Project Report

SOCCER ANALYTICS

INDEX

Table of Contents

I. Problem Setting

With an estimated audience of 12 million people every game, the English Premier Competition is the most watched professional soccer league on the planet. In comparison, La Liga in Spain draws an average of just over 2 million spectators per game.

There are 20 teams competing for first place in the EPL. The team with the most points at the end of the season is the victor, with three points granted for a win, one point for a tie, and none for a loss. The bottom three teams are relegated and replaced with better-performing teams from lesser divisions. Every team plays each other twice, once at home and once on the road. Thus, there are a total of 380 games per season. A season runs from August to May of the following year.

Soccer analytics is attracting an increasing interest of academia and industry, thanks to the availability of sensing technologies that provide high-fidelity data streams extracted from every match. Soccer is played by 250 million players in over 200 countries (most popular sport globally). Using this information along with team and player specific statistic we are working on soccer match prediction along with expected goal models.

II. Problem Definition

In this project we will be using information of matches over last 20 years and work on XGBoost, SVM and other regression classifiers for soccer match prediction. Challenges that we would be facing will be understanding on which features to use and how events can provide information for creating expected goal models which would show the probability of shot being converted into a goal.

Results in football, more so than any other sport, can be greatly influenced by random moments and “luck.” Near misses, deflected shots, goalkeeping errors, and controversial refereeing decisions alone can dictate the final result. Football is a game of inches.

Expected Goals Model(xG) measures the probability that a shot will result in a goal based on a number of factors. Such factors include the distance from where the shot was taken, angle with respect to the goal line, the game state (what is the score), if it was a header, if the shot came during a counter attack and other factors. For the purpose of simplicity, our exploration will focus on just three of these factors. We can use this metric to sum over all the chances in a match to determine how many goals a team should have scored.

III. Data Source

- The Event data is extracted from https://figshare.com/articles/dataset/Events/7770599?backTo=/collections/Soccer_match_event_dataset/4415000
- The English Premier Competition match results are taken from https://datahub.io/sports-data/english-premier-league#resource-season-1819

- We have used a paper as reference from https://www.imperial.ac.uk/media/imperial-college/faculty-of-engineering/computing/public/1718-ug-projects/Corentin-Herbinet-Using-Machine-Learning-techniques-to-predict-the-outcome-of-profressional-football-matches.pdf

IV. Data Description

The dataset contains football league data of European Premier League from 2000-18 for 18 seasons. The dataset is in different excel files(18 files) so we have used "FOR" loop to download the list of files and merged them into one complete file. After conducting some data pre-processing and dropping missing values we had a set of 6040 rows and 40 Columns.

The variables below give a detailed description of all the columns:

- Div = League Division
- Date = Match Date (dd/mm/yy)
- Time = Time of match kick-off
- HomeTeam = Home Team
- Away team = Away Team
- FTHG and HG = Full Time Home Team Goals
- FTAG and AG = Full-Time Away Team Goals
- FTR and Res = Full-Time Result (H=Home Win, D=Draw, A=Away Win)
- HTHG = Half Time Home Team Goals
- HTAG = Half Time Away Team Goals
- HTR = Half Time Result (H=Home Win, D=Draw, A=Away Win)

Match Statistics (where available)
Attendance = Crowd Attendance

- Referee = Match Referee
- HS = Home Team Shots
- AS = Away Team Shots
- HST = Home Team Shots on Target
- AST = Away Team Shots on Target
- HHW = Home Team Hit Woodwork
- AHW = Away Team Hit Woodwork
- HC = Home Team Corners
- AC = Away Team Corners
- HF = Home Team Fouls Committed
- AF = Away Team Fouls Committed
- HFKC = Home Team Free Kicks Conceded
- AFKC = Away Team Free Kicks Conceded
- HO = Home Team Offsides
- AO = Away Team Offsides
- HY = Home Team Yellow Cards

- AY = Away Team Yellow Cards *HR = Home Team Red Cards AR = Away Team Red Cards HBP = Home Team Bookings Points (10 = yellow, 25 = red) ABP = Away Team Bookings Points (10 = yellow, 25 = red)

V. Data Pre-processing

We first retrieved data for all the 18 seasons, then found goal scored and conceded for both home and away teams, which were organized by teams and match week. Moreover we have add the outcomes of each match by comparing the home team goal and away team goal for that match, as these datasets did not contain the match result. If the home team scores more goals than the away team, the match is won by the home team; otherwise, the match is won by the away team. If both goals are equal, the match ends in a tie. We assigned a score of 1 for a win, a score of 0 for a tie, and a score of -1 for a loss, and then attached it to the table. We added a few extra columns to the final data frame, such as the home team's and away team's continuous match streaks, as well as the continuous loss of each home and away team. Finally, we included goal differentials for the away and home teams, as well as point differentials between the two.

We discovered that several columns are significantly associated, such as away team goal scored and home team goal scored, away team goal allowed and match win, and so on, based on the correlation between each column.

