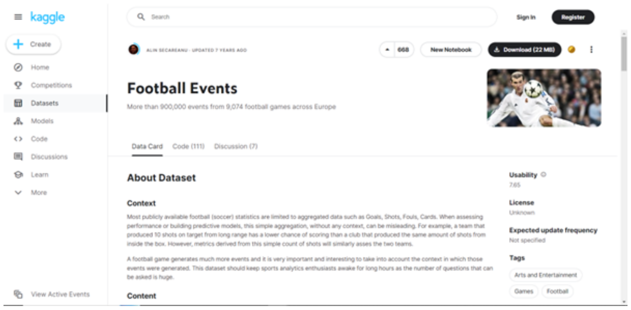
MTH 765P Mini-project

George Massey

15/1/24

1 Introduction

My mini-project is making use of a dataset comprised fully of football data which I found on Kaggle, the web page is shown in Figure 1, where you can see the black download button. The nice thing about this dataset is that it contains nearly 1 million footballing events from 9,000 football games across Europe which allows for a wide scope of potential analysis. The creator of this dataset has scraped text commentary from football matches across the top 5 European leagues from the 2011/2012 season to the 2016/2017 season. This is to add further context to the limited aggregated data that most public football data is comprised of which can result in misleading analysis. The type of data provided not only includes the most basic footballing events (e.g. shots, goals, fouls) but also more in depth information like the location of a shot, body part used, point of time in the match and many more.



Football Events

More than 900,000 events from 9,074 football games across Europe

About Dataset

Context

Most publicly available football (soccer) statistics are limited to aggregated data such as Goals, Shots, Fouls, Cards. When assessing performance or building predictive models, this simple aggregation, without any context, can be misleading. For example, a team that produced 10 shots on target from long range has a lower chance of scoring than a club that produced the same amount of shots from inside the box. However, metrics derived from this simple count of shots will similarly asses the two teams.

A football game generates much more events and it is very important and interesting to take into account the context in which those events were generated. This dataset should keep sports analytics enthusiasts awake for long hours as the number of questions that can be asked is huge.

Content

Usability

7.65

License

Unknown

Expected update frequency

Not specified

Tags

Arts and Entertainment

Games

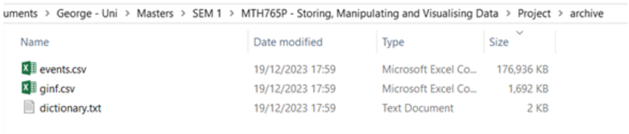
Football

Figure 1: Football Events Data Kaggle Webpage.

2 Obtaining/Acquiring the Data

I obtained the data from the website Kaggle which is freely available to download, at `https://www.kaggle.com/datasets/secareanualin/football-events/data?select=dictionary.txt`. The dataset was from 7 years ago but I felt that the data was so interesting it would still be useful. Also, up to date football data is very hard to find nowadays, normally you would have to pay as the data is in such high demand due to betting implications, and football teams wanting data on themselves and their oppositions.

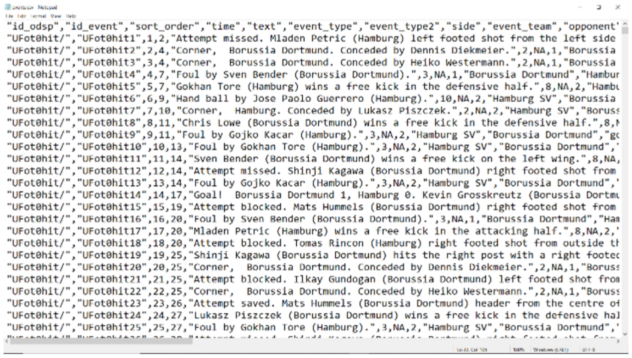
Figure 2 shows the unzipped archive folder that is the downloaded from the Kaggle web page, which contains three files: the first one is a small dictionary txt file containing a description of the categorical variables coded with integers, the second one is a large csv file containing all the events data about each game (shown in Figure 3), which was scraped from bbc.com and other sites, the third is another but smaller csv file containing aggregated game information and odds from oddsportal.com.



uments > George - Uni > Masters > SEM 1 > MTH765P - Storing, Manipulating and Visualising Data > Project > archive

Name	Date modified	Type	Size
events.csv	19/12/2023 17:59	Microsoft Excel Co...	176,936 KB
ginf.csv	19/12/2023 17:59	Microsoft Excel Co...	1,692 KB
dictionary.txt	19/12/2023 17:59	Text Document	2 KB

