Data Analytics

Premier League Football Results; Using ML and Football DataStats

Ismaël CISSE

June, 2023

Table of content

Introduction

The 38th matchday of the Premier League has just ended... It will be hard for us to forget the incredible scenarios witnessed during this season, from the battle for the title between the men of Spanish coach Mikel Arteta, coach of Arsenal and former player, against those of Josep 'Pep' Guardiola, coach of Manchester City. The same can be said about the relegation fight involving several historic clubs, who have been present in the top-five for many seasons now.

These emotions, sometimes taking our breath away during this intense and suspenseful atmosphere, are what make football so charming. Yes, football, the most popular sport in the world, let's not shy away from that fact. Look around the globe, it is the most lucrative sport in terms of media coverage, especially in England. The English Premier League generates the highest revenue worldwide. At the start of the season, there were already talks of English clubs surpassing 7 billion euros in revenue for the 2022-23 season. This sum is mainly attributed to television rights, particularly from the United States and the Middle East, allowing English clubs to attract the best talents from around the world.
The presence of numerous football superstars in many English teams such as Manchester City, Liverpool, Chelsea, Arsenal, and others makes it challenging to predict the next champion of the league each year. City's famous "dominance" is occasionally challenged by unexpected changes, like Liverpool's triumph in 2020 during the pandemic or Chelsea's victory in 2017. Predicting the outcome of a match is not always an easy task, even when betting on the favorite. There is always the possibility of an unforeseen event playing the role of a party pooper and defying predictions. This practice has always been associated with sports, and we are witnessing it more and more in the world of football, with numerous match previews, videos, and advertisements related to sports betting.

To predict the name of the team that will win a match, there are several aspects we can rely on, primarily the football-related factors. The players who make up both teams have a significant influence on the result. The team composition, the stadium, the atmosphere, and other factors also play a crucial role. Let's focus on the 11 players on the field, including female football players, as women's football is also an integral part and can sway the outcome of the match. Their performances throughout the season can give us indications of whether victory will be achieved.
However, football is a sport that unfolds in the present moment. While calculations can be made in advance by

numerous scientists, analysts, researchers, or even bettors to predict the winner, the exact score, or the future champion, the game itself is dynamic and unpredictable. The objective of this project will be to predict the final results of matches in the English Premier League that took place during the 2022-23 season.
As someone who has been following this championship for twenty years, I have witnessed wonderful moments in front of the TV, witnessing the passage of the greatest international stars and the emergence of prominent names such as Thierry Henry, Cristiano Ronaldo, Steven Gerrard, and Alan Shearer. Combining my acquired skills during this bootcamp with my passion for sports, particularly football, motivates me even more in my willingness to work as a data analyst in this field.

What makes this project unique is that it will primarily rely on the overall team performances as well as the individual performances of each player at the beginning of the 2022-23 season. The goal is to compare the predicted results with the actual outcomes by utilizing the historical encounters between two teams and the performances of players in the league. Additionally, the project will consider the impact of new players who have joined the league during the summer transfer window of 2023.

Indeed, sports, like us, bear witness to the numerous technological advancements that surround us. Victory, driven by pushing the boundaries of performance, now compels us to calculate and obtain key aspects for improvement. Given its success, it was not very difficult for me to acquire the necessary data to implement my project.

Data and data sources

Taking a journey through the history of the Premier League, named as such since 1993, I was able to obtain a record of the matches played by all the participating teams since that mentioned year. Acquired from the Kaggle website, I was able to study the matches and results of numerous English teams, with some still competing in the current championship, while others now play in lower divisions.

History Transfers PL since 1993

The second data source is the website Fbref.com, which is a highly comprehensive site in terms of detailed and collective statistics. Partly created by Statsbomb, it allowed me to retrieve personal data of players joining the English league during the 2022-23 season. It is important to note that to predict the results of this season, I had to place myself in a timeline close to August 2022. This involved considering the squad compositions that best matched that period, as well as considering the individual performances of each player at that same date.

Column Name	**Description**
Season_End_Year	Season played year
Wk	Week the of Matchday
Date	Date when match were played
Home	Name of the home team
HomeGoals	Goals scored by the home team

AwayGoals	Goals scored by the away team
Away	Name of the away team
FTR	Full-Time Result

The second data source is the website Fbref.com, which is a highly comprehensive site in terms of detailed and collective statistics. Partly created by Statsbomb, it allowed me to retrieve personal data of players joining the English league during the 2022-23 season. It is important to note that to predict the results of this season, I had to place myself in a timeline close to August 2022. This involved considering the squad compositions that best matched that period, as well as considering the individual performances of each player at that same date.