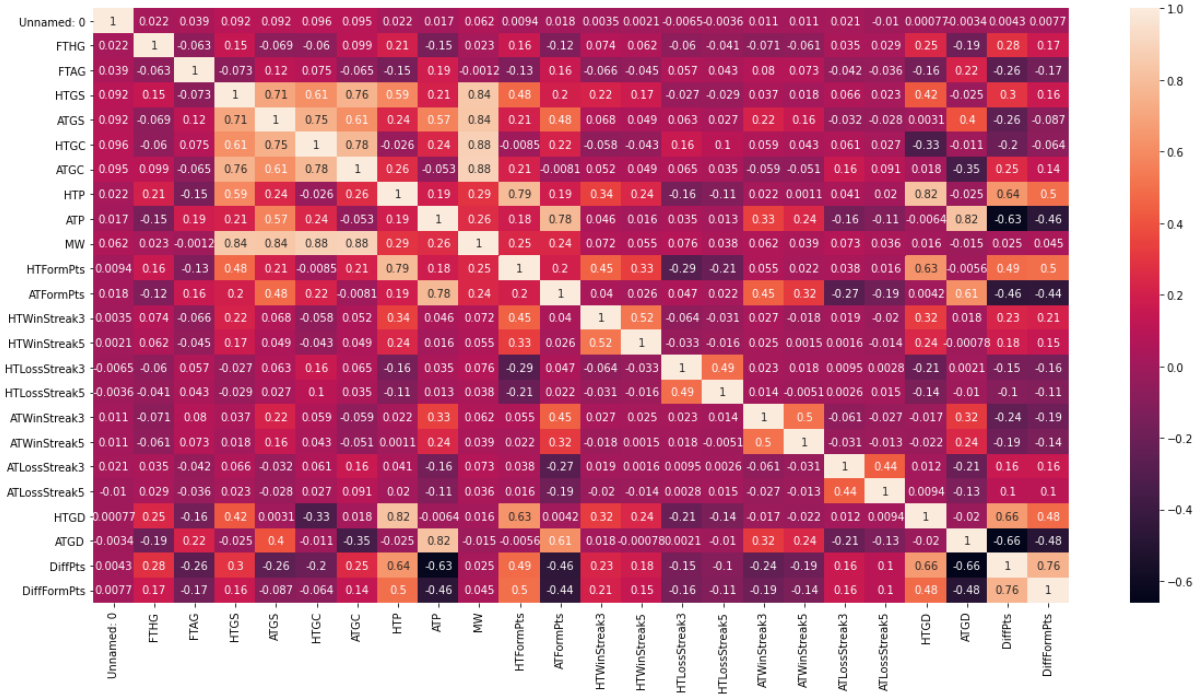


Fig 1: - Correlation

Let's imagine two features are highly or perfectly connected, meaning that increasing one causes the other to increase. This signifies that the information in both characteristics is relatively comparable, and there is little or no variation in the information. This is referred to as Multicollinearity since both carry nearly identical information.

Because our final data frame has a lot of columns, we'll have to employ the PCA data reduction approach. We decreased our data frame's columns from 39 to 22 after normalizing and doing PCA.

VI. Data visualization

On our pre-processed data, we've done some visualization. We did data visualization for Data Exploration so that we could better understand the data. The use of graphs and charts aids in the better analysis of data. During our investigation, we discovered several interesting facts. So, for the first analysis, we utilized a bar graph to find the total number of winners for the entire season, and in the end, we determined that Man City, Arsenal, Man City, Chelsea, and Tottenham were the top winners for the entire season.

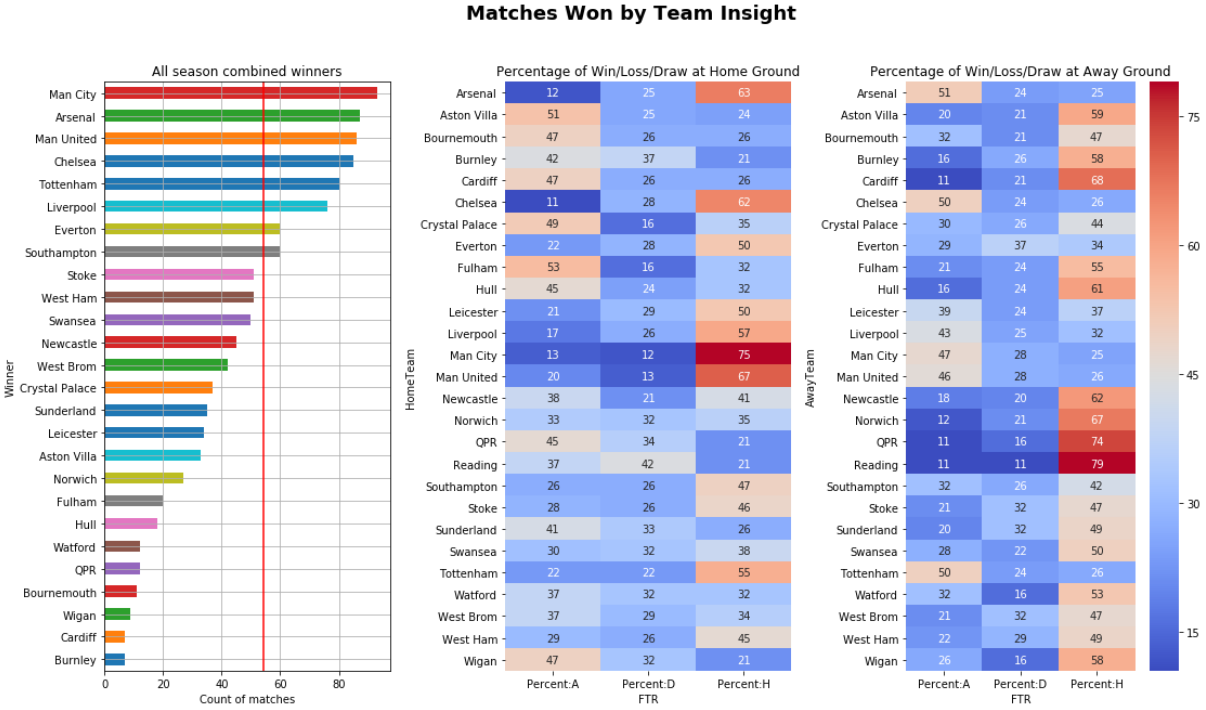


Fig 2:- All season combined winners and team insight

We kept looking at match winners vs. match count, and for the next visualization, we looked at total goals scored by each team, both at home and away. Man City, Arsena, and Chelsea are clearly the top goal scorers at home ground, while Man City, Arsenal, and Liverpool are the top scorers at the away ground .