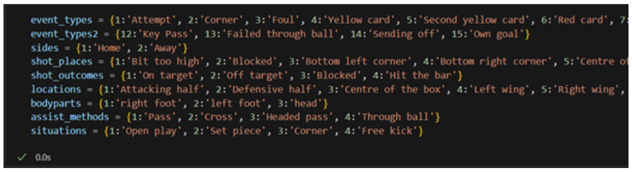
Figure 2: Unzipped Downloaded Archive Folder from Kaggle Web page.



```
"id_odsp","id_event","sort_order","time","text","event_type","event_type2","side","event_team","opponent'
"UFot0hit/","UFot0hit1",1,2,"Attempt missed. Mladen Petric (Hamburg) left footed shot from the left side
"UFot0hit/","UFot0hit2",2,4,"Corner,  Borussia Dortmund. Conceded by Dennis Diekmeier.",2,NA,1,"Borussia
"UFot0hit/","UFot0hit3",3,4,"Corner,  Borussia Dortmund. Conceded by Heiko Westermann.",2,NA,1,"Borussia
"UFot0hit/","UFot0hit4",4,7,"Foul by Sven Bender (Borussia Dortmund).",3,NA,1,"Borussia Dortmund","Hambur
"UFot0hit/","UFot0hit5",5,7,"Gokhan Tore (Hamburg) wins a free kick in the defensive half.",8,NA,2,"Hambu
"UFot0hit/","UFot0hit6",6,9,"Hand ball by Jose Paolo Guerrero (Hamburg).",10,NA,2,"Hamburg SV","Borussia
"UFot0hit/","UFot0hit7",7,10,"Corner,  Hamburg. Conceded by Lukasz Piszczek.",2,NA,2,"Hamburg SV","Boruss
"UFot0hit/","UFot0hit8",8,11,"Chris Lowe (Borussia Dortmund) wins a free kick in the defensive half.",8,N
"UFot0hit/","UFot0hit9",9,11,"Foul by Gojko Kacar (Hamburg).",3,NA,2,"Hamburg SV","Borussia Dortmund","gc
"UFot0hit/","UFot0hit10",10,13,"Foul by Gokhan Tore (Hamburg).",3,NA,2,"Hamburg SV","Borussia Dortmund",'
"UFot0hit/","UFot0hit11",11,14,"Sven Bender (Borussia Dortmund) wins a free kick on the left wing.",8,NA,
"UFot0hit/","UFot0hit12",12,14,"Attempt missed. Shinji Kagawa (Borussia Dortmund) right footed shot from
"UFot0hit/","UFot0hit13",13,14,"Foul by Gojko Kacar (Hamburg).",3,NA,2,"Hamburg SV","Borussia Dortmund",'
"UFot0hit/","UFot0hit14",14,17,"Goal!  Borussia Dortmund 1, Hamburg 0. Kevin Grosskreutz (Borussia Dortmu
"UFot0hit/","UFot0hit15",15,19,"Attempt blocked. Mats Hummels (Borussia Dortmund) right footed shot from
"UFot0hit/","UFot0hit16",16,20,"Foul by Sven Bender (Borussia Dortmund).",3,NA,1,"Borussia Dortmund","Ham
"UFot0hit/","UFot0hit17",17,20,"Mladen Petric (Hamburg) wins a free kick in the attacking half.",8,NA,2,'
"UFot0hit/","UFot0hit18",18,20,"Attempt blocked. Tomas Rincon (Hamburg) right footed shot from outside tl
"UFot0hit/","UFot0hit19",19,25,"Shinji Kagawa (Borussia Dortmund) hits the right post with a right footec
"UFot0hit/","UFot0hit20",20,25,"Corner,  Borussia Dortmund. Conceded by Dennis Diekmeier.",2,NA,1,"Boruss
"UFot0hit/","UFot0hit21",21,25,"Attempt blocked. Ilkay Gundogan (Borussia Dortmund) left footed shot from
"UFot0hit/","UFot0hit22",22,25,"Corner,  Borussia Dortmund. Conceded by Heiko Westermann.",2,NA,1,"Boruss
"UFot0hit/","UFot0hit23",23,26,"Attempt saved. Mats Hummels (Borussia Dortmund) header from the centre of
"UFot0hit/","UFot0hit24",24,27,"Lukasz Piszczek (Borussia Dortmund) wins a free kick in the defensive hal
"UFot0hit/","UFot0hit25",25,27,"Foul by Gokhan Tore (Hamburg).",3,NA,2,"Hamburg SV","Borussia Dortmund",'
```

Figure 3: Raw Football Events CSV File opened in Notepad.