Transferktmarkt

Standard Stats

Column Name	Description
Player	Name of the player
Position	Player position
Squad	Player age
MP	Player price if is in transfer list, calculate from his overall performance in the field
Starts	Former country's league were player played
Min	Former league were player played
90s	Former players club
Gls	Location where player is now playing

G+A	League where player is now playing
G-PK	Club where player player is now playing
PK	Price paid by the club where player is now playing
PKatt	If player in on loan (bool)
CrdY	Yellow Card
CrdR	Red Card
Gls.1	Goals per 90 minutes
Ast.1	Assists per 90 minutes
G+A.1	Goals + Assists per 90 minutes
G-PK.1	Non Penalty Goals per 90 minutes
G+A-PK	Non Penalty Goals + Assists per 90 minutes

Shooting Stats

Column Name	Description
Player	Name of the player
Pos	Player position
Squad	Player age
Gls	Player price if is in transfer list, calculate from his overall performance in the field
Sh	Former country's league were player played
SoT	Former league were player played

G/Sh	Former players club
G/SoT	Location where player is now playing
Dist	League where player is now playing
PK	Club where player player is now playing
PKatt	Price paid by the club where player is now playing

Miscellaenous Stats

Column Name	Description
Player	Name of the player
Pos	Player position
Squad	Player age
CrdY	Player price if is in transfer list, calculate from his overall performance in the field
CrdR	Former country's league were player played
2CrdY	Former league were player played
Fls	Former players club
Fld	Location where player is now playing
Off	League where player is now playing
Crs	Club where player player is now playing
Int	Price paid by the club where player is now playing
TklW	If player in on loan (bool)

PKwon	
PKcon	
OG	

The Transfermarkt website was also necessary to update the squads of English clubs that have acquired or sold several of their players. The data was obtained through web scraping, collecting all the transfers that took place during the summer of 2022. This allowed me to identify players who are no longer part of the league, such as Sadio Mané and Paul Pogba. On the other hand, players like Erling Haaland and Anthony have joined this league this season.

Column Name	Description
name	Name of the player
position	Player position
age	Player age
market_value	Player price if is in transfer list, calculate from his overall performance in the field
country_from	Former country's league were player played
league_from	Former league were player played
club_from	Former players club
country_to	Location where player is now playing
league_to	League where player is now playing
club_to	Club where player player is now playing
fee	Price paid by the club where player is now playing

loan	True If player in on loan, False otherwise

After obtaining the transfer table through web scraping, data cleaning was necessary. First, regarding the maket_value column, age, and fee, in order to interpret them better during visualizations. Once the types were modified, the players' positions also needed to be changed. This was done to match perfectly with the positions in other tables that also have a dedicated column for player positioning. The club names were then modified.
After that, it was necessary to filter the data frame elements to retrieve player arrivals and departures in England, specifically in the Premier League, so a new dataframe was created for departures:

```
#Create a new dataframe that only displays all departures from PL
df_departures = df.drop('loan', axis=1)

df_departures = df.loc[(df["league_from"] == "Premier League") & (df["country_from"] == "England")]

df_departures.head()
```

	name	position	age	market_value	country_from	league_from	club_from	country_to	league_to	club_to	fee	loan
2	Wesley Fofana	Defender	21	40	England	Premier League	Leicester City	England	Premier League	Chelsea FC	80	False
7	Raheem Sterling	Forward	27	70	England	Premier League	Manchester City	England	Premier League	Chelsea FC	56	False
8	Sadio Mané	Forward	30	70	England	Premier League	Liverpool FC	Germany	Bundesliga	Bayern Munich	32	False
9	Romelu Lukaku	Forward	29	70	England	Premier League	Chelsea FC	Italy	Serie A	Inter Milan	7	True
10	Marc Cucurella	Defender	24	28	England	Premier League	Brighton & Hove Albion	England	Premier League	Chelsea FC	65	False

Another for the arrivals :

```
#Create a new dataframe that only displays all arrivals to the PL

df_arrivals = df.loc[(df["league_to"] == "Premier League") & (df["country_to"] == "England")]

df_arrivals.head()
```

	name	position	age	market_value	country_from	league_from	club_from	country_to	league_to	club_to	fee	loan
0	Erling Haaland	Forward	21	150	Germany	Bundesliga	Borussia Dortmund	England	Premier League	Manchester City	60	False
1	Antony	Forward	22	35	Netherlands	Eredivisie	Ajax Amsterdam	England	Premier League	Manchester United	95	False
2	Wesley Fofana	Defender	21	40	England	Premier League	Leicester City	England	Premier League	Chelsea FC	80	False
4	Casemiro	Midfielder	30	40	Spain	LaLiga	Real Madrid	England	Premier League	Manchester United	70	False
5	Alexander Isak	Forward	22	30	Spain	LaLiga	Real Sociedad	England	Premier League	Newcastle United	70	False

An equivalent data cleaning procedure was performed on the arrivals table. Next, the data from the transfermarkt.fr website was loaded, and the three tables underwent the same treatment. Missing values were checked, which were present because some players had insufficient or no data, and goalkeepers were present, especially in the "Shooting" table. For missing values, the average of the column elements was used, and for the shooting table, the goalkeepers were removed. Changes were made to certain data types, club names, positions, etc.

