Analyzing English Premier League 2020/21 Data

By Yasemin Gunal

Motivation

I enjoy watching soccer, particularly the English Premier League. However, it is not just a passive hobby; I find myself actively wondering and hypothesizing about the strategies employed by teams and players in different situations. For this project, one goal was to analyze different variables that could potentially impact the outcomes of matches throughout the 2020-2021 season. Specifically, I wanted to study whether or not the total average age of a club, the total number of yellow cards the club received, or the total number of successful passes has an evident impact on the number of wins that a team experiences during a given season.

Another goal of mine for this project was to analyze the performance of specific positions and how they might impact the outcome of matches. First I looked at the defensive line of each club; I wanted to measure whether or not aggression played a role in the number of goals conceded. To achieve this, I compared the total number of yellow cards each defensive line of each club received with the number of goals that club conceded for the season. As for the midfielders, I wanted to measure if there was a correlation between the number of assists or goals midfielders had for each team and the number of wins that team experienced. Lastly, I wanted to analyze the offensive line's performance in relation to their salaries. To accomplish this, I compared their salaries to the number of goals they scored and the number of matches they played during the season to determine the extent to which an offensive player's performance has an impact on their yearly salary.

Lastly, I wanted to leverage the data to calculate the probability of the home and away teams winning a match based on the half-time score of the match. If this yields significant disparities in the probabilities, then this insight could be used for predicting the outcome of future matches based on their half-time scores.

Data Sources

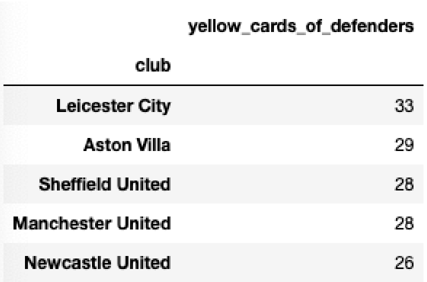
The first dataset I accessed was entitled 'English Premier League(2020-21)' from Kaggle. This dataset was a 16kB CSV file containing a variety of information on each individual player in the league during this season, including the club, nationality, position, age, number of goals scored, number of assists, number of attempted passes, penalty goals, yellow cards, red cards, etc. I downloaded this file and navigated it through Microsoft Excel.

The second resource I used was a website which displays an HTML table with the names, weekly wages, yearly wages, teams, countries of origin, positions, and ages of the highest paid EPL players from the 2020-2021 season. These values were ordered highest to lowest in terms of wage. I specifically only used the names of the players and their yearly salaries.

The third resource I used was a database entitled ‘World Soccer DB: archive of odds [01-JUN-2021]’ which was a 51MB SQLite database that I downloaded and navigated with DB Browser. This database contains the specific match information of every soccer game over the past few seasons and can be filtered by multiple leagues, countries, divisions, and teams. The dataset provides the home and away teams and the goals scored by each of those teams at half time and full time.

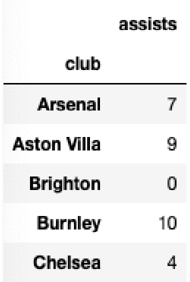
Data Manipulation Methods

For the English Premier League(2020-21) dataset, I used pandas to read in the CSV file and dropped the columns I didn’t plan on utilizing (penalty goals attempted, goals scored by players away, and goals scored by players at home). Then, I renamed the columns to account for spaces and capitalization to be easier to work with (e.g. ‘Passes Attempted’ became ‘passes_attempted’). Then, using the player position column, I filtered this DataFrame and created separate DataFrames only consisting of specific positions–for defenders (‘defenders_only’), midfielders (‘mids_only’), and strikers (‘forwards_only’). I also sorted and summed each of these by club name to include the second variable I was interested in using later for analysis/visualizations (for defenders and forwards, this was yellow cards, and for midfielders this was assists). For example, the ‘defenders_only’ DataFrame was manipulated to become the DataFrame below (Figure 1) and the ‘mids_only’ DataFrame became Figure 2:



	yellow_cards_of_defenders
club	
Leicester City	33
Aston Villa	29
Sheffield United	28
Manchester United	28
Newcastle United	26