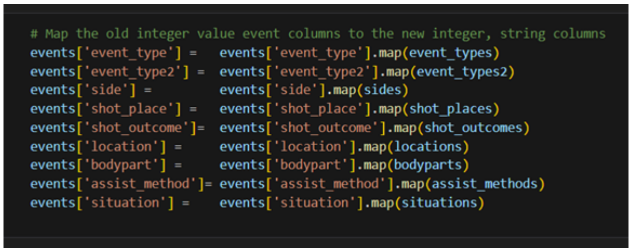
As you can see in Figure 3 each value is separated with a comma with most columns being string values denoted by the ”” connotations. This screenshot just shows the first few columns in which there are around 40 but it gives you an idea of how the data is stored. The data was very clean already so I only needed to do minimal manipulation. The main issue was that some of the columns are coded with integers, for example ’event_type’ is coded with a number between 0 and 11 which correspond to a certain event, e.g. ’Foul’. Therefore, my preprocessing started with attaching the integers with their meaning so that when I carry out my analysis I can use the actual definition instead of numbers. To do this I first read in the three files into separate pandas data frames. Then I encoded the integer values columns to python dictionaries in order to more easily refer to the data and mapped the new dictionary values to the old integer value columns, this is shown in Figure 4 and Figure 5.



```
event_types = {1:'Attempt', 2:'Corner', 3:'Foul', 4:'Yellow card', 5:'Second yellow card', 6:'Red card', 7:
event_types2 = {12:'Key Pass', 13:'Failed through ball', 14:'Sending off', 15:'Own goal'}
sides = {1:'Home', 2:'Away'}
shot_places = {1:'Bit too high', 2:'Blocked', 3:'Bottom left corner', 4:'Bottom right corner', 5:'Centre of
shot_outcomes = {1:'On target', 2:'Off target', 3:'Blocked', 4:'Hit the bar'}
locations = {1:'Attacking half', 2:'Defensive half', 3:'Centre of the box', 4:'Left wing', 5:'Right wing',
bodyparts = {1:'right foot', 2:'left foot', 3:'head'}
assist_methods = {1:'Pass', 2:'Cross', 3:'Headed pass', 4:'Through ball'}
situations = {1:'Open play', 2:'Set piece', 3:'Corner', 4:'Free kick'}
```

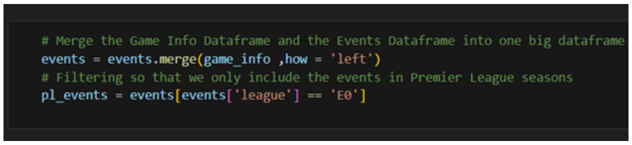
Figure 4: Python Code for Encoding Integer Value Columns.

Next, Figure 6 shows how I merged the game info data frame and the events data frame in order to create my subset of the data frame which is where I will be focusing on Premier League Seasons, which is encoded ’E0’. When I had a look at the merged dataframe I found lots of missing values in the columns that I was interested in so, as shown in Figure 7, I replaced them with a new ’unknown’ class abbreviated ‘UNK’. After using the .info() function it could be seen that a lot of the columns were of type object when they needed to be either category or boolean, for example the ’is_goal’ column needs to be a boolean variable as it is either True or False. Therefore, I fixed this using the .astype() function which is shown in figure 8.



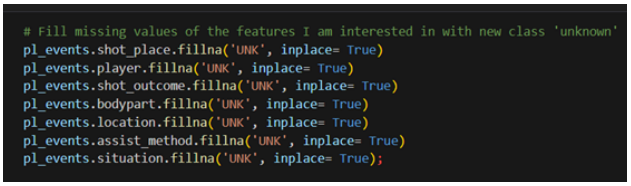
```python
# Map the old integer value event columns to the new integer, string columns
events['event_type'] =   events['event_type'].map(event_types)
events['event_type2'] =  events['event_type2'].map(event_types2)
events['side'] =         events['side'].map(sides)
events['shot_place'] =   events['shot_place'].map(shot_places)
events['shot_outcome']=  events['shot_outcome'].map(shot_outcomes)
events['location'] =     events['location'].map(locations)
events['bodypart'] =     events['bodypart'].map(bodyparts)
events['assist_method']= events['assist_method'].map(assist_methods)
events['situation'] =    events['situation'].map(situations)
```

Figure 5: Python Code for Mapping Integer Value Columns.



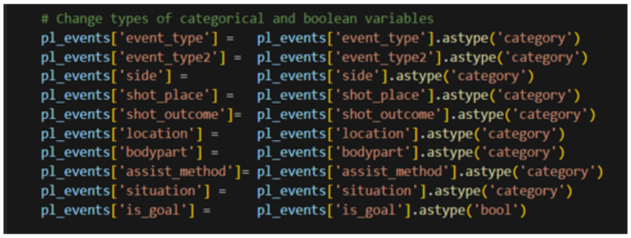
```python
# Merge the Game Info Dataframe and the Events Dataframe into one big dataframe
events = events.merge(game_info ,how = 'left')
# Filtering so that we only include the events in Premier League seasons
pl_events = events[events['league'] == 'E0']
```

