Score Predictor

Data Scraping

American University of Armenia

Instructor: Hrant Davtyan

04/08/2020

Mher Vahramyan

Ruben Ghandilyan

table of contents

**ABSTRACT** 3

**INTRODUCTION** 3

**RESEARCH** 3

**DATA COLLECTION** 4

**xG BASED PREDICTION METHOD** 4

**OUR ALGORITHM** 4

coefficients 5

testing 5

**RESULTS** 5

**FUTURE THOUGHTS** 6

**SOURCES** 8

**PROJECT LINK** 8

**ABSTRACT**

This paper explores football score prediction techniques and introduces a new algorithm. Having considered the algorithms and different methods used to predict final scores of the matches, we came up with an idea to create our Algorithm and assess the accuracy that we can get with predicting final scores. We used the ideas of attack score and defense score, which are the main deciding values in our Algorithm. Also, to automate and optimize the process, we introduced coefficients optimized by our Algorithm to obtain better results.

**INTRODUCTION**

Football is the most popular sport on the planet. That already explains the driving force behind the idea of predicting match results. As we know, sports have "passed the line" of being just a hobby. It has become a part of life for many people, so predicting the results of a game has been a challenge that interests many people. Score prediction has never been an easy task and will never be as the factors that affect a football game cannot be easily identified and are highly unpredictable. For example, suppose you consider only the team performance to predict the outcome of the game. In that case, you will have a problem when some of the key players are missing. While getting the information that a player will miss the match is still possible, calculating the impact of a player on the team's performance is a bigger problem. Moreover, there are factors in the game that can hugely impact the result, that are not predictable as getting a red card and playing with fewer players than the opponent. That is why you cannot expect a program to have a very high accuracy rate. Even in predicting the match result (win, lose or draw), getting a 50% accuracy rate is considered very good. We tried to get our predictions done with team performance data and see how we can optimize our Algorithm to get a better accuracy rate.

**RESEARCH**

Football predictions have become very common nowadays with the improvements in machine learning and many big betting companies interested. Obviously, the most accurate and great working algorithms are secret, and you can never know what the actual implementation is in behind. However, there are some notes and known methods to make football predictions. One of them uses Expected Goals (xG), a metric that analyses the quality of the chances to score a goal in a match. Each shot in a game has a value that shows the probability that it will be converted into a goal. For instance, penalty kick has a value of 0.76, which means that there is a 76% chance that it will end with a goal. To be used in predictions, the probabilities can be added up to get how many goals the team should have scored in the match considering its chances. Sometimes, this is a more useful parameter than the number of goals the team scored. Other predicting algorithms use the notions of attack score and defense score to predict the final score. These values represent the attacking force and defensive strength of a team, and it cannot be represented with one value since, in that case, you cannot know the number of goals scored in a match. Also, there is a statistical model based on the Poisson distribution used to analyze the number of goals scored in a match by home and away teams, which shows that in general, teams score more when they are playing at home, so this fact is considered in predicting the score in algorithms. We get a basic idea of how to use the notions found and what data we will need for our predictions (1, 2, 3, 4).

**DATA COLLECTION**

Data collection is one of the most significant parts for prediction algorithms as, without proper data, the prediction is not possible. We needed to get the data of previous season tables to get the initial attack and defense scores that we will base on. We also needed the finished fixtures data of the current season to update the attack and defense scores after each match and the upcoming matches to predict. For this information, we scraped the SoccerStats.com website using Scrapy in Python. Moreover, we decided to have a separate prediction function to guess the score using live xG data for home and away games, so we scraped FootyStats.org using Selenium and Scrapy to get it. After scraping from two different websites, we noticed that there are some data matching problems, so we matched the names to have clean data to work on.

**xG BASED PREDICTION METHOD**

As mentioned above, xG is the number of expected goals, and we have the data for each team. To predict the number of goals scored by the home team, we rounded the average home team scoring xG at their stadium and away team defending xG at the opponent's stadium. In a similar way to predict away team results, we rounded the average of away team home scoring xG and home team away defending xG. This way, we used the xG data calculated by advanced algorithms and considered the difference of away and home performance for each team.

**OUR ALGORITHM**

Our Algorithm starts with initializing defense and attack scores. We use the historical data of the previous season and take the average amount of scored goals in last season matches as the attack score. The average amount of conceded goals in the last season matches the defense score. Each season, some teams get promoted, and others get relegated, so we need to take the data from a lower division and somehow interpret it for the top division (we will talk about it in coefficients). After initializing, the next step is to update the data accordingly after a match is played to keep track of the team's performance and predict the next results accurately. In the end, we predict the scores according to attack and defense values and considering some coefficients.

