

# Forecasting the Best Teams for the Remainder of the 2020-2021 Premier League Season

Neelansh Visen

22 December 2020

## Abstract

The advent of analytics and advanced metrics in association football (soccer) has led to the adoption of more sophisticated models used to determine the outcomes of games across a season. Recently, expected statistics (particularly expected goals) have gained popularity in their ability to provide a balanced evaluation of team performance in a match. In this paper, I create a predictive model of team performance based on a club's expected goals for and expected goals against against from the first third of the 2020-2021 Premier League Season. Based on these findings, a Monte Carlo simulation is run on the remaining fixtures of the 2020-2021 season, with Liverpool slightly edging out the competition to claim their second title in as many years.

**Keywords:** Association Football (Soccer), Premier League, Poisson Distribution, Expected Goals, Monte Carlo Simulation

Github repository can be found at: <https://github.com/neelvisen/STA304-PS5-xG-Model>

## Introduction

Whether it be through gambling, fantasy sports, or simple banter with colleagues, predicting the outcome of a match or season has long been a subject of interest for even the most passive of football (soccer) fans. In recent times, what was once an topic of fascination has evolved into a hotly discussed area of study. With many teams in the Premier League sporting a betting company sponsorship, the discussions between match predictions and the game itself are inexplicably connected.

In the following study, I create a predictive model for simulating the 2020-2021 Premier League season from gameweek 13 onward. As of the writing of this report, Tottenham Hotspur have enjoyed a hot start to the season, and currently sit tied for first in the league with current champions Liverpool. Although past results are not the strongest predictors of future results, one can apply expected results (that is, statistics based on what should have happened in the game, not necessarily what actually happened) for a more nuanced judgment of a team's actual performance and form over an extended period of matches.

The simulation model for this analysis relies on two key assumptions. First, the accuracy of the model relies on the quantified abilities assigned to each team in the Premier League. In this paper, teams are assigned an offensive and defensive rating based on the relative strength of each team with regards to the expected number of goals scored and conceded during the first 12 gameweeks of the season. The second assumption this model relies on is the recent notion that goals scored in a football match closely resemble the Poisson distribution (Dyde and Clarke [1999]; Chu [2003]). After constructing each teams relative offensive and defensive strengths, an additional parameter is included in the model based on where the game is being played. Although the impact of match location has been declining for over a century in English football, home field advantage still exists to this day (Smith [2017]).

These three parameters (relative offensive strength, relative defensive strength, home field advantage) are then applied to every remaining fixture of the season. Each competing team in a match is assigned a

predicted number of goals scored based on their competition. This value is used as the lambda (mean) of the Poisson process that is used to determine the probability of any given fixture result. From there, a Monte Carlo simulation is conducted to simulate 100 iterations of each remaining fixture this season. Of these 100 simulations, each team’s average number of points is calculated and added to the current Premier League standings to determine the final rankings of this season. I find that Liverpool manage to hang on to their first place ranking, with Chelsea, Manchester City, and Tottenham not far behind. These results heavily penalize teams like Southampton (falling from 4th to 10th) that according to the model have been punching above their weight thus far, and favour perennial favourites Manchester City (rising from 9th to 3rd) that have struggled to secure positive results despite heavily outplaying their competition. At the end of this paper, I address certain shortcomings of the model, including sample size, recency bias, and its failure to address the dynamism of factors including managerial and personnel changes. Furthermore, I discuss some of the implications and applications of this model, similar models, and expected statistics at-large.

This analysis is grateful for the works of Mark Taylor, Robert Hickman, and David Sheehan, whose analyses and simulations for other teams, leagues, and time periods helped formulate some of the code for this paper. Additionally, this paper could not have been made without the data collected by Understat (Understat 2017) and the work of Vaastav Anand, whose repository was used to collect the Understat data in CSV. form. Finally, this paper was created using R (R Core Team 2019)], as well as the packages ‘tidyverse’ (Wikham et al. [2019]), ‘ggrepel’ (Slowikowski [2020]), and ‘kableExtra’ (Zhu [2019]).

