Group Members: Clayton Hebert, Tom Wallenstein

CPSC 452

May, 19th, 2021

Deep Learning Final Project:
Predictions of Game Outcomes and Underdog Wins in Soccer

Link to Github: https://github.com/tw98/dl_soccer

Project Description & Motivation

The most popular sport in the world, soccer may also hold the title as the most difficult to predict. As little as a single chance can decide the outcome of a game, providing underdogs with the opportunity to draw or even defeat the world's biggest clubs each and every weekend. And the numbers back this conclusion. In a book published in 2013 entitled "The Numbers Game", two statisticians found that the betting favorite only won around half the time in soccer, as opposed to three-fifths in baseball and two-thirds in football and basketball.[1]

As alluded to above, much of soccer's beauty derives from the ability of small clubs to squeak out wins or draws against clearly superior foes. So, for our project we wish to bring the power of neural networks to try to understand what tactical and personnel decisions correlate with successful outcomes across the top soccer leagues in Europe. Many, including those who published the book mentioned above, have utilized the power of big-data to make arguments for the supremacy of one tactical decision over another. Our group, however, will be honing in specifically on underdogs. We want to examine the tactical situations and scenarios in which an underdog is able to pull off an unexpected win or draw. What makes this question unique to soccer as opposed to other sports, is that a team can win despite being statistically dominated by another team. In a match, one team may have significantly more shots, corner kicks, possession, and free kicks, but can still lose, as the opposing side may simply need one chance to pull out a win. This is what gives soccer its excitement and drama. Are these instances where a team, being statistically dominated in a match, wins purely a matter of chance? Or are there certain features of that match that more often than not allows an underdog to pull out a win?

There are numerous challenges associated with our project, many of which derive from the dataset itself. The dataset we are looking at contains the match details of the top 5 European domestic soccer leagues for 2017/2018, as well as the World Cup of 2018 and the European

[1] https://www.nytimes.com/2014/07/08/science/soccer-a-beautiful-game-of-chance.html

Championship of 2016. Our first challenge, where we spent much of our time, was deciding on and then properly computing the features we wanted to incorporate into our network. Second, we had to make numerous design decisions on what type of networks to use and what constitutes success with our results. These will all be explained in more detail in the subsequent sections.

As avid soccer fans, this question is of much importance to us and the community as whole. It will put to the test long held philosophies from leading clubs and coaches around the world on what tactical decisions lead to success.

Problem Formulation

As mentioned above, we have extracted numerous features about each game in our dataset, and our goal is to gain some insight into which features best predict a game's outcome. To do this, we have first created a network which takes the features we extracted and, using those, tries to predict the outcome of a match. We attempt to optimize the accuracy to predict the right outcome by minimizing the binary cross entropy between the predicted and the true outcomes of the training data. After optimising our network, we use Shapely Additive Explanations and integrated gradient to analyze feature importances to conclude which factors have the most influence on the predicted game outcomes.

Methods Used and Justification

The dataset we used consisted of all matches of the five major European club leagues (England, Germany, Italy, France). In total, these are 1826 matches. We decided to focus only on these league games because we wanted to analyze games from similar competitions. Thus, we did not include national team games from the World Cup and the European Championship because from our soccer background we reasoned that the game dynamics of club soccer and national games are slightly different.

With the help of another dataset that included betting odds of multiple services for each of the games, we determined which of the games were underdog match-ups where the bookmakers favored one team more than the other. We defined an underdog match as a match, where the average betting odds for the two teams had an absolute distance of more than 3 (i.e., team 1 - 1.1 vs. team 2 - 4.0). Using this definition of underdog matches, we end up with a data set of 775 matches. We decided to use 3 as favorite-underdog-threshold because this value offered

the best balance between having a clear favorite-underdog-matchup and keeping a considerably big enough dataset.

We developed methods to extract relevant features from the given dataset. We collected all the features in a data frame, where each row represents a game and the columns represent the features. To track the occurrence or the intensity of all the features, we used bins where each bin represented 5 min of the game time.

In the end, our constructed dataset contained the following features for each of the two teams:

- average height of the starting players
- average age of the starting players
- number of shots
- number of corners
- number of free-kick shots
- number of red cards
- number of yellow cards
- percentage of passes in the final/middle/defensive third
- average pass length
- average acceleration index (measure of how fast a team reaches the closest position to the opponent's goal)[2]
- average invasion index (measure of how close to the opponent's goal a team plays during a match)[2]

In total, this dataset has 1826 rows and 432 columns.

Lastly, we created a feedforward neural network, which takes as input the previously computed z-scores of each extracted feature and predicts the winner of the game. The network contains four hidden layers. The size of the hidden layers are 1024, 512, 128, and 64 neurons, respectively. We use Tanh as the activation function between the hidden layers and the sigmoid function on the output layer to bound the output value between 0 and 1. We chose this architecture setup because it seemed to perform best on the data. Moreover, while we experimented quite long with using dropout (with probability p=0.1) between the first and second

[2] Link, D. & Weber, H. Using individual ball possession as a performance indicator in soccer. Workshop on Large-Scale Sports Analytics (2015)

and second and third layers, we eventually ended up only using L2 regularization to encourage the neural network to better generalize.

