels in our training set, we'll prioritize the F1 Macro score when comparing model performance as well as tuning the model hyperparameters.

6.2 K-Nearest Neighbors

6.2.1 All features. Evaluating a 10-fold cross-validation for our KNN model to determine the most tuned hyperparameter yielded a K value of 10, which resulted in the best classification performance in terms of F1 Macro. Figures 3 and 4 show how the average F1 Micro and standard deviation of F1 Micro changed in terms of K, and how the average F1 Macro and standard deviation of F1 Macro changed in terms of K (respectively).

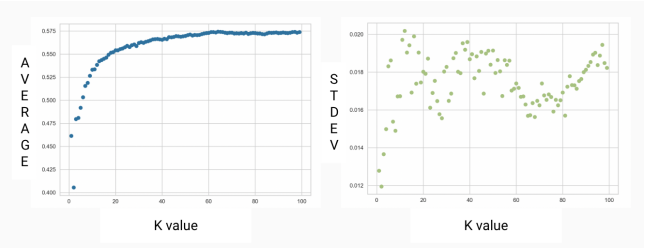

Figure 3: F1 Micro average and stdev vs. K

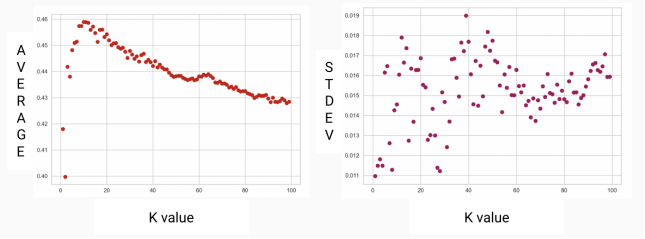

Figure 4: F1 Macro average and stdev vs. K

Table 1 shows the confusion matrix of this model for a singular fold. Figures 5 and 6 show the ROC and Precision-Recall curves respectively.

	DRAW	LOSS	WIN
DRAW	96	156	314
LOSS	86	344	260
WIN	125	189	822

Table 1: KNN (K = 10) confusion matrix

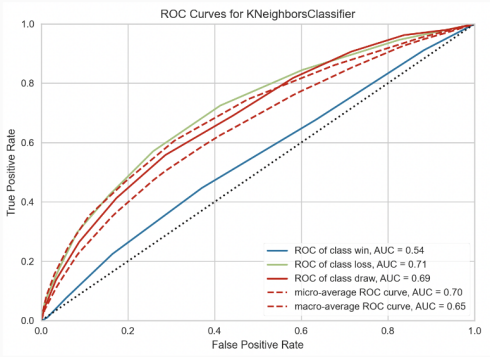

Figure 5: ROC curve for KNN (K = 10)

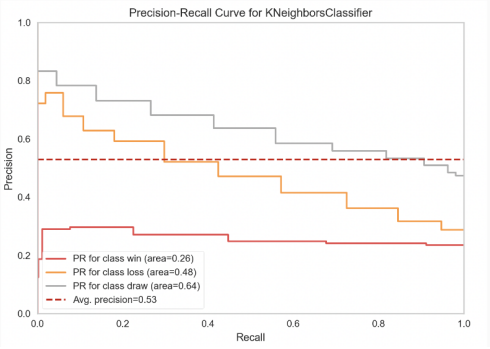

Figure 6: Precision Recall curve for KNN (K = 10)

6.2.2 10 most correlated features. Evaluating a 10-fold cross-validation for our KNN model (only considering the 10 most correlated features determined in section 5.3.3) to determine the most tuned hyperparameter yielded a K value of 14, which resulted in the best classification performance in terms of F1 Macro. Figures 7 and 8 show how the average F1 Micro and standard deviation of F1 Micro changed in terms of K, and how the average F1 Macro and standard deviation of F1 Macro changed in terms of K (respectively).

