6/9/2021

Introduction To Machine Learning Project

Data: English Premier League Soccer Results 2010-2021

Dataset: **https://www.kaggle.com/pablohfreitas/all-premier-league-matches-20102021**

Resources & Tools Used: Kaggle, Jupyter Notebook, Python, Pandas, Numpy, sklearn

Goal: See if we can build a respectable classification model that will predict wether the home team will Lose, Draw or Win a match based on features in the dataset?

Process

The curator of the dataset did a good job of organizing the different data points. Some cleaning was required, but I did not have to drop too many columns or filter out many NaN records. The data dictionary was also well formatted for understanding the ~4000 x 120 entries and features.

sklearn was the machine learning library that I utilized for this project. The problem statement/ goal required classification models because I wanted to show based on xyz features being fed into the model, the home team either 0:Lost, 1:Drew, 2:Won. With this in mind, the models utilized included a Decision Tree, Random Forest, KNN and SVM.

Results

The results were initially quite surprising because I was achieving 100% accuracy for most of the models. My error was due to not properly indexing the columns for the X_train subset. Instead of indexing all the features up until the target column, my indexing included the target column. After rectifying this, I was able to get more reasonable results.

When removing all columns related to goals scored/conceded by the home and away teams, the results of each classification model were mediocre. Included in the appendix section are 2 tables with the results from goals included and goals excluded. Naturally goals included lead to mostly 100% accuracy around the board.

The goals excluded table is located at the bottom of the Jupyter Notebook. The model that performed the best was the support vector machine which gave me an accuracy of ~62%. In terms of precision, it was decent at predicting occasions where the home team would win ~73%, but awful at predicting when the home team would draw ~37%. In terms of recall, the model was again quite decent with winning and losing ~70% for both, but awful for a draw ~27%. Lastly,

with regards to R^2 the SVM had the only positive value out of all the other models. However, the value was extremely low at 0.13 indicating that this model only accounts for ~13% of the variance in the data.

Unsurprisingly, it is very hard to predict the outcome of a sporting event in the English Premier League based on historical features included in this dataset. Taking a random event containing a home team and away team in the BPL, the model has ~2/3 chance of predicting the outcome. This is significantly better than just randomly guessing in which case you would have a 1/3 chance, but the model is not going to make you a millionaire if you use this to make sporting bets unfortunately…