Since we essentially have a binary classification, we used binary cross-entropy as our loss function. As our optimizer, we chose Adam. We used a learning rate of 0.00001.

We used this network for two purposes: 1) predicting the winner of a normal game and 2) predicting the winner of a favorite-underdog-matchup. In the first version, a value of 1 corresponded to a win of the home team (team 1), while a value of 0 signaled a win for the away team (team 2). In the second version, a value of 1 corresponded to a win of the underdog while a value of 0 signaled a win for the favorite.

Quantitative and qualitative evaluation

We assess the performance of our neural network architecture by evaluating the accuracy of the dataset to predict the right game outcome. We compared our achieved accuracy against two baselines: picking a random winner and picking the odds-favored team as winner. Following the strategy of choosing the winner randomly would lead to an accuracy of 50%. If one would follow the strategy of always betting on the favorite, one would achieve a prediction accuracy of around 60-65% on the given dataset (see notebook *Comparison_Benchmarks* for validation).

Moreover, we compare our achieved accuracy against the works from previous research projects. Using neural networks, one group from Egpyt and the UK tried to predict a team's results at the 2018 world cup as a win or a loss. They were able to correctly predict 83.3% of wins and 72.7% of losses[3]. Another team of researchers from Bangladesh used a neural network to again predict the 2018 world cup and achieved an accuracy of 63.3% [4].

We then selected two methods to analyze how important the features are to our network. The first is the Shapely Additive Explanations, or SHAP. This method is based on the assumption that each possible combination of features should be considered in order to determine the relative importance of a feature. So where N is the number of features being fed into our network, there are 2^(N) iterations of our network that correspond to each possible combination of features. So to assess the marginal contribution of a feature, you compare the performance in the iterations when that feature is and is not present. Obviously, the runtime for calculating this metric would quickly become outrageous, so we utilize a library which provides some more

[3] file:///Users/claytonhebert/Downloads/sensors-20-03213.pdf
[4] https://link.springer.com/article/10.1007/s42452-019-1821-5

efficient approximations.[5] The second method we selected to analyze the feature importance in our network was integrated gradient. From a high level, the integrated gradient technique takes a straight line path from one of our input vectors to a baseline input. Along this path, the gradient with respect to each input feature that our network sets is calculated at every point. As the name of the technique suggests, this gradient is summed for each feature across the path to get a metric for how much weight it holds.[6]

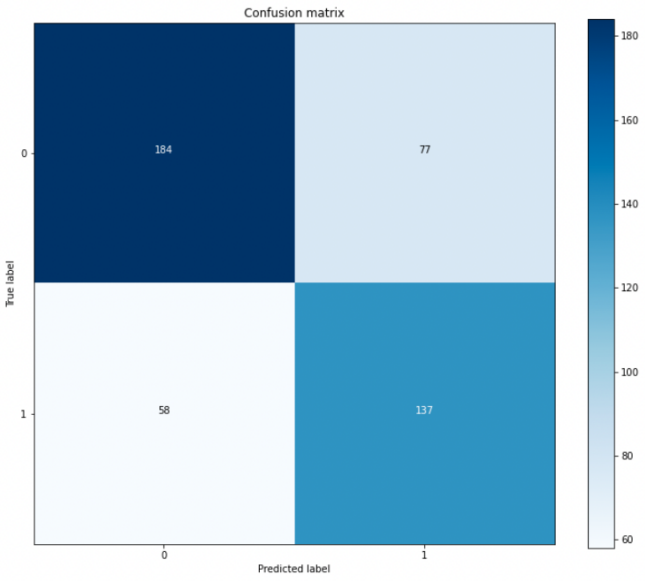
Results and Insights

Predicting Game Winners

First, we investigated if we can predict the game winner with our extracted features and our network architecture. More specifically, we asked the network to predict if team 1 would emerge successfully from the game. To do this, we split the dataset into a training and a test set. The test set consists of 25% of the samples in the dataset. The remaining 75% of samples we used to train.

