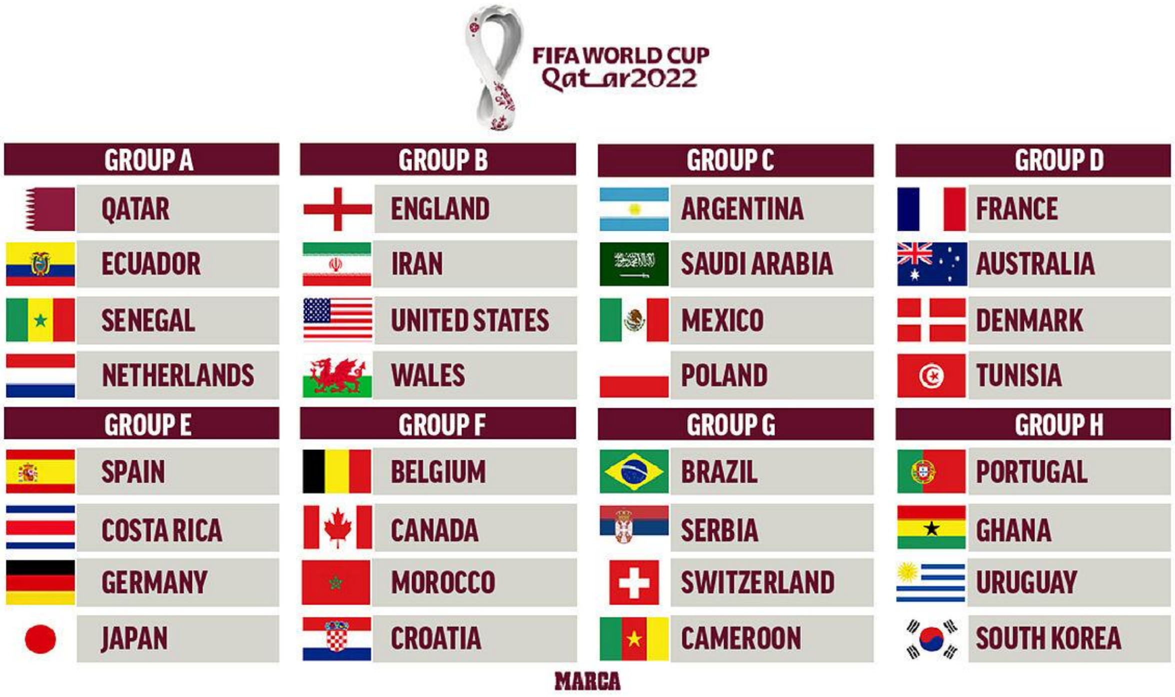
FIFA World Cup Qatar Predictions

By: Kian Golestaneh, Sean Slavich, Taehyung (Tyler) Kim, Hee Soo (Amy) Kim


FIFA WORLD CUP
Qatar2022
MARCA

GROUP A	GROUP B	GROUP C	GROUP D
QATAR	ENGLAND	ARGENTINA	FRANCE
ECUADOR	IRAN	SAUDI ARABIA	AUSTRALIA
SENEGAL	UNITED STATES	MEXICO	DENMARK
NETHERLANDS	WALES	POLAND	TUNISIA

GROUP E	GROUP F	GROUP G	GROUP H
SPAIN	BELGIUM	BRAZIL	PORTUGAL
COSTA RICA	CANADA	SERBIA	GHANA
GERMANY	MOROCCO	SWITZERLAND	URUGUAY
JAPAN	CROATIA	CAMEROON	SOUTH KOREA

Motivation

In light of the World Cup happening right now, our team felt that being able to predict World Cup results would be an interesting task to both test our machine learning modeling skills and something fun to be able to verify our results as we continue to watch this year's world cup unfold. The main problem we are trying to solve is to see how we can most accurately predict matches that happen in the group stage and knockout stage of the World Cup. There is an abundance of data available online on a range of granularity including international match-to-match data, team data, and player data.

Our main task was to decide which type of data would be best for accomplishing the task of predicting the outcome of matches. Our predictions will ideally be in the format of a probability for a given match of whether each team will win, lose, or draw. These probabilities will then be fed into a simulation function to be able to choose a winner (or draw for group stages) for each match. We will also ideally incorporate some form of random noise to account for upsets that can occur when an underdog team beats one of the giants.

Data

Data Sources:

Our data came from multiple sources as we aimed to incorporate as many game statistics to observe feature importances later in EDA. We ultimately decided that match-to-match data was the best in terms of the granularity of our data. Additionally, based on our expertise on how soccer starting lineups change throughout the years, we decided that filtering our data to be matches from 2018 to 2022 made the most sense.