Data collection

```python
def modif_txt(text):
    regex = re.compile(r'[\n\r\t]')
    text = regex.sub('', text)
    return " ".join(text.split())


def format_currency(value):
    value = value.replace('€', '')
    value = value.replace('-', '0')
    value = value.replace('Loan fee:', '')
    value = value.replace('-', '0')
    value = value.replace('?', '0')
    value = value.replace('loan transfer', '0')
    value = value.replace('free transfer', '0')

    if value[-1] == 'm':
        value = value.replace('m', '')
        return float(value)

    if value[-1] == '.':
        value = value.replace('.', '')
        if value[-2:] == 'Th':
            value = value.replace('Th', '')
            return float(value) / 1000
    return value


def loan_transform(value):
    if bool(re.match('loan', value, re.I)):
        bool_value = True
        return bool_value
    else:
        bool_value = False
        return bool_value
```

Data cleaning and Exploratory data analysis

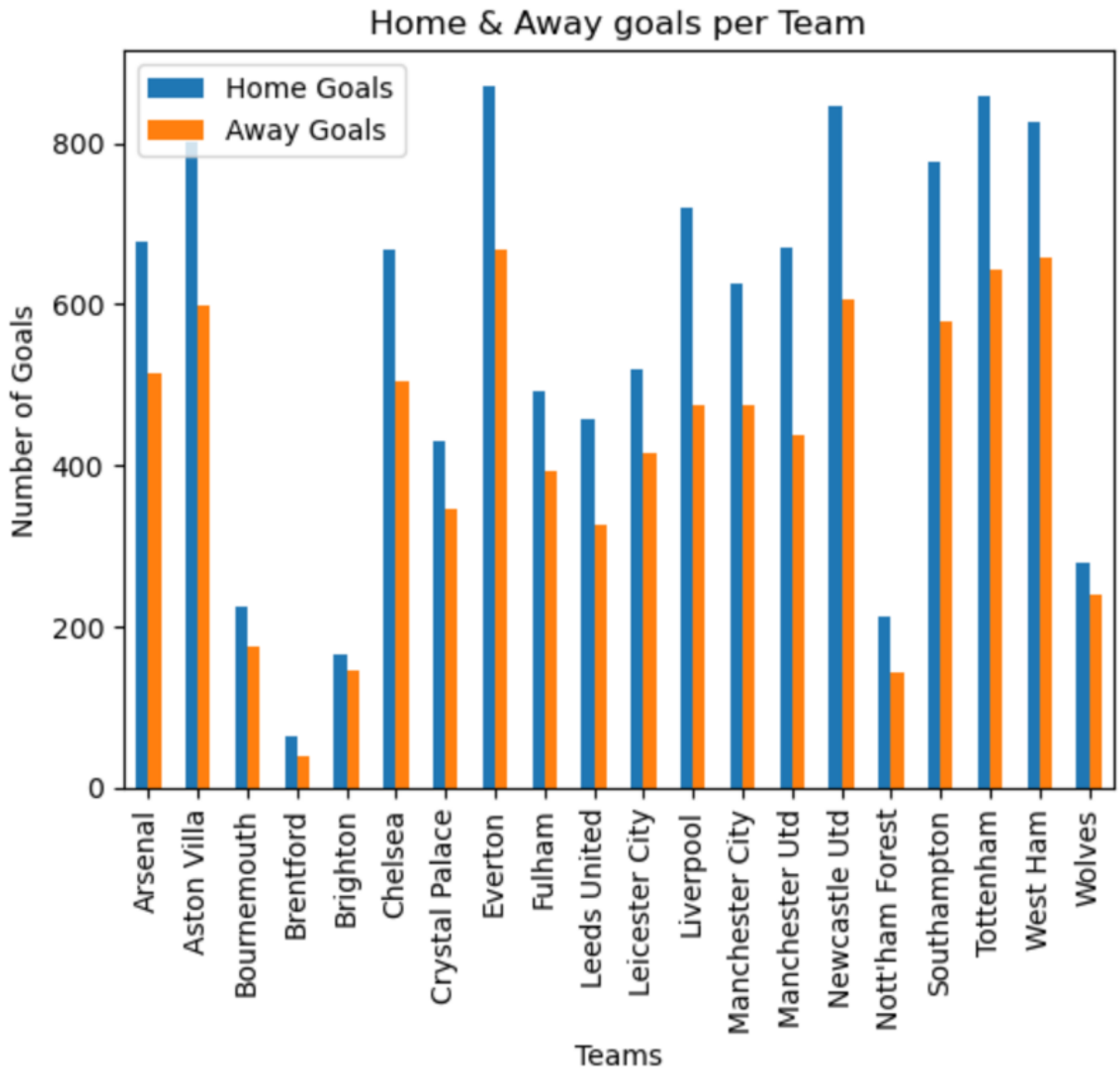
For data collection, I first considered the teams present during the first matchday of the league, which can be seen by entering the date "2023" in the " Season_End_Year " column. From there, I was able to obtain the names of the teams currently in the Premier League.

```
# Collecting Teams present during Premier League 2022-2023 Season
df_plmatches23 = pl_df.loc[(pl_df["Season_End_Year"] == 2023) & (pl_df["Wk"] == 1)]
```

```
df_plmatches23
```