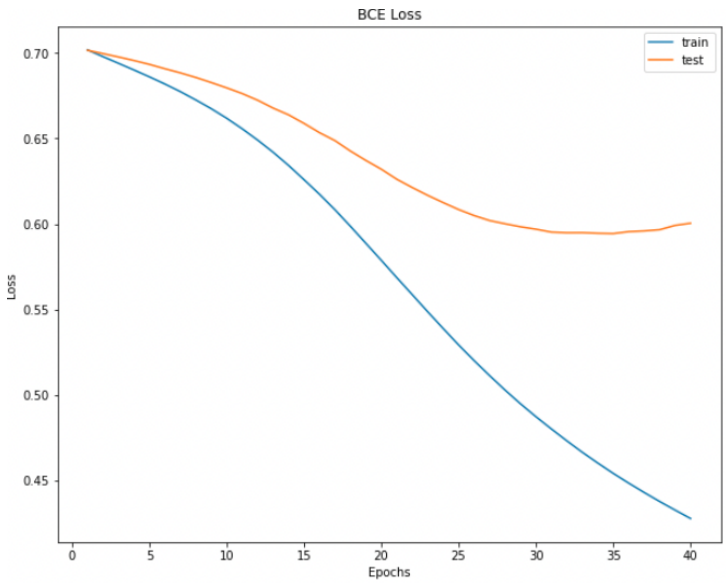
Accuracy

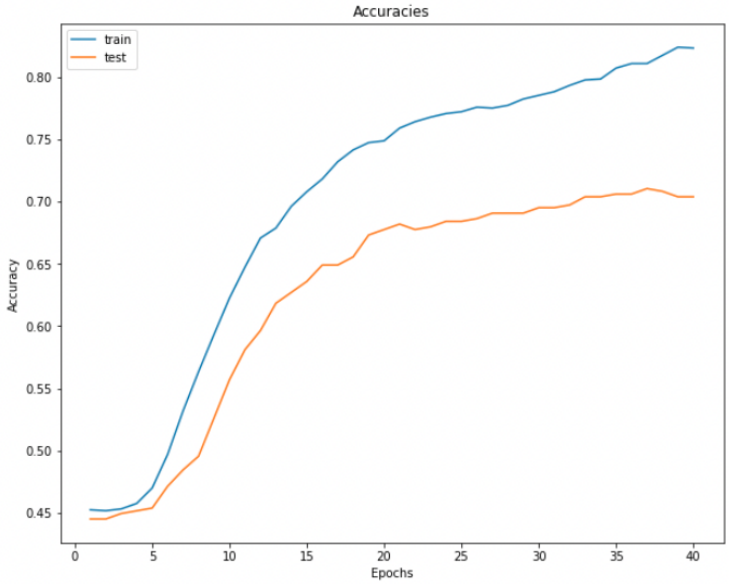
The network's accuracy to predict the right game winner is around 70-72%. We achieve these results after training the network for 40 epochs with a training batch size of 32. Comparing this result to other benchmarks shows that the network learns to find patterns which make it more likely for a team to win. For example, the network clearly outperforms the simple baseline of picking the winner of the game randomly and the more sophisticated baseline of always picking the team with better bookmaker odds. In comparison to the other research works, our network performs moderately well.



5 https://towardsdatascience.com/shap-explained-the-way-i-wish-someone-explained-it-to-me-ab81cc69ef30

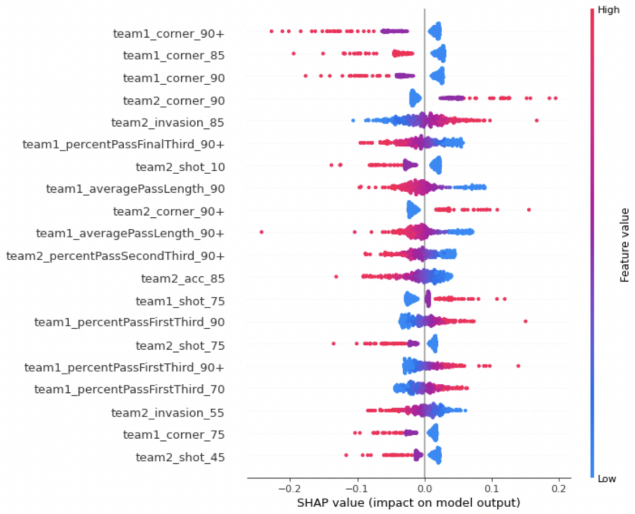
6 https://towardsdatascience.com/should-you-explain-your-predictions-with-shap-or-ig-9cabe218b5cc





Feature Importance

Next, we explored which features had the most contributions to the network's predictions. For that, we analyzed the results of the Integrated Gradients method and the SHAP values.



Especially, the analysis of the SHAP values gives interesting insights. Looking at the distribution of high and low values and their corresponding impact on the network's predictions, one can reason that the network's behavior is coherent with general soccer assumptions and observations. For example, the rather negative impact of high feature values for (team1_corner_90+, team1_corner_85, team1_corner_90) on the probability of team 1 winning can be explained quite easily. If team 1 is trailing late in the game, the team is likely to play

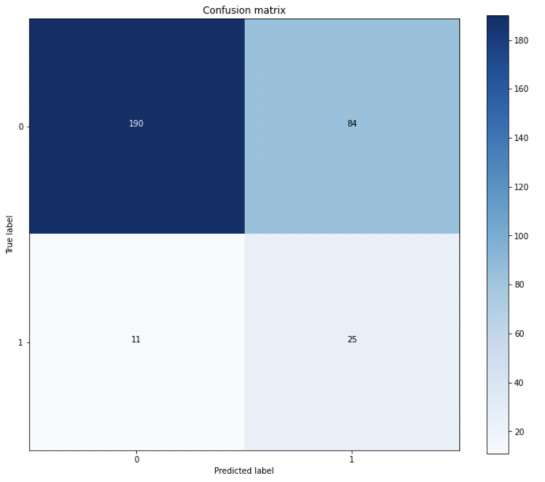
offensive to level the game. Since they attack, it is also likely that they get more corner kicks. Thus, the network uses this feature probably as indication that team 1 trails back and is likely to score. The opposite distribution for the feature team2_corner_90+ reaffirm this hypothesis. An similar argumentation can be made for the features (team1_averagePassLength_{90 / 90+}). If a team is trailing at the end of the game, they tend to play more long balls to quickly reach the dangerous areas of the field. Thus, this team's average pass length is higher than during other periods of the game. Overall, one can conclude that both of the tools show that the network does reasonable predictions which are mostly in line with the experiences of regular soccer viewers.