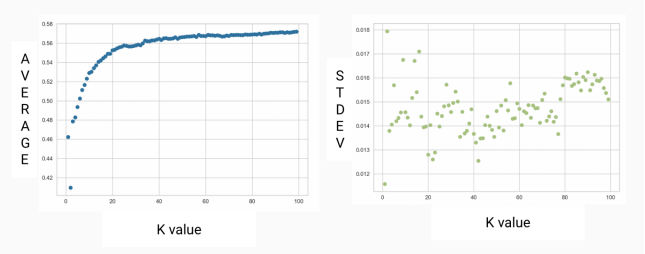

Figure 7: F1 Micro average and stdev vs. K

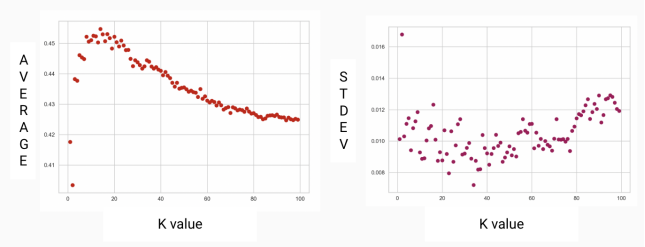

Figure 8: F1 Macro average and stdev vs. K

Table 2 shows its confusion matrix. Figures 9 and 10 show the ROC and Precision-Recall curves respectively.

	DRAW	LOSS	WIN
DRAW	83	163	320
LOSS	101	338	251
WIN	92	177	867

Table 2: KNN (K = 14) confusion matrix

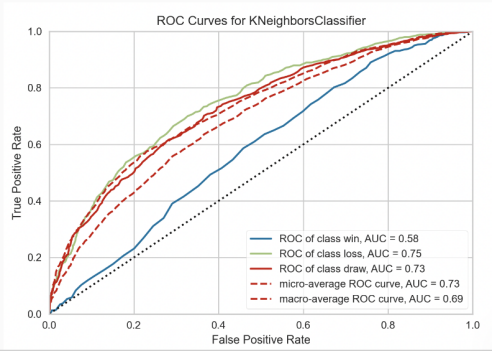

Figure 9: ROC curve for KNN (K = 14)

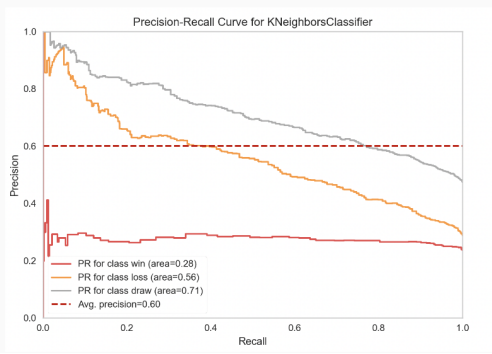

Figure 10: Precision Recall curve for KNN (K = 14)

Comparing the performance of both KNN models yields the following performance metrics recorded in Table 3:

Model	F1 Micro	F1 Macro
KNN (K = 10)	0.5332 ± 0.0167	0.4589 ± 0.0146
KNN (K = 14 - reduced)	0.5406 ± 0.0167	0.4548 ± 0.0108

Table 3: KNN Model comparison

Comparing the F1 Macro scores displayed in Table 3, we can conclude that the KNN model performs better if all 33 features are used.

6.3 Random Forest

6.3.1 All features. Evaluating a 10-fold cross-validation for our Random Forest classifier to determine the most tuned hyperparameter yielded a number of estimators (N) equal to 21, which resulted in the best classification performance in terms of F1 Macro. Figures 11 and 12 show how the average F1 Micro and standard deviation of F1 Micro changed in terms of N, and how the average F1 Macro and standard deviation of F1 Macro changed in terms of N (respectively).

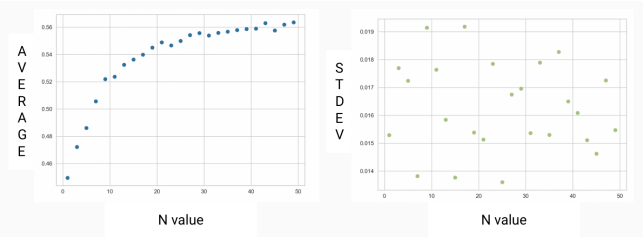

Figure 11: F1 Micro average and stdev vs. N

Table 4 shows its confusion matrix. Figures 13 and 14 show the ROC and Precision-Recall curves respectively.

