Predicting Soccer Team Performance From Unintuitive Predictors

Rhys Logan, Kourosh Hassibi, and Caden Kalinowski

Project Overview

The sport of soccer has gained significant traction within the United States in recent years with the growing popularity of the MLS (Major League Soccer). On a high-level, soccer is about scoring more than your opponent. The complexity of the game lies in the fact that the upper body cannot be used aside from the head (hence its original and more widely-used name "Football"). Understanding how to create a goal and properly defend against the attack is no small feat, and all managers have their own theses on what types of actions are important for success. The goal of this project is to test the precision of machine learning techniques for predicting where teams will fall in the standings at the end of the year based on unobvious signals. The results should give 1) an indication of how important the minutiae of soccer is in the success of professional teams and 2) an idea of which soccer actions,or features, are most predictive of performance – what should managers be focusing on if their team is having trouble scoring or defending?

Clearly, teams that score (and therefore assist) more while also conceding fewer goals will finish higher in the table -- these types of features have extremely strong correlations with league rank. Because we wanted to consider less obvious feature variables, we chose to leave out the more obvious predictors of league rank (a discussion of the removed features can be found in the appendix) in order to determine if there is meaningful predictive power in more granular features such as types of passes, tackles, clearances, headers won, dribbles, and much more. This project focuses on classification using Random Forest, Neural Network, and SVM classification algorithms as well as a k-nearest neighbor algorithm, and predicts how well a team performed given the features considered.

Introduction to the Dataset

The dataset includes numerous (95 feature variables) team statistics for 4 seasons going back to the 2017-2018 season. There is data from 3 different leagues: the Italian Serie A, the English Premier League, and Spanish La Liga were considered with each league having 20 teams (these are considered the top 3 leagues in the world).

For each team's season, there were 95 feature variables that were considered along with the label variable describing where they finished in the table. To accurately capture the standards

by which teams assess their success for a season, we decided to use the following label variables: “Top 4”, “Mid-Table”, and “Fighting Relegation” (denoted by 1, 2, and 3 respectively). Finishing in the Top 4 is crucial for teams in these leagues as it means the squad has qualified for the Champions League (the top club competition in the world). Conversely, a team that is Relegated (finished in the bottom 3) will be sent down to the lower division in their respective country. This labeling system is easier on the algorithms because they don’t have to predict 20 distinct labels, and it’s more accurate to how soccer clubs evaluate their seasons as successful, mediocre, or poor.

It may prove extremely rewarding to examine this data because, while the end goal is to always score and not get scored on, focusing on more clear and specific actions may significantly improve a club’s chances of performing better than they would have otherwise. For each major category of data (Passes, Tackles, Saves, Headers, Shots, etc) there are subcategories that specify things like specific locations, percentages, expected vs actual, etc., thus leading to the 95 feature variables for each label.

Because of the granularity of the data, it is a perfect opportunity to utilize machine learning techniques to understand how impactful this seemingly over-specific set data is on predicting what actually matters: is your team going to Champions League, will your team stay in the league for another season, or will your team be relegated to the second division.

Teams can rely on data analysis like this to better understand the reasons behind their success or failure and what, on a lower level, they can improve on to finish higher in the table. Furthermore, the growth of sports gambling affords the opportunity to monetize an accurate machine learning model.

Gathering the Data

To gather the data, we scraped FBref.com using the league seasons url and pd.read_html() to query the datasets for each season and league. The code is as follows:

```
numbers_prem = [10728, 3232, 1889, 1631]
numbers_serie = [10730, 3260, 1896, 1640]
numbers_liga = [10731, 3239, 1886, 1652]
numbers_bundes = [10737, 3248, 2109, 1634]
numbers_ligue = [10732, 3243, 2104, 1632]

years = ["2020-2021", "2019-2020", "2018-2019", "2017-2018"]

frames_prem = {}
frames_serie = {}
frames_liga = {}
frames_bundes = {}
frames_ligue = {}

for p, s, l, b, one, y in zip(numbers_prem, numbers_serie, numbers_liga, numbers_bundes, numbers_ligue, years):
    frames_prem[y] = pd.read_html(f"https://fbref.com/en/comps/9/{p}/{y}-Premier-League-Stats")
    frames_serie[y] = pd.read_html(f"https://fbref.com/en/comps/11/{s}/{y}-Serie-A-Stats")
    frames_liga[y] = pd.read_html(f"https://fbref.com/en/comps/12/{l}/{y}-La-Liga-Stats")
    frames_bundes[y] = pd.read_html(f"https://fbref.com/en/comps/20/{b}/{y}-Bundesliga-Stats")
    frames_ligue[y] = pd.read_html(f"https://fbref.com/en/comps/13/{one}/{y}-Ligue-1-Stats")
```

