CS229 Final Project

Dec. 10, 2019

Gaining a Statistical Edge in Soccer Prediction using Machine Learning: Role of Meta Statistics in Match Prediction

Varun Harbola[1] [varunh] & Kyuho Lee[1] [kyuho]

Sports match prediction is a huge industry, spanning from pre-/post-match analysis to sports betting and sports management. In predicting the outcome of a match between two teams, statistics such as historical match results between the two teams have been conventionally employed. However, a match outcome is strongly dependent on a much wider spectrum of team statistics and tactics employed during the match. To navigate in this complex feature space, we employed machine learning algorithms on various sets of relevant features for English Premier League (EPL) soccer match prediction. We achieved prediction accuracy of up to 58.89%, surpassing the literature values by ~10%.[1] Interestingly, we observed that expanding the feature space results in overfitting on the training set to improve predictions on draw games, resulting in poorer prediction overall. These results call for further developments in learning algorithms to improve draw game predictions.

Soccer is one of the most popular sports in the world, and, with current advancements in data collection and computing power, it becomes possible to analyze and predict the outcome of matches with reasonable accuracy.[2,3] The use of such analysis has become ubiquitous, not only in the sports betting industry, but also in match analysis, television broadcasting, and team management. Traditionally, the features involved in match prediction have been the past match results between the two competing teams. However, this does not represent the full complexity involved in a soccer match. Team styles, home advantage, along with overall positional strength are only some of many features which can deeply impact the results of a soccer match.[3,4] Handling such high-order feature space becomes limited using traditional analytical approaches. On the other hand, machine learning provides the ideal tools to deduce targeted information from dense feature space. Previous reports on soccer match prediction using neural networks have reported accuracy of up to 53.25%,[1,5] with the choice of features dramatically influencing the prediction accuracy of the algorithms.[1]

In this project, we implemented several different classification techniques for match prediction (win, draw, or loss) in the EPL soccer tournament. Given the nature of match prediction, we generated the models with features which are determined before the match takes place. Examples of such features include overall team rating, positional average of player ratings, and individual player ratings, which have minimal variations within a season. Implementing the models in various sizes of the feature space, we were able to achieve the highest prediction accuracy of 58.89%, which is higher than literature values[1] and significantly better than blind guess (33.33%) or home-team-win guess (46.19%). Meanwhile, due to the closely overlapping distribution of the match outcomes, we observed that draw game prediction is quite challenging for models with lower-dimensional feature space. Increasing the size of the feature space resulted in overfitting on the draw games in the training set, resulting in poorer prediction on the test set. This analysis elucidates the difficulty in match prediction induced by draw games, motivating further improvements in the learning algorithms for draw game predictions.

While the dataset on EPL match results and statistics are easily accessible, a comprehensive dataset including match lineup and individual player statistics is not readily available. Hence, we directly constructed the needed dataset by importing the relevant data from *https://www.whoscored.com*. Using the *selenium* and

[1]Department of Physics, Stanford University, Stanford, CA, USA.