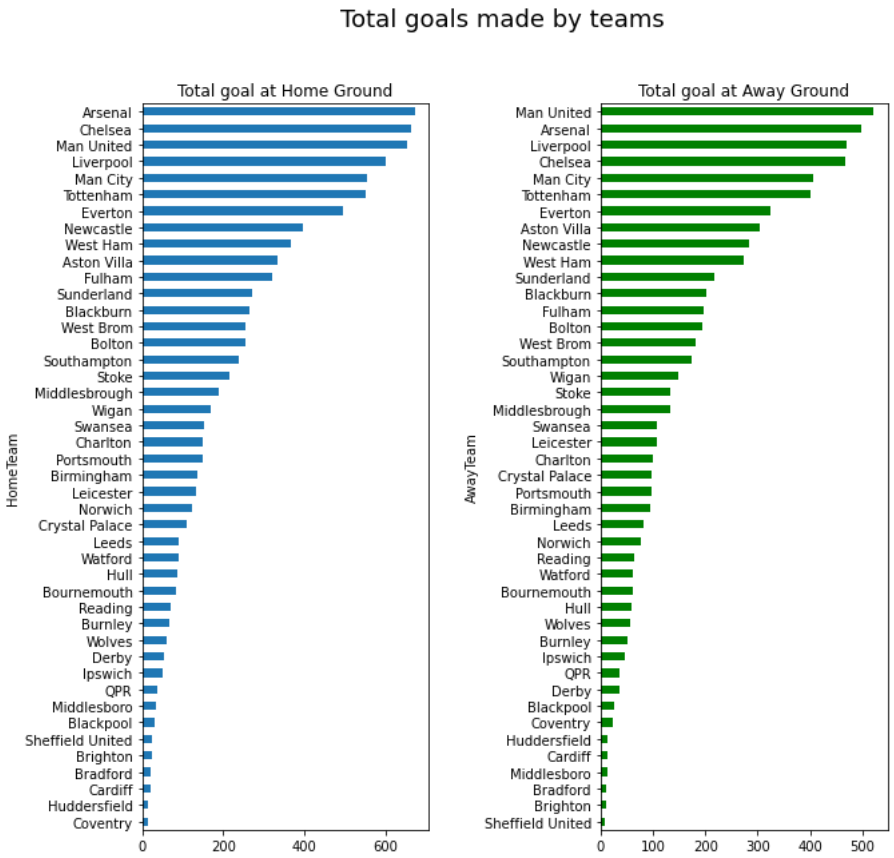


Fig 3: - Total Goals scored by teams

We have constructed a heatmap to find team score how many goals against which team on both home and away ground.

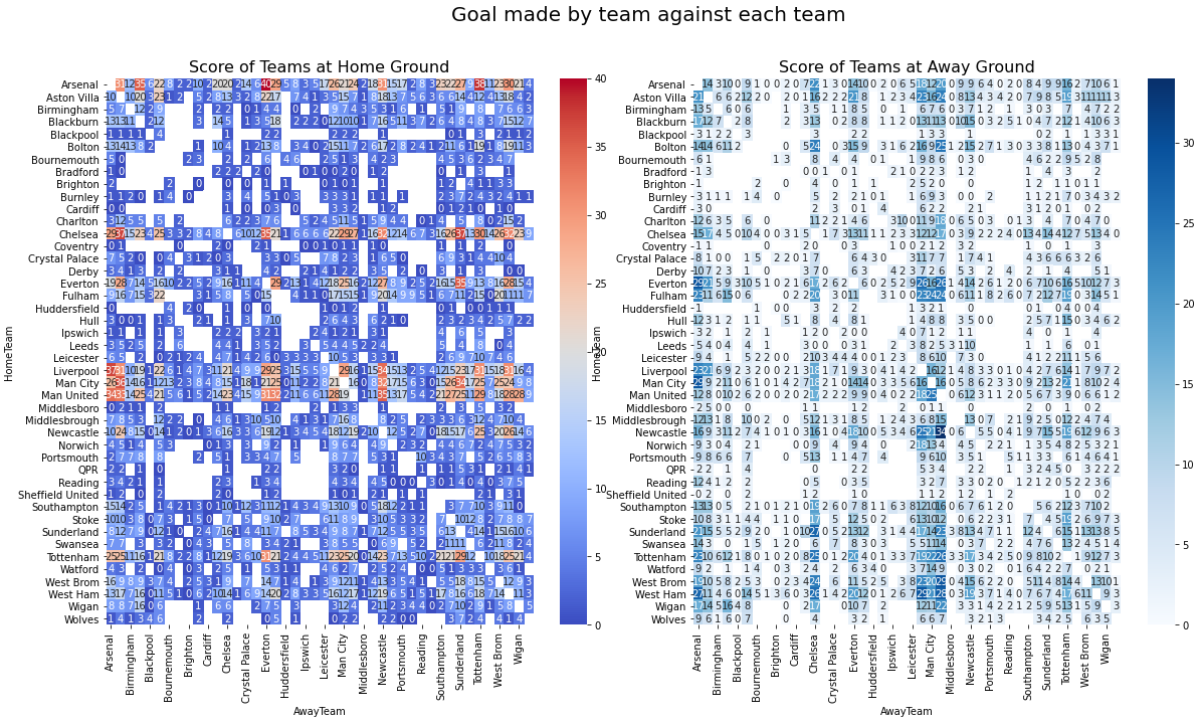


Fig 4:- Goal Scored by teams against each team

Now, using count win on the x-axis and total goal on the y-axis, we've produced a Win versus Score corelation. We may conclude from the graph below that goals scored and matches won have a strong positive correlation.

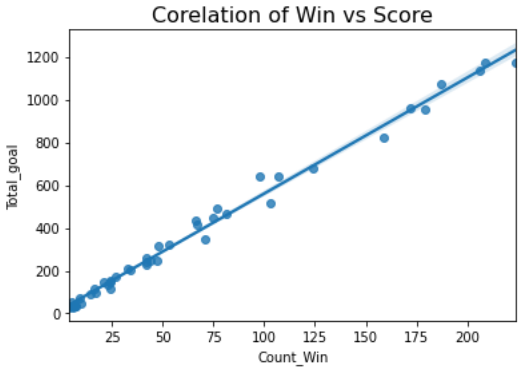


Fig 5:- Corelation of win vs Score

VII. Data Mining Models

We have used 8 models in total: -

1. Logistic Regression
2. SVM
3. Random Forest Classifier
4. Decision Tree Classifier
5. K nearest Neighbours
6. Neural Networks
7. AdaBoost Classifier
8. XGBoost Classifier

To begin developing the models, we split the entire dataset into two parts: one for the predictor variables (X) and the other for the response variable (Y). Then we divided both X and y into training and validation sets of 80 percent and 20 percent, respectively. The training set values were then sent to a model for the data to be trained and used to predict any future data, and the validation sets were used to forecast the results and to see the predicted accuracy. Model performance judgments were made by calculating accuracy for both the training and validation sets.

We have performed hyper-parameter tuning on 6 models out of the 8 models we are using and saw that:

- For KNN, we performed hyper-parameter tuning using GridSearchCV and discovered Manhattan as metric, using distance as metric to assign weights ,and with 29 neighbours suited to be the best fit for our model.
- After performing GridSearchCV we found out impurity split measurement is best using entropy, going till a max depth of 25 nodes and using the best splitting method to be done at each node gave using a good accuracy for that model.
- We obtained that splitting done based on gini and balanced class weights give us a good accuracy score for Random Forest Classifier model.

- For Neural Networks we found that using ‘Rectified Linear Unit Layer’(ReLu) as the activation layer for the input and hidden layers, and Adam as the optimizer gives best accuracy among all the combinations.
- Using SAMME.R as the parameter algorithm and keeping the learning rate at 1.11 to meet the local minimum in gradient descent gave us good accuracy for AdaBoost Classifier model.
- For XGBoost Classifier, we got that when learning rate is 0.75 and when the parameter booster is either gbtree or dart gives us good results when using that model.