Table 1: Premier League Standings As of Gameweek 12

team	points	goal_difference
Tottenham Hotspur	25	14
Liverpool	25	9
Leicester	24	9
Southampton	23	7
Chelsea	22	13
West Ham	20	5
Everton	20	3
Manchester United	20	2
Manchester City	19	6
Aston Villa	18	8
Crystal Palace	17	1
Newcastle United	17	-2
Wolverhampton Wanderers	17	-5
Leeds	14	-5
Arsenal	13	-5
Brighton	10	-6
Burnley	9	-12
Fulham	8	-10
West Bromwich Albion	6	-16
Sheffield United	1	-16

# Data

The data used in this project is sourced from Understat, a free online resource that analyzes, collects, and publishes tabulated football match data for the English Premier League, as well as the Spanish, German, Italian, French, and Russian leagues. According to the message published on their homepage, “In a low-scoring game such as football, final match score does not provide a clear picture of performance... this is why more and more sports analytics turn to the advanced models like [expected goals], which is a statistical measure of the quality of chances created and conceded” (Understat [2017]). Essentially, the expected goals metric takes on several parameters associated with a shot in football, including the type of pass from which the shot came from (aerial cross, through ball, etc.), shot angle and distance from goal, and type of chance (established possession, rebound, volley, position of the defense, etc.).

In a football match, each shot taken by a team is compared to thousands of other shots with similar shot parameters as those mentioned above. Essentially, what expected goals shows is the probability (between 0 and 1) that a given shot would result in a goal, based on the goal history of similar shots across time. For instance, a screamer hit from 35 yards away from goal with the weak foot might generate an expected goal value close to 0, whereas a tap-in in front of an open goal might result in a expected goal value close to 1. If during a certain offensive action a team generates multiple shots in a brief period with an xG above 1, then the calculation simply becomes 1 - the probability of the offensive team NOT scoring during that passage of play. Because the process of comparing a particular shot to several thousand others is incredibly exhaustive, in order to generate their expected goals tallies, Understat, “[trains] neural network prediction algorithms with a large dataset (>100,000 shots, over 10 parameters for each)” (Understat).

There are several advantages to using expected goals data versus actual goals data. Perhaps most evidently, is that expected goals can help reduce the impact of luck in any given match result. In such a low scoring game like professional football, game results can often be decided by a fluke deflection towards or away from the net. Beyond fan analysis, teams themselves have adopted expected goals as a strong indicator of real match or season performance, leading to more informed decisions with regards to managerial and player personnel changes. Furthermore, teams (and amateur statisticians) can assess certain parameters of expected goals in order to determine things like favourable passing patterns and specific team strengths and weaknesses (goals from set pieces, certain sides of the pitch, etc.).

At the same time however, expected goals can sometimes be misleading for the best players and teams. Although expected goals addresses several factors leading into a shot, the player taking the shot is not addressed. For example, Tottenham forwards Harry Kane and Heung Min Son are widely regarded as two of the best finishers in world football. As statistically unrigorous as this statement is, the sentiment is backed up in their career expected and actual goal output. Season after season, the two players have constantly outperformed their expected goals tallies by a healthy margin. Although outperforming expected goals in some cases is a sign of a luck, its impossible to maintain that luck for an extended period of time.

With regards to this paper, I take expected goals data from Understat for the first 12 gameweeks of the 2020-2021 Premier League season. In this paper, I focus on expected goals for (xG) and expected goals against (xGA) to create the offensive and defensive relative strength metrics of each team. Because fixtures historically trend in favour of the home team, I create a table summarizing average points per game, average expected points per game (based on xG and xGA), and average expected goals per game for home and away teams.

Table 2: Home and Away Summary Statistics

home_or_away	avg_pts	xG_per_game	avg_xpts
a	1.479	1.338	1.343
h	1.325	1.441	1.425

Surprisingly, away teams this season have fared better than home teams in terms of actual points won this season. However, when we look at xG and xPts (expected points per match based on xG and xGA), we

find that home teams in fact have been outplaying away teams thus far. Whether the discrepancy is due to low sample size, poor finishing stretches, or some other factor, we will later use quotient of home and away xG to create a home field advantage factor for our model. We continue examining the Understat xG data below, proceeding to rank each team in the Premier League based on xG, xGA, and xGD (the difference in expected goals scored and conceded per game). (Figure 1, Figure 2, Figure 3 in the appendix).