From here, we cleaned each data frame for each season of each league and concatenated them together. After that, we removed the highly predictive features. A full list of removed features can be found in the appendix.

Data Processing

We took a fairly straightforward approach to processing our data after we gathered it. For each season in each league, we standardized the data. In other words, we had each league season standardized with respect to itself (all the teams during that season were standardized against each other). This was important to do because when the algorithms are evaluating a season, they are looking at just those 20 teams against their rank (1,2,or 3), so we didn't want the data to be relative to anything except other data points in the same season and league. We also decided to standardize because we wanted everything to be relative and not thrown off by huge outliers if, say, one team had an unprecedented season in a certain league.

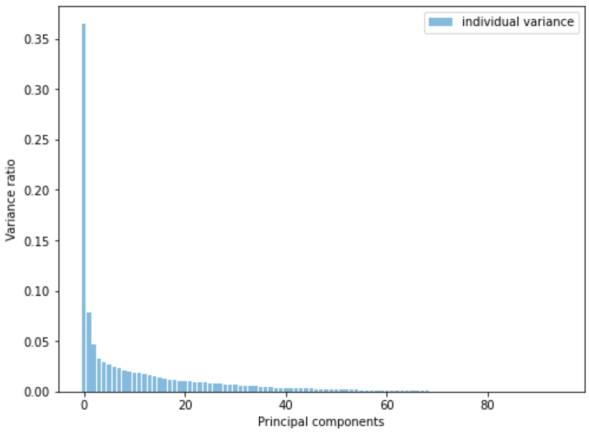
We split the data into train and test data at the point at which each league had 3 seasons of recorded data, so that's 180 data points in the training set and 60 in the testing set (one more season for each league). This allowed us to predict the labels for each league once, to consider the most amount of data (given that we didn't have much to begin with due to soccer data only recently becoming this granular).

The Process

Given the large number of feature variables, a Principal Component Analysis was the first obvious step to take. The following two graphics explain the variance explained by the components and the spread of the top 2 principal components respectively:

Figure 1 shows that a large amount of the variance is explained by the first 10 Principal Components (about 66% of the variance)

Figure 1

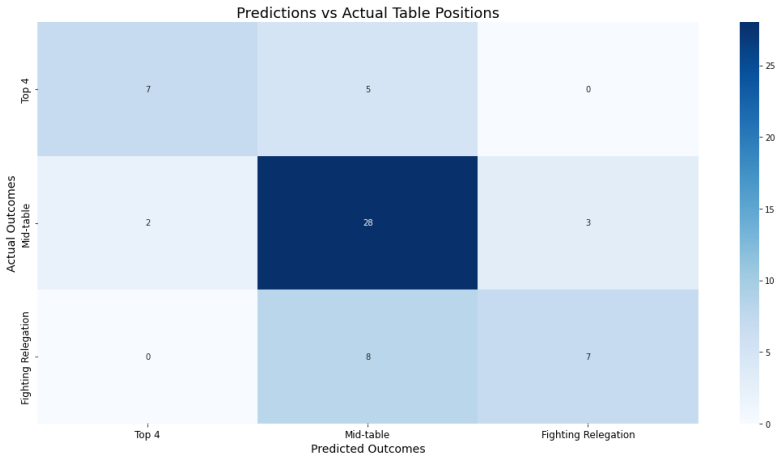


Random Forest Classification Performance

Our algorithm was very straightforward. We produced the following results, predicting labels with 70% accuracy. As can be seen, the model was good at predicting which teams would fall in the middle of the table. This makes sense as it had the widest range of table positions (rank 4 to 17). The model accurately predicted 7/12 of the top 4 squads, and only 7/15 of the teams fighting for relegation.