Figure 6: Python Code for Merging Events and Info Dataframes.



```python
# Fill missing values of the features I am interested in with new class 'unknown'
pl_events.shot_place.fillna('UNK', inplace= True)
pl_events.player.fillna('UNK', inplace= True)
pl_events.shot_outcome.fillna('UNK', inplace= True)
pl_events.bodypart.fillna('UNK', inplace= True)
pl_events.location.fillna('UNK', inplace= True)
pl_events.assist_method.fillna('UNK', inplace= True)
pl_events.situation.fillna('UNK', inplace= True);
```

Figure 7: Python Code for Filling Missing Values with new 'UNK' class.



```python
# Change types of categorical and boolean variables
pl_events['event_type'] =   pl_events['event_type'].astype('category')
pl_events['event_type2'] =  pl_events['event_type2'].astype('category')
pl_events['side'] =         pl_events['side'].astype('category')
pl_events['shot_place'] =   pl_events['shot_place'].astype('category')
pl_events['shot_outcome']=  pl_events['shot_outcome'].astype('category')
pl_events['location'] =     pl_events['location'].astype('category')
pl_events['bodypart'] =     pl_events['bodypart'].astype('category')
pl_events['assist_method']= pl_events['assist_method'].astype('category')
pl_events['situation'] =    pl_events['situation'].astype('category')
pl_events['is_goal'] =      pl_events['is_goal'].astype('bool')
```

Figure 8: Python Code for Changing Column types to Boolean or Categorical.

3 Purpose of the Dataset

The dataset holds about 90% of the football matches played in the top 5 European Leagues between 2011-2017 allowing for a great scope of research. The author has used this dataset for creating predictive models in order to bet on future footballing outcomes but also welcomes other sports enthusiasts to answer other questions that you may be interested in, for example:

- Which teams are the best or sloppiest at holding the lead?
- What is the probability of a shot being a goal given it's location, shooter, assist method, shot placement, gamestate, number of players on the pitch, time, etc..
- Which teams or players make the best use of set pieces?
- In which leagues is the referee more likely to give a card?
- And many many more questions...

There are lots of ways this type of data can assist people betting on football matches but one basic example is that if their model shows that one player is committing a lot of fouls then they may bet on that player receiving a yellow card next game. Additionally, there could be lots of other users of this data, for example football teams themselves. Football teams will have a data analysis team to produce data on their own team, opposition teams and any players they are interested in buying.

4 Analysis

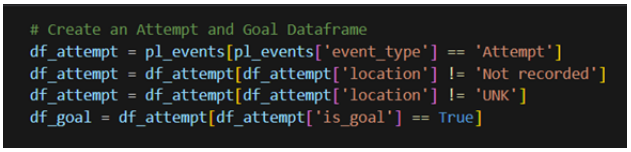
As mentioned earlier I filtered my events dataframe down to only the Premier League Seasons which is the top division in England, I did this because this is the league I am most interested in and would therefore, know the most about which I thought would help when producing my analysis. I am someone who is very interested in football statistics due to my interest in fantasy football games and in particular, data that models the probability of shots being a goal - known as expected goals (xG) models. Therefore, I decided to tackle one of the prompted questions above which is:

- What is the probability of a shot being a goal given it's location, shooter, league, shot placement, assist method, gamestate, number of players on the pitch, time, etc...

However, I limited my analysis to the ones I was most intrigued in which was: location of the shot, the shooter, shot placement and point of time in the game.

I started off my analysis by creating two separate dataframes, shown in Figure 9. One dataframe for all the events that were attempts at goal, whether they missed or scored, and a dataframe for only the attempts that resulted in a goal (where is_goal == True). Additionally, I made sure to remove any rows that were 'UNK' or 'Not recorded'. The first question I looked at was:

- What is the probability of a shot being a goal given its location?



```
# Create an Attempt and Goal Dataframe
df_attempt = pl_events[pl_events['event_type'] == 'Attempt']
df_attempt = df_attempt[df_attempt['location'] != 'Not recorded']
df_attempt = df_attempt[df_attempt['location'] != 'UNK']
df_goal = df_attempt[df_attempt['is_goal'] == True]
```

Figure 9: Python Code for Creating My Attempts and Goals Dataframes.