Looking at the graphs of the integradient gradients (see appendix), one observes similar trends for the influence of features on the network's outcome. The three features most positively correlated with the prediction of team 1 winning are team2_acc_85, *team1_averagePassLength_90+ and team1_averagePassLength_85*. Here again, one could conclude that it is likely that team 1 is leading and thus wants to keep the ball away from its goal by hitting more long balls. At the same time, team 2's acceleration index has a large impact because whether or not team 2 wants to get in dangerous positions as soon as possible to make up a deficit seems to be a good indication for the score . The three most negatively correlated features are *team2_invasion_85*, *team1_percentPassFirstThird_90*, and *team1_shot_10*. Finding a good explanation for these influences is not as straightforward as for the feature importances mentioned before.

Predicting Underdog Wins

Predicting underdog wins proved to be difficult. The difficulty mainly arose because the dataset of the labelled underdog matches (e.g., where there is a clear favorite and underdog) was heavily skewed. While there were 775 of such match-ups overall, the underdogs only won in 88 of these 775 cases. Attempting to create a separate, independent network which made good predictions was, therefore, challenging and not very successful. In most cases, the network simply predicted a favorite win because choosing the favorite to win was favorable for the large majority of samples. Thus, in fear that an independent network does not find and derive any reasonable patterns, we tried out some kind of transfer learning where we used our pretrained model from the first task to solve the challenge of predicting underdog wins. We did this by first training the previously described network for 30 epochs on the task of predicting whether team 1 wins the game. Then, we transferred the model over and trained it for additional 15 epochs asking to predict whether an underdog won the game given the features. We attempted this

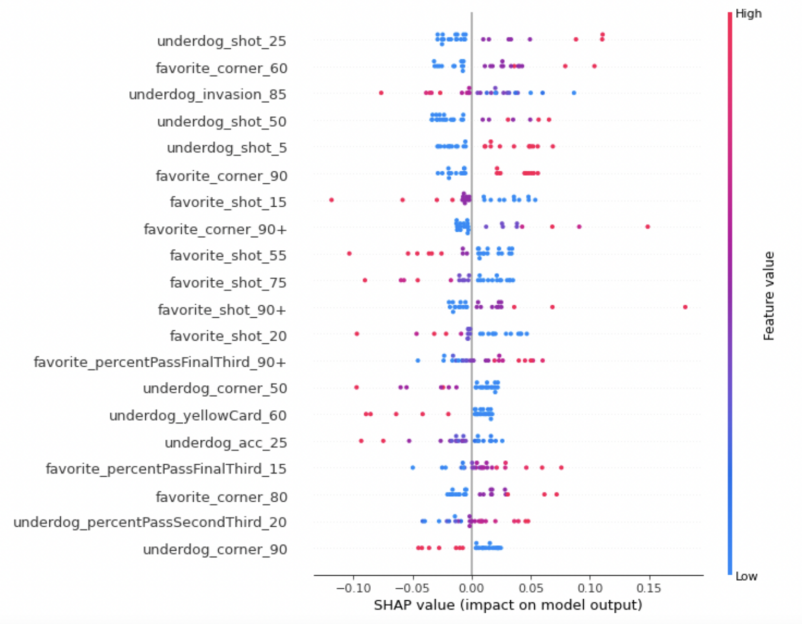
approach because we hoped this would cause that the network learns first the winning patterns for a general soccer game and then fine-tunes these patterns to underdog matches.



Accuracy

Using the method of pretraining on all game outcomes and then specifically focusing on the underdog matches, we achieved an accuracy of up to 70%. The confusion matrix shows well the skewness of the data. The network predicts the right outcomes for most games. If the network makes a mistake, it is more likely to label a match where the favorite ends up winning as an underdog win while the other mistake, predicting a favorite win vs. underdog win in reality, is not as common.

Feature Importance



We also looked at the reported feature importances in this case. To be more precise, we were interested to see if there were any changes with respect to the feature importances and if the reported effects were reasonable. By looking at the SHAP values, one sees that the network

makes credible decisions again. For example, one can look at influences of features describing the number of shots of the favorite during the specific time interval. For time intervals earlier in the game (favorite_shot_15, favorite_shot_55, favorite_shot_75), high values in these features have a negative effect on the likelihood of the underdog winning. This can be explained by concluding that in these examples the favorite likely approached the game seriously bringing its full potential to the field, making it more difficult for the underdog to pull off a surprise. In contrast, for time intervals later in the game, this trend is opposite (favorite_shot_90+ b). Here, high values in these features influence positively the decision for predicting an underdog win because the dynamics within the game are flipped. A trailing favorite usually plays offensively and pushes forward at the end of the game to prevent an upset. Thus, it is more probable that the trailing favorite has more shots, more corners, and more passes in the final third than a leading favorite. The network detects this trend and uses it as an indicator for the score (underdog is leading). This argumentation also holds for other features describing the dynamics of the game and team's tactics such as number of corners during certain time intervals or percentage of passes in the final third.