COEFFICIENTS

We have considered several coefficients to see whether they affect the prediction, or we can ignore them. The considered coefficients are:

* Goal Intervals

These are the numbers that decide what the value generated by attack score, defense score, and other coefficients convert into the exact goals scored. For instance, let our interval be [0, 0.5, 1.6, 3.0]. If our generated value is 0.4, we predict the team to score 0 goals. If it is 1.3, we predict the team to score 1 and so on.

* Game Importance Coeff

This is the coefficient used when updating the attack and defense scores for a team. It decides how the games played should affect the initial scores after every match. The scores are updated based on the predicted result for the game, and the actual game result difference.

* League Difference Coeff

This is the number that decides how big the difference is between high and low divisions. We need it to properly transfer the information about promoting teams to the higher division standards. However, after some testing, we noticed that changing this coefficient only affects the Goal Intervals and does not help with the accuracy, so we decided to fix it to be 0.6.

* Home Game Coeff

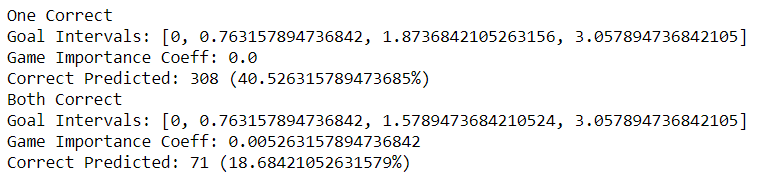
This is the number that decides the advantage given to the home team in a match. It is involved in calculating the final score of the match. However, after testing, we noticed that this coefficient is often 0 in the optimal cases of our Algorithm, so we decided to ignore it.

TESTING

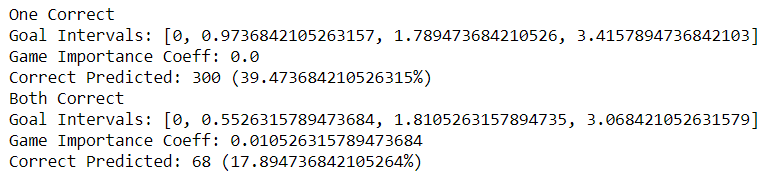
The coefficients we talked about above need to be optimized and chosen by the Algorithm to get better results. For this reason, we created a testing process that would test coefficients' different values and choose the combination that produces the most accurate final score predictions. This process is conducted by taking values from a range of numbers that we optimized, so we created a parameter step for the main function which has a default value, but can be increased for more accurate testing of the coefficients.

**RESULTS**

To assess the accuracy of our Algorithm, we tested in on English Premier League and Italian Serie A. First, we tested the xG method to compare it with our Algorithm and noticed that the predictions are very similar for every match and did not get high accuracy in the results. For instance, for the last five games in Serie A xG method did not predict any of the games correctly, while our Algorithm predicted one of the scores. The problem with testing this method was that the leagues were mostly over and these method does not allow us to get the predictions for already finished games as we get the xG scores in the current state and cannot apply it to the previous games. Finally, to test the accuracy of our Algorithm, we introduced a parameter for our main function called playedMatches that lets as pass the amount of played matches we want in the current season and get the results for that moment. So, after testing for different steps, we got different results.

For instance, for all 380 matches in Serie A 2019/2020 we got this result:****

For English Premier League 2019/2020 season we got this result:



Here we separated the results for two cases. In one case, we consider the optimized coefficients the one that guessed more exact scores right. In the other, we consider the result of each team separately and consider it correct if, for example, the match ended 1-1, but we predicted it to end 2-1. In the second case, we must consider each match twice to get the accuracy for every team correctly.

Observing the results, we can see that the prediction rate for One Correct is greater than 38%, while for Both Correct, it is greater than 17%. These can be considered good results, but there is always a way to advance and improve.

**FUTURE THOUGHTS**

Considering the results and the research we have done, our Algorithm can be improved in the future and produce even better results. We understand that many factors can significantly affect the match score, so trying to include them may give us better results. We also have an idea to use the xG data combined with our Algorithm and test the accuracy. Moreover, there are other outcomes like the result of the match (win, lose or draw) or BTTS (both teams to score) that we can apply our Algorithm for. Finally, this Algorithm can be transferred into an ML model and give more analyzable data.

**SOURCES**

1. Expected Goals (xG): what it is and how it works. (2020, July 30). Retrieved from https://www.driblab.com/analysis-team/expected-goals-xg-what-it-is-and-how-it-works/
2. Predicting Football Results With Statistical Modelling. (2017, June 4). Retrieved from https://dashee87.github.io/football/python/predicting-football-results-with-statistical-modelling/
3. The KickOff Algorithm | Football Predictions, stats, betting and tips - KickOff.co.uk. (2020). Retrieved from https://www.kickoff.co.uk/the-kickoff-algorithm/
4. xG Explained. (2020). Retrieved from <https://fbref.com/en/expected-goals-model-explained/>

**PROJECT LINK**

https://github.com/mhervah/ScorePredictor