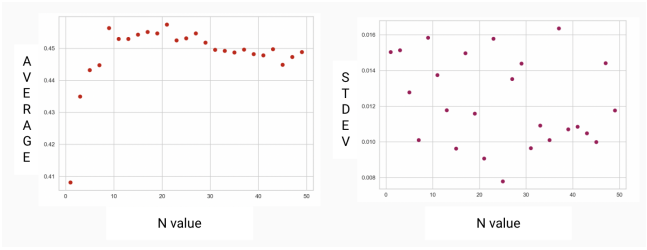

Figure 12: F1 Macro average and stdev vs. N

	DRAW	LOSS	WIN
DRAW	67	156	343
LOSS	67	354	269
WIN	92	164	880

Table 4: Random Forest (N = 21) confusion matrix

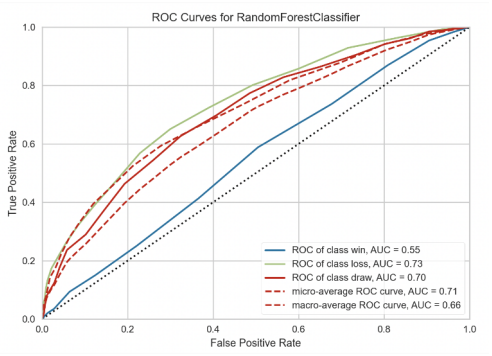

Figure 13: ROC curve for Random Forest (N = 21)

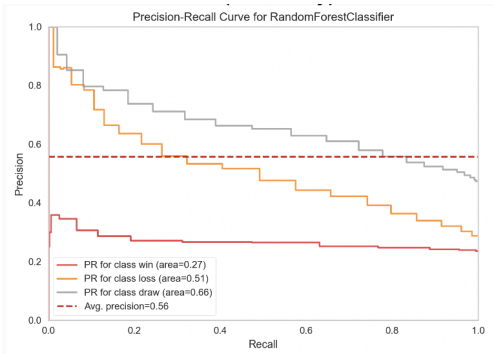

Figure 14: Precision Recall curve for Random Forest (N = 21)

6.3.2 10 most correlated features. Evaluating a 10-fold-cross-validation for our Random Forest classifier (only considering the 10 most correlated features determined in section 5.3.3) to determine the most

tuned hyperparameter yielded a number of estimators equal to 13, which resulted in the best classification performance in terms of F1 Macro. Figures 15 and 16 show how the average F1 Micro and standard deviation of F1 Micro changed in terms of N, and how the average F1 Macro and standard deviation of F1 Macro changed in terms of N (respectively).

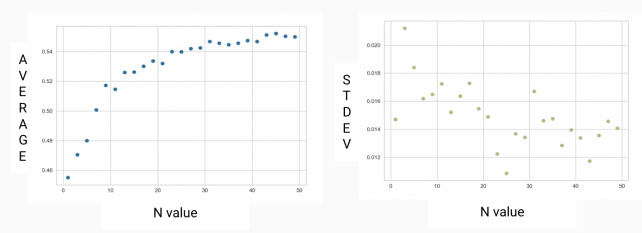

Figure 15: F1 Micro average and stdev vs. N

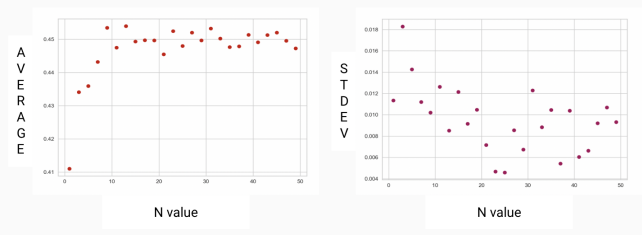

Figure 16: F1 Macro average and stdev vs. N

Table 5 shows its confusion matrix. Figures 17 and 18 show the ROC and Precision-Recall curves respectively.