Conclusion

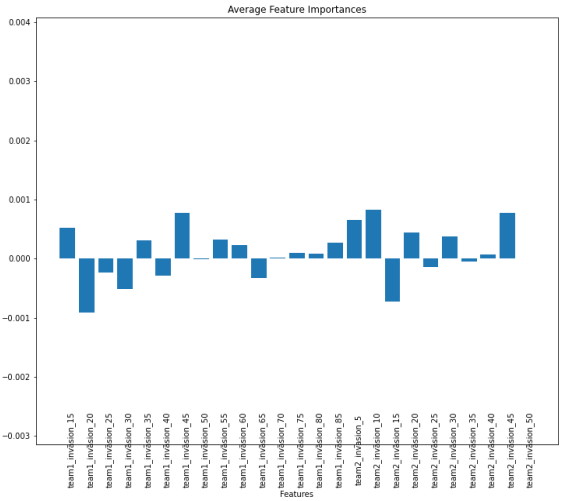
Firstly, we were able to construct a fairly successful network that predicts outcomes based on different features of those matches in question. The predictive capacity of our network may not seem very useful at first glance, as we were predicting the results of the match based on the statistics of that match itself. First and foremost, however, it allowed us to examine our initial goal of understanding which game features best predict the outcome of a match. Furthermore the structure of the network and the input data itself provide the capacity for many interesting additions to this project in the future. One specifically that we had in mind was to create an RNN where we would feed in the 5 minute chunks of game information and have the network predict who the next team to score would be.

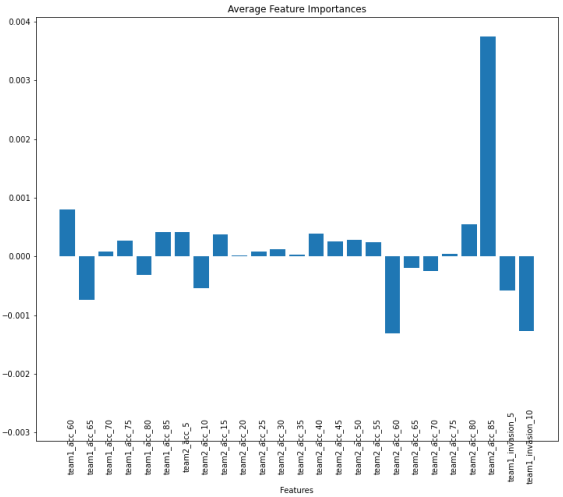
Furthermore, as mentioned above, the analysis of feature importance for the general dataset and the underdog-specific dataset allowed us some insight into our original question as to what game features may best determine a match's results. Given what our metrics give us, however, we did not discover that the conclusions derived from the network contradict conventional wisdom. As explained in the previous section, the features that coincide with what one might expect over the course of a match. When completing this part of the project, we began to realize

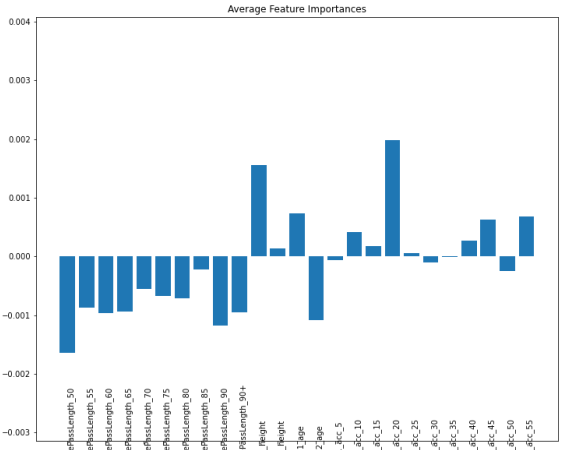
the limitations of our data set. We initially had hoped to only look at the “underdog matches” and then tried to understand which features our network relied on to predict whether an underdog won or not. Even across all these different European leagues, however, there were not nearly enough games that fit this definition to get any reliable or insightful results. So the next step in improving our results would undoubtedly be increasing our dataset, by perhaps looking at leagues across many years and outside of Europe.