Not unexpectedly, we find the same four teams at the bottom of both the current league table and the expected goal difference per game chart (Figure 3). Despite finishing 7th in the league last year, Wolverhampton has struggled to put the ball in the net without their two main goal threats Raul Jimenez (injured) and Diogo Jota (transferred to Liverpool). On the top half of the chart, we see the usual contenders Liverpool and Manchester City thoroughly out-creating and defending their oppositions. Interestingly, quality in play has not transferred over favourably for the latter, who currently sit 9th in the league table. Similarly, Chelsea maintain a healthy lead over the competition in terms of xGD despite sitting 5th in the table. On the other hand, despite their 5th place standing on the xGD chart, Tottenham currently lead the league in points (in large part due to the world class contributions of the aforementioned Harry Kane and Heung Min Son). We continue the exploratory analysis by creating a scatterplot of xG against xGA and actual goals, as well as xGA against actual goals against. (Please see graphs in the appendix).

Evidently, there is a somewhat strong linear relationship between expected and actual goal statistics (Figure 4 and Figure 5). Perhaps more surprisingly, there hardly any linear relationship (if at all) between xG and xGA (Figure 6). Usually, expected goals and assists are used to project under and over performing teams over a stretch of games. To visualize this idea, we create another scatterplot below mapping the differences between expected and actual goals for and against (Figure 7 in the appendix).

The horizontal and vertical dotted lines indicate whether teams are over/under performing a given metric. Teams above the horizontal line are currently overperforming with regards to goal scoring. Likewise, teams below the horizontal are teams currently underperforming with regards to goal output. Teams to the right of the vertical line are teams currently overperforming with regards to defending. These teams are conceding at a lower rate than what the shots they allow would suggest. Likewise, teams to the left of the vertical line have been conceding at a higher rate than what the shots they allow would suggest. If we assume that the xG/xGA models hold true in the long term, (that is, teams in the long term score and concede at a similar rate to what xG/xGA expects), then we should expect teams like Brighton to improve defensively, Sheffield United to improve offensively, Southampton to regress offensively, and Tottenham to regress both offensively and defensively.

## Model

Ultimately, our goal is to create an offensive and defensive rating per team, and then to apply those ratings to each remaining fixture of the 2020-2021 season. Since we are looking at relative strengths, we simply take each team's xG and xGA and divide (standardize) them by the league-wide average for each metric respectively. As equations, these metrics can be seen as:

$$\begin{aligned} \text{OffensiveRating}_{team} &= \frac{xG_{pergame}_{team}}{xG_{pergame}_{league}} \\ \text{DefensiveRating}_{team} &= \frac{xGA_{pergame}_{team}}{xGA_{pergame}_{league}} \end{aligned}$$

Additionally, we create a third metric accounting for whether a team is playing at home.

$$\text{HomeAdvantage} = \frac{\text{avg.leaguwide.xG}_{pergame}_{home}}{\text{avg.leaguwide.xG}_{pergame}_{away}}$$

To apply the equations above, we use Liverpool as a working example. Given that each team in the league has been assigned a relative offensive and defensive metric, and that we know each of Liverpool's remaining

fixtures and where they will occur (Home or Away), we can predict how many goals Liverpool and their opponent will score in a given match. Using the formulas from earlier, we create two new formulas calculating Liverpool's predicted home and away goals against a given opponent:

$$Pred.LiverpoolGoals_{home} = (OffensiveRating_{Liverpool})(DefensiveRating_{Opponent})(HomeAdvantage)$$

$$Pred.LiverpoolGoals_{away} = (OffensiveRating_{Liverpool})(DefensiveRating_{Opponent})$$

Using these equations, we generate the "Lambda" of the Poisson process of our simulations. Below, we see the predicted number of goals for each of Liverpool's remaining games. By applying this lambda" to both teams in every remaining fixture, we can generate simulations for every remaining fixture.