	DRAW	LOSS	WIN
DRAW	104	155	307
LOSS	120	317	253
WIN	157	184	795

Table 5: Random Forest (N = 13) confusion matrix

Comparing the performance of both Random Forest models yields the following performance metrics recorded in Table 6:

Model	F1 Micro	F1 Macro
Random Forest (N = 21)	0.5488 ± 0.0151	0.4575 ± 0.0091
Random Forest (N = 13 - reduced)	0.5230 ± 0.0152	0.4540 ± 0.0085

Table 6: Random Forest Model comparison

Comparing the F1 Macro scores displayed in Table 6, we can conclude that the Random Forest model performs better if all 33 features are used.

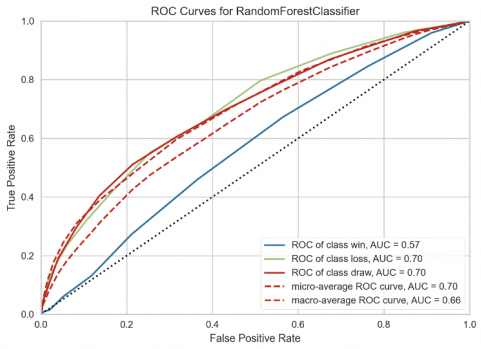

Figure 17: ROC curve for Random Forest (N = 13)

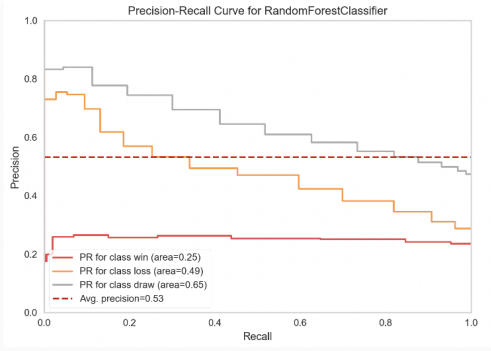

Figure 18: Precision Recall curve for Random Forest (N = 13)

6.4 Multilayer Perceptron

6.4.1 Tuning the hyperparameters of the Multilayer Perceptron. A multilayer perceptron can be tuned in multiple different ways. A few major ways to tune a model are with the number of hidden layers, the number of neurons in each layer, the activation function, the solver for weight optimization, alpha, and the learning rate. We leveraged GridSearchCV in order to search through the combinations of the possible values of each tuning. We found the default values from Scikit Learn gave the best results for the multilayer perceptron. The model has one hidden layer with 100 neurons, alpha value of 0.0001, the learning rate is 10.

6.4.2 All features. Table 7 shows its confusion matrix. Figures 19 and 20 show the ROC and Precision-Recall curves respectively.

