# Introduction

## Description of the club Arsenal

Arsenal Football Club is an English professional soccer club based in London and plays in the Premier League, the top tier of English soccer. The club has won 13 league titles, a record 14 FA Cups, two League Cups, 16 FA Community Shields, one European Cup Winners' Cup and one Inter-Cities Fairs Cup. In terms of trophies won, they are the third most successful club in English soccer (Wikipedia n.d.).

However, despite being one of the most successful teams, one of the richest, always counting with highly recognized players and having one of the largest fans in England, its football performance has been decreasing over the years and this is evidenced by the fact that the last time Arsenal was champion of the Premier League was in the 2003/2004 season.

## Description of the database

The database to be used has been obtained from the Kaggle website. This database has important statistics of Premier League matches from 2010 to mid-2021 (Kaggle 2021). Similarly, this data is available for free on the English Premier League website and allows tracking of the last few seasons. The original and updated archive is available on the official Premier League website.

# Objective of the project

## Main Objective

Develop a data storytelling in order to explain how the level of the football club Arsenal FC has deteriorated during the last decade.

## Specific objectives

* Explore and clean the database to obtain key parameters that help determine the performance of a football team.
* Creation of a variable indicating the number of points obtained at the end of a match
* Creation of the final rankings of each season.
* Compare the performance of Arsenal against the best teams
* Compare the performance of Arsenal against the champions of the premier league.
* Develop a prediction model to determine the winner of the match

# Methodology

## Importing libraries and database

The first step is to import the necessary libraries to be able to explore, analyze and visualize the information of the database. For this purpose, the libraries to be used are shown Figure 1.

Text

Description automatically generated

Figure 1. Libraries used for the project

Once the libraries have been imported, the next step is to import the database. The database is a csv file which can be imported using the function “read\_csv” of the pandas’ module.

This database has a size of 4070,114. In other words, it has a total of 4070 entries(matches) and also has 114 columns (variables).

## Removing unnecessary columns

The database has a large number of variables, which contain important information on every game played since 2010. However, it is necessary to reduce this number in order to perform an adequate analysis of the teams' performance.

The column 'Unnamed: 0' is identical to the index. Since the index is already integrated in the table, there is no need to store it in a separate column. The column 'link\_match' represents links that directs you to the match information. There is no need to keep this column since most of the data is already stored in other columns.

To further study the performance of a team we need information related to offensive and defensive indicators. The database contains too much information on these aspects. This information is contained in different columns that represent halftime and fulltime match results data. For this analysis, the final score of the match is required, which helps to determine who was the winner.

Therefore, the following columns will not be considered are shown on the Table 1 :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Columns to delete | | | | |
| Unnamed: 0 | link\_match | result\_full | result\_ht | goal\_home\_ft |
| goal\_away\_ft | sg\_match\_ft | goal\_home\_ht | goal\_away\_ht | sg\_match\_ht |
| clearances\_avg\_H | corners\_avg\_H | fouls\_conceded\_avg\_H | offsides\_avg\_H | passes\_avg\_H |
| possession\_avg\_H | red\_cards\_avg\_H | shots\_avg\_H | shots\_on\_target\_avg\_H | tackles\_avg\_H |
| touches\_avg\_H | yellow\_cards\_avg\_H | goals\_scored\_ft\_avg\_H | goals\_conced\_ft\_avg\_H | sg\_match\_ft\_acum\_H |
| goals\_scored\_ht\_avg\_H | goals\_conced\_ht\_avg\_H | sg\_match\_ht\_acum\_H | performance\_acum\_H | clearances\_avg\_A |
| corners\_avg\_A | fouls\_conceded\_avg\_A | offsides\_avg\_A | passes\_avg\_A | possession\_avg\_A |
| red\_cards\_avg\_A | shots\_avg\_A | shots\_on\_target\_avg\_A | tackles\_avg\_A | touches\_avg\_A |
| yellow\_cards\_avg\_A | goals\_scored\_ft\_avg\_A | goals\_conced\_ft\_avg\_A | sg\_match\_ft\_acum\_A | goals\_scored\_ht\_avg\_A |
| goals\_conced\_ht\_avg\_A | sg\_match\_ht\_acum\_A | performance\_acum\_A | clearances\_avg\_home | corners\_avg\_home |
| fouls\_conceded\_avg\_home | offsides\_avg\_home | passes\_avg\_home | possession\_avg\_home | red\_cards\_avg\_home |
| shots\_avg\_home | shots\_on\_target\_avg\_home | tackles\_avg\_home | touches\_avg\_home | yellow\_cards\_avg\_home |
| goals\_scored\_ft\_avg\_home | goals\_conced\_ft\_avg\_home | sg\_match\_ft\_acum\_home | goals\_scored\_ht\_avg\_home | goals\_conced\_ht\_avg\_home |
| sg\_match\_ht\_acum\_home | performance\_acum\_home | clearances\_avg\_away | corners\_avg\_away | fouls\_conceded\_avg\_away |
| offsides\_avg\_away | passes\_avg\_away | possession\_avg\_away | red\_cards\_avg\_away | shots\_avg\_away |
| shots\_on\_target\_avg\_away | tackles\_avg\_away | touches\_avg\_away | yellow\_cards\_avg\_away | goals\_scored\_ft\_avg\_away |
| goals\_conced\_ft\_avg\_away | sg\_match\_ft\_acum\_away | goals\_scored\_ht\_avg\_away | goals\_conced\_ht\_avg\_away | sg\_match\_ht\_acum\_away |
| performance\_acum\_away |  |  |  |  |