Overall, the model was relatively poor at predicting the more important labels of "Top 4" and "Fighting Relegation."

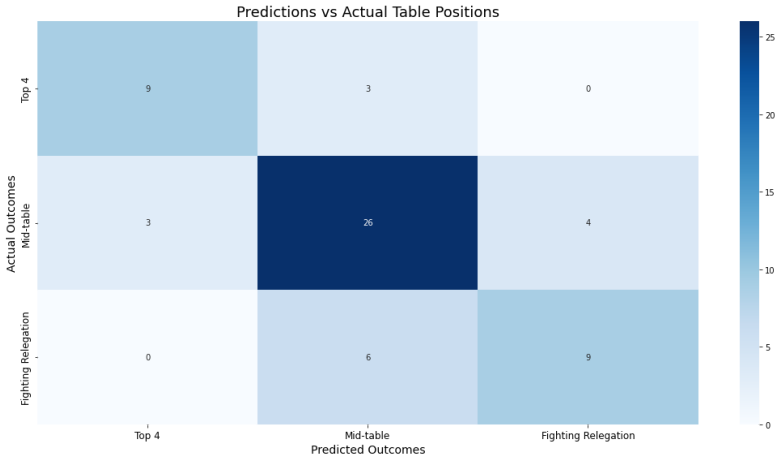
Figure 2



Neural Network Performance

The Neural Network, in a similar way, was only able to generate an accurate prediction about 73% of the time (only slightly better than the Random Forest) over different hidden-layer environments. The confusion matrix is as follows:

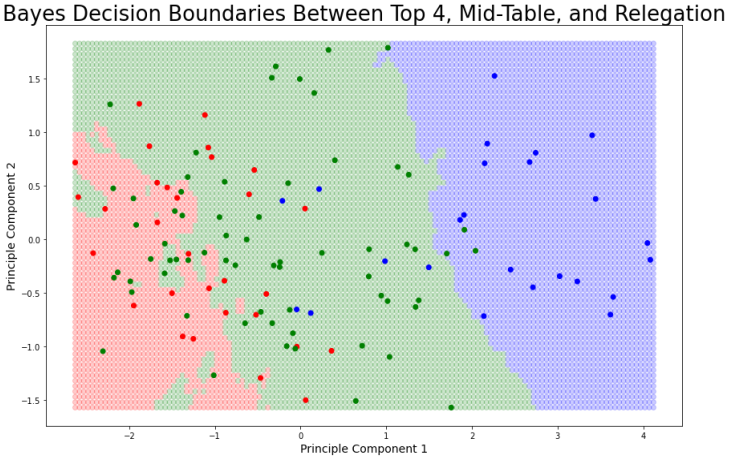
Figure 3



K-Nearest Neighbors Algorithm Performance

Our KNN algorithm outperformed the Random Forest algorithm. To produce the graph, we used the 6 largest Principal Components from the PC dataset. After fitting to the training set, it was determined that 17 neighbors would be our best bet for predicting (Figure 5). Indeed, the model returned a max accuracy score of 76.66% which outperformed the Random Forest by a little over 6%.

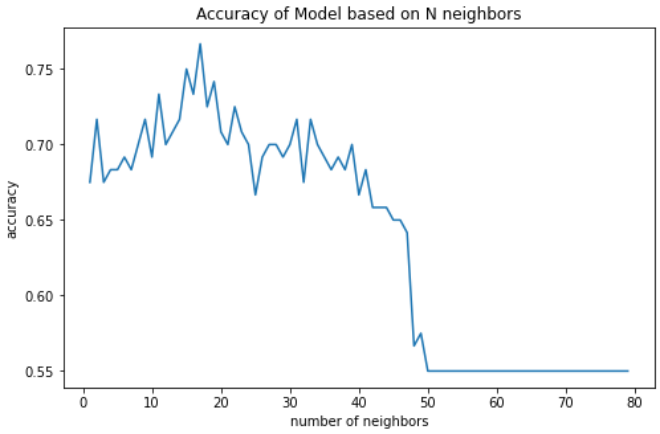
Figure 4 shows that the KNN model was better than the Random Forest at predicting whether a team was going to make top 4 or not, but also struggled with



classifying the teams that would be fighting against relegation.