In addition, pipeline was utilized to calculate training and testing accuracy, which helped us discover the best model for the provided data. Then we segregated our data into a feature set and a target variable, standardised the data, and looked into each feature column, converting all categorical data types to dummy variables. Finally, the revised columns are saved into new data frame.

We have fitted the logistic regression, SVM and Random forest on training data and constructed a confusion matrix

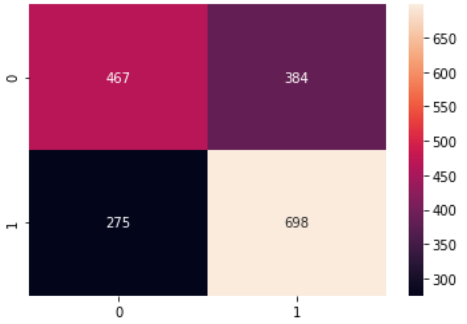


Fig 6:- (Logistic Regression)

	precision	recall	f1-score	support
H	0.63	0.55	0.59	851
NH	0.65	0.72	0.68	973
accuracy			0.64	1824
macro avg	0.64	0.63	0.63	1824
weighted avg	0.64	0.64	0.64	1824

Table 1:- Classification report Logistic Regression

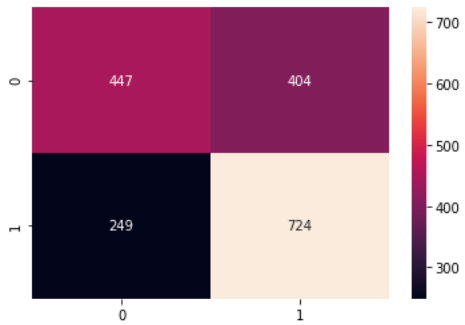
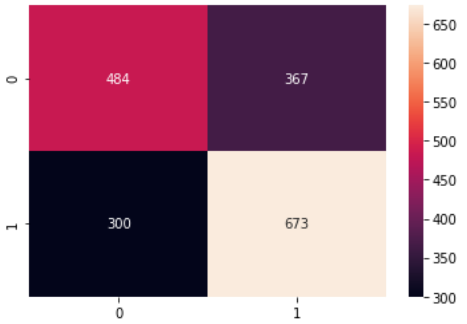


Fig 7:- (SVM)

	precision	recall	f1-score	support
H	0.64	0.53	0.58	851
NH	0.64	0.74	0.69	973
accuracy			0.64	1824
macro avg	0.64	0.63	0.63	1824
weighted avg	0.64	0.64	0.64	1824

Table 2 :- Classification report SVM



	precision	recall	f1-score	support
H	0.62	0.57	0.59	851
NH	0.65	0.69	0.67	973
accuracy			0.63	1824
macro avg	0.63	0.63	0.63	1824
weighted avg	0.63	0.63	0.63	1824

Fig 8:- (Random Forest) Table 3:- Classification report Random Forest

We also ran grid searches for artificial neural networks and Ada-Boost and discovered that parameter relu was the best fit for the model.

VIII. Performance Evaluation

The model performance before standardization is represented by the following training and testing accuracy Table below

Model	Training Accuracy Score	Test Accuracy Score
Logistic	0.996945	0.997807
KNN	1.000000	0.828399
Decision Trees	1.000000	0.809759
Random Forest	1.000000	0.889254
SVC	0.996006	0.987390
ANN	1.000000	1.000000
Ada	0.948778	0.918311
XGBoost	1.000000	0.963268

Table 4: - (Accuracy Before Stand)

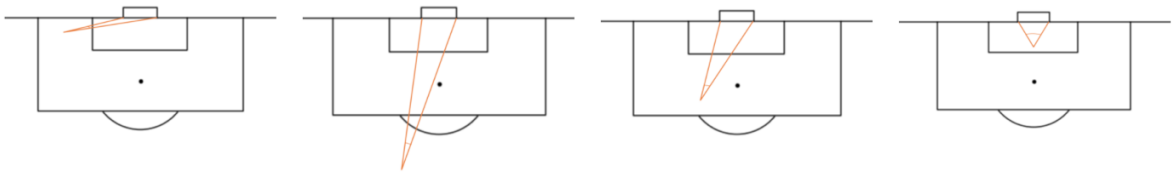
The below tables show the training and testing accuracy after standardizing the data:

Model	Training Accuracy Score	Test Accuracy Score
Logistic	0.996711	0.997807
KNN	1.000000	0.848136
Decision Trees	1.000000	0.805921
Random Forest	1.000000	0.888706
SVC	0.996945	0.986294
ANN	1.000000	1.000000
Ada	0.948778	0.918311
XGBoost	1.000000	0.963268

Table 5: - (Accuracy after Stand)

IX. Expected Goals Model (xG Model)

The xG (expected Goals) is the most utilized metric in today's Football Analytics area. In layman's terms, it's the chance (from 0 to 1) that a shot will result in a goal. As a result, a likelihood of 1 indicates a goal, while a probability of 0 indicates no goal. These probabilities are derived using a statistical model that has been trained with a sufficient amount and diversity of shot data, as well as its associated geographical, temporal, and context information.



We can deduce that the most important factors we need would be the distance from goal when the shot was taken, the angle with respect to the goal and what part of the body the shot was taken with.

Football data is normally split into two forms: event data and tracking data. Event data records all on-ball events and where on the pitch they happened (such as shots, passes, tackles, dribbles), whereas tracking data records the positions of players and the ball throughout the game at regular intervals.

The shot position inside the field (e.g. X,Y coordinates), which may be used to calculate the shot distance to goal and the shoot viewable angle, is one of the most crucial features to include in an xG model. The xG values from a rudimentary model that just examines these two properties are shown in the graph below. The higher the xG value, as expected, the closer the shots are made:

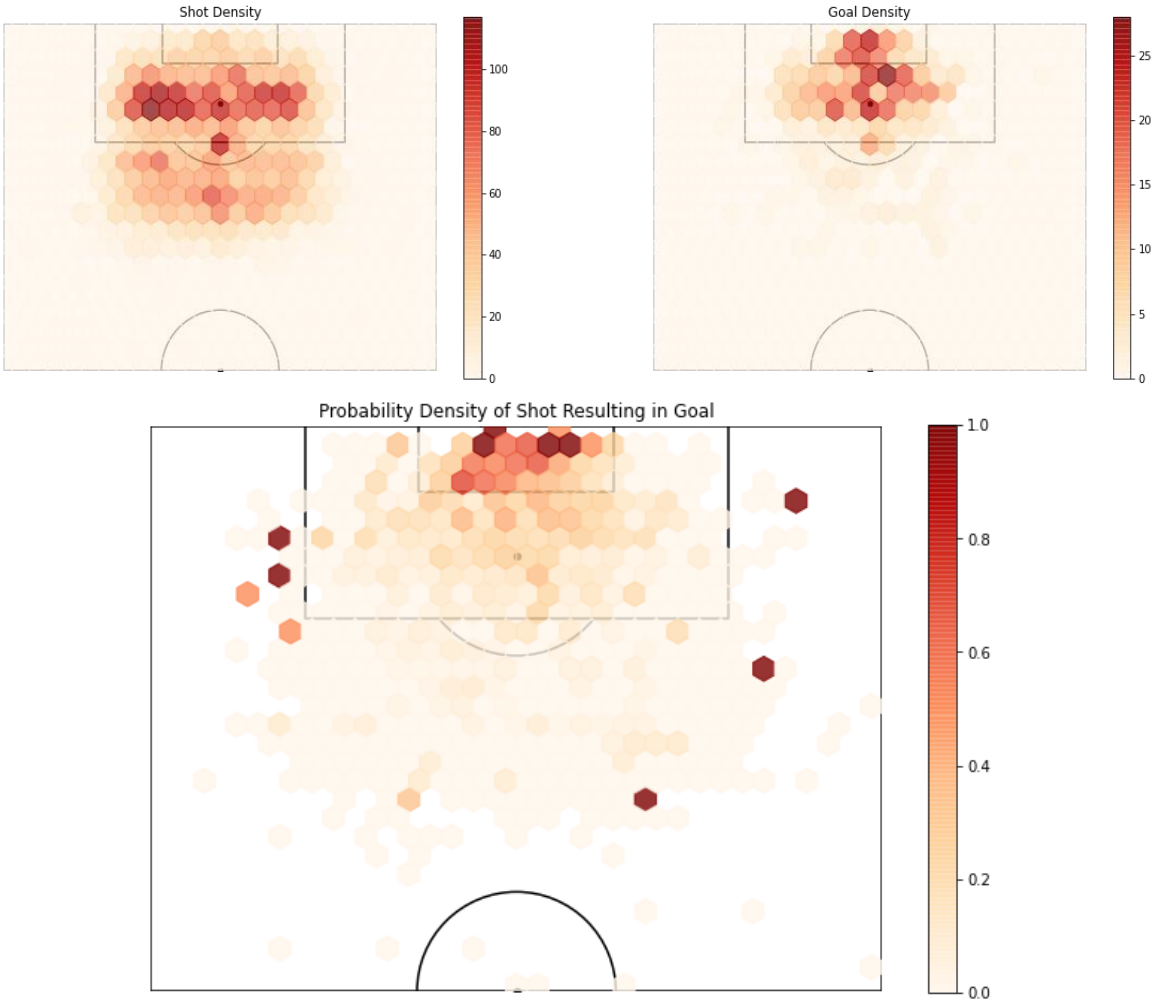


Fig 9:- Probability Density of shot resulting in goal

Similarly, we discovered the likelihood of a header being converted into a goal. It can be observed in the graph below that if a player hits a header from close to the goal post, the chances of it being converted into a goal are great. The red coloured boxes indicate a high likelihood of a shot being converted into a goal.

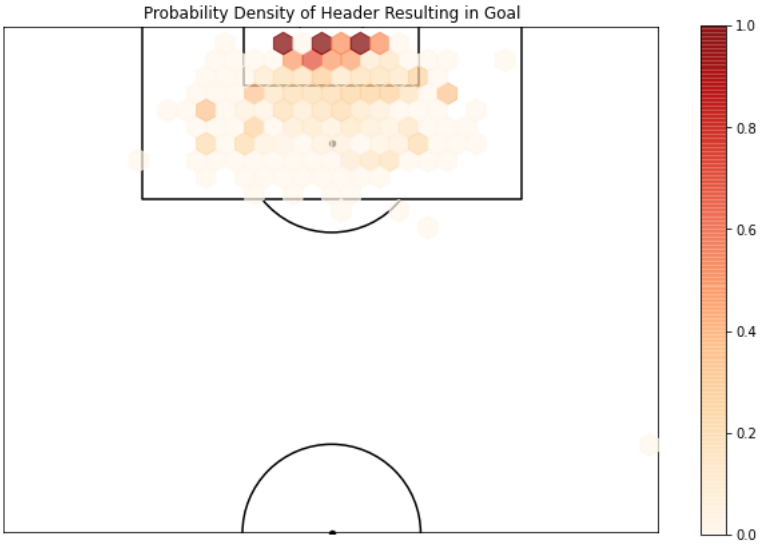


Fig 10: Probability density of header resulting in goal

1. Data Visualization

The histogram below depicts the total number of goals scored in each of the 18 seasons from various distances and angles in degrees. Furthermore, it is evident that the Angle histogram has been left skewed.

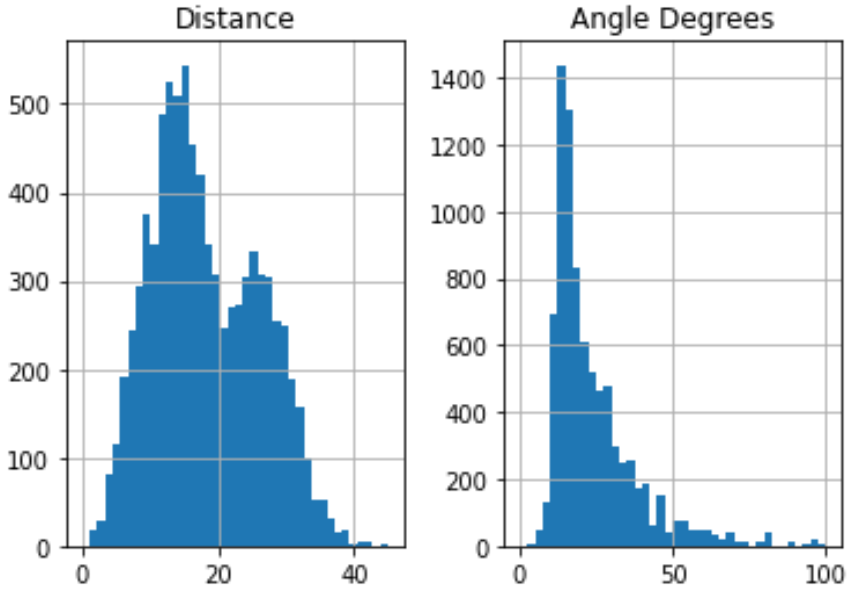


Fig 11: Goal scored all season from Various Distance and Angle

The pair plots below show the distance of the shot from the goal vs. the result, as well as the angle of the shot vs. the result. It may be established that if a shot is made from less than 20

meters away, it has a good chance of being converted into a goal. Similarly, if a player shoots from a 50-to-25-degree angle, it has a good possibility of being converted into a goal.