	DRAW	LOSS	WIN
DRAW	16	197	306
LOSS	17	472	215
WIN	14	173	983

Table 7: MLP confusion matrix

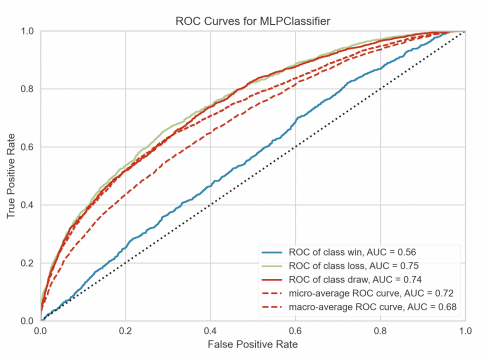

Figure 19: ROC curve for MLP

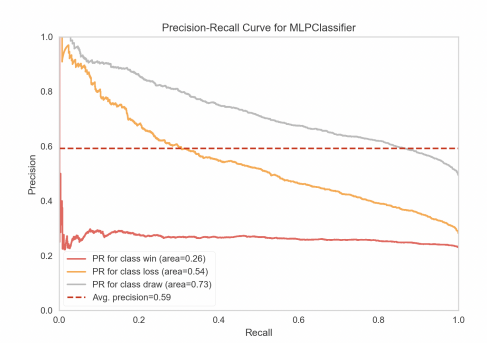

Figure 20: Precision Recall curve for MLP

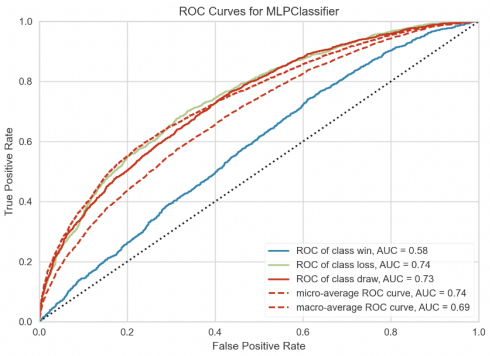

Figure 21: ROC curve for MLP

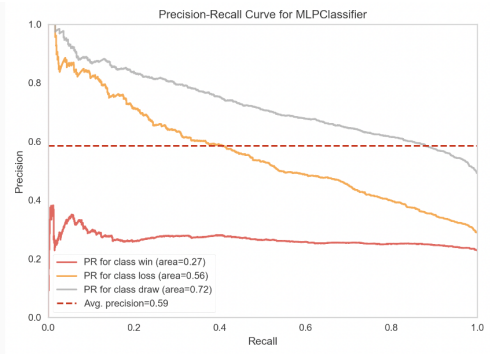

Figure 22: Precision Recall curve for MLP

6.4.3 10 most correlated features. Table 8 shows its confusion matrix only considering the 10 most correlated features determined in section 5.3.3. Figures 21 and 22 show the ROC and Precision-Recall curves respectively.

	DRAW	LOSS	WIN
DRAW	0	178	341
LOSS	0	439	265
WIN	0	140	1030

Table 8: MLP confusion matrix

Comparing the performance of both MLP models yields the following performance metrics recorded in Table 9:

Model	F1 Micro	F1 Macro
MLP	0.6173 ± 0.036	0.4432 ± 0.064
MLP (reduced)	0.6124 ± 0.029	0.4534 ± 0.029

Table 9: MLP Model comparison

Comparing the F1 Macro scores displayed in Table 9, we can conclude that the MLP model performs better if only the 10 most correlated features are used.

6.5 Support Vector Machine

6.5.1 Tuning the hyperparameters of the Support Vector Machine. A support vector classification has a few different hyperparameters but one of the main ones is the C value. The C value in the model dictates how large the margin will be when determining the hyperplane. When the size of C is increased there is more of a chance for over-fitting. When the size of C is decreased there is more of a chance for under-fitting. We found that the best C value was 1.0 which is the default for Scikit Learn.

6.5.2 All features. Table 10 shows its confusion matrix. Figures 23 and 24 show the ROC and Precision-Recall curves respectively.

6.5.3 10 most correlated features. Table 11 shows its confusion matrix only considering the 10 most correlated features determined in section 5.3.3. Figures 25 and 26 show the ROC and Precision-Recall curves respectively.

Comparing the performance of both SVM models yields the following performance metrics recorded in Table 12.

Comparing the F1 Macro scores displayed in Table 12, we can conclude that the SVM model performs better if all 33 features are used.