Figure 5 shows that 30-40 neighbors was the most logical number for KNN on the testing data.

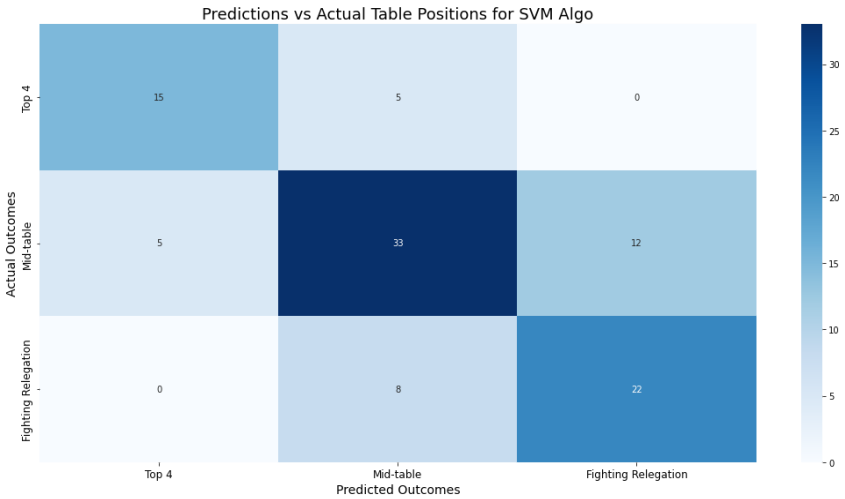
Figures 4 (above page) and Figure 5 (right)



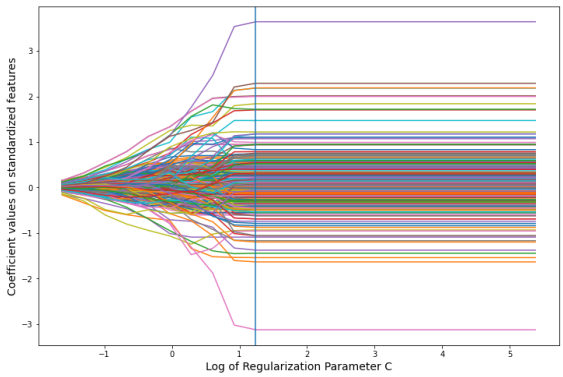
Support Vector Machine Algorithm Performance

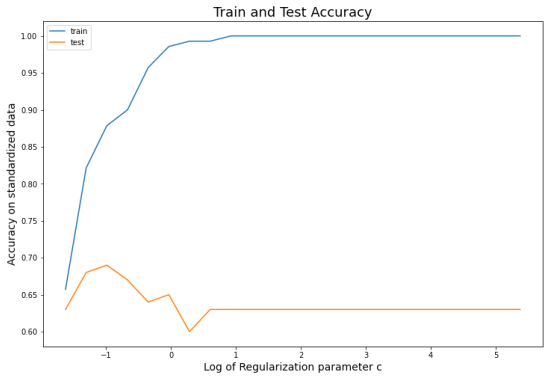
Our SVM algorithm performed in a similar manner to the others, returning a max accuracy of 70%. The confusion matrix is as follows: The base accuracy was 65%, but we regularized the data and used the optimal regularization parameter, which was C = 0.0103.

Figure 6



The regularization graphs are as follows:





Figures 7,8

Conclusion

It seems that, based on our results, there's a pretty clear ceiling for predictive accuracy based on the features considered. It seems like the alternative stats in soccer that we considered can really only be about 75% predictive as to how well a team is going to perform in a given season. It makes sense then, that the more obviously predictive stats in soccer can push the accuracies past our current ceiling to a higher one. Based on our results, though, the top 10 most predictive features were the following:

```
Total Saves / Percentage of Shots Saved
Progressions to opponents goal in less than 6 passes
Carries that Enter Opponent's Final Third
Total Saves
Passes into the opponent's box
Passes with the right foot
Shots on Target
Carries (running with the ball)
Completed Passes
Completed Passes of 15–30 yards
```

While we admit many of these features intuitively make sense, there are a few on here that are pretty enlightening. Firstly, the most predictive feature: "Total saves / Percentage of Shots Saved" is pretty unintuitive at first glance. It's odd that this feature would be more predictive than "Shots on Target" and "Passes into the Opponent's Box." If you think about it, though, teams that have a high number of Saves have the ball less and play worse defensively. Conversely, teams that have a high percentage of shots saved either have excellent goalkeepers or play well defensively and only allow non-threatening shots. So a team with a relatively low Total Saves / Percentage of shots Saved Ratio is more likely to be above a team with a relatively

higher one, because they are not having to save many shots (they play good defense) and when they do concede a shot, it is unlikely to result in a goal.