Table 1. Columns to delete

Additionally, 5 new columns are added containing the home team goals, away team goals, home team points, away team points and the winner of the match. To determine the points obtained, it is enough to simply compare the number of goals scored by each team. The team that scored the most goals at the end of the match gets 3 points and is the winner. Both teams get 1 point if they scored the same number of goals. The team with fewer goals scored gets 0 points and is the loser.

## Identifying and/or deleting duplicate values and NaN management

Datasets that contain duplicates and missing values may contaminate the training data with the test data or vice versa as well as lead to wrong interpretation of the different variables. A good way to get a quick feel for the data is to take a look at the information of the DataFrame. As shown on the Table 2 the Dataframe has a total of 4070 entries, 33 columns and non-null values were found.

|  |  |
| --- | --- |
| RangeIndex | 4070 entries, 0 to 4069 |
| Data columns | total 33 columns |
| Column | **Non-Null Count** |
| season | 4070 non-null |
| date | 4070 non-null |
| home\_team | 4070 non-null |
| away\_team | 4070 non-null |
| home\_clearances | 4070 non-null |
| home\_corners | 4070 non-null |
| home\_fouls\_conceded | 4070 non-null |
| home\_offsides | 4070 non-null |
| home\_passes | 4070 non-null |
| home\_possession | 4070 non-null |
| home\_red\_cards | 4070 non-null |
| home\_shots | 4070 non-null |
| home\_shots\_on\_target | 4070 non-null |
| home\_tackles | 4070 non-null |
| home\_touches | 4070 non-null |
| home\_yellow\_cards | 4070 non-null |
| away\_clearances | 4070 non-null |
| away\_corners | 4070 non-null |
| away\_fouls\_conceded | 4070 non-null |
| away\_offsides | 4070 non-null |
| away\_passes | 4070 non-null |
| away\_possession | 4070 non-null |
| away\_red\_cards | 4070 non-null |
| away\_shots | 4070 non-null |
| away\_shots\_on\_target | 4070 non-null |
| away\_tackles | 4070 non-null |
| away\_touches | 4070 non-null |
| away\_yellow\_cards | 4070 non-null |
| home\_goals | 4070 non-null |
| away\_goals | 4070 non-null |
| points\_home | 4070 non-null |
| points\_away | 4070 non-null |
| winner | 4070 non-null |

Table 2. DataFrame information Missing values

Entries with missing values will lead models to misunderstand features, and outliers will undermine the training process – leading your model to “learn” patterns that do not exist in reality. To identify duplicate values the "duplicated" function is used in combination with the "sum" function. If the sum is greater than 0 it means that duplicates have been found. For this specific database no duplicate values were found.

## Formatting the date

For a better analysis of the data, dates are represented in a time format. Python allows to do this in a simple way using the "date time" module. As shown in Figure 2, we use the to\_datetime function of pandas using as input the "date" column of the database.

A picture containing shape

Description automatically generated

Figure 2. Formatting the date

## Adjusting the data

For the elaboration of the graphs a new dataframe will be created. This dataframe will contain the statistics per season from 2010 to 2020. The 2021 season will not be considered because the available data only has information up to the middle of the season. This is done in order to have the information more summarized and easier to interpret visually.