	DRAW	LOSS	WIN
DRAW	0	112	407
LOSS	0	355	349
WIN	0	62	1108

Table 10: SVM confusion matrix

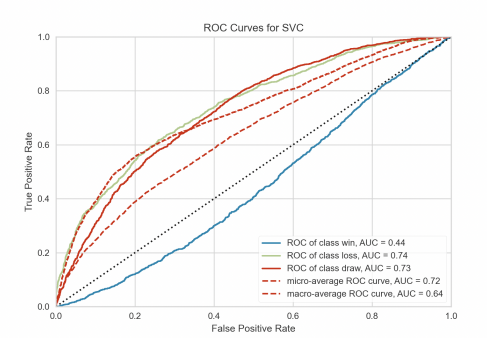

Figure 23: ROC curve for SVM

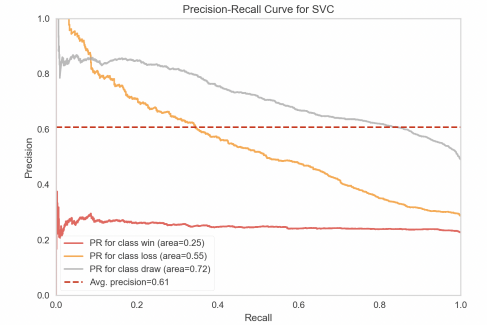

Figure 24: Precision Recall curve for SVM

	DRAW	LOSS	WIN
DRAW	1	115	404
LOSS	0	360	344
WIN	0	71	1099

Table 11: SVM confusion matrix

6.6 Analysis

From Table 13 and Figure 27, we can observe that KNN and Random Forest outperform MLP and SVM in terms of F1 Macro. Moreover,

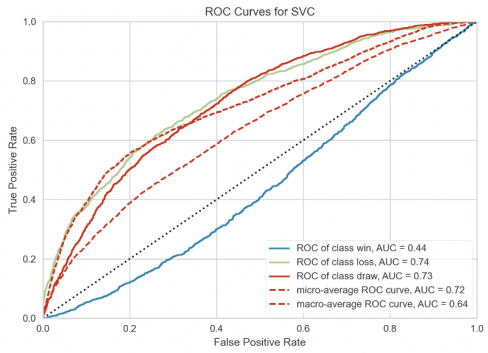

Figure 25: ROC curve for SVM

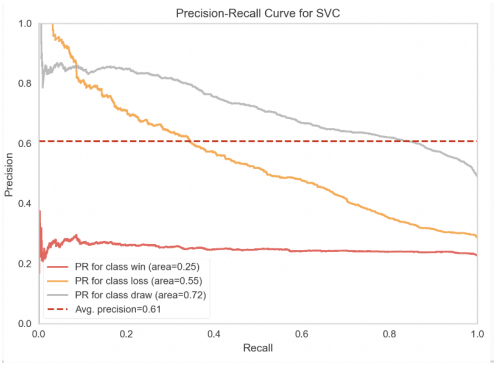

Figure 26: Precision Recall curve for SVM

Model	F1 Micro	F1 Macro
SVM	0.6185 ± 0.017	0.4491 ± 0.013
SVM (reduced)	0.6193 ± 0.016	0.4356 ± 0.013

Table 12: SVM Model comparison

all 4 models classify very few matches to be Draws, with the MLP and SVM doing so the least. Therefore in the following section, we'll be conducting an in-depth analysis of the KNN and Random Forest Performance.

6.6.1 KNN. As seen in figures 3 and 4, once the K value went over 14, there was a decrease in F1 Macro, but still an increase in the F1 Micro. This is largely due to the fact that there is a class imbalance in our data set. As the K value increased, there was a much higher rate of the model falsely predicting games to be home team wins and the number of draws predicted was very low. This was part of our decision to choose to select the best K value based on the F1 Macro, which provides equal weights to each class instead of equal weight to each data object as in F1 Micro. A similar phenomenon happened in the case of the reduced features as seen in Figures 7 and 8, where the F1 Micro increases first sharply and then flattens

out as the k value increases and the F1 Macro has a sharp incline and then declines.

Ultimately reducing the number of features didn't prove to create much change in terms of performance as seen in Table 3. The F1 Micro value for the case where all of the features were used had an accuracy averaging 0.5332 with a standard deviation of 0.0167, and the model with reduced features had an F1 Micro averaging 0.5406 with a standard deviation of 0.0167. Both measures were within one standard deviation of one another and therefore pretty similar. The same is true for F1 Macro where the model with all features having an average score of 0.4589 with a standard deviation of 0.0146, and the model with reduced features having a score of 0.4548 with a standard deviation of 0.0108. Again both scores are within one standard deviation of one another.