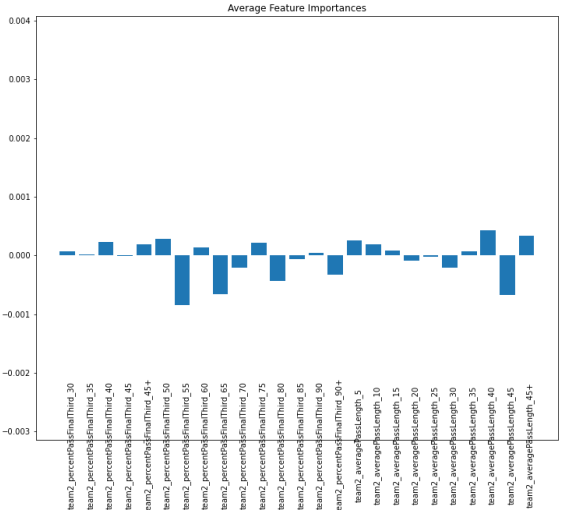
Appendix

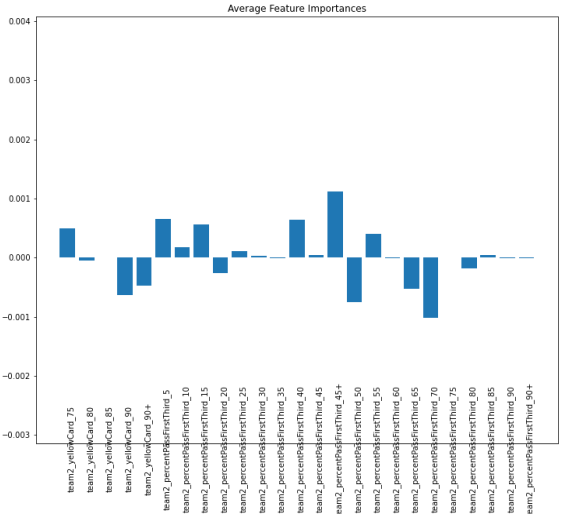
Integrated Gradients

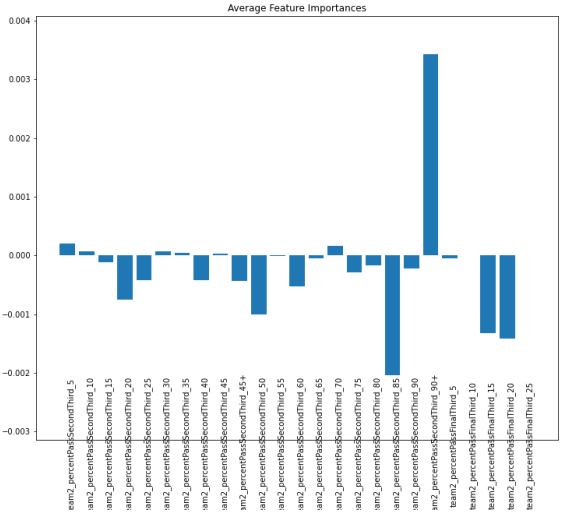


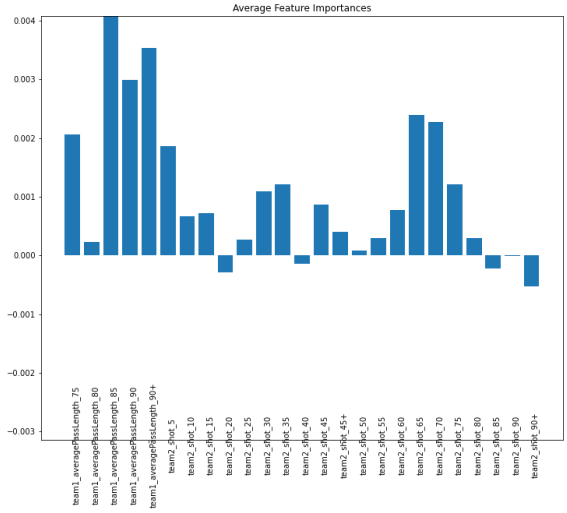


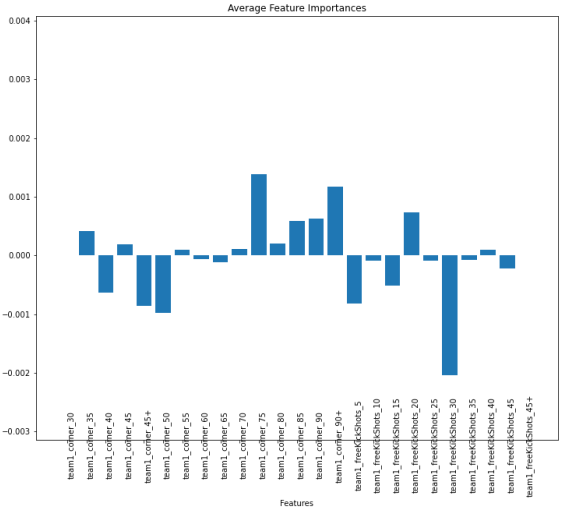


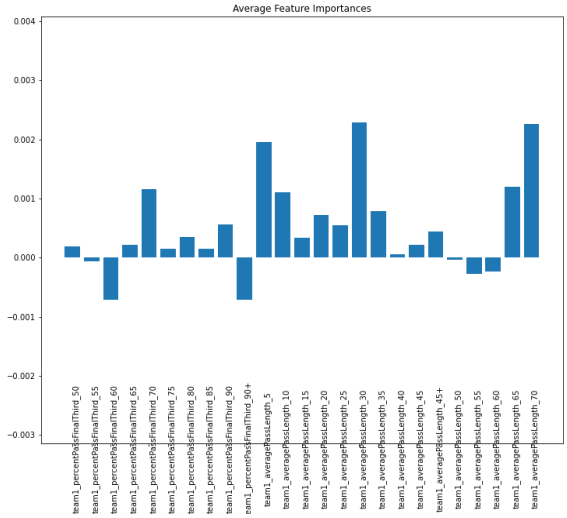


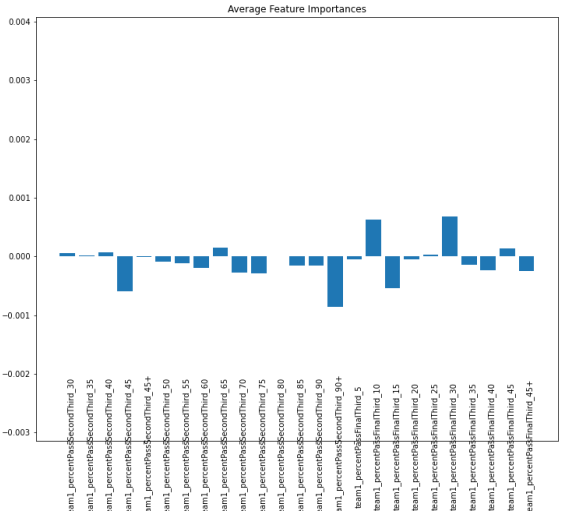


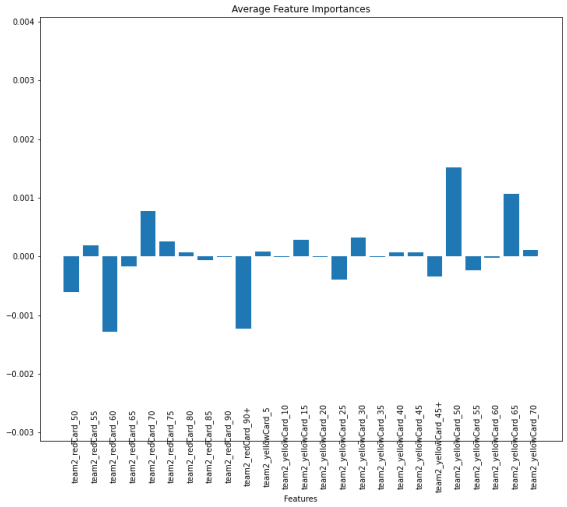


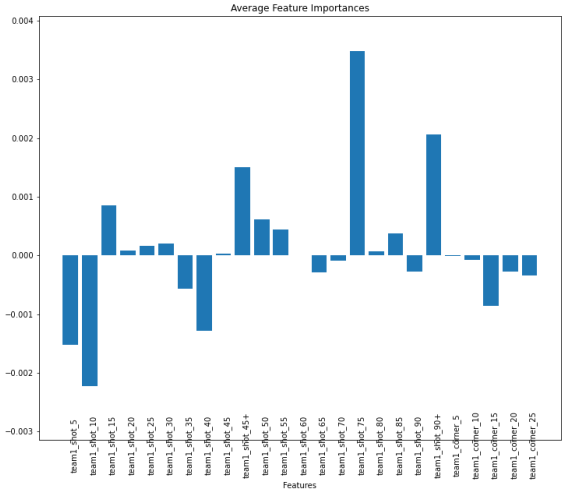

Average Feature Importances
0.004
0.003
0.002
0.001
0.000
−0.001
−0.002
−0.003
team1_averagePassLength_75
team1_averagePassLength_80
team1_averagePassLength_85
team1_averagePassLength_90
team1_averagePassLength_90+
team2_shot_5
team2_shot_10
team2_shot_15
team2_shot_20
team2_shot_25
team2_shot_30
team2_shot_35
team2_shot_40
team2_shot_45