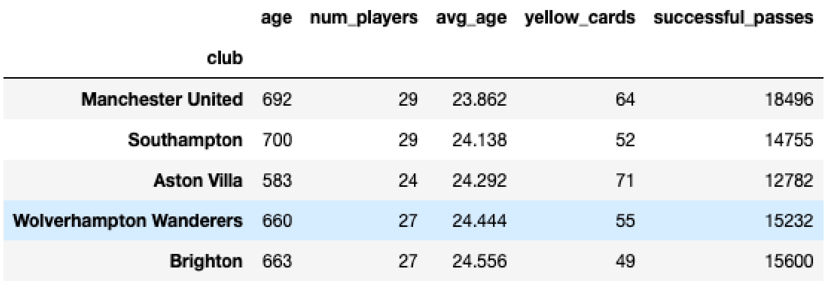
Figure 1



	assists
club	
Arsenal	7
Aston Villa	9
Brighton	0
Burnley	10
Chelsea	4

Figure 2

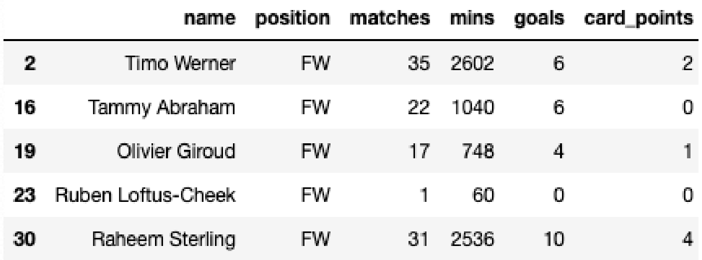
Additionally, I used existing columns in the original ‘epl_csv’ DataFrame to compute simple calculations used later for analysis. Specifically, I used a for loop to multiply the ‘passes_attempted’ and ‘perc_passes_complete’ columns to compute the number of successful passes per player and created a new column based on these values called ‘successful_passes’. I then grouped, summed, and sorted this column per each club to get a DataFrame of the number of successful passes completed by each club during the season. Next, I used the ‘.value_counts()’ function to count how many players there are per team and the ‘.sum()’ function to get the sum of the player’s ages per team. Then, I used a for loop to divide the sum by the number of players to retrieve the average per team and put this information in a new column called ‘avg_age’. This was all stored in the DataFrame called ‘merged_df’. I used similar calculations to get the total number of yellow cards per team into a DataFrame titled ‘cards_per_team’. I merged each of these separate DataFrames so all of the information would be easily accessible and well organized (‘team_summed_info’ DataFrame in Figure 3).



club	age	num_players	avg_age	yellow_cards	successful_passes
Manchester United	692	29	23.862	64	18496
Southampton	700	29	24.138	52	14755
Aston Villa	583	24	24.292	71	12782
Wolverhampton Wanderers	660	27	24.444	55	15232
Brighton	663	27	24.556	49	15600

Figure 3

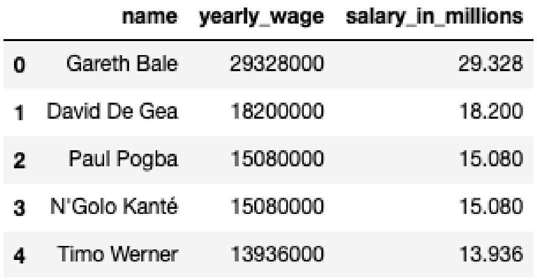
Lastly, I created a new DataFrame (called ‘player_influence’) based off of the original DataFrame and used a for loop to add a ‘card_points’ column, which adds the number of yellow and red cards together for each player. From this DataFrame, I was able to create one only for the forwards (called ‘forwards_influence’ - Figure 4) by filtering the player position to be forwarders. This was used for analysis later in the notebook.