	Season_End_Year	Wk	Date	Home	HomeGoals	AwayGoals	Away	FTR
11646	2023	1	2022-08-05	Crystal Palace	0	2	Arsenal FC	A
11647	2023	1	2022-08-06	Fulham FC	2	2	Liverpool FC	D
11648	2023	1	2022-08-06	Tottenham Hotspurs	4	1	Southampton FC	H
11649	2023	1	2022-08-06	Newcastle United	2	0	Nottingham Forest	H
11650	2023	1	2022-08-06	Leeds United	2	1	Wolverhampton Wanderers	H
11651	2023	1	2022-08-06	Bournemouth	2	0	Aston Villa	H
11652	2023	1	2022-08-06	Everton FC	0	1	Chelsea FC	A
11653	2023	1	2022-08-07	Leicester City	2	2	Brentford FC	D
11654	2023	1	2022-08-07	Manchester United	1	2	Brighton & Hove Albion	A
11655	2023	1	2022-08-07	West Ham United	0	2	Manchester City	A

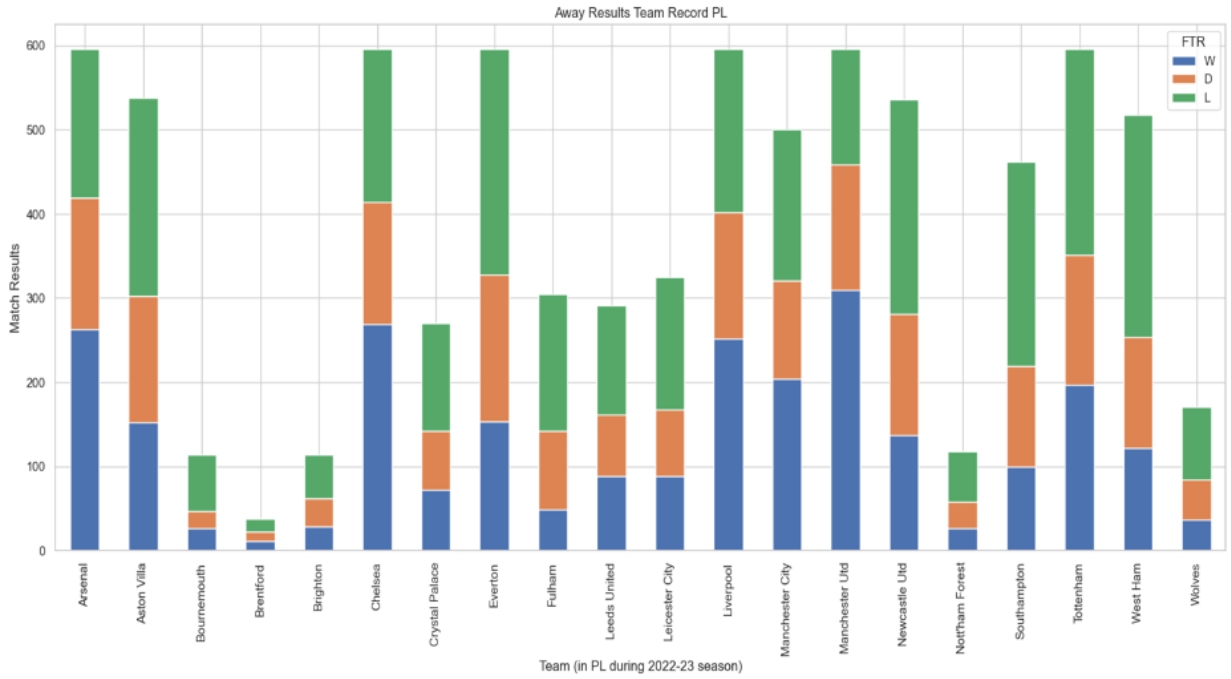
Regarding the cleaning process, this table did not have any missing data, null values, or duplicates. However, in order to join the tables later on, it was important to ensure that the club names were spelled consistently (e.g., Arsenal FC instead of Arsenal, or Newcastle United instead of Newcastle Utd).