A picture containing timeline

Description automatically generated

Figure 3. Custom made functions for total and average statistics per season

As shown in Figure 3, two functions were created to obtain the total statistics per season as well as the average per season. These functions receive as input the database, the columns to be considered and the name to be assigned to them.

For example, to calculate the total points per season we will use the following inputs df, 'points\_home','points\_away','Points(home)','Points(away)'. This will create a new dataframe containing the total home points and away points per season for each team. Analogously, the total and average statistics will be obtained for the following information: goals scored, goals conceded, passes, possession, shots and shots on target.

As shown in Figure 4, the total and average statistics dataframes per season will be merged into one big dataframe called "performance\_per\_season" with the help of the "merging" function. This database will be the basis for the creation of graphs.

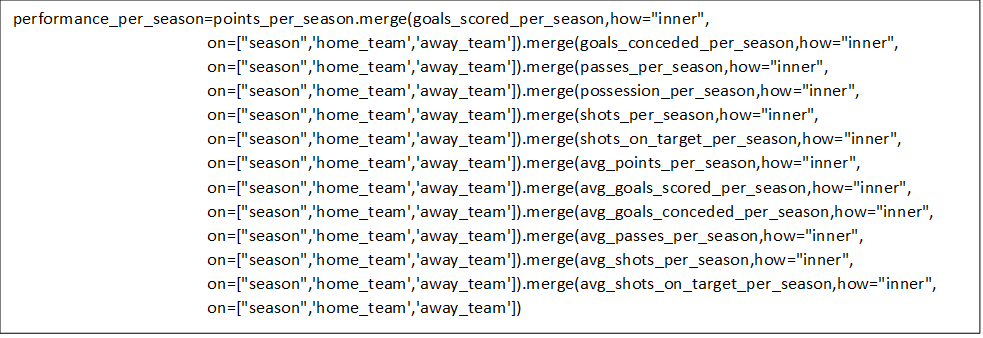


Figure 4. Creation of the dataframe “performance\_per\_season”

Once the dataframes are merged, the rows in which the team name is the same in the columns "team home" and "team away" will be deleted. This is because when the merging is done, rows are created between the matches of each team against the others. This information is not relevant because the objective is to determine the statistics of each team per season and not how they performed against other teams.

New columns were created to summarize the performance information of each team. To obtain the totals, simply add the home and away columns for each aspect of the match. For example, for the total number of points, the Points(home) and Points(away) columns will be added together. Similarly, this will be done for goals scored, goals conceded, passes, shots and shots on target.

For the average, the home and away columns are added together, and the result is divided by 2. Finally, the result is rounded and expressed with two decimal places. Additionally, a new column called "Table\_Standing” will be created. This column will show the position occupied by each team at the end of the season. In order to determine who is the season champion, the data is drawn according to the number of points. In case two or more teams have the same number of points, they are drawn according to the goal difference.

In order to evaluate a team's performance, it is necessary to compare its statistics against the best teams of each season. For this report, the best teams are defined as the teams that placed in the top four. The top four teams are the ones that qualify for the Champions League, the most important club championship in Europe. The best teams are shown in Table 3.

|  |
| --- |
| **Best teams** |
| * Manchester United * Chelsea * Manchester City * Arsenal * Tottenham Hotspur * Liverpool * Leicester City |

Table 3. Best and worst teams of the Premier League from 2010 to 2020

## Predictive modeling

Predictive modeling is a mathematical process used to predict future outcomes by analyzing patterns in a given set of input data. It is a type of data analysis that uses current and historical data to predict activities, behaviors and trends (TechTarget n.d.).

For this research report, a predictive model will be developed to determine the winner of a game using information on the offensive and defensive aspects of each team. The problem in question is a classification problem. The developed model must be able to determine if a team is a winner, loser, or was a draw.

For this, three machine learning algorithms will be implemented in Python using Scikit-learn. The machines learning algorithms to be trained are LogisticRegression, KNeighborsClassifier and RandomForestClassifier. Then, the objective is to identify the machine learning algorithm that best fits the problem by comparing their performances and selecting the one with the best score.