##	Liverpool-Tottenham Hotspur	Crystal Palace-Liverpool
##	"1.35-1.11"	"0.95-1.48"
##	Liverpool-West Bromwich Albion	Newcastle United-Liverpool
##	"2.31-0.43"	"0.8-1.42"
##	Southampton-Liverpool	Liverpool-Burnley
##	"0.93-1.38"	"1.65-0.57"
##	Liverpool-Manchester United	Tottenham Hotspur-Liverpool
##	"1.62-1.01"	"1.19-1.26"
##	West Ham-Liverpool	Liverpool-Brighton
##	"1.17-1.47"	"1.36-1.04"
##	Liverpool-Manchester City	Leicester-Liverpool
##	"1.18-1.24"	"1.28-1.45"
##	Liverpool-Everton	Sheffield United-Liverpool
##	"1.63-1.2"	"0.8-1.85"
##	Liverpool-Fulham	Wolverhampton Wanderers-Liverpool
##	"2.08-0.88"	"0.76-1.46"
##	Liverpool-Chelsea	Arsenal-Liverpool
##	"1.04-1.3"	"0.86-1.33"
##	Liverpool-Aston Villa	Leeds-Liverpool
##	"1.35-1.29"	"1.24-2.05"
##	Liverpool-Newcastle United	Manchester United-Liverpool
##	"1.52-0.74"	"1.09-1.51"
##	Liverpool-Southampton	West Bromwich Albion-Liverpool
##	"1.49-0.86"	"0.47-2.15"
##	Burnley-Liverpool	Liverpool-Crystal Palace
##	"0.61-1.53"	"1.59-0.88"

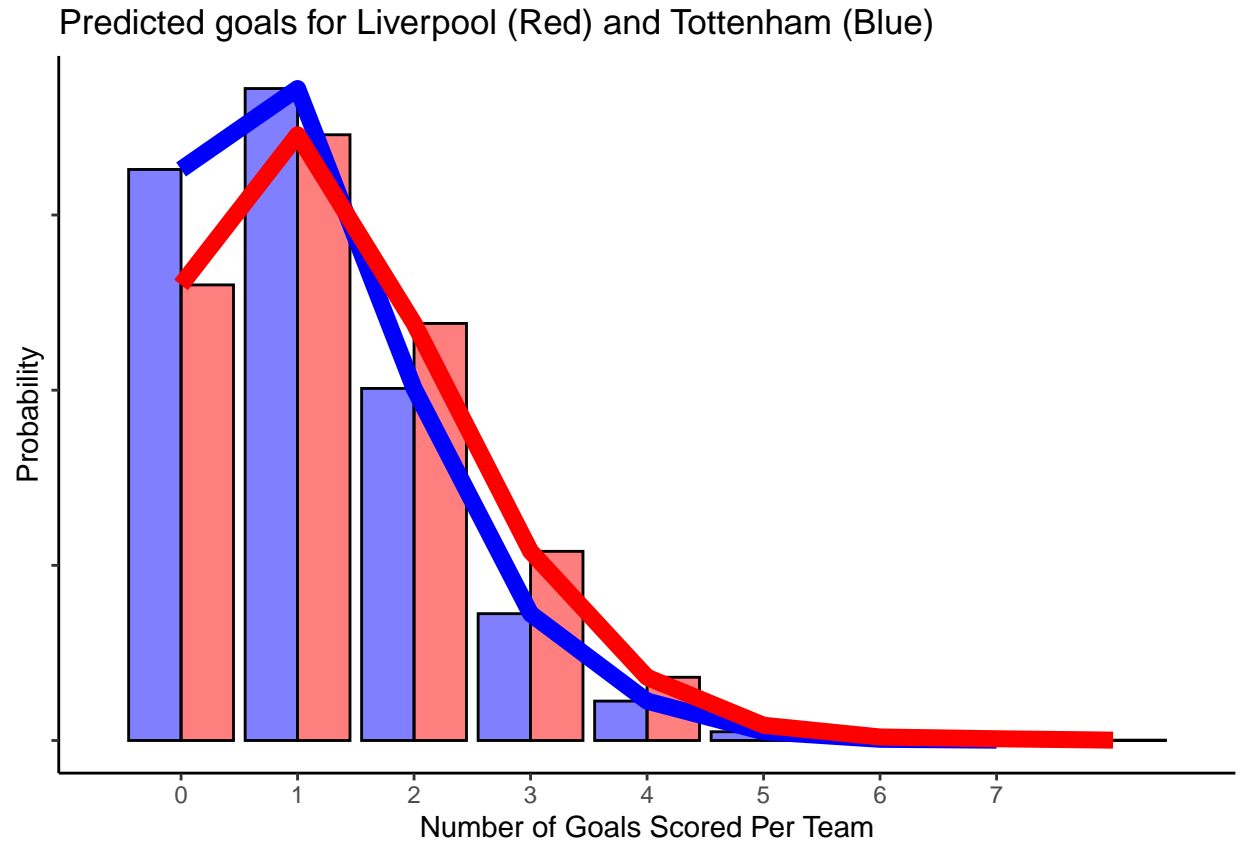
```
## # A tibble: 1 x 4
##   home      predicted_home_goals predicted_away_goals away
##   <chr>          <dbl>          <dbl> <chr>
## 1 Liverpool            1.36            1.11 Tottenham Hotspur
```