I first made a attempt and goal location dataframe by grouping by location, but as there was around 20 different locations I filtered so that the dataframes only included locations where more than 10 goals had been scored over the period, which mostly removed location from very long range, i.e. more than 40 yards. I then plotted a bar chart for the number of attempts and the number of goals overlayed on each other to start seeing which shooting locations are the most promising. However, this led to a poor visulisation as there was a great variety in the number of attempts across the locations, shown by the pie chart in Figure 10. So to truly find which shooting locations had the highest probability of scoring from, I proportioned the goals and attempts which led to the visualisation shown in Figure 11:

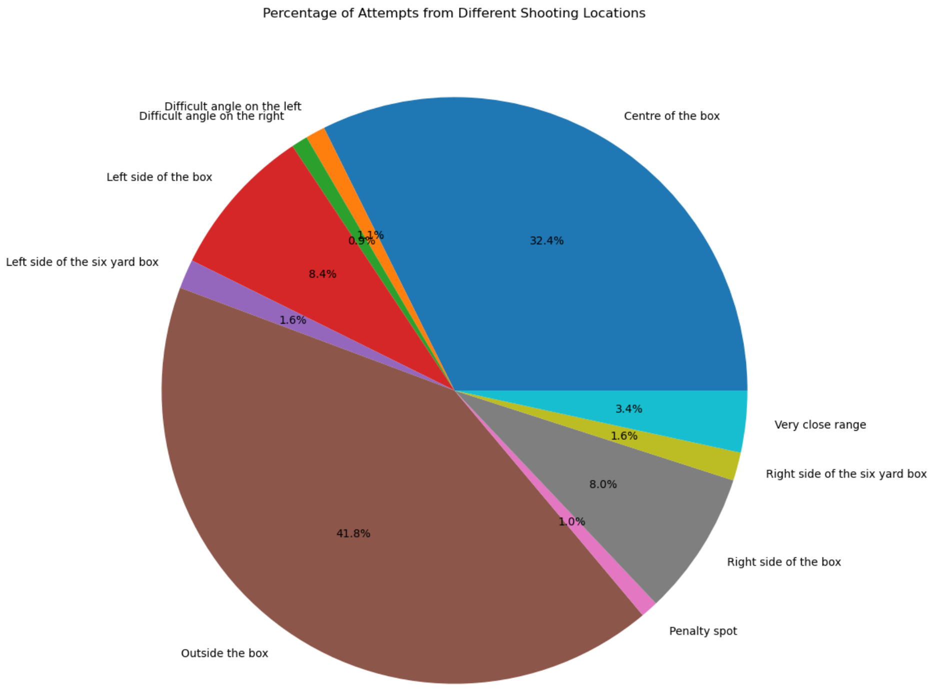


Figure 10: Pie Chart for Percentage of Attempts taken from each Shooting Location.

Figure 11 clearly represents that shots from the penalty spot are the most probable of being a goal and shots from very close the second which is to be expected, but what is interesting is that this data suggests that around 1 in 5 penalties are missed giving penalties a value of 0.8 goals (this is in line with most expected goal models). Interesting finds from this figure is that shots from the left side and right side of the pitch seem to more effective than from the centre of the pitch which could be explained by the goal keeper being able to make the angle harder when the player is in central positions. Shots from outside the box and longer range are obviously the least effective shooting positions even though over 40% of shots are taken from these locations, as shown in Figure 10, which suggests teams should perhaps try and work the ball into more effective shooting locations.

- What is the probability of a shot being a goal given its placement?

So this question refers to the placement of the shot, i.e. the centre of the goal. Similarly to the previous question I filtered out the placements where zero goals were scored which included shot placements like,

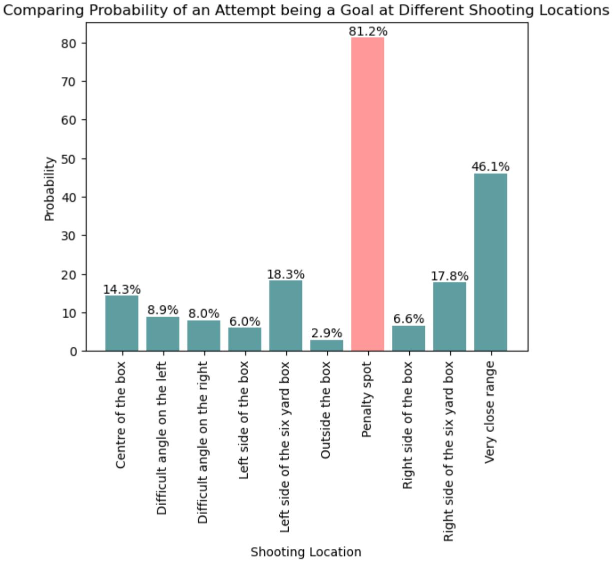


Figure 11: Bar Chart for Probability of an Attempts being a Goal from each Shooting Location.

'misses to the left'. This left 5 different placements: Bottom left, Bottom right, Centre of the goal, Top left, Top right. The pie chart in Figure 12 shows that the first 3 placements make up approximately 80% of the shots that resulted in goals. Figure 13 then shows the probability of shots in each placement resulting in a goal. As expected here the shots placed at the centre of the goal are the least effective which can mostly be explained by the shots being easier to save as the keeper normally stands in the middle of the goal. Then the other 4 placements are all fairly equal with shots at the bottom left corner being the slightly more effective placement. It suggests that if you are shooting at the left side of the goal it is more effective to keep the ball low where as on the right hand side shooting the ball high is slightly more effective.