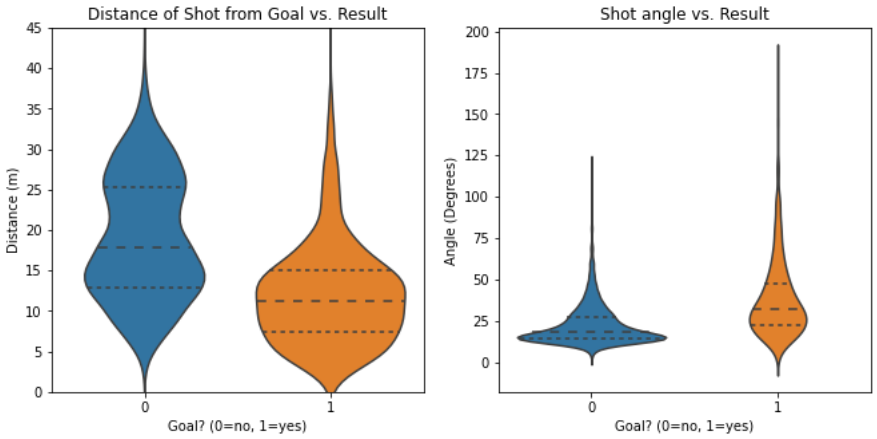


Fig 12: A shot from Distance & angle converting in a Goal

2. Goal Analysis

Below we can see that there is a clear distinction between the blue and orange clusters. If we assume this is the true nature of the relationship between shots and distance, and we want to predict future shot results based on this data, what model could we adopt?

Since the data is easily separable, we can draw a boundary line between the two clusters. This line will represent a discriminant function, which we will use to classify each shot outcome as a goal or miss. In the case above, the discriminant function maps out a distance from the center of the goal.

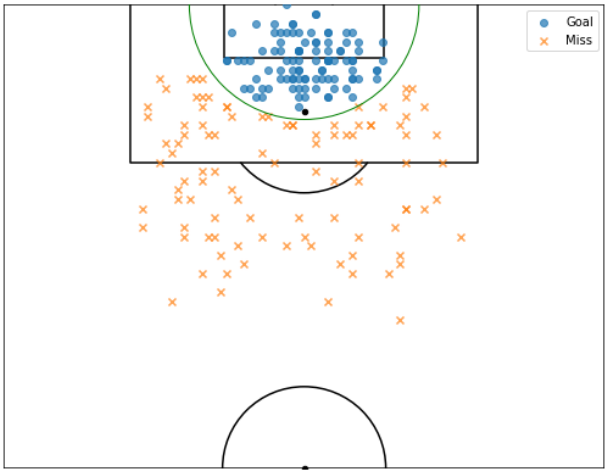


Fig 13: Number of Goals and Miss

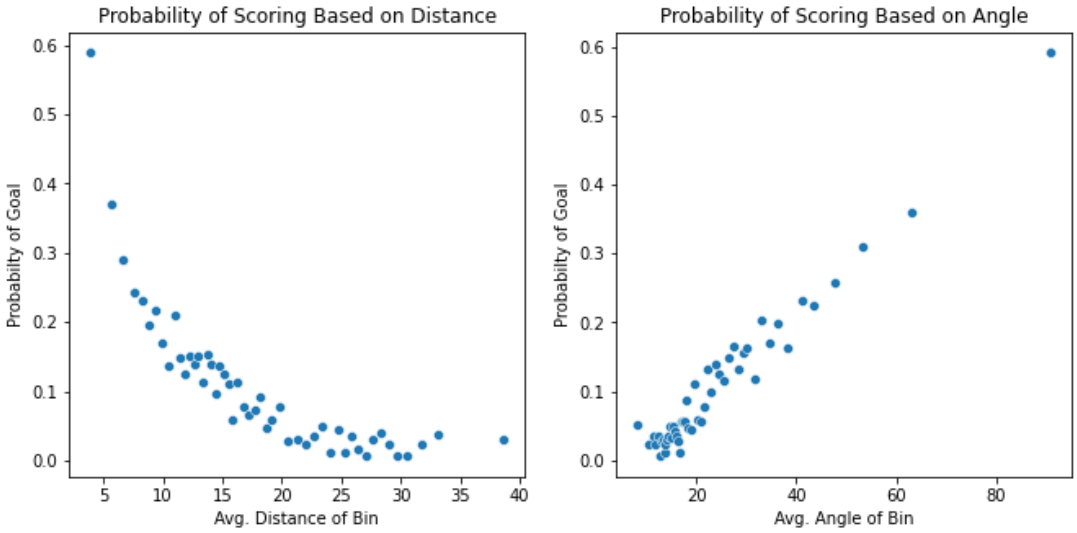


Fig 14: Probability of scoring based on distance and angle

The first thing that pops out and is quite intriguing is that as we move further away from goal, the probability of scoring becomes exponentially more difficult. Now that is profound because it vastly diminishes the value of shots from distance. We can hypothesize that this is because as we increase the distance a shot is taken from, it not only has a longer distance to travel but the target also becomes smaller.

3. Data Modelling

Heaviside Function Classification for Shots

The Heaviside function, often written as H(x), is a non-continuous function whose value is zero for a negative input and one for a positive input. The function is used in the mathematics of control theory to represent a signal that switches on at a specified time, and which stays switched on indefinitely.