The first step is to create a standardizer because the variables to be considered have different scales. In parallel, a correlation matrix is created to determine the relationship between the different variables. ‘season’,’date’,’home\_team’,’away\_team’,'home\_goals','away\_goals','points\_home' and 'points\_away' will not be considered in the correlation matrix. The information contained in these columns directly determines the winner or loser of a match because they have the final result of the match.

Then the ‘data’ and ‘target’ variables are created, which will be used to train the model. The ‘data’ variable contains all the variables discarding those mentioned above and also discarding the winner column. The ‘target’ variable will only consider the information of the winner column.

Finally, the model is trained so that it can be used with the algorithms. The variables ‘lr’,’knn’ and ‘clf’ are created, which represent respectively the algorithms LogisticRegression, KNeighborsClassifier and RandomForestClassifier. Now, the score of each algorithm is evaluated and the best one is chosen. This algorithm is optimized to improve its performance in order to obtain better classification results.

# Analysis and results

## Performance of the English Premier League’s Teams

The performance of the best and worst teams is analyzed according to points obtained, goals scored (offensive aspect) and goals conceded (defensive aspect).

As shown in Figure 5, the team with the best performance is Manchester City. This team has managed to stay in the top positions since 2010. Arsenal on the other hand has been in decline. From the 10/11 to 15/16 season they managed to stay in the top 4. However, it began to occupy lower positions, reaching the eighth position in the table in the 19/20 season. Leicester was the surprise team winning the premier league in 14/15. However, apart from this achievement, the team has had a regular performance.

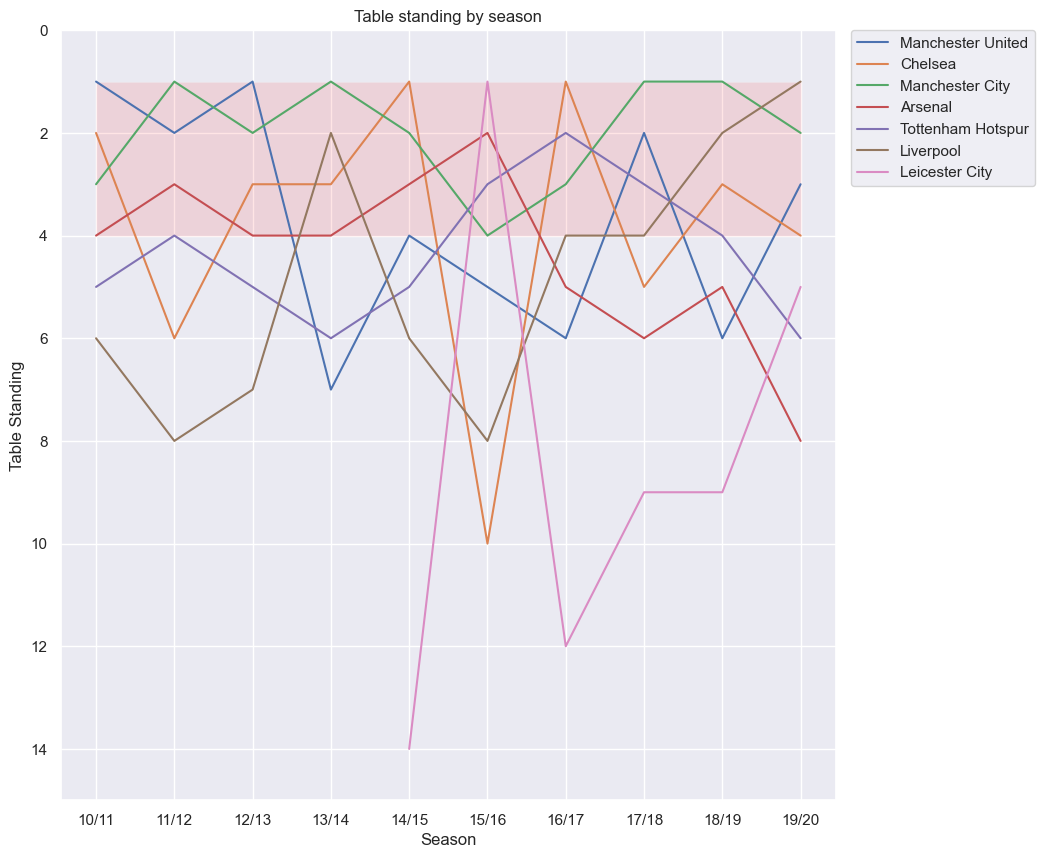


Figure 5. Table standing of the best teams by season

As shown in Figure 6, Manchester city is the team with the best offensive performance (goals scored). The rest of the teams show ups and downs. Arsenal's performance in this aspect is a little above (excluding Manchester city). However, from 16/17 season onwards, the number of goals scored by Arsenal started to decline.