The objective with this table is to obtain a historical record of each team in the English top division. By comparing the number of goals scored at home with those scored away, we can see that teams tend to score more goals at home than away. This information is not insignificant, as playing at home always provides an advantage for the host team. The environment plays an important role, as fatigue during away games, familiarity with the pitch, home weather conditions, and the support of fans cheering for their favorite team are all significant variables in predicting a result. Therefore, a team generally has a higher chance of scoring goals at home than away.

After investigating the number of goals scored by Premier League teams in the 22/23 season, I turned my attention to analyzing the number of wins, losses, and draws for each selected team.

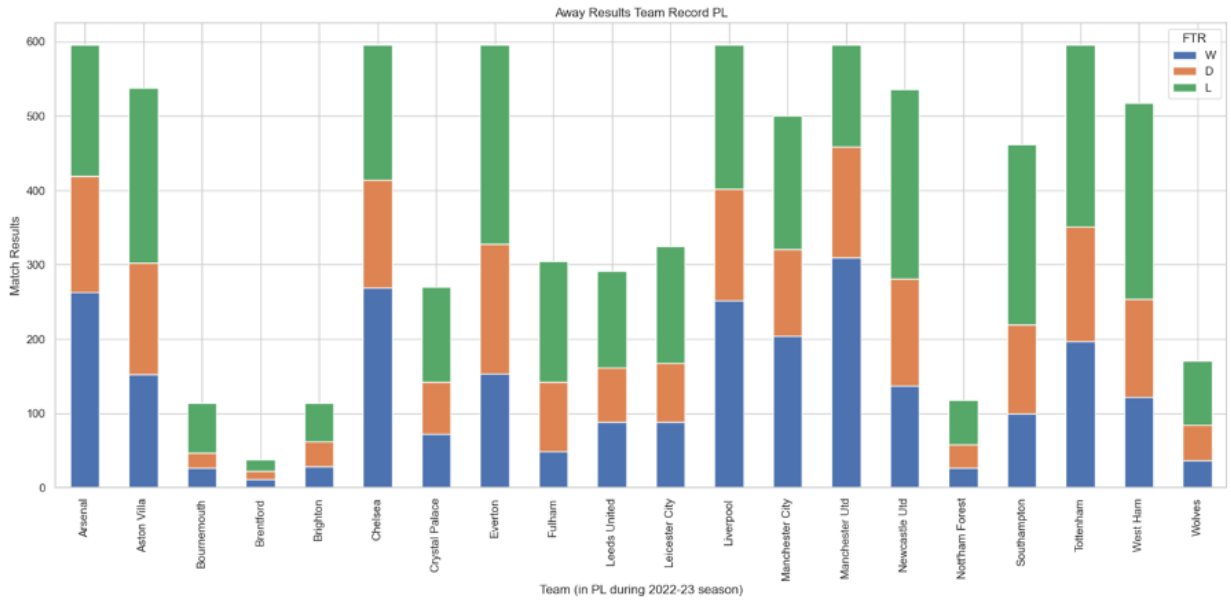
By observing the visualization, we can see that the number of away defeats is higher for a large number of teams during their away matches. The share of defeats in the overall matches of the teams below confirms this, with a few exceptions. It is important to note that we focused on the share rather than the number of wins, defeats, or draws away from home because not all teams have played the same number of matches in the Premier League.




```
away_results_pl23.plot(kind='bar', stacked=True,
        title='Home Results Team Record PL')
plt.xlabel("Team (in PL during 2022-23 season)")
plt.ylabel("Match Results")

```

On the contrary, this observation confirms the "home advantage," where the number of victories for the home team is significantly higher than the other results.