- What is the probability of a shot being a goal given its shooter?

I first created two dataframes for the shot counts and goal counts for each player and in order to make a meaningful analysis I only included players with more than 15 goals across all the premier league seasons in the database. Figure 14 compares the attempts and goals of the players, one glance at this bar chart would show that Sergio Aguero has the most goals over this period (52). However, we need to add the context that he has also had the most attempts on goal (329), so is he the best shooter of the ball in the Premier League?

Figure 15 shows that the majority of effective goal scorers fall between 11-16% goal conversion rate, where conversion rate is a players total goals divided by their total attempts. Diego Costa has the best goal conversion rate at 19.2% with Mesut Ozil closely behind at 18.9% which shows that they need less shots to score each goal. However, Ozil had the least amount of attempts at goal out of all the players (90) with 15+ goals which suggests that his conversion rate may not be sustainable as the sample size isn't large enough. Whereas someone like Harry Kane who had far more attempts (278) still held a strong conversion rate at 15.1% suggesting he might be a more consistent finisher. There was one player who was clearly the least clinical which was Phillipe Coutinho with a conversion rate of 7.2%, the only player with a conversion rate under 10%. This could be explained by him being a midfielder, so his shots are more likely to be further out from goal and less likely to result in a goal.

- What is the probability of a shot being a goal given the time in the game?

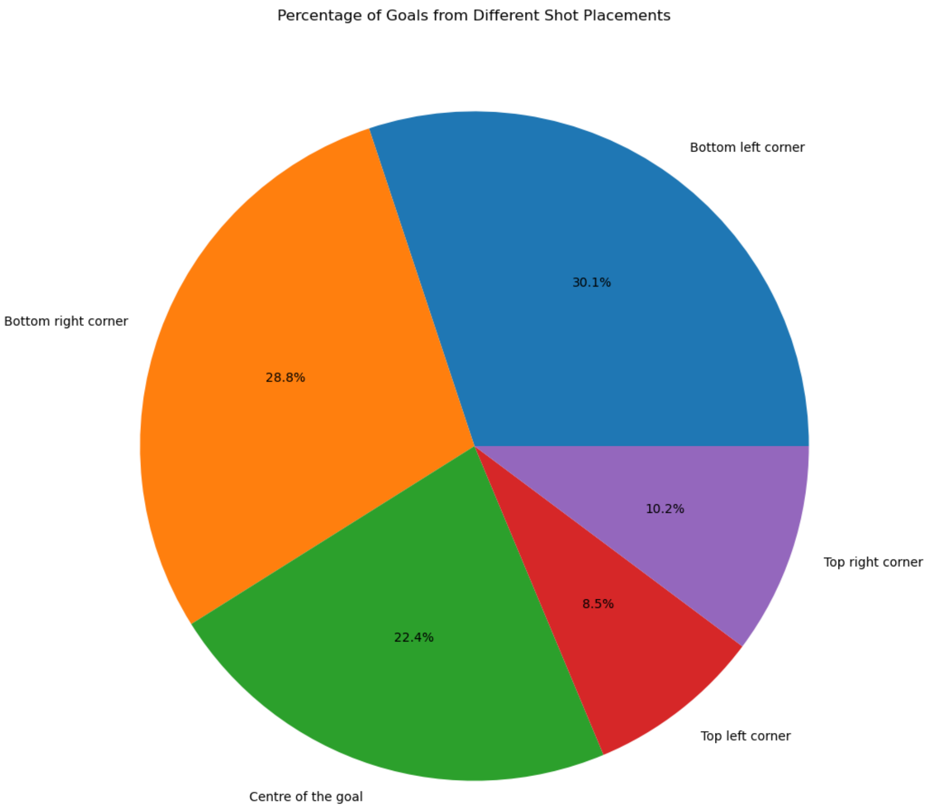


Figure 12: Pie Chart for Percentage of Attempts taken from each Shooting Location.