	name	position	matches	mins	goals	card_points
2	Timo Werner	FW	35	2602	6	2
16	Tammy Abraham	FW	22	1040	6	0
19	Olivier Giroud	FW	17	748	4	1
23	Ruben Loftus-Cheek	FW	1	60	0	0
30	Raheem Sterling	FW	31	2536	10	4

Figure 4

The first obstacle I encountered when searching for data to scrape from a website was that certain websites do not allow scraping. However, this was easily overcome due to salary rankings of professional athletes being easily accessible information online due to high levels of interest surrounding it. This made it plausible to try different websites until one allowed web scraping. Once I was able to successfully scrape a website using the pandas ‘.read_html()’ function, I converted the information into a DataFrame called ‘salaries2’. Next, I needed to convert the objects in the cells to the correct types. All cell values were Object data types upon scraping, and the salary column had values with commas to indicate large values and included the English Pound symbol in front (e.g. ‘£29,328,000’). First, I changed the salary column values to strings and then using indexing, I was able to get rid of the Pound symbol. Then, I replaced all commas in the string with empty strings using ‘.replace()’. This yielded a string value that I could change into an integer. Next, I rounded and divided the integer values by 1,000,000 in order to reformat them into their decimal format in millions (e.g. ‘29328000’ became 29.33 million). Lastly, I converted the type of the ‘Name’ column values into strings as well. One issue I encountered was that a few rows had no values. However, because the data was scraped from a website, the objects were not of ‘NaN’ types. To solve this, I used the ‘.drop()’ method to drop the specific rows that had missing values. The last manipulation tactic I used on the ‘salaries2’ Dataframe was renaming the columns to be easier to access from ‘Name’ and ‘Yearly Wage’ to ‘name’ and ‘yearly_wage’, respectively. The final Dataframe is represented by Figure 5.

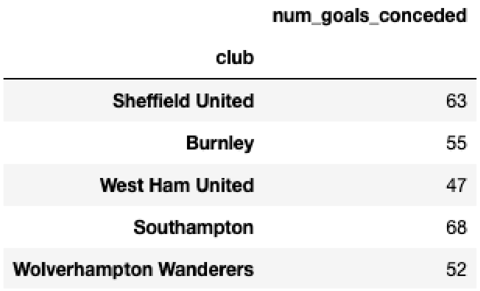


	name	yearly_wage	salary_in_millions
0	Gareth Bale	29328000	29.328
1	David De Gea	18200000	18.200
2	Paul Pogba	15080000	15.080
3	N'Golo Kanté	15080000	15.080
4	Timo Werner	13936000	13.936

Figure 5

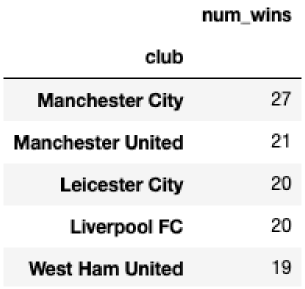
Next, in order to read in and clean the Database data from SQLite, I used a combination of Pandas and SQL (specifically, I used the connect() function to read the database, and then pandas.read_sql() to leverage SQL commands to create a DataFrame of the data. This allowed me to filter out the columns I did not plan on using and also only retrieve the season, league, and country I was interested in (2020/2021, Premier League, England, respectively). Once I had inserted that data into a DataFrame (called 'epl_db_df'), I ran into an obstacle with the names of the clubs. In the data resources I had already obtained and cleaned, the team names were the full-length, official club names, whereas in this database, the club names had been shortened (e.g. "Manchester City" was shortened to "Man City" and "Wolverhampton Wanderers" was "Wolves"). This was a consistent issue with a handful of the twenty teams in the league, meaning I would not be able to join this DataFrame with the other DataFrames I had previously created. To solve this issue, I used the '.rename()' function on the index of the DataFrame I created from this database and renamed all of the club names that were inconsistent with the other data sources.