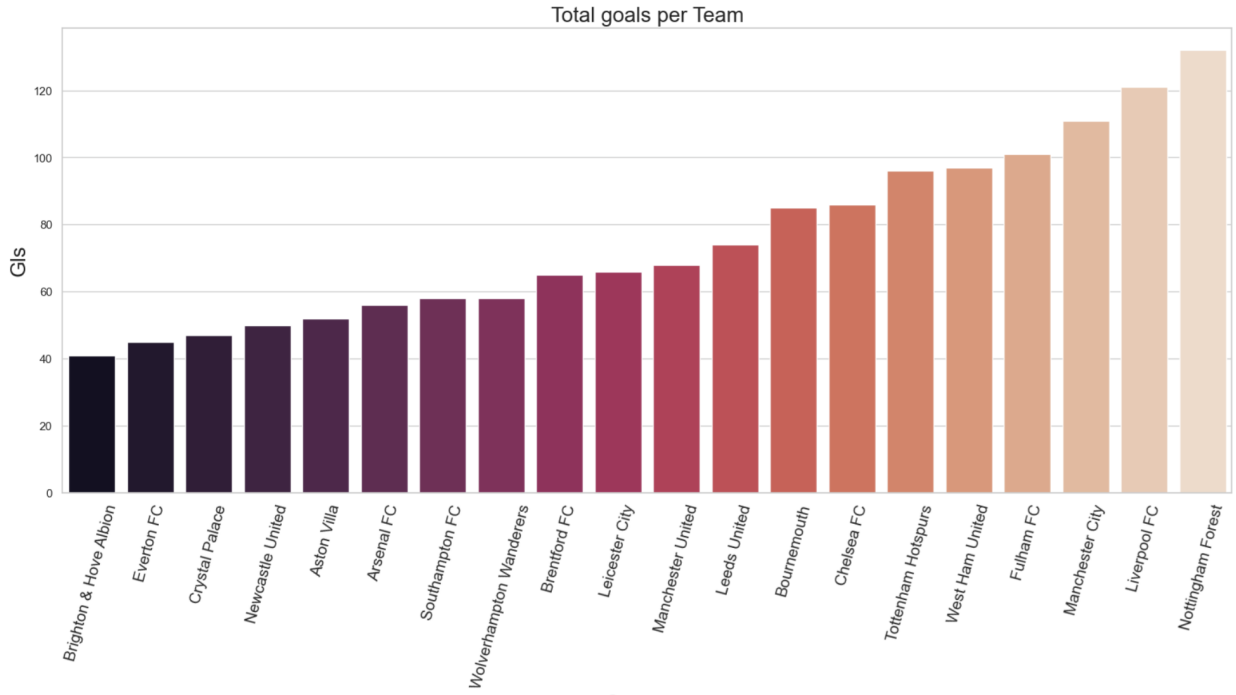
For the table concerning transfers, web scraping was used to retrieve all the transactions made during the summer transfer window. It is important to specify the transfer window period, as football has two sessions, one in the summer and another in the winter, allowing teams to manage contingencies such as injuries during the first half of the season. The timing is crucial and should not be overlooked, as the prediction starts at the beginning of the season. However, it does not begin from the first matchday because the arrival or departure date of the player was not included in our database. Therefore, I decided to start the prediction session from the 6th matchday, by which time all the transfers would have been completed, with the deadline day being August 31, 2022

```
# Deadline Week Summer Transfer Window - 08/31/22
pl_df.loc[(pl_df["Season_End_Year"] == 2023) & (pl_df["Wk"] == 5)]
```

	Season_End_Year	Wk	Date	Home	HomeGoals	AwayGoals	Away	FTR
11686	2023	5	2022-08-30	Fulham	2	1	Brighton	H
11687	2023	5	2022-08-30	Crystal Palace	1	1	Brentford	D
11688	2023	5	2022-08-30	Southampton	2	1	Chelsea	H
11689	2023	5	2022-08-30	Leeds United	1	1	Everton	D
11690	2023	5	2022-08-31	Bournemouth	0	0	Wolves	D
11691	2023	5	2022-08-31	Manchester City	6	0	Nott'ham Forest	H
11692	2023	5	2022-08-31	Arsenal	2	1	Aston Villa	H
11693	2023	5	2022-08-31	West Ham	1	1	Tottenham	D
11694	2023	5	2022-08-31	Liverpool	2	1	Newcastle Utd	H
11695	2023	5	2022-09-01	Leicester City	0	1	Manchester Utd	A

This plot highlights the number of shots on target per team and can be interpreted as a statistic related to the team's accuracy level on the field. A team that takes a high volume of shots does not necessarily increase its chances of victory compared to a team that takes fewer shots but has better accuracy. The goal is to put the ball in the back of the net at least once during a football match. The graph indicates that teams such as Manchester City, Liverpool, Leeds, and Tottenham are the most threatening in terms of goal-scoring. The absence of Bournemouth and Nottingham Forest is justified. These two teams were promoted to the top division during the 2022-23 season, and the collective and individual statistics are from the second division championship. The difference in level between the two leagues is significant, and their statistics outweigh those of teams that were in the Premier League last season. I have taken this situation into account for the machine learning stage. Introducing a multiplier to decrease the statistics of promoted clubs could be a hypothesis before using different models. Without this adjustment, the stats could be completely skewed and mislead the prediction.