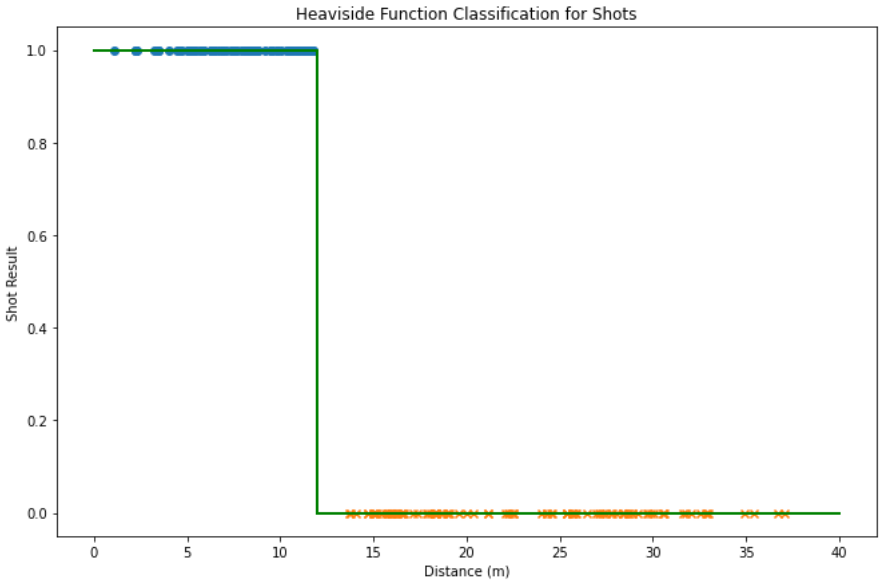


Fig 15: Heaviside Function Classification for Shots

The xG Model

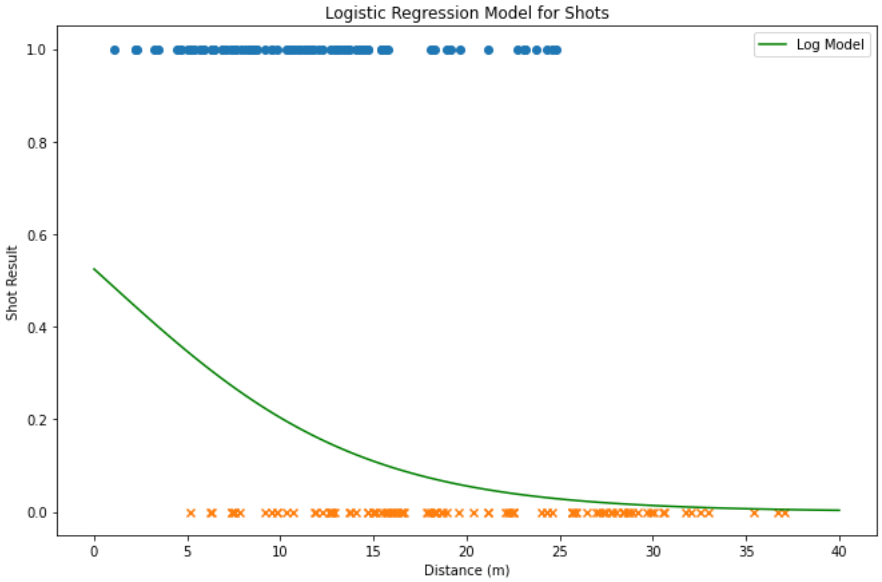


Fig 16: Logistic regression model for shots

While the graph above produces some certification that our coefficients produce a reasonable fit, it is not very clear graphically how good this fit is, especially considering this is just a small sample of our training data.

Fitting the model with data

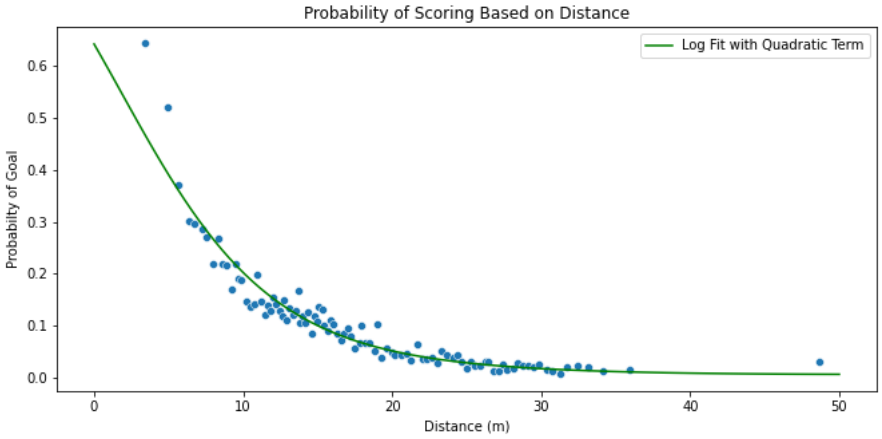


Fig 17: Probability of scoring based on distance

The purpose of these graphs is to gauge where our model performs well and where it does not. Plotting 32,000 points on a graph is not great for visualization, so we decide to plot a sample representation of the population. Used pipeline function from sklearn to mesh a polynomial function with logistic regression.

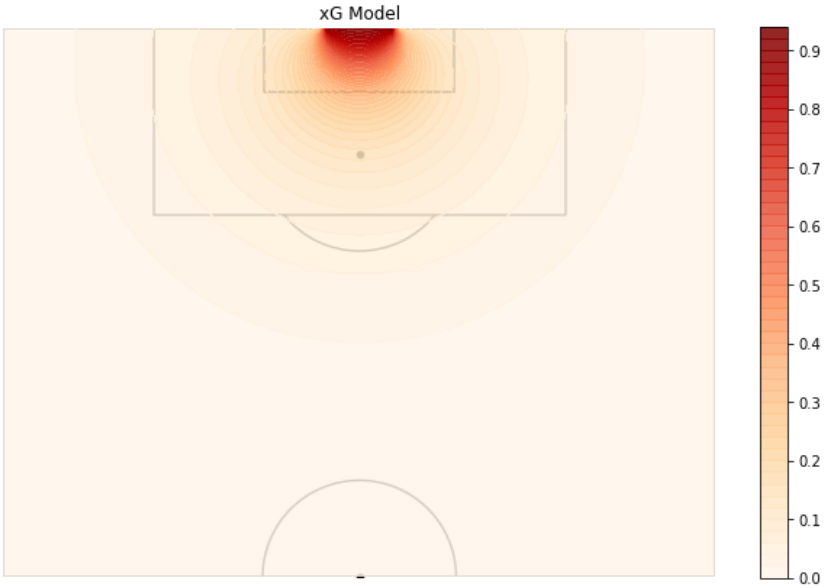


Fig 18: xG Model

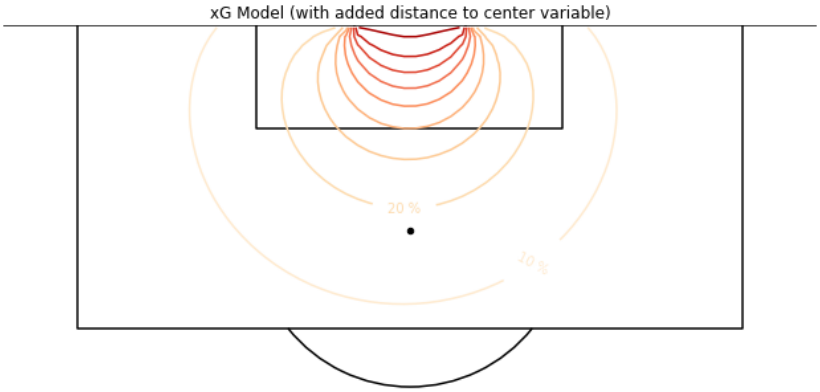


Fig 19: xG Model(with added distance variable)

Adding this new variable decreases the chance of scoring from small angles close to goal, but does little for further distances. This is probably because there are few shots taken from those positions and of the shots recorded, some might be miss-hit crosses that resulted in unlikely goals. Players rarely take shots from these wide, low angle positions. If we had 10 seasons worth of data, we would see the model begin to undervalue those types of shots.