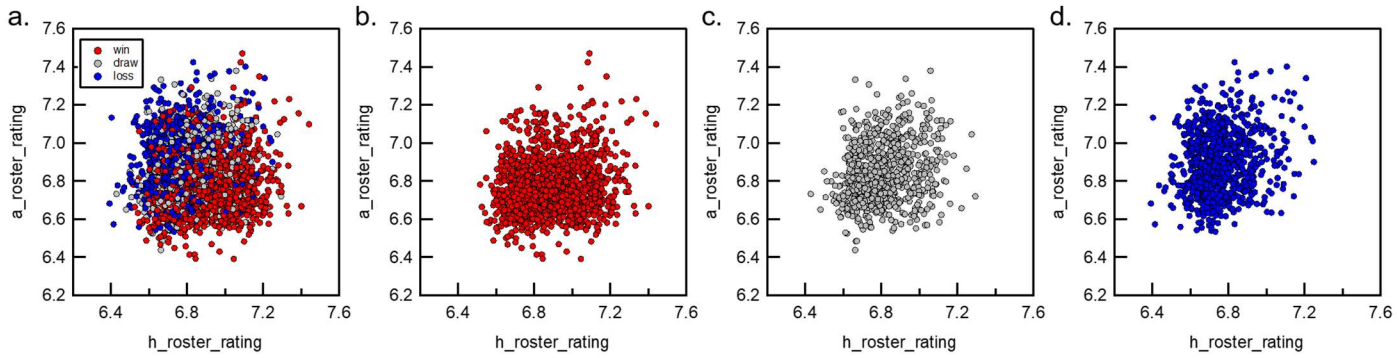

Figure 1. a) Plot of the training data with the home and away average roster ratings on the *x*-axis and the *y*-axis, respectively, along with individual plots of **b**) home win, **c**) draw, and **d**) home loss datapoints.

bs4.BeautifulSoup packages in Python, the data mining was automated to collect the match results, match statistics, match lineup, and individual player statistics of EPL matches from 2009/2010 to 2018/2019 season.

We tested the models with three feature spaces with different dimensionality. The first feature set F1 is the simplest, consisting only of the home and away team roster average ratings. The second feature set F2 is generated by adding the positional (attack, midfield, defense, and goalkeeper) player average ratings on top of the overall team average ratings for home and away teams. The third feature set F3 consists of the overall rating, pass success percentage, passes per game, shots per game, key passes per game, blocks per game, and interceptions per game for all players in the home and away team. By employing these features, which are essentially constant for each season with minimal time dependence, we assume that the model is also time-independent. Namely, given any game between any two teams at any season, the model should generate predictions only based on the given features. With this being the case, we randomly sorted and designated 80% and 20% of the entire dataset (380 matches per season, 3800 matches in total) to training and test sets, respectively.

With three possible match outcomes (win, draw, or loss), or classes, a blind guess on a soccer match would result in 33.33% accuracy. However, surveying over the dataset reveals that the match outcome is nontrivially biased in favor of the home team, with home win, draw, and loss probability being 46.19%, 24.80%, and 29.02% respectively. This implies that blindly guessing home-team-win would result in 46.19% accuracy. The requirement for the models therefore is to surpass the accuracy of at least 46.19%, and hopefully overcome the literature report value of 53.25%.

For the classification algorithms, we implemented Gaussian discriminative algorithm (GDA), support vector machines (SVM) with linear, degree-5 polynomial, and radial basis function (RBF) kernels, SoftMax regression with linear and quadratic features, and neural networks with single hidden layer. Details on these algorithms are discussed in *Methods*.

Before implementing the above learning algorithms, we first inspected the dataset to observe the distribution of the labels and understand the relevant challenges.

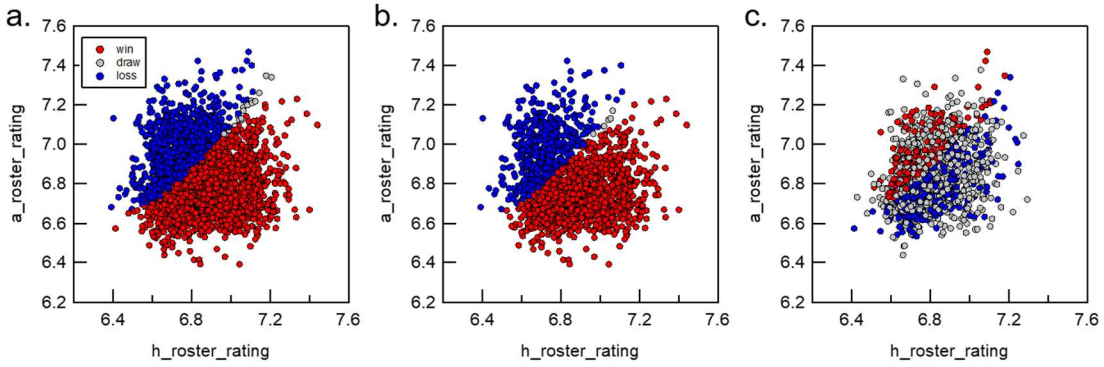

Figure 2. Plots of **a)** total fit, **b)** correctly fitted datapoints, and **c)** incorrectly fitted datapoints of the training set using GDA with feature set F1.