Once this was solved, I used a for loop to calculate the number of goals conceded by each team. This entailed adding the away goals to the home team's goals conceded, and the home goals to the away team's goals conceded (using columns: 'HomeTeam', 'AwayTeam', 'FTAG', and 'FTHG'). First, I put all of this information into a Python dictionary, and then inserted the keys and values of that dictionary into separate columns of the same DataFrame (called 'goals_conceded'), which is represented by Figure 6 below. Using a similar for loop/calculations, I created a dictionary and then a DataFrame of the number of wins each team had based on how many times the 'FTR' (full-time match winner) was the Home or Away teams. This provided me with the 'wins_per_team' DataFrame in Figure 7.



	num_goals_conceded
club	
Sheffield United	63
Burnley	55
West Ham United	47
Southampton	68
Wolverhampton Wanderers	52

Figure 6

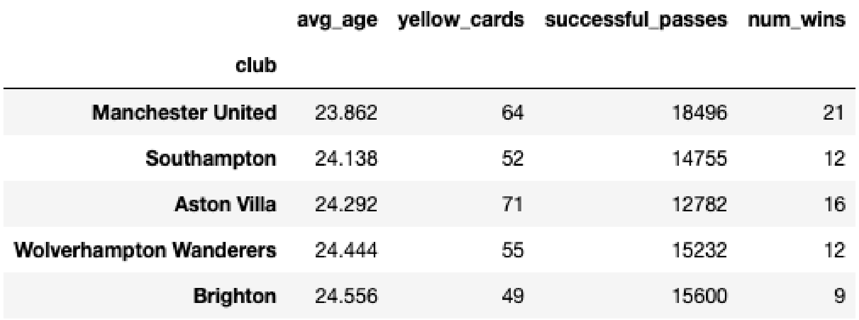


	num_wins
club	
Manchester City	27
Manchester United	21
Leicester City	20
Liverpool FC	20
West Ham United	19

Figure 7

Combining Data:

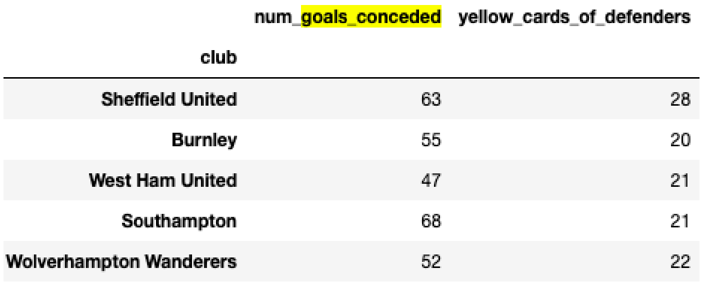
Based on the information from team_summed_info (from the EPL CSV file) and the information on the number of wins of each club from the wins_per_team DataFrame (from the Database), I created one single DataFrame called 'merged_db_csv' using the '.merge()' function (Figure 8).



club	avg_age	yellow_cards	successful_passes	num_wins
Manchester United	23.862	64	18496	21
Southampton	24.138	52	14755	12
Aston Villa	24.292	71	12782	16
Wolverhampton Wanderers	24.444	55	15232	12
Brighton	24.556	49	15600	9

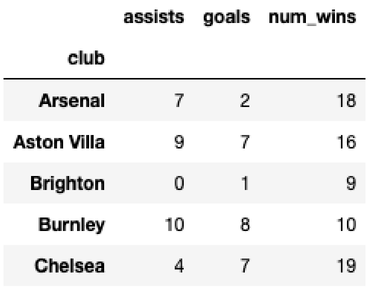
Figure 8

Next, I created a merged DataFrames using the existing DataFrames created from the CSV file containing separate information depending on the positions of the players (e.g. yellow cards of defenders, assists of midfielders, and yellow cards of forwards) and the goals_conceded and num_wins DataFrames to get the following DataFrames:



club	num_goals_conceded	yellow_cards_of_defenders
Sheffield United	63	28
Burnley	55	20
West Ham United	47	21
Southampton	68	21
Wolverhampton Wanderers	52	22