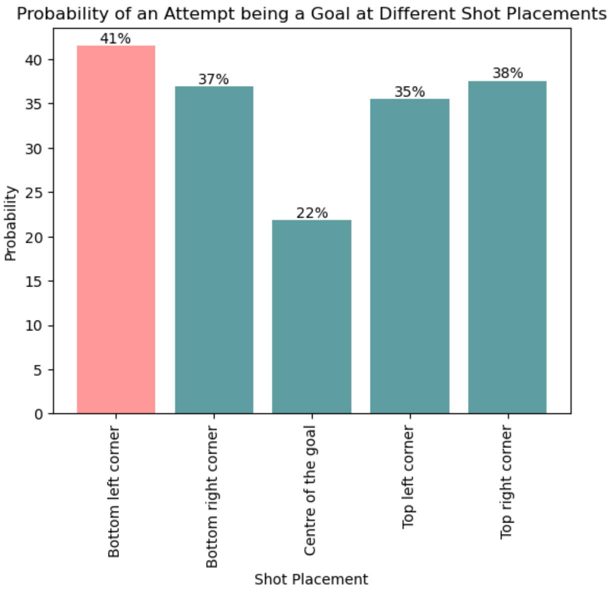


Figure 13: Bar Chart for Percentage of Attempts taken from each Shooting Location.

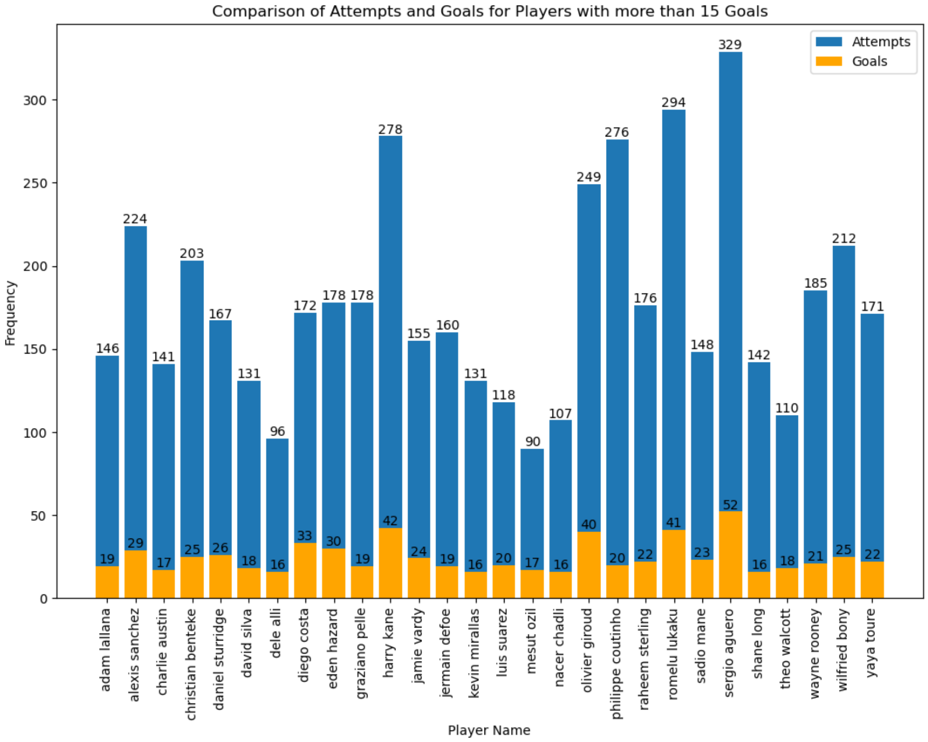


Figure 14: Bar Chart for Comparison of Goals and Attempts for Players with More than 15 Goals.