To show the Poisson process at work, Liverpool's gameweek 13 fixture against Tottenham is simulated 10000 times, with each team's goal distribution plotted below. Assuming that goals scored follows a Poisson distribution, the probability that a team scores n goals in one match is defined as:

$$P(n) = \frac{\lambda^n e^{-\lambda}}{n!}$$

It should also be noted that the maximum number of goals any team will score in the fixture has been set to 7 arbitrarily, given the extremely low probabilities of scoring 8+ goals in a top-level domestic match. Given

that Given that Liverpool's lambda is 1.35 and Tottenham's lambda is 1.11, we would expect Liverpool's distribution to be pushed further towards the right.



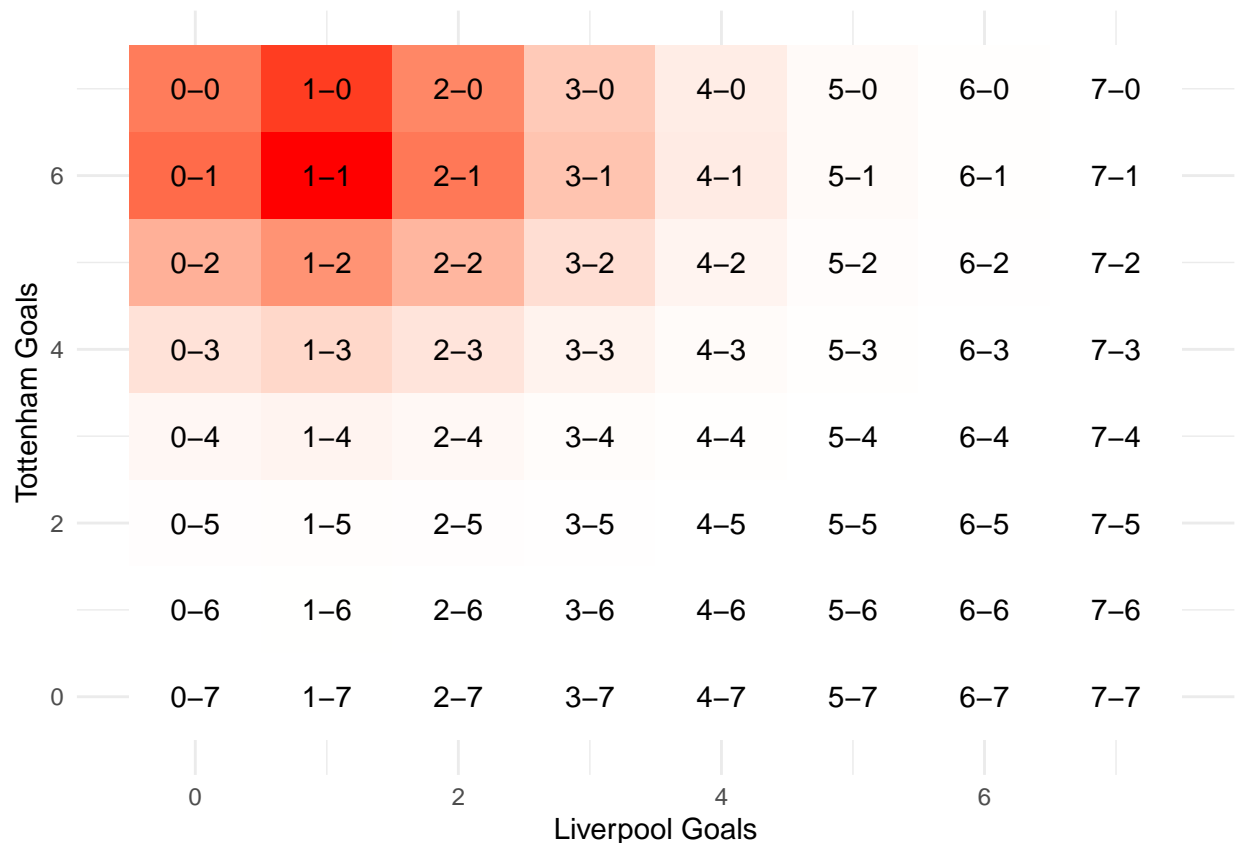
As predicted, Liverpool's goal distribution (in Red) is shifted rightward compared to the Tottenham goal distribution. Although the probability of Tottenham scoring 0 or 1 goal is greater than that of Liverpool, the probability of Liverpool scoring 2 or more goals exceeds that of Tottenham. Because football is a game of competing totals, a matrix can be made to show the probability of a particular scoreline occurring. If we define the highest row to correspond to the number of goals Liverpool scores, and the leftmost column to correspond to the number of goals Tottenham scores, then we can determine the probability that a particular scoreline occurs. By adding up the upper triangle of the probability matrix, we can calculate the probability that Liverpool wins the match. Likewise, we can calculate the probability that Tottenham wins the match (lower triangle) and the probability of a draw occurring (main diagonal).

```
## home win %:
## 42.49608
```

```
## draw %:
## 26.93578
```

```
## away win %:
## 30.55724
```