6.6.2 Random Forest. As seen in Figures 11, 12, 14, and 15, our Random Forest models do not seem to experience the same overfitting when increasing the number of estimators as the KNN model saw as K increased. This is due to the fact that as more trees are added to the forest, a greater ensemble is created and because of the random aspect of the model, no overfitting should occur.

There was a big difference in performance between the model with reduced features and the model containing all of them for the F1 Micro score, where the model with all of the features had a score of 0.5488 with a standard deviation of 0.0151 and the model with reduced features having an average F1 Micro of 0.5230 with a standard deviation of 0.0152. They are not within one standard deviation of another and the model with all of the features performed better in terms of F1 Micro. In terms of F1 Macro, both models are comparable with the model with all features and the model with reduced features having average scores of 0.4585 and 0.4540 respectively, and standard deviations of 0.0091 and 0.0085 respectively.

Model	F1 Micro	F1 Macro
KNN (K = 10)	0.5332 ± 0.0167	0.4589 ± 0.0146
KNN (K = 14 - reduced)	0.5406 ± 0.0167	0.4548 ± 0.0108
Random Forest (N = 21)	0.5488 ± 0.0151	0.4575 ± 0.0091
Random Forest (N = 13 - reduced)	0.5230 ± 0.0152	0.4540 ± 0.0085
MLP	0.6173 ± 0.036	0.4432 ± 0.064
MLP (reduced)	0.6124 ± 0.029	0.4534 ± 0.029
SVM	0.6185 ± 0.017	0.4491 ± 0.013
SVM (reduced)	0.6193 ± 0.016	0.4356 ± 0.013

Table 13: Evaluation Metrics by Model

7 DISCUSSION

Based on the above analysis the study yielded the best model to be the KNN model with K = 10 when all features are used. Interestingly enough, all models except for the MLP model performed slightly worse on the reduced feature set containing the top 10 most correlated features than on the full feature set, even if their F1 macro scores were pretty close. This leads us to believe that unlike current conceptions of the ranking predicting the match outcome, other variables play a role in determining the outcome of a match.

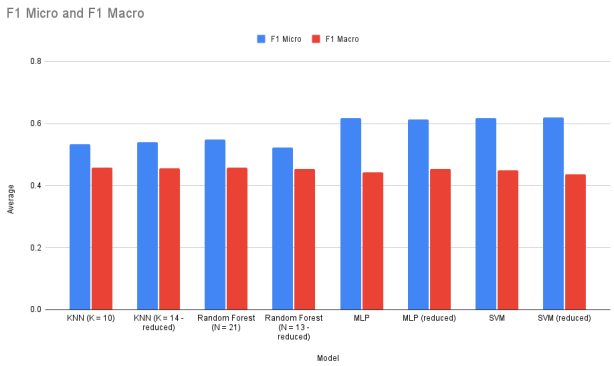

Figure 27: Evaluation Metrics by Model

That being said, all of our models performed significantly better than the baseline model whose F1 Micro score was 0.3852 and whose F1 Macro score was 0.3328. All of our models performed at least 10 percentage points greater than the baseline for both metrics, indicating a significant improvement for all models that we trained. While we never achieved any super drastic results, this increase in F1 Micro and Macro would allow for one to much better predict the outcome of the game to be used to help inform sports betting or for other purposes.

8 FUTURE WORK

If provided more time and resources, there are several strategies that this project could take in the future in order to achieve a higher accuracy in predicting the winner of these highly-touted matches. For starters, there are new FIFA ratings being used to measure the performance of each national team that are updated on a monthly basis. These FIFA points ratings are also now used for the official FIFA World Rankings. Furthermore, there are now ratings made by the EA FIFA video games that measure the performance of the offense, midfield, defense, and goalkeepers of each national team. These could be utilized and adjusted to reflect the overall importance of each part of the team in various soccer match settings (i.e. how goalkeeping might matter more in the FIFA World Cup since draws are not allowed during the knockout stage). With more of these measurements available from future games, these measurements would be dense enough to be included as features for our predictor.