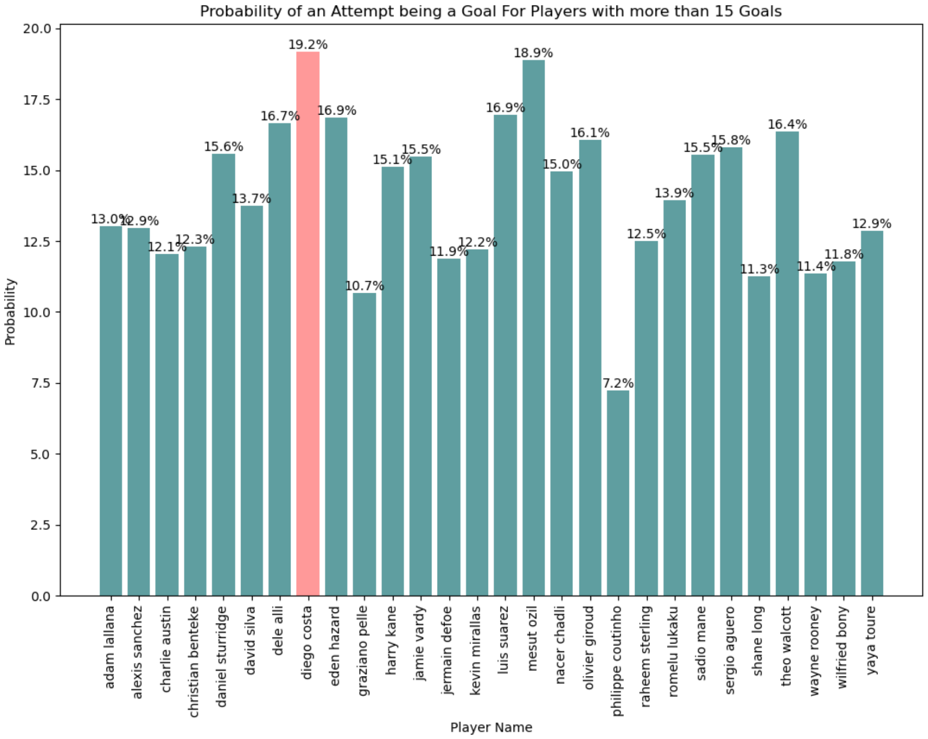


Figure 15: Bar Chart of Conversion Rate for Players with More than 15 Goals.

This visualisation was slightly different as I split the minutes of the game into 10 minute intervals using the .cut() function. The figure 16 shows that as the game goes on there is a slight increase in the number of attempts and the number of goals, even though the last time interval has the least amount of attempts this is because the matches usually end before the 100th minute. Furthermore, the most attempts and goals were in the 40-50 and the 80-90 minute time intervals which is likely to do with tiredness and teams playing more reckless towards the end of the half. Additionally, the same is seen when I visualise the probabilities (figure 17), which shows that maybe the defender and keepers may start to lose concentration as the game wears on. However, note that there is only a very slight increase in the probabilities.

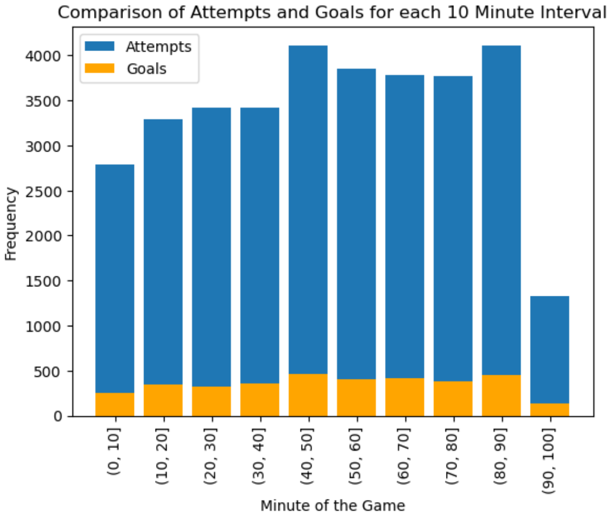


Figure 16: Bar Chart of Frequency of Attempts and Goals for each 10 Min Time Interval.

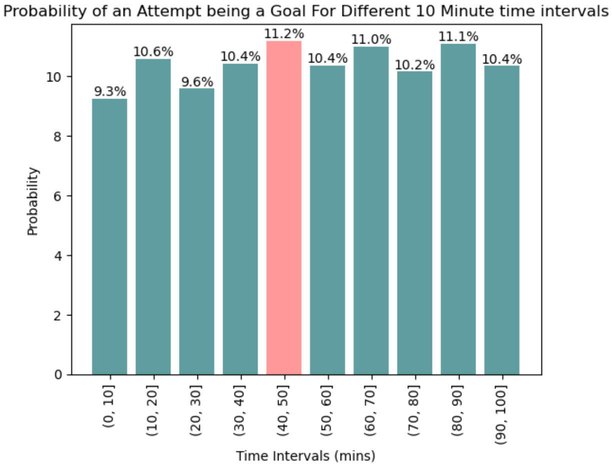


Figure 17: Bar Chart of Probability of an Attempt being a Goal for each 10 Min Time Interval.

The Libraries I have used in this section are:

- Seaborn - for bar charts
- Numpy - for numerical operations
- Pandas - for storing in dataframes
- Matplotlib.pyplot - for pie and bar charts

5 Questions

What would you do if you had to do more analysis on the dataset? Do you need additional data?

If I was to have longer with this project I would extend my analysis to look at even more features of a shot which may affect its probability of being a goal, which is where I may need to look for additional data. Once I have looked at all potential features I could create a model where if I was given the characteristic of a shot my model would produce its probability of being a goal (this is known as xG). I could then use this model to predict a team or players performance based on whether they are under or over performing their expected goals, i.e. they have scored less or more goals than they should have given the chances they have created. This could be used for fantasy football purposes, i.e. bringing a player into your team who has scored less goals than they should have in the hope that they will start scoring more based off the chances they are getting. However, this comes with the risk that the player continues to under perform and is just a poor shooter.

Are there interesting questions that one could try to answer?

As outlined above in section 3 there are many other questions one could try to answer with this dataset as I have just gone down one of the many paths of analysis. Personally, I think some of the more interesting questions would be looking at patterns of attack for different teams as this would be useful analysis for opposing teams.