Michael Buchar
MAIS 202

Bundesliga Predictor

Deliverable 2

1/ Problem Statement
The aim of this final project is to predict with high accuracy (cca 70%) the upcoming winners of future Bundesliga games, which is the highest German soccer league. This program outputs the probabilities of a home team win, a draw or an away team win.

2/ Data Preprocessing
I have come to the conclusion to use a different dataset, which includes the pre-game odds by the most popular bookmakers of the game played. The dataset can be obtained at http://www.football-data.co.uk/germanym.php, and again, I will use the datasets from the past 5 years. On top of that, I would like to add a feature that would get as input all the statistics from the first half and again determine the final winner/draw. Moreover, I have decided to incorporate all of this on a website since in my circumstances it would be simpler than an iOS app.

3/ Machine Learning Model
As inspired by the YouTube tutorial on Predicting the Winning football team by Siraj Raval (https://github.com/llSourcell/Predicting_Winning_Teams/blob/master/Prediction.ipynb), I will be using XGBoost classifier to train the data. There aren't many parameters known before a given football game and so I will try to incorporate all of these features in my program: complete analysis of the previous games (date, result, goals scored etc.), bets before the game (most likely received in the background by running a suitable program that will load the bookmaker's webpages), the home team (there is an obvious home team advantage - based on my testing and data in Bundesliga_Predictor.ipynb, there is an approx. 44% chance that the home team will win). Moreover, I would like to implement a "last three games result" as an additional feature. I am likewise trying to visualize the data and discover trends between features. Furthermore, I am planning on using around 60 features since my dataset provides many detailed statistics.

4/ Preliminary Results
As mentioned in Siraj Raval's tutorial, the XGBoost has an accuracy score of around 74% and it performed better than Logistic Regression and SVC. I would like to test all of this by myself as well, yet I am planning on pursuing my project with the XGBoost classifier.

5/ Next Steps
Firstly, I need to test by myself the accuracy of my model and see whether it is a good fit. Secondly, I would like to incorporate the betting odds before the game as additional features and think about implementing the known half-time results algorithm as well. Thirdly, I need to look at some optimization techniques to achieve the highest accuracy score possible. Lastly, I need to figure out how to implement an algorithm that will output all the possible outcome "H", "A", "D" with their respective probabilities. I would then like to compare these

results with the google probabilities, which they have available before every football game on google search.