team2_shot_45+
team2_shot_50
team2_shot_55
team2_shot_60
team2_shot_65
team2_shot_70
team2_shot_75
team2_shot_80
team2_shot_85
team2_shot_90
team2_shot_90+

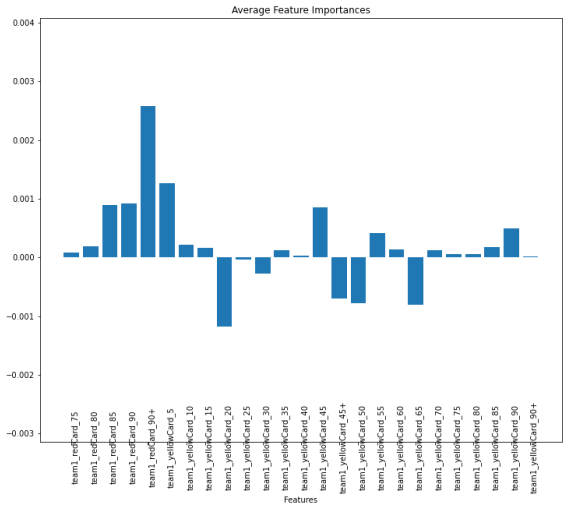

Average Feature Importances
0.004
0.003
0.002
0.001
0.000
−0.001
−0.002
−0.003
team1_corner_30
team1_corner_35
team1_corner_40
team1_corner_45
team1_corner_45+
team1_corner_50
team1_corner_55
team1_corner_60
team1_corner_65
team1_corner_70
team1_corner_75
team1_corner_80
team1_corner_85
team1_corner_90
team1_corner_90+
team1_freeKickShots_5
team1_freeKickShots_10
team1_freeKickShots_15
team1_freeKickShots_20
team1_freeKickShots_25
team1_freeKickShots_30
team1_freeKickShots_35
team1_freeKickShots_40
team1_freeKickShots_45
team1_freeKickShots_45+
Features