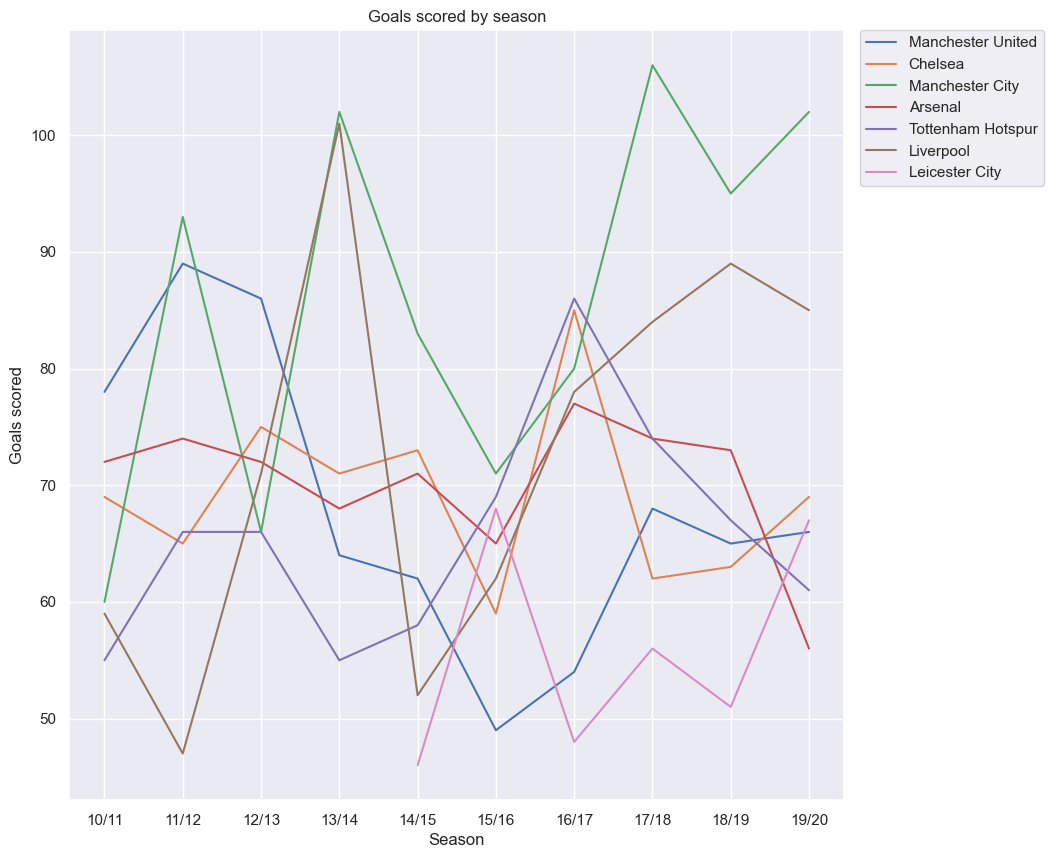


Figure 6. Goals scored by season (best teams)

Manchester City is the team with the best defensive performance (goals conceded). Arsenal's performance has worsened in this aspect. As shown in Figure 7, from the 15/16 season onwards, Arsenal's number of goals conceded started to increase. It can be inferred that Arsenal has been scoring fewer and fewer goals and conceding more and more goals. This is a clear indication of the decline in their performance. As a team begins to get more conceded more goals than scored, it means that they are not winning enough games to challenge for top spot in the table.