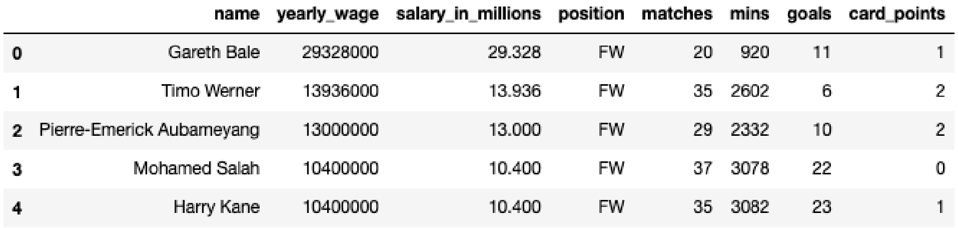
Figure 9



club	assists	goals	num_wins
Arsenal	7	2	18
Aston Villa	9	7	16
Brighton	0	1	9
Burnley	10	8	10
Chelsea	4	7	19

Figure 10

My next goal was to determine the influences on an offensive player's salary. To do this, I needed to combine the data scraped from the website with the information I had on the performance of forwards. Using '.merge()', I combined the 'forwards_influence' DataFrame and the 'salaries2' DataFrame to obtain the table in Figure 11:



	name	yearly_wage	salary_in_millions	position	matches	mins	goals	card_points
0	Gareth Bale	29328000	29.328	FW	20	920	11	1
1	Timo Werner	13936000	13.936	FW	35	2602	6	2
2	Pierre-Emerick Aubameyang	13000000	13.000	FW	29	2332	10	2
3	Mohamed Salah	10400000	10.400	FW	37	3078	22	0
4	Harry Kane	10400000	10.400	FW	35	3082	23	1

Figure 11

This table provided me with a direct comparison of offensive player performance and their yearly wages, which made the analysis and visualization process simpler to navigate. I particularly focused on the goals and matches columns.

Analysis and Visualization

Based on the data exploration and combinations described above, I was able to create multiple visualizations using MatPlotLib and Seaborn. These visualizations allowed me to compare variables that can potentially be used to provide insight into what attributes of a team/player/position impact their likelihood of success in a match, what influences a striker's salary, and the likelihood of a team winning a match based on the half-time score.

Firstly, using the combination of the Database data and the CSV data (combination and exploration explained in exploration section) and the I studied the impact of different variables (e.g. average team age, team yellow cards, and team passes) on the number of wins each team experienced in the 2020-2021 EPL season. These are displayed below:

1. Number of Club Wins VS. Average Club Age:

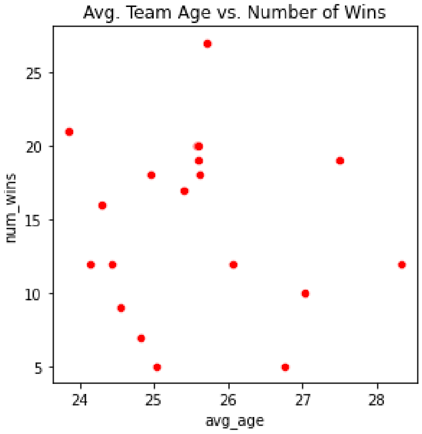


Figure 12

Based on Figure 12, there is not a strong enough relationship displayed in the visualization between age and wins in order to conclude that the age of players impact match outcomes. This may be due to the small range between the highest and lowest average ages (ranging between approximately 23 years old to approximately 30 years old). The small range reinforces the idea that professional soccer players typically reach their peak in their careers between the ages of 20 and 30 years old, and most retire around the older end of that range. So, most athletes participating in the EPL are in this age range and are in their prime, meaning they all perform relatively well and have minimal differences in their ages. In order to extend this analysis, I would be interested in looking at the specific players at each end of the age spectrum rather than the average age of the team.