According to the probability matrix, Liverpool wins ~42.5% of the time. Tottenham wins 30.6% of the time, and a draw occurs ~26.9% of the time. The cleaned up results of this probability matrix are published below in a tile plot.



The tile plot above indicates what our intuition would tell us about a soccer match between two relatively even teams (tied for 1st in the league as of their gameweek 13 matchup). With redder squares corresponding to more likely outcomes, the Poisson process expects 1-1 to be the most common result between the two teams, followed by a 1-0 Liverpool win and a 0-1 Tottenham win.

A similar Poisson process is applied to every remaining fixture this season. It is here where the Monte Carlo simulation is applied to the model. Since every fixture has been assigned a distribution of results (essentially a probability matrix and the tile plot above are showing), we can sample from each distribution a certain number of times (100 in the case of this paper) and generate the average result and goal difference of a sampled fixture. After summing each team's average result and goal difference across all of their remaining fixtures, these results can be combined with the current league standings to generate a final predicted league table, shown in the results section below.

## Results

The results of the Monte Carlo simulations of the Poisson process are tabulated below. After simulating each fixture from gameweek 13 onward, Chelsea end up earning the most points per fixture by a solid margin ahead of current champions Liverpool, Manchester City and Aston Villa. As for average goal difference across the simulations, the same four teams are represented at the top, with West Bromwich Albion, Burnley, and Sheffield United in the relegation zone.

As for the combined standings (between the simulated and current league standings), Liverpool finishes as champions once again, with Manchester City and fairytale team Aston Villa rounding out the top 4. As for relegation, Fulham manage to barely edge out Burnley, who are sent down to the Championship with Sheffield United and league newcomer West Bromwich Albion.

Table 3: Simulated Premier League Standings Gameweek 13 Onward

team	average_points	average_goal_difference	points_per_game
Chelsea	48.72	18.12	1.8738462
Liverpool	47.18	15.97	1.8146154
Manchester City	47.84	15.36	1.7718519
Aston Villa	48.54	14.17	1.7335714
Tottenham Hotspur	41.07	7.88	1.5796154
Leicester	40.15	6.31	1.5442308
Brighton	40.02	5.97	1.5392308
Everton	39.68	4.95	1.5261538
West Ham	38.76	4.55	1.4907692
Manchester United	37.70	1.84	1.3962963
Southampton	34.27	-1.86	1.3180769
Leeds	33.60	-4.33	1.2923077
Arsenal	32.28	-2.30	1.2415385
Crystal Palace	31.75	-3.91	1.2211538
Newcastle United	31.39	-5.77	1.1625926
Wolverhampton Wanderers	28.37	-8.57	1.0911538
Fulham	27.68	-11.85	1.0646154
Sheffield United	25.34	-13.83	0.9746154
Burnley	25.42	-14.06	0.9414815
West Bromwich Albion	15.34	-28.64	0.5900000