Our match-to-match data includes data from multiple international leagues including the Asia Cup, Gold Cup, Arab Cup, Confederations Cup, Nations League, Euro Cup, and 2018 World Cup matches and World Cup 2022 qualification matches. We also added international friendlies into match-to-match data. These leagues' data were concatenated together to display all of the international matches from 2018 to the present.

This data includes many game-time statistics (possession, expected goals, corner kicks, fouls, etc) that we believe are important to predict the result of a match. Next, we grabbed data from FIFA rankings which are calculated monthly. We joined the FIFA ranking data to our existing match to match data to incorporate FIFA's ranking (a numerical rank of each country), FIFA total rank points per country, and rank change. Finally, we collected data regarding each country's goalkeeper, defense, midfield, and offense scores which we joined to our match-to-match data to fully understand the different capabilities of each country.

Data Cleaning and Engineering:

In terms of data cleaning, we followed a few procedures. First, some of our data had different names for countries such as “IR Iran” and “Iran.” This was accounted for by normalizing country names to unique names through all the different datasets. Next, we noticed we had some NaN values in our possession data which we felt would be an important feature in the future for predictions. To remedy this, we took an average of each country's overall possession that we had available in the dataset through all different leagues and imputed these values into our missing values for possession. We also found that we did not have we did not have positional scores for Iran, US, South Korea, and Qatar for the year 2022. To address this issue, we decided to impute values from positional scores from previous years as an average for each of those countries.

Next, we worked to engineer new features. We did this with our “time since” feature where we took the exact date of each match and computed how many days since that match. We did this to indicate to the model that more recent games have more influence on the outcome since starting lineups and performance changes occur very frequently in soccer.

We also added a “first_country_score” feature as a categorical variable created from the “goal timings” home and away columns which indicate the minute a team scores in a game. The “first_country_score” column displays which country scored the first goal in each game. We also had to create the column “who_won” based on the home team goal count and away team goal count manually and append it to the table as a 1, 0, or -1 signifying a win, draw, or loss.

Feature Selection:

Since we had a large number of features to sift through we decided to start with VIF to observe the multicollinearity of our different features. We started by excluding features that were a VIF above 10 and worked our way down to include 16 features out of our 75 original features. Some features that we initially expected to retain here but did not retain include possession, average goals per match, and previous team points. With our selected features, we decided to run multinomial logistic regression with statsmodel to output the p-values but we noticed that only 5 out of our 16 features had a significant p-value lower than 0.05.

We made a decision that the features that were not significant according to the stats model were in fact very important based on our soccer expertise so we decided to continue with the 16 features we had to run our models. With many of the features that we excluded, (ex: corners) we found that our probabilities would become very skewed and unreasonable when predicting win, loss, and draw.

After training and testing our model with multiple combinations of features, we found that by keeping only positional rankings (ex: goalkeeper score), team rank, and various metrics measuring scoring, our predicted probabilities for all of our models were much more reasonable.

Analytics Models

Model Tuning:

For each model we made sure to set a random state to ensure reproducible results to go back and tune. For both decision tree classification and random forest classification, we ran a grid search cross validation to be able to observe the best parameters to go back and feed into our model fitting. We attempted to run grid search cross validation for SVM but this resulted in the code running for too long for all attempted parameter grid search values so we decided to omit it.

Modeling:

In search of which model would perform the best on our data we decided to run multiple classification models on our data to predict the result of each match and output specific probabilities of each result occurring. The models we decided to use include and their respective accuracy on the test set (sorted by accuracy):

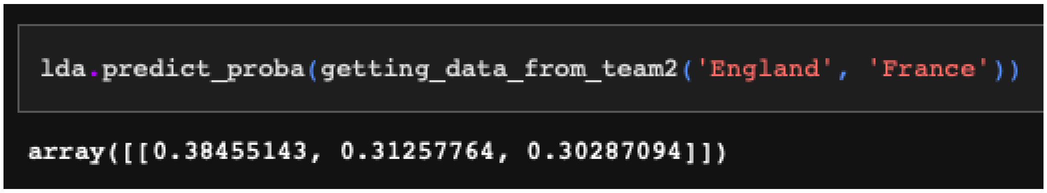
1. Support Vector Machines → 63%
2. Random Forest Classification → 62%
3. LDA → 61%
4. Decision Tree Classification → 60%
5. Logistic Regression → 56%
6. Naive Bayes → 53%
7. Neural Network → 53%
8. Baseline* → 46%

*Baseline model predicts a win for every outcome since this was the most common outcome in the training data

Outcome Prediction Probabilities:

The next step in our project was to use our models to create predictions for each possible match outcome. To achieve this we created a function that takes in two different countries for its input and then outputs a single row of data that includes the