4. Model Evaluation

If we want to assess the accuracy of our model, we have to test how well it can predict future events. But this raises another concern. How do we classify a shot with our new expected goals model? Unlike the Heaviside function, which gives hard classifications, the logistic regression model returns a probability of a shot resulting in a goal.

In order to make a classification, we have to define a threshold. This threshold essentially splits the logistic function, assigning goals for where the model lies above the threshold and misses below.

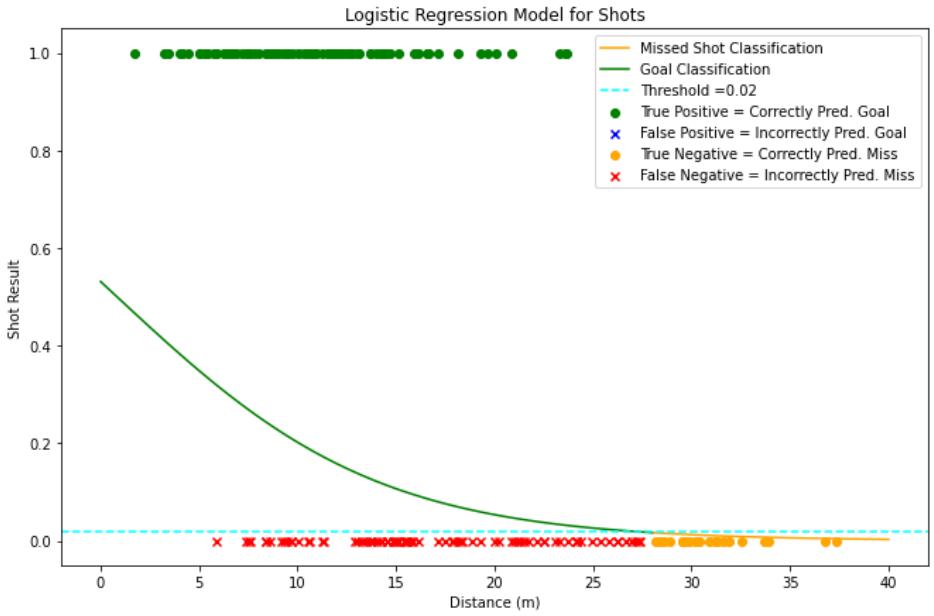
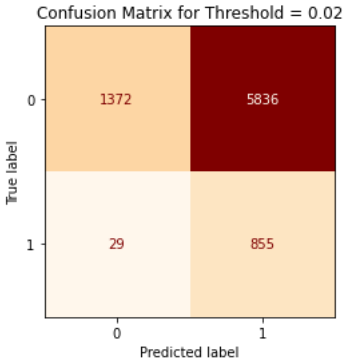


Fig 20: Logistic Regression Model for Shots

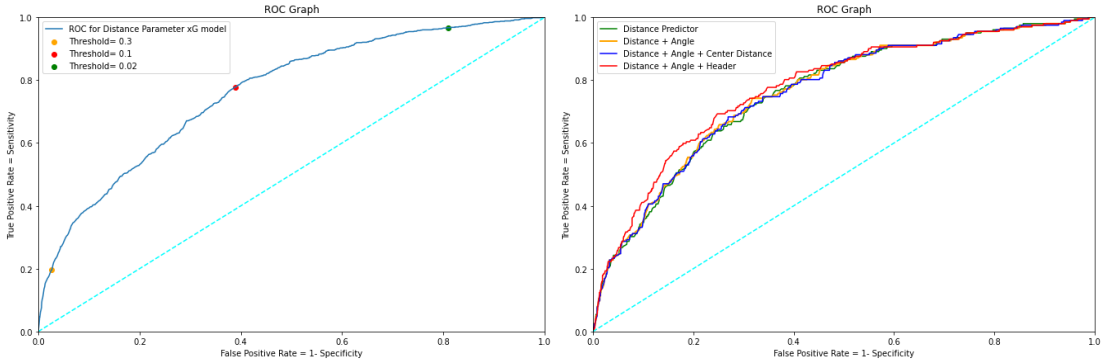
We now predict goals and misses at a more balanced rate. We have seen that three different thresholds produce vastly different predictions from our model. Where they vary is in their specificity and sensitivity; that is, their ability to correctly predict goals and misses. It's a trade-off.

Confusion Matrix



ROC Graph

Using a small step size, the ROC curve plots the model's ability to predict goals correctly versus its ability to incorrectly predict misses for different threshold values. As you move up the y-axis, the model better predicts goals, and as we move to the left along the x-axis, the model better predicts misses. It essentially maps the trade-off between predicting goals and predicting misses. The dashed line represents a model that has no predictive power and is essentially useless because for every correct classification, it also predicts an incorrect classification. Therefore, the further away our ROC curve is from the 45 degree line, the better overall job it does at classifying the test data. Another way of looking at it is that the larger the area under the curve, the better our model is in describing the test data. This is useful to us because we can use it to compare different models and see if there is any substantial advantage to adding more variables to our model.



Now we have something concrete when assessing our model. If we look closely, a model with distance and angle as input variables produces the same area under the curve (AUC) as the same model but with an added "distance to center" parameter.

Model	Test Accuracy
KNN	0.862371
Decision Trees	0.868285
Logistic Regression	0.892371
Neural Networks	0.885643
Naïve Bayes	0.321703

X. CONCLUSION

Ada boost performed best with an accuracy of 95% for predicting home team winning a soccer match.

Logistic Regression performed with an accuracy of 89% for classifying every goal-scoring chance, and the likelihood of scoring.

Goals, as we know, are mostly random. We need to remember that every shot is unique, comprised of hundreds of different variables. We did not consider where the goalkeeper is positioned, if the shot was taken with a weak foot or strong foot, shot height at point of contact, the game state, home field advantage, if there are many bodies between the goal and the shot, etc.