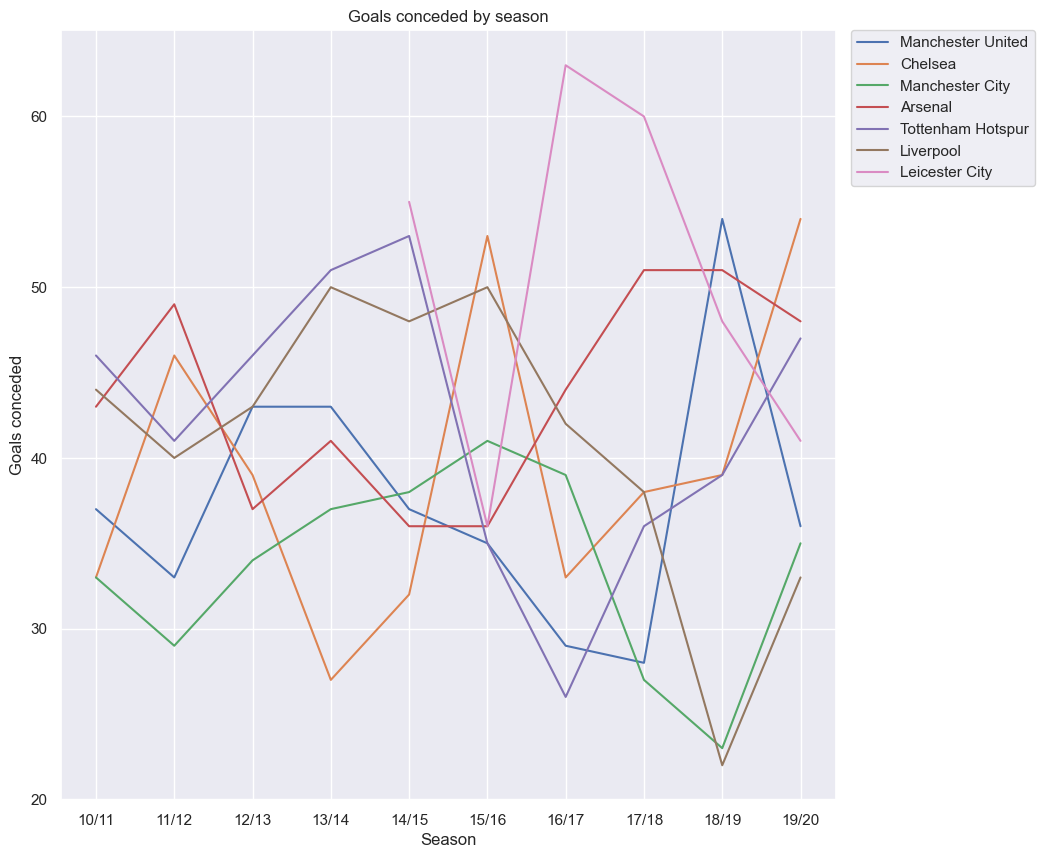


Figure 7. Goals conceded by season (best teams)

An important aspect to evaluate the performance of a football team is the number of points obtained at the end of each season. Table 4 shows the English league champions since 2010. It is important to note that according to the data provided, for a team to be crowned champion, it must obtain more than 80 points at the end of the season and the number of goals scored must be greater than the number of goals conceded (around 50%).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Season | Team | Table standing | Total points | Total goals scored | Total goals conceded |
| 10/11 | Manchester United | 1 | 80 | 78 | 37 |
| 11/12 | Manchester City | 1 | 89 | 93 | 29 |
| 12/13 | Manchester United | 1 | 89 | 86 | 43 |
| 13/14 | Manchester City | 1 | 86 | 102 | 37 |
| 14/15 | Chelsea | 1 | 87 | 73 | 32 |
| 15/16 | Leiscester City | 1 | 81 | 68 | 36 |
| 16/17 | Chelsea | 1 | 93 | 85 | 33 |
| 17/18 | Manchester City | 1 | 100 | 106 | 27 |
| 18/19 | Manchester City | 1 | 98 | 95 | 23 |
| 19/20 | Liverpool | 1 | 99 | 85 | 33 |

Table 4. Champions of the premier league

From 10/11 onwards, arsenal has not scored more than 80 points per season. As shown in Figure 8, Arsenal's best season was in 13/14, in which the team scored a total of 79 points. Nevertheless, this was not enough, and Arsenal ranked fourth in the table. The champion of this season was Manchester City and they obtained 86 points.