The same principle applies to the number of goals scored by each club. We can still observe a high number of goals for clubs like Nottingham and Fulham. Following them are Liverpool and Manchester City. The latter two consistently feature prominently in most of the highlighted graphs

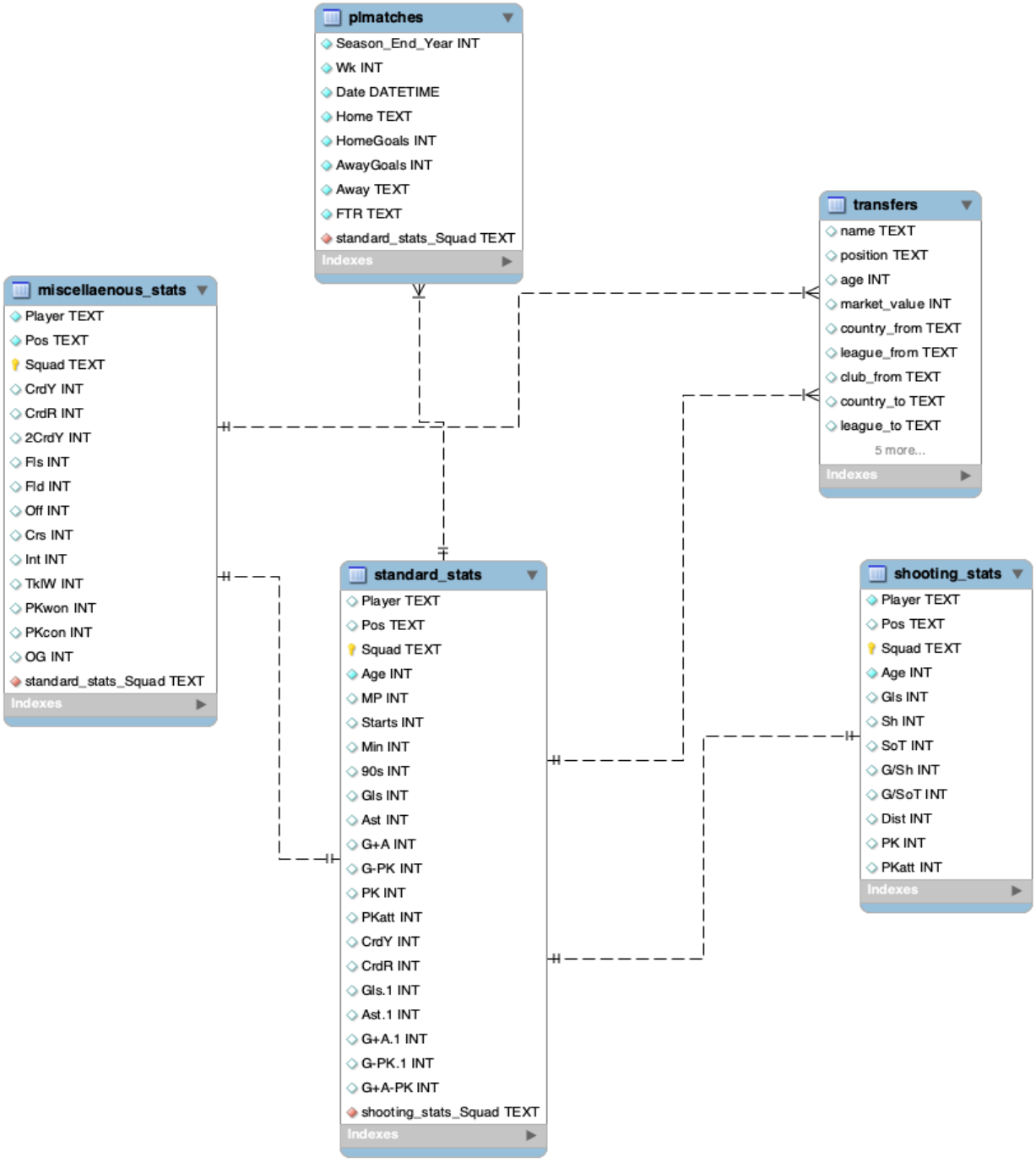


Data base type selection

SQL databases are built on the foundation of a relational model, which empowers you to establish connections between tables using foreign keys. This capability facilitates streamlined querying and retrieval of data based on these relationships, thereby enabling intricate joins and aggregations spanning multiple tables.