2. Number of Club Wins VS. Number of Club Yellow Cards:

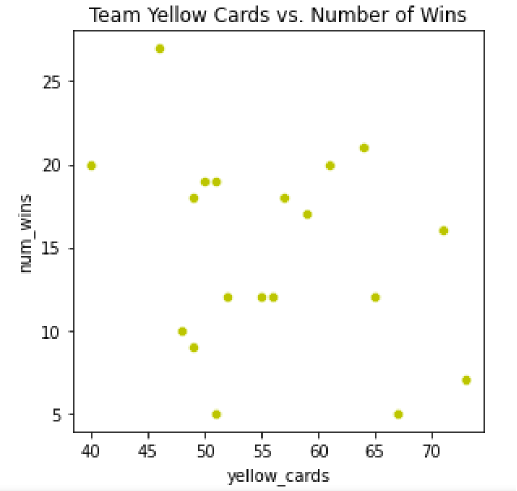


Figure 13

Figure 13 displays a very loose negative correlation. This indicates that there may be grounds to conclude that when teams are more aggressive–and risk collecting yellow cards by being aggressive–they decrease their likelihood of winning matches. It is commonly seen during matches that players obtaining yellow cards anger athletes and may impact the morale of the team to the extent of poor performance. However, in order to confirm these assumptions, more data would be necessary to ensure that there is a more significant negative correlation between these two variables. In order to do this, I am interested in including the data from all past seasons to see if the relationship changes. Additionally, it would be interesting to see if performance (based on number of goals scored, athlete energy/speed, etc.) decreases after receiving the yellow card. However, this would require information on the time that the yellow card was given..

3. Number of Club Wins VS. Number of Club Successful Passes:

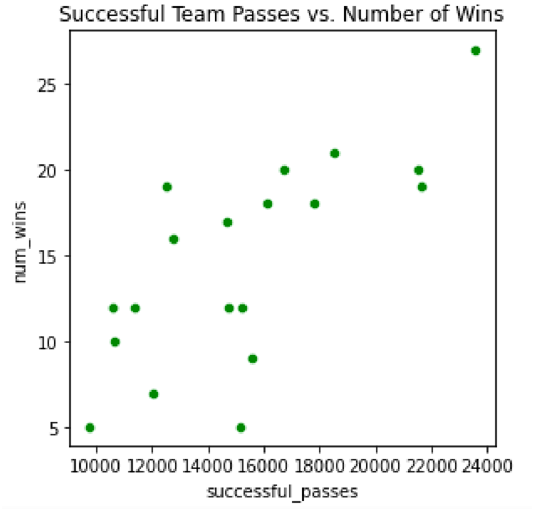


Figure 14

Figure 14 shows the strongest influence on the number of team wins out of the three variables inspected in this section. There is a very clear positive correlation between the number of successful team passes completed during a game and a team's likelihood of winning that match. This is likely due to the fact that increased passes correspond with increased teamwork in a match, which is bound to yield positive results for the team. This is grounds to conclude that teams should increase their possession of the ball during a match by passing to their teammates.

This section of the analysis focused on the specific positions of the teams and how the performance of the players in those positions impact the outcomes of matches. Specifically, I wanted to test the hypothesis that a more aggressive defense would decrease the number of goals conceded by a team. To do this, I created a scatter plot of the number of yellow cards that the defenders of each team received during the season (based on the CSV data) and compared it to the number of goals that team conceded throughout the season (based on the Database data). This is seen in Figure 15:

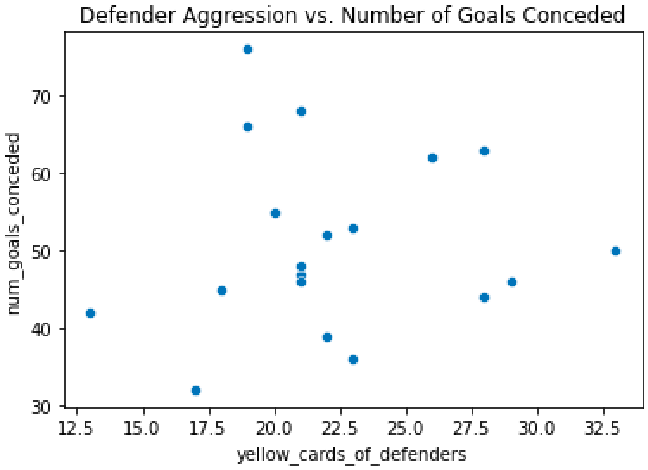


Figure 15

Evidently, there is not as strong of a relationship between the two variables as I initially assumed. Generally, it seems that both sides of the spectrum (the team with the lowest number (13) of defender yellow cards, and the team with the highest number (33) of yellow cards conceded a very similar amount of total goals (between 40-50 goals). However, many teams that had yellow cards between those two extremes conceded significantly more goals–the highest being 80 goals conceded for a team with 20 defender yellow cards. Overall, this relationship is not strong enough to be conclusive. This is interesting, because oftentimes commentators will mention that defensive players make sacrifices by defending aggressively and risking receiving cards in order to protect the goal, but this data indicates that those risks may not be worth taking in certain situations. It would be interesting to analyze this relationship further by incorporating red cards and the aggressiveness of the goalie as well, since the goalie plays an important role in the number of goals conceded.

Next, using similar exploration methods, I created a line plot to explore how midfielders can influence the number of wins of a team. In blue, I plotted the relationship between the number of team wins and the number of midfielder assists, and in orange I plotted the relationship between the number of team wins and the number of midfielder goals (Figure 16).

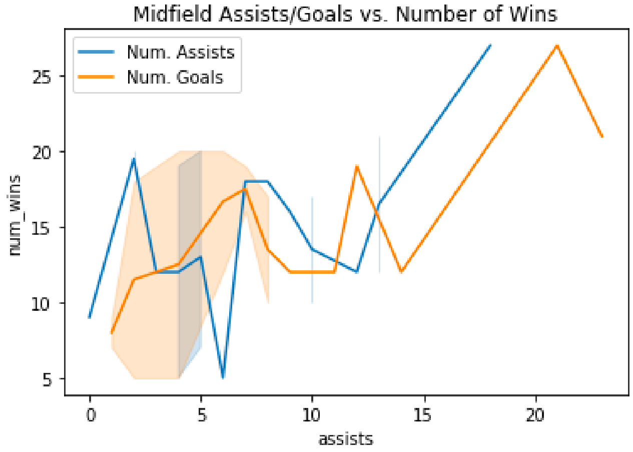


Figure 16

Based on the visualization, there is an evident positive correlation between both the number of wins versus midfielder goals and between the number of wins versus midfielder assists. However, upon closer examination, the influence of the number of midfielder goals evidently has less extreme dips and variation in the data. However, both variables share a mostly-positive correlation with the number of wins, which makes sense since assists typically lead to goals, and more goals lead to a higher probability of winning a match. This could indicate that midfielders should balance their focus between simply getting the ball to the striker, and taking shots to score themselves. This analysis could be furthered by looking at specific teams–it is possible that certain offensive players change between playing as a striker and playing in the midfield (e.g. Mason Mount from Chelsea F.C. has played in both positions, which may increase the chances that he can score a goal from the midfielder position, due to having scoring experience as a striker).

My next analysis was based on the combined data between the website and the CSV. I wanted to learn whether or not the number of goals they score or the number of matches they play has a significant impact on their yearly salary. To study this, I created the visualization in Figure 17.