Table 4: Final Predicted Premier League Standings

team	predicted_points	predicted_goal_difference
Liverpool	72.18	24.97
Chelsea	70.72	31.12
Manchester City	66.84	21.36
Aston Villa	66.54	22.17
Tottenham Hotspur	66.07	21.88
Leicester	64.15	15.31
Everton	59.68	7.95
West Ham	58.76	9.55
Manchester United	57.70	3.84
Southampton	57.27	5.14
Brighton	50.02	-0.03
Crystal Palace	48.75	-2.91
Newcastle United	48.39	-7.77
Leeds	47.60	-9.33
Wolverhampton Wanderers	45.37	-13.57
Arsenal	45.28	-7.30
Fulham	35.68	-21.85
Burnley	34.42	-26.06
Sheffield United	26.34	-29.83
West Bromwich Albion	21.34	-44.64



# Discussion

## Model Implications

In this paper, league results were simulated based on expected statistics collected by Understat from the first 12 gameweeks of the Premier League season. Of the displayed results, perhaps most notable is the relatively low values of total average points and total goal difference in the simulated Premier League table. In recent years, the Premier League winner has finished on 99, 98, 100 and 93 points dating back to 2017. In fact, over the last 25 years, the average league winner has accumulated just over 87 points in a crowning season. While this is not to say that the results of this model are unfounded (Manchester United and Arsenal have won the league at 75 and 78 points respectively), there is an indication that this xG model has increased the parity of most teams in the league, resulting in more ties and thus fewer points (as was the case in the Liverpool vs. Tottenham example).

One explanation for this points deflation is that the weights assigned to each team in the league were each scaled based on the league average for xG. Since dividing team xG tallies by the league average reduced the absolute strength of each team offensively and defensively, it therefore reduced the lambdas used to predict the number of goals scored by each team in every fixture. While this scaling may have been ideal for predicting one off odds of a team winning, applying a Monte Carlo simulation to these figures likely lead to an increased selection of close, low-scoring affairs between most of the teams.

## Limitations

More than anything else, this expected goals model is likely best served as a starting point for more intricate predictive models. Football is an incredibly dynamic game which changes from minute to minute, let alone game to game or season to season. As such, the results of this paper are limited primarily by three things: sample size, recency bias, and its failure to address the dynamism of the league itself.

There are a few reasons why this model relies solely on data from only the first 12 gameweeks of the 2020-2021 Premier League season. One reason is because this is the only season to have these specific 20 teams. The promotion/relegation system of English football makes it impossible for the same 20 teams to play one another from one season to the next (unlike the North American franchise model). Another reason is because teams from one season to the next are given time to sign and release players, at times dramatically changing the lineup and therefore results of a team (see: Chelsea). While the limited sample size manages to account for current teams and their rosters, it creates problems for the study with regards to recency bias.

There is a fine balance between focusing on recent events over historical trends. One issue this model faces is that it does not account for outlier fixture results that have already occurred this season. For example, although Aston Villa trounced Liverpool 7-2 (xG: 3.08-1.66) in gameweek 4, it is becoming more and more plausible that this result was nothing more than a freak occurrence, an onslaught brought on by a few early goals and some questionable goalkeeping. Beyond statistical anomalies, the 12 week sample size this model uses may have disproportionately affected teams on cold streaks (Manchester City) and teams with injury troubles (Burnley). While xG models in general tend to account for underperforming and overperforming teams, the model in its current state is not able to account for injury riddled teams.

This dilemma leads to perhaps the biggest limitation of the model, the fact that it tries to apply a relatively static model to an incredibly dynamic game. Having been without key defender Ben Mee at the start of the season, Burnley struggled to keep the ball out of the net. Since his return however, Burnley have reverted back to their usual formidable defensive block. Another prime example of dynamism in football is with regards to player and coaching personnel. Just as Bruno Fernandes did last year, one new signing, or one player returning from injury can completely change a team's season around. Additionally, a team with several new pieces may take time to gel together as a cohesive unit.

## Future Studies

Future works may wish to incorporate different expected stats to build their offensive and defensive ratings for teams in the Premier League. For example, substantial data exists with regards to non-penalty xG and xGA, a metric that can help reduce the variance and unpredictability of penalties in a football match. Two other metrics that Understat provides that may be of use include passes allowed per defensive action and opponent passes allowed per defensive action, measures that can help proxy a team's defensive pressure and ability to win the ball back quickly.