Secondly, the “Progressions to the opponent's goal-area in less than 6 passes” is super interesting: it had a positive relationship. This type of feature is consistent with what in soccer is called a “counterattack” where the opponent is on the offensive and suddenly loses possession while they are weak in defense. The team who wins possession then advances the ball quickly up the field in as few passes as possible.

This concept is common amongst teams that don’t have a lot of possession, which is strange, considering higher ranking teams seem to have a higher frequency. This may simply be due to the fact that top teams simply get that many more opportunities to counterattack because they are better defensively, and that when they do, they actually complete the action of progressing to the opponent’s goal-area much more frequently.

Other features that I find interesting are “Passes of 15-30 yards” and “Passes with the right foot”, the latter of which we have absolutely no intuitive explanation for. The rest of the features make a lot of intuitive sense.

Prior/Related Work

In a similar study titled “Predicting English Premier League Winners Using Machine Learning”, Divyesh Harit and Rishi Mody of the University of Massachusetts useed six different models to attempt to predict season results in the English Premier League. They utilized 103 feature variables tied to factors that influence the result of a soccer game and evaluated various models including k-nearest neighbors, random forest, support vector machines, stochastic

gradient descent, logistic regression, and neural network. The researchers did not remove any data because it was too “obvious", like we have.

Throughout 11 seasons from 2004 to 2014, they found that the Random Forest resulted in the best accuracy in almost every season. The random forest showed an accuracy of 0.5 or greater four times. SVM had the highest individual season accuracy with 0.6 once, and KNN also performed well for certain seasons. Furthermore, they found that increasing the number of seasons trained on each year had no impact on performance. Therefore, more seasons and more data was not better for prediction purposes.

Connections to Our Study

As mentioned, we found that our predictive accuracy was “hitting a ceiling” at around 70-75%. This seems to also be the case for the related study as well. Also, much like our results, they found that the KNN algorithm performed the best when there was more train data than the other algorithm. The study also found that Crosses were an extremely important feature for predicting table position. The study’s other important features were all ones that we tried to avoid to isolate the less obvious features.

Appendix

Removed Data:

- Anything that measured Goals, Assists, Key Passes, Shots on Target, Shots, or Penalty Kicks.
- Goal-Creating Actions, Shot-Creating Actions
- Wins, Losses, Draws, Goal Difference, Goals For, Goals Against, Points
- Any "Expected" metrics that had to do with Goals, Assists, Wins, Losses, Draws, Points
- All irrelevant features like Matches Played, Starts, Minutes Played (teams all play the exact same amount of matches)

Git Repository with Project Data/Code/Visuals:

https://github.com/Rhys2024/DATA-221-Midterm-Project

Works Considered/Cited

1. "Serie A Stats." *FBref.com*, https://fbref.com/en/comps/11/Serie-A-Stats.
2. "La Liga Stats." *FBref.com*, https://fbref.com/en/comps/12/La-Liga-Stats.
3. "Premier League Stats." *FBref.com*, https://fbref.com/en/comps/9/Premier-League-Stats.
4. "Predicting English Premier League Winners Using Machine Learning", Divyesh Harit and Rishi Mody, University of Massachusetts, https://rmmody.github.io/pdf/589Project.pdf, Spring 2017.

Group Member Contributions:

1. Gathering the Data: Kourosh, Caden
2. Data Processing: Rhys, Kourosh
3. Random Forest Classification Algorithm: Rhys, Kourosh
4. Neural Network: Rhys
5. K-Nearest Neighbors Algorithm: Rhys, Caden
6. Support Vector Machine Algorithm: Rhys
7. Project Writeup: Caden, Rhys, Kourosh