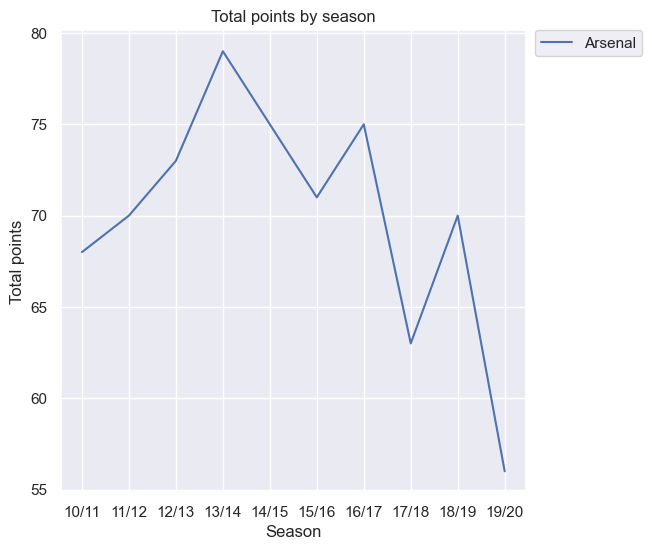


Figure 8. Total points obtained by Arsenal

To compare more concretely the performance of the best teams and the arsenal, a polar chart was made. It is used to demonstrate data in two-dimensional for two or more data series. The axes start on the same point in a radar chart. This chart is used to compare more than one or two variables. For this report, this chart is composed of variables goals scored, goals conceded, passes, points, shots and shots on target. The variables were standardized on a scale of 0 to 3 so that they can be better represented visually.

Arsenal's offensive performance is very similar to the performance of the last Premier League champions. The number of goals scored, and points scored is very similar between Arsenal, Liverpool, Manchester United and Chelsea. Manchester city is the team that clearly has the best score in each variable.

On the other hand, the defensive aspect is definitive to understand the performance of the teams. As shown in Figure 9, despite having similar statistics on the offensive performance, Arsenal concedes more goals than the champions. This translates to arsenal's defensive performance being below that of other teams. The exception is Leicester City whose performance is below what would normally be expected of a champion.

Chart, shape, radar chart, polygon

Description automatically generated

Figure 9. Performance of Arsenal vs Champions.

## Predictive modeling

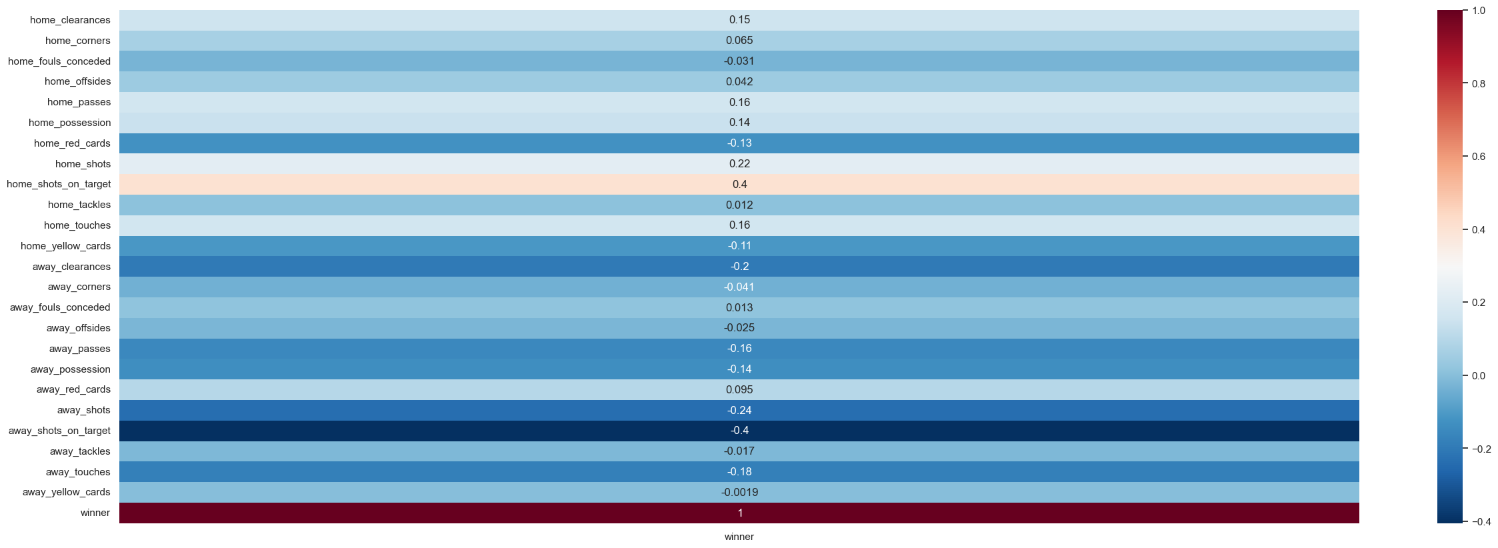
The first step was to develop the correlation matrix to determine the relationship between the variables. As shown in Figure 10, the variables most closely related to the winner of a match are: 'home\_clearances','home\_passes','home\_possession','home\_red\_cards','home\_shots','home\_shots\_on\_target','home\_touches','home\_yellow\_cards','away\_clearances','away\_passes','away\_possession','away\_shots','away\_shots\_on\_target','away\_touches','winner'