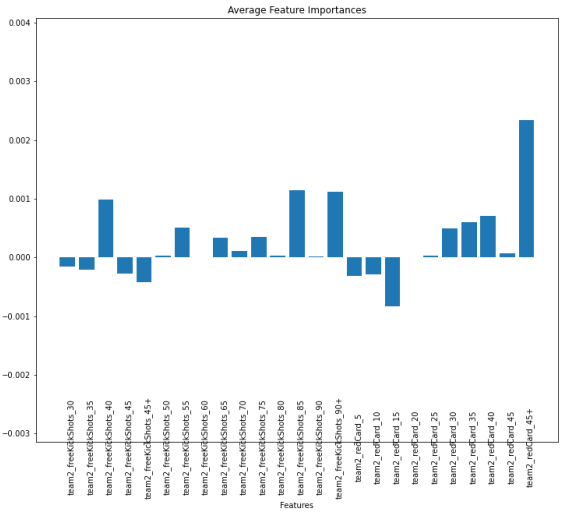

Average Feature Importances
0.004
0.003
0.002
0.001
0.000
−0.001
−0.002
−0.003
team1_percentPassFinalThird_50
team1_percentPassFinalThird_55
team1_percentPassFinalThird_60
team1_percentPassFinalThird_65
team1_percentPassFinalThird_70
team1_percentPassFinalThird_75
team1_percentPassFinalThird_80
team1_percentPassFinalThird_85
team1_percentPassFinalThird_90
team1_percentPassFinalThird_90+
team1_averagePassLength_5
team1_averagePassLength_10
team1_averagePassLength_15
team1_averagePassLength_20
team1_averagePassLength_25
team1_averagePassLength_30
team1_averagePassLength_35
team1_averagePassLength_40
team1_averagePassLength_45
team1_averagePassLength_45+
team1_averagePassLength_50
team1_averagePassLength_55
team1_averagePassLength_60
team1_averagePassLength_65
team1_averagePassLength_70

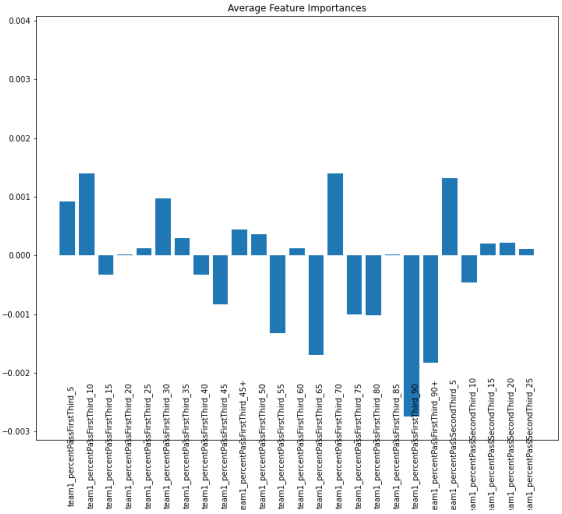

Average Feature Importances
0.004
0.003
0.002
0.001
0.000
−0.001
−0.002
−0.003
team1_percentPassSecondThird_30
team1_percentPassSecondThird_35
team1_percentPassSecondThird_40
team1_percentPassSecondThird_45
team1_percentPassSecondThird_45+
team1_percentPassSecondThird_50
team1_percentPassSecondThird_55
team1_percentPassSecondThird_60
team1_percentPassSecondThird_65
team1_percentPassSecondThird_70
team1_percentPassSecondThird_75
team1_percentPassSecondThird_80
team1_percentPassSecondThird_85
team1_percentPassSecondThird_90
team1_percentPassSecondThird_90+
team1_percentPassFinalThird_5
team1_percentPassFinalThird_10
team1_percentPassFinalThird_15
team1_percentPassFinalThird_20
team1_percentPassFinalThird_25
team1_percentPassFinalThird_30
team1_percentPassFinalThird_35
team1_percentPassFinalThird_40
team1_percentPassFinalThird_45
team1_percentPassFinalThird_45+

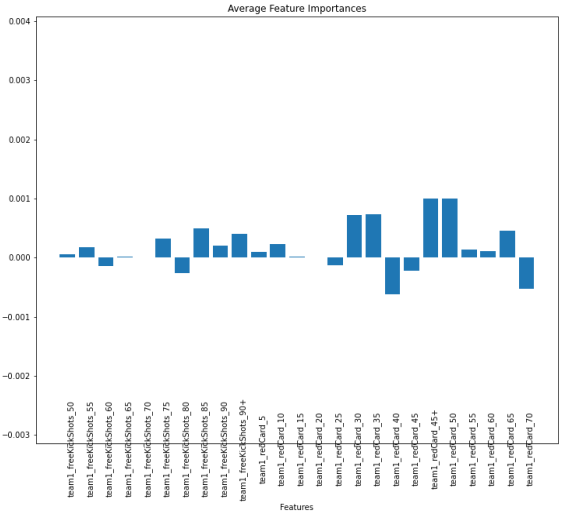

Average Feature Importances
Features


Average Feature Importances
Features


Average Feature Importances
Features


Average Feature Importances
Features


Average Feature Importances


Average Feature Importances