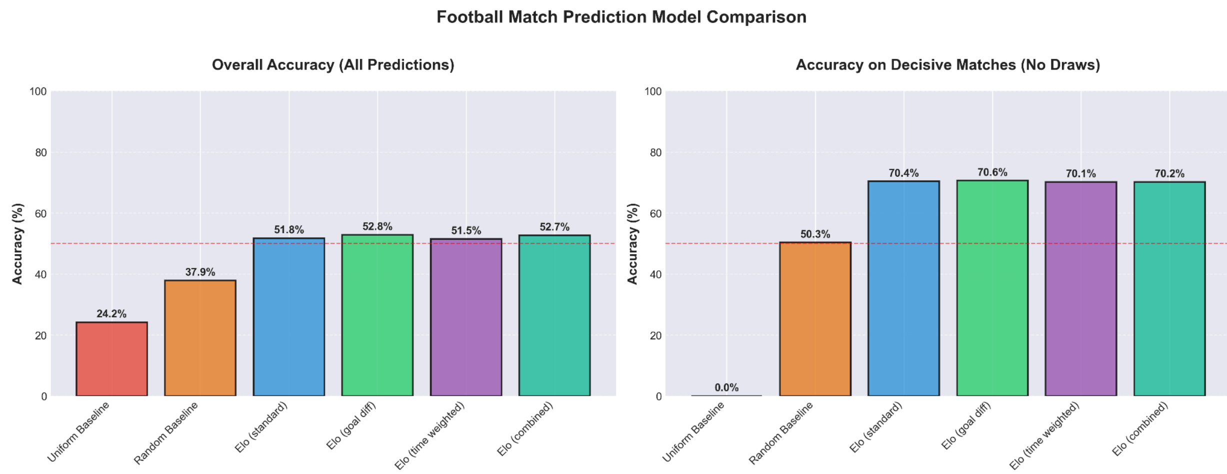
Raghav Punnam, Sara Ziadat

Github Link: https://github.com/RaghavPu/am215-miniproj-1

Abstract

The FIFA World Cup takes place every four years and is the most widely viewed sporting event globally. Given the scale of interest in this tournament, and in betting on its outcomes, there is significant value in developing a reliable model to predict the probability of each team advancing through the various stages of the tournament. To do this, we modify traditional Elo rating calculations to incorporate additional features such as win margin (how decisively a team won). The dataset used includes approximately 23,900 matches played since 1993. The training set was used to tune the Elo model's hyperparameters, and the resulting formula was applied to the validation set. Monte Carlo simulations were then run to estimate the probabilities of each team reaching each round of the tournament. When compared to baseline models using random or uniform probabilities, our approach demonstrates improved predictive performance, suggesting that it may be a more effective tool for forecasting World Cup outcomes.



For our evaluation, we split up our ~30 years of data into sequential groups of 5. So, for each group of 5, we train our models on the first 4 years and then evaluate them in the fifth year. We do this in 6 groups and then average our results. The averages in accuracy are shown in the graphs. Additionally, with soccer there is a nuance that teams can win, lose, or DRAW. So, we evaluated both of these scenarios. As the results show, we can see that the model performs way better when draws are out of question, because it takes away an extra class to predict.

We used multiple versions of ELO in addition to the standard version. We have a goal differential version, which increases the K (uppercase) factor for score updates depending on the severity of the win/loss (win margin). This multiplication factor scales logarithmically as a function of the margin. Additionally, we have a time weighted version, which emphasizes the most recent game updates to account for team growth/development. Finally, we combine both of these as well. Now, we are working on a deep learning solution that could learn the update factor K as a function of many different features (i.e. win margin, goalie score, offense score, home/away etc.) which are also in the dataset. This is an interesting problem because you are learning these factors for updates, but don't necessarily have these features for inference purposes, which would solely depend on the final ratings.