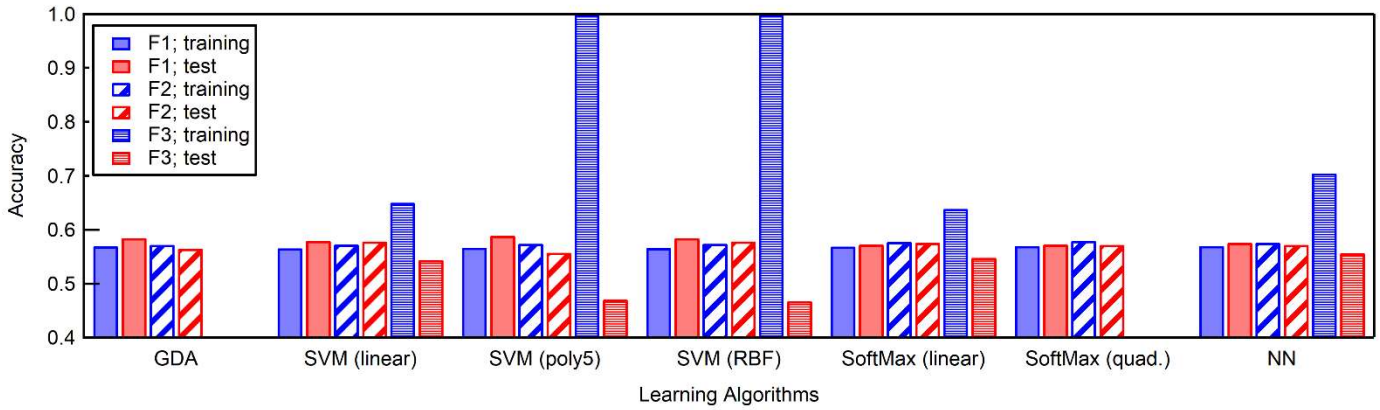

Figure 3. Fit accuracy on training set and prediction accuracy on test set for various learning algorithms, trained on different feature sets. GDA on feature set F3 failed due to singular matrix formation caused by larger feature space than the span of datasets. SoftMax with quadric features on feature set F3 was terminated due to long convergence time.

Figure 1 shows the plot of the training data on the feature space of away *vs.* home average roster rating. While the datapoints are clustered tightly to their respective classes, they are also closely overlapping with each other. Resultantly, difficulties in class identification is expected. This is apparent in the optimized training set fit using GDA, which is shown in Figure 2. The optimized fit essentially misses the entirety of the draw games and is only able to give reasonable decision boundary on win/loss games. This is clearly shown by comparing the correctly and incorrectly fitted points (Figure 2b, c).

Meanwhile, even with this simplistic approach of feature and model selection, we already achieved prediction accuracy of 58.50% (with training set fit accuracy of 56.93%), which is a ~10% improvement on the previous report.[1] Given that the previous study focuses on the time-dependence of team ratings,[1] our work suggests that the overall squad performance is a more important feature than the historical team performance in soccer match prediction. In other words, squad selection is more important than team momentum (i.e. winning or losing streaks).