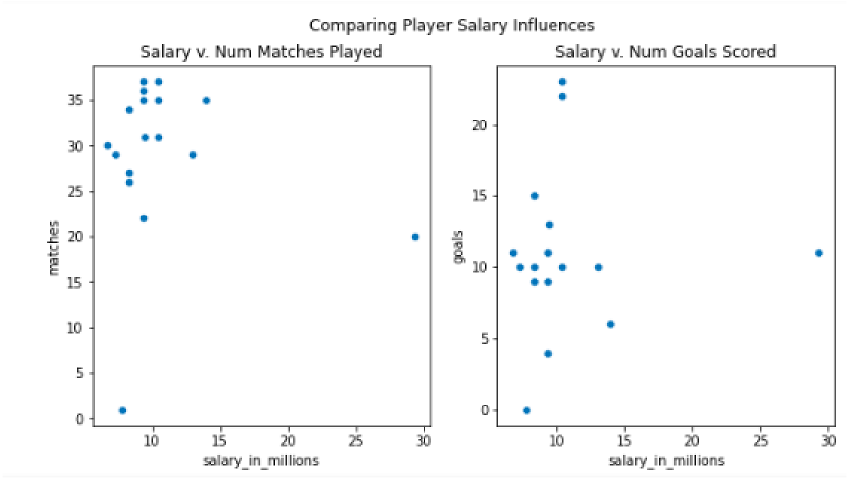


Figure 17

Based on these graphs, there is not a strong enough relationship to draw significant conclusions about the salaries of the top forwards in the EPL last season. Specifically, looking at the graph comparing the number of matches played with salaries, besides two outliers, most top players seemed to play around 25 - 40 matches total and made somewhere between 5 million to 15 million pounds. However, this graph indicates that there are very likely other variables that are influencing their salaries, since there are players who played up to 10 matches more and made 5-15 million pounds less than other players.

Looking specifically at the salaries versus the number of goals scored by forwards, this graph also indicates that there are likely other influences on player salaries. There is no significant relationship between the variables, indicating that the influence that these strikers have in specific games on increasing the score of their team is not a substantial factor in determining their salaries.

It is possible that the reason there is a weak relationship between each of these variables is due to the players in the website data being the top-earning players in the league. This could indicate that these players have been participating in the league for several years and have an excellent history and track record as forwards, meaning that their previous seasons/matches/statistics (that are not accounted for in this analysis) might be influencing their current wages.

My last analysis was based only on the Database data and values computed during the exploration phase of this project. Specifically, I wanted to utilize the half-time scores of all matches from the 20/21 season to determine the probability of the home or away team winning. I did this by calculating all of the following situations:

1. My first finding was based on the half-time score ending in a tie. If a match is a tie at half-time, it is most likely that the away team would win (38.56% chance), second most likely that the match

would remain a tie (34.64% probability), and least likely that the home team would win (26.80% probability).

2. If the half-time score was an away team lead, it was most likely that the away team would win (79.44% probability), second most likely that the end result would be a tie (12.15% probability), and least likely that the home team would win (8.41% probability).
3. Lastly, if the half-time score was a home team win, it was most likely that the home team would win (78.33% probability), second most likely that the result would be a tie (14.17% probability), and least likely that the away team would win (7.5% probability).

One surprising element of these findings, is that when the half-time score is a tie, it is most likely that the away team would win the match. This is not what I had initially expected, since it contradicts the advantage that teams experience when playing on their home field. However, both situations where the half-time score is led by the home or away team have more predictable outcomes for the full-time score.

Reflection

It is evident that without substantial amounts of data and multiple variables considered at a time, it is nearly impossible to draw conclusive conclusions about professional sports teams and players. There are so many different variables all influencing performance and matches at a given time, that all need to be simultaneously accounted for. This requires substantially larger datasets than I had access to for this project. However, based on my preliminary analysis and research/exploration conducted through this project, I have learned significantly more about what I can and cannot assume during soccer matches. I also have a much better understanding of sports analytics and the importance of data science within athletics.

Direct Links to Data Sources:

https://www.kaggle.com/datasets/rajatrc1705/english-premier-league202021

https://salarysport.com/football/premier-league/highest-paid/

https://www.kaggle.com/datasets/sashchernuh/european-football