Figure 10. Correlation Matrix

The model is trained after standardizing the data and creating the data and target variables. Then, the scores of each algorithm are evaluated to determine the best one. As shown in Table 5, the logistic regression algorithm has a score of 0.66 which is higher compared to the other algorithms. Therefore, this algorithm is preferred because it can better classify whether a team is a winner.

|  |  |
| --- | --- |
| Machine learning algorithm | Score |
| LogisticRegression | 0.6611001964636543 |
| KNeighborsClassifier | 0.5618860510805501 |
| RandomForestClassifier | 0.6208251473477406 |

Table 5. Scores of the machine learning algorithms

The parameters used in the initial analysis were the default parameters provided by Python. To obtain better results, the model will be optimized using different parameters. Figure x shows the parameters that were used to optimize the model. Using the python GridSearchCV module it was determined that the best parameters are: {'C': 0.05, 'penalty': 'none', 'solver': 'saga'}

Text

Description automatically generated

Figure 11. Parameters for logistic regression.

Despite optimizing the model, the score did not improve considerably. However, it is a good score and it can be concluded that the algorithm performs an adequate classification of which team will be the winner. Figure 12 shows the cross matrix in which the classification performed by the algorithm can be observed. The letter A represents that the visiting team will be the winner, the letter D indicates that it will be a tie and the letter H indicates that the home team will be the winner.

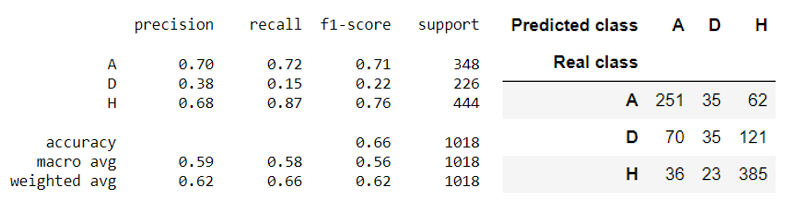


Figure 12. Score of optimized logisitc regression model and crosstab matrix.

# Conclusions and recommendations

Arsenal has been one of the best teams in the Premier league and has been at the top of the table. However, over the years their performance has been declining and they have not been able to become champions. The database of the Premier league matches from 2010 to 2021 has been adjusted to obtain the relevant information to evaluate a team's performance. New variables were created to determine the winning team of each match as well as the points obtained.

Arsenal's performance has been compared against the top teams (defined as the teams in the top 4 of the table) as well as the champions. Arsenal have been in the top four during the 10/11 to 15/16 seasons. However, from this season onwards they started to occupy lower positions resulting in their non-participation in the champions league.

Offensive and defensive aspects have been analyzed to evaluate arsenal's performance. On the one hand, arsenal has similar statistics in the offensive aspect compared to the best teams and champions. Their goals scored ratio is similar. However, their offensive aspect has been declining causing them to concede more goals than they score. This decline is due to their defensive performance. On average, champion teams concede 33 goals while scoring 87 goals. Arsenal on average concede 44 goals while scoring 70 goals.

Another important aspect is the number of points obtained by the arsenal. To be titled champion of the premier league, a team must obtain at least 80 points. Arsenal's best season was 13/14 with 78 points. Its worst season was 19/20 in which it obtained 56 points.

Class "A" represents the visiting team as the winner. The model used 348 data points, of which it predicted 351. In other words, it correctly assigned the class about 72% of the time. Class "H" represents the home team as the winner. The model used 444 data points, of which it predicted 385, i.e. it correctly assigned the class about 86% of the time. Class "D" representing tie was the class with the least accuracy. The model used 226 data, of which it predicted 35. It correctly assigned the class about 15% of the time.