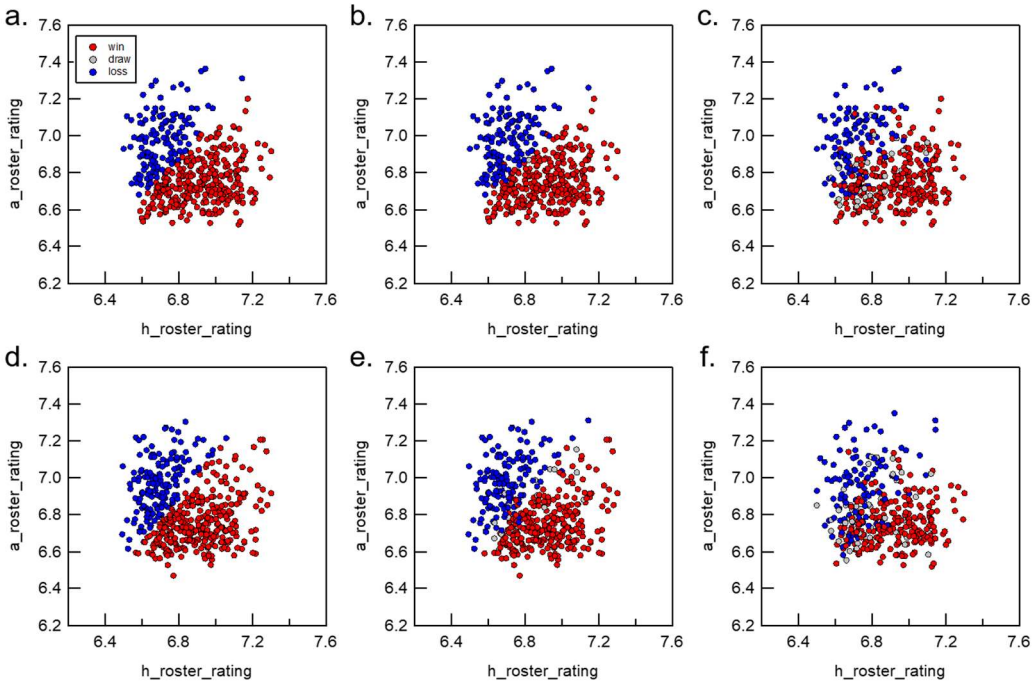

Figure 4. Plots of correctly predicted test set datapoints for feature sets **a)** F1, **b)** F2, and **c)** F3 using neural network. Below are the plots of correctly predicted test set datapoints for feature sets **d)** F1, **e)** F2, and **f)** F3 using SVM with degree-5 polynomial kernel.

The evaluation of other mentioned models on the three feature sets is summarized in Figure 3. Interestingly, test set prediction accuracy of ~58% is achieved on all models with the simplest feature set F1. However, the models are limited in their prediction of draw games with this small feature space. This is visually verified from the lack of draw game points in the plots of the correctly predicted test set points (Figure 4a, d). The reason behind the poor predictions on draw games can be deduced from the distribution of the dataset. Although still close to each other, the win and loss game datapoints are separated enough for the models to comfortably draw the decision boundary between win and loss games. However, the tie games are closely positioned to both win and loss games, causing the model to face difficulties in distinguishing the tie games from win/loss games. Also, draw is the least likely outcome of a match (24.80%), so the models' classifications on the dataset are biased towards the more heavily favored win and loss games.

Meanwhile, the similar accuracy on training and test sets with feature set F1 suggest that the models might be at high-bias regime (i.e. underfitting), for which increasing the feature space can help improve the models. Employing the larger feature set F2 results in small improvement in the training set fit, but the test set accuracy goes down for most models, indicating that the models are entering the high-variance regime (i.e. overfitting). This trend continues when using the largest feature set F3 (Figure 3). Some models end up completely overfitting to the training set, resulting in significant decrease in the test set accuracy.

We can further understand the effect of enlarging the feature space from visual comparison of the models' performances on the three feature sets. Figure 4 shows the correctly predicted test dataset points using the three feature sets with neural network and SVM with degree-5 polynomial kernel. Initially with feature set F1, both models fail to predict any draw games (Figure 4a, d). As the feature set increases, the models try to improve their fits on the draw games, resulting the models to predict the draw games more frequently (Figure 4b, c, e, f). However, this also results in the models to overfit on the draw games, resulting the overall test set prediction accuracy to decrease.

Overall, the highest test set accuracy of 58.89% was reached for SVM with degree-5 polynomial kernel using feature set F1.