16 features for both countries. To access each country's features as a single row we aggregated our dataset, grouping by country to get the averages of each feature by country. Once we had this aggregated data, we were able to run each model on any match (or potential match) we desired and get the probabilities of a win, loss, or draw. For example:



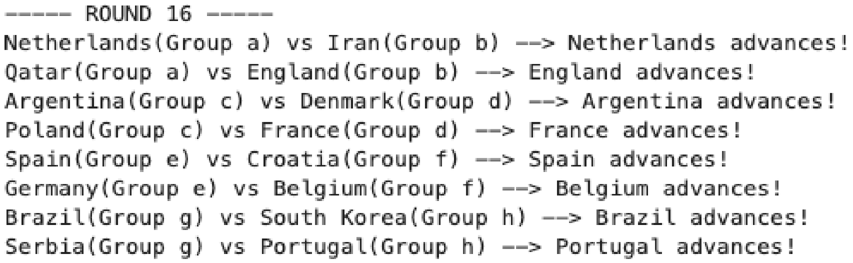
```
lda.predict_proba(getting_data_from_team2('England', 'France'))

array([[0.38455143, 0.31257764, 0.30287094]])
```

Here we are feeding the match between England and France into our LDA model and we are able to observe that France has a 38% chance of winning, there is a 31% chance of a draw, and a 30% chance of England winning.

Simulation:

The final step in our prediction is to incorporate these probabilities into a simulation that is able to choose the outcome of each match based on these probabilities. First, we run our function for simulating the group stage. In this stage the result of the matches can be either a win, loss, or draw. A win gives a team 3 points, a loss gives a team 0 points, and a draw gives a team 1 point. Each of the 4 teams in each group plays each other once. These points are summed at the end of all matches to determine which top two teams advance to the knockout stage. To simulate this, we ran a function that gives the team with the highest win probability for each game 3 points, and if the win probability percentage for each team is within 5% of the draw probability percentage then the match results in a draw (or if the draw probability is outright greater than the win or loss probability).

Once we have the results for the top two teams that will advance from each group then we start the simulations for the knockout stage. In this stage, the result of the game can only be a win or loss. The winner of each game advances to the next round. The simulation function here takes the outright winner by choosing which team has a higher probability of winning that match:



```
----- ROUND 16 -----
Netherlands(Group a) vs Iran(Group b) --> Netherlands advances!
Qatar(Group a) vs England(Group b) --> England advances!
Argentina(Group c) vs Denmark(Group d) --> Argentina advances!
Poland(Group c) vs France(Group d) --> France advances!
Spain(Group e) vs Croatia(Group f) --> Spain advances!
Germany(Group e) vs Belgium(Group f) --> Belgium advances!
Brazil(Group g) vs South Korea(Group h) --> Brazil advances!
Serbia(Group g) vs Portugal(Group h) --> Portugal advances!
```

Results:

- LDA predicts France vs. Belgium in the final with France winning.
- CV Random Forest predicts England vs. Belgium in the final with England winning.
- SVM predicts France vs. Belgium in the final with France winning.

Impact

As mentioned above, the main problem we are trying to solve is to see how we can most accurately predict matches that happen in the group stage and knockout stage of the World Cup. Not only sports fans but also the general public love to make bets on the final outcome, whether it be between friends or in an official sports betting market. Therefore, being able to better predict the game outcome through the use of machine learning is an exciting project that has the potential to impact tons of people.

The scope of the analysis can be expanded in many ways. First, we can further advance our model to take in real-time data as the games go on, so that the model can make real-time adjustments to its winner predictions. This would be an even more interesting model as we can see the impact of an event - such as the first goal - on winning probabilities. We can also attempt to expand the scope of the analysis to different sports and different events such as swimming at the Olympics or basketball. There are many approaches possible in terms of data sources, model selection, feature engineering, and more.

After all, it is impossible to achieve 100% accuracy with a sports model. Sports is a random game by its nature, and that is one of many reasons why fans love the game. Comparing our model outcomes with the real results during the 2022 World Cup season was very interesting, and we will be able to find out the final winner in a few days.

Appendix:

Links to Data Sources:

1. https://footystats.org/international
2. https://www.kaggle.com/code/sslp23/predicting-fifa-2022-world-cup-with-ml/data
3. https://www.kaggle.com/datasets/amineteffal/qatar2022worldcupschudule
4. https://www.kaggle.com/code/mahmoudelhusseni/fifa-world-cup-winner-2022-part1/data

Link to Notebook:

1. https://github.com/seanslavich1/WorldCupPrediction2022/blob/main/142_Project_Final_Draft%20(3).ipynb