Thus, SQL, being a standardized language, allows for seamless portability of code across different database systems without significant modifications. This inherent compatibility facilitates smooth transitions between various DBMS vendors and simplifies the integration of multiple databases into a unified system.

Entities. ERD



Entrée [1185]:
```
1 sql_pass = getpass.getpass()
```

.....

Entrée [1187]:
```
1 #MySQL database
2 connection_string = 'mysql+pymysql://root:' + sql_pass + '@localhost:3306/footballbase'
3 engine = create_engine(connection_string)
```

Entrée [1193]:
```
1 df_plmatches.to_sql('plmatches', engine,
2                     'footballbase', if_exists='replace', index=False)
```

Out[1193]: 12026

Entrée [1194]:
```
1 df.to_sql('transfers', engine,
2                     'footballbase', if_exists='replace', index=False)
```

Out[1194]: 2000

Entrée [1195]:
```
1 pl_standard.to_sql('standard_stats', engine,
2                     'footballbase', if_exists='replace', index=False)
```

Out[1195]: 596

Entrée [1196]:
```
1 pl_shooting.to_sql('shooting_stats', engine,
2                     'footballbase', if_exists='replace', index=False)
```

Out[1196]: 553

Entrée [1197]:
```
1 pl_miscellaenous.to_sql('miscellaenous_stats', engine,
2                     'footballbase', if_exists='replace', index=False)
```

Out[1197]: 596

MySQL Queries

```
SELECT Squad, AVG(`G+A`) AS Goals_and_Assists_Average
FROM standard_stats
GROUP BY Squad
ORDER BY Goals_and_Assists_Average DESC
LIMIT 5;
```

Squad	Goals_and_Assists_Average
Liverpool FC	6.8621
Manchester City	6.7143
Fulham FC	6.6207
Tottenham Hotspurs	5.6000
Chelsea FC	5.4615

```
SELECT name, position, league_to, fee
FROM transfers
WHERE league_to = "Premier League" and fee < 200
ORDER BY fee desc
Limit 10;
```

name	position	league_to	fee
Antony	Forward	Premier League	95
Wesley Fofana	Defender	Premier League	80
Darwin Núñez	Forward	Premier League	80
Casemiro	Midfielder	Premier League	70
Alexander Isak	Forward	Premier League	70
Marc Cucurella	Defender	Premier League	65
Erling Haaland	Forward	Premier League	60
Richarlison	Forward	Premier League	58
Lisandro Martínez	Defender	Premier League	57
Raheem Sterling	Forward	Premier League	56

```sql
SELECT AVG(`G/SoT`) as moy ,Squad
FROM standard_stats ss
LEFT JOIN shooting_stats s using (Squad)
LEFT JOIN miscellaenous_stats m using (Squad)
GROUP BY ss.Squad
ORDER BY moy desc
Limit 5;
```

Squad	moy
Fulham FC	0.1154
Newcastle United	0.1071
Arsenal FC	0.1053
West Ham United	0.1000
Crystal Palace	0.0952

```sql
SELECT Squad, SUM(Gls) as GOALS, SUM(Sh) as Shoot, SUM(SoT) as Shoot_on_Target
,SUM(Dist) as Distance, SUM(PK) as Penalty_Kick, SUM(PKatt) as Penalty_Attempted
FROM shooting_stats
GROUP BY Squad
ORDER BY Squad, GOALS;
```

Squad	GOALS	Shoot	Shoot_on_Targ...	Distance	Penalty_Kick	Penalty_Attempt...
Arsenal FC	56	498	161	356	3	3
Aston Villa	52	489	168	389	2	2
Bournemouth	85	732	251	546	4	5
Brentford FC	65	559	187	397	6	6
Brighton & Hove Albion	41	406	128	373	5	7
Chelsea FC	86	569	216	419	10	14
Crystal Palace	47	367	128	356	6	8
Everton FC	45	536	161	535	4	6

```sql
SELECT name, position, league_from, fee
FROM transfers
WHERE league_from = "Premier League"
ORDER BY fee desc;
```

name	position	league_from	fee
Nabil Dounga	Midfielder	Premier League	775
Tanguy Ndombélé	Midfielder	Premier League	500
Kenedy	Forward	Premier League	500
Wesley Fofana	Defender	Premier League	80
Marc Cucurella	Defender	Premier League	65
Raphinha	Forward	Premier League	58
Richarlison	Forward	Premier League	58
Raheem Sterling	Forward	Premier League	56
Gabriel Jesus	Forward	Premier League	52
Kalvin Phillips	Midfielder	Premier League	49
Oleksandr Zinch…	Defender	Premier League	35
Sadio Mané	Forward	Premier League	32