In summary, we explored EPL soccer match prediction using various machine learning algorithms and multiple sets of feature space. Even on the smallest feature set explored and simplistic learning algorithms, prediction accuracy of ~58% was reached, surpassing the naïve home-team-win prediction accuracy of 46.2% and outperforming previously reported predictions by ~10%.[1] Such nontrivial improvement in the accuracy compared to the previous report is attributed to the difference in the employed features, suggesting that squad selection is more important than team momentum. The maximum accuracy of 58.89% was achieved with the simplest feature set F1 by using SVM with degree-5 polynomial kernel. On the other hand, modeling with these low-dimensional feature space displayed limited prediction on draw games. Enlarging the feature space enabled more frequent predictions on the draw games. However, this was due to the models overfitting on the draw games, resulting in the decrease of the test set accuracy. Further developments and improvements on the learning algorithms are called for to improve draw game predictions while maintaining overall prediction accuracy.

Methods

Gaussian Discriminative Algorithm (GDA) GDA is a generative learning algorithm which assumes that each class comes from a Gaussian distribution independent of all the other classes. The optimized Gaussian distribution can be worked out from the below equations:[6]

$$\mu_i = \frac{\sum_j \delta_{iy^{(j)}} x^{(j)}}{\sum_j \delta_{iy^{(j)}}}, \Sigma_i = \frac{\sum_j \delta_{iy^{(j)}} x^{(j)} x^{(j)^T}}{\sum_j \delta_{iy^{(j)}}}, \phi_i = \frac{\sum_j \delta_{iy^{(j)}}}{n}, \quad (1)$$

where μ_i, Σ_i, and ϕ_i are the mean, covariance and probability of the i'th class and the sum is run from 1 to n, the total number of datapoints.

Support Vector Machine (SVM) SVM is a classifier which maximizes the geometric margin of the decision boundary. Mathematically,[6]

$$\min_{\gamma,w,b} \frac{1}{2}|w|^2 + C\sum_i \xi_i \text{ s.t. } y^{(i)}\left(w^T x^{(i)} + b\right) \geq 1 - \xi_i;$$
$$i = 1, \ldots, n \text{ and } \xi_i \geq 0, \quad (2)$$

where (w, b) are the fit parameters, ξ_i is the error term, and C is the regularization constant.

SoftMax Regression SoftMax regression is an iterative learning algorithm which maximizes the joint likelihood given by[6]

$$L(\theta) = \prod_i \prod_j \left(\frac{\exp\left(\theta_j^T x^{(i)}\right)}{\sum_l \exp\left(\theta_l^T x^{(i)}\right)}\right)^{\delta_{jy^{(i)}}}, \quad (3)$$

where the product i runs over all the data points and the product j runs over all the classes. The log-likelihood is

optimized via gradient descent or cross-entropy minimization.

Neural Network The neural network was constructed with one hidden layer. Grid search was used to find the optimal number of hidden nodes in the architecture. Forward propagation was used for prediction, and backward propagation was used for optimization using regularized stochastic gradient descent.[6]

All methods were implemented using the *scikit* package in Python except for GDA, which was coded manually. The relevant codes and data files can be found in *https://github.com/varunharbola/EPL_match_prediction*.

Contributions

V.H. and K.L. conceived of this project. V.H. and K.L. contributed equally to data collection and scraping. V.H. implemented SoftMax regression and neural network, while K.L. implemented GDA and SVM.

Reference

1 Aslan, B. G. & Inceoglu, M. M. in *Seventh International Conference on Intelligent Systems Design and Applications (ISDA 2007).* 545.

2 Lopez-Gonzalez, H. & Griffiths, M. D. Understanding the Convergence of Markets in Online Sports Betting. *International Review for the Sociology of Sport* **53**, 807 (2018).

3 Rue, H. & Salvesen, O. Prediction and Retrospective Analysis of Soccer Matches in a League. *Journal of the Royal Statistical Society: Series D (The Statistician)* **49**, 399 (2000).

4 Dixon, M. & Robinson, M. A Birth Process Model for Association Football Matches. *Journal of the Royal Statistical Society: Series D (The Statistician)* **47**, 523 (1998).

5 Cheng, T., Cui, D., Fan, Z., Zhou, J. & Lu, S. in *Proceedings Fifth International Conference on Computational Intelligence and Multimedia Applications. ICCIMA 2003.* 308.

6 Charikar, M., Ng, A. & Ré, C. *CS229: Machine Learning*, <http://cs229.stanford.edu/> (2019).