Moreover, given the single winner outcome in the bracket stage of the World Cup, a further consideration would include how to determine the winner of a match that results in a Draw. As we know it is currently decided by tie-breaking methods including overtime and/or penalty shoot-out. Therefore, incorporating a method to deterministically predict the winner of a bracket game would be of great advantage, as currently, we decided what team won the

tie-break by looking at the second most popular label for the Home team.

In addition, our group could also look at exploring other model methods to see if they provide more accurate results. For instance, our group considered using regression models in an attempt to predict the actual score of each of these international matches, which would thus return the winner. We could also use different evaluation metrics such as weighted-average F1 score, which takes into account the class imbalance between the proportion of wins, losses, and draws in the dataset. This weighted-average could then be used as the parameter with which we tuned our model hyperparameters and compare it to our F1 Macro score models to see which one yields the greater accuracy.

9 CONCLUSIONS

All 4 of the models have a better performace when compared to the baseline model's F1 Macro. The KNN model with all original features has the highest accuracy of the three models with an F1 Macro of 0.4589 when K = 10. The Random Forest model has the second best accuracy with an F1 Macro of 0.4575 when N = 21. The SVM model has the third best accuracy with an F1 Macro of 0.4591. These 3 models perform best when all 33 features are used for training and testing purposes. The MLP model has the lowest accuracy with an F1 Macro of 0.4434 when only the 10 most correlated features were used. Moreover, as can be observed throughout the confusion matrices for the different models, draws were the least predicted outcome for matches due to the class imbalance between the three labels (Win, Loss, Draw). After running the models on this year's head-to-head World Cup match ups, we used the results to predict the bracket in Figure 28.

Figure 28: FIFA World Cup Bracket based on KNN (K = 10)

One interesting thing to note is how our model was used to predict whether matches end in a Win, Loss, or Draw. Some major international championship cups use round-robin group stages like the FIFA World Cup which allow for draws. Others use 2-legged aggregate scores to determine the winner, with each team playing a game at home. However, when it comes to the FIFA knockout stages, games that end in a tie go to an overtime period. If this still ends in a Draw, then the teams go to a shoot-out, with the team that makes the most goals out of 5 shots winning. For running our model on the FIFA bracket, there were a few matches that were predicted to end in a Draw in the knockout stages. In each of these cases, we took the next most likely label to fill out the rest of our FIFA bracket. For instance, in the knockout match between Japan and Croatia, our model predicted a draw, so we took the next most likely result predicted by the model, which was Japan winning. It is worth mentioning this match itself did end in a draw after 120 minutes of play, but Croatia emerged victorious in the shoot-out.

Our bracket predicted the Netherlands to defeat England in the championship. The bracket initially performed well, predicting 6 out of the 8 teams to correctly reach the quarterfinal round. The 2 teams mispredicted in this round won games that were decided by a penalty shoot-out. However, the bracket did not correctly predict any of the teams that reached the semifinals, as this was after a contentious quarterfinals in which 2 matches were decided by a shoot-out and the other 2 were decided by a single goal.

10 REFERENCES

[1] https://en.wikipedia.org/wiki/FIFA_World_Cup
[2] R. Bunker, F. Thabtah (2019) A machine learning framework for sport result prediction. https://doi.org/10.1016/j.aci. 2017.09.005.
[3] A. Groll, C. Ley, G. Schauberger, H. Eetvelde (2018) Prediction of the FIFA World Cup 2018 - A random forest approach with an emphasis on estimated team ability parameters. https://doi.org/10.48550 /arXiv.1806.03208 [4] https://cambridge-intelligence.com/fifa-world-cup-2022-prediction/
[5] https://tgmresearch.com/predicting-fifa-world-cup-2022.html
[6] https://www.kaggle.com/datasets/ brenda89/FIFA-World-Cup-2022