A significant complication of this model is that it assumes that every team will play the same way against every other team in the league. Even the most causal football fan can watch a game and immediately conclude which side is playing a riskier attacking style and which side is parking the bus in front of their goal. Stylistic match ups play an important role in determining match results, so accounting for certain match ups may help improve the efficacy of any future predictive models.

Finally, it should be noted that the Poisson model for goal distributions may not be the most ideal to represent goals. Since the distribution assumes that the likelihood of an event happening is the same throughout an entire time period, it assumes that the rate of goals being scored in a football match is distributed relatively evenly. Research has shown however that goal frequency increases the further one goes into a game (perhaps due to substitutions, pressure, and physical and mental fatigue). Future research may address this shortcoming by mapping goals to a different distribution, such as a bivariate Poisson or Weibull distribution.

## Appendix

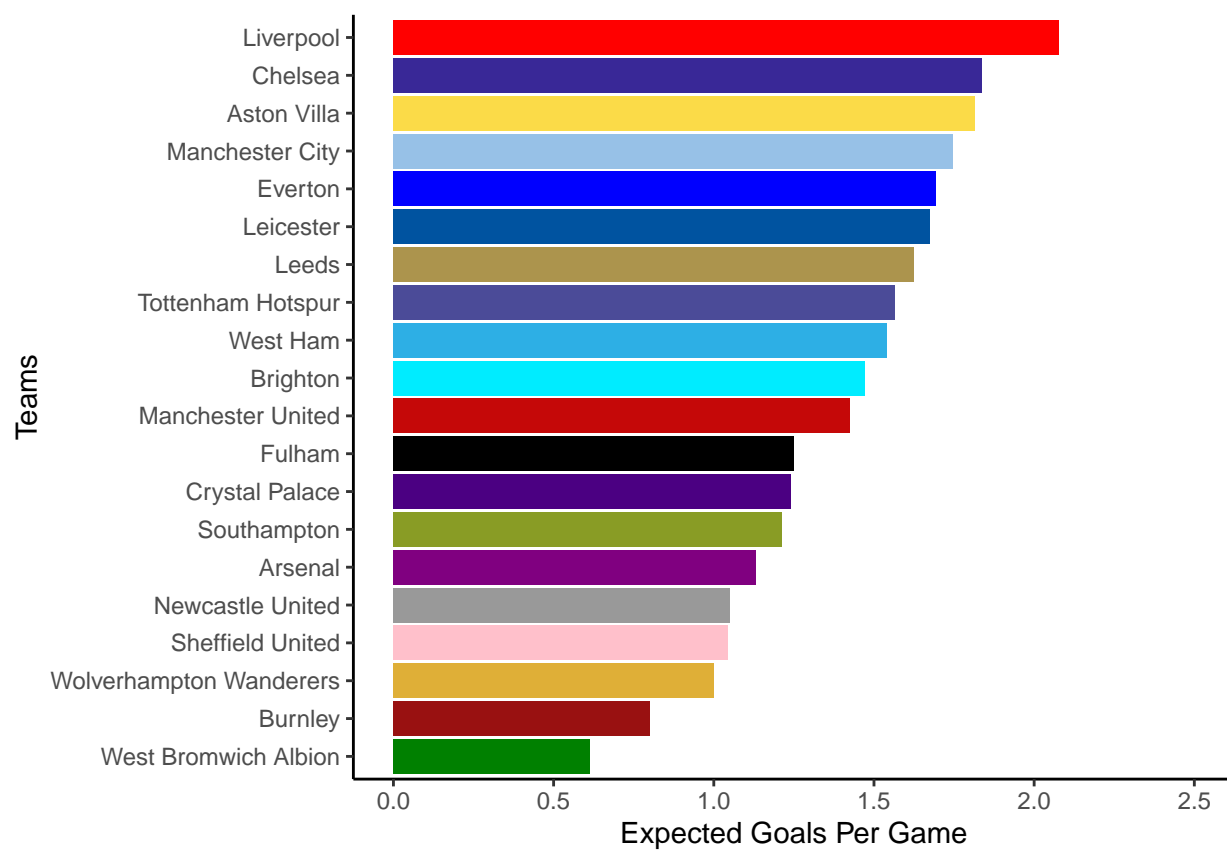


Figure 1: PL Teams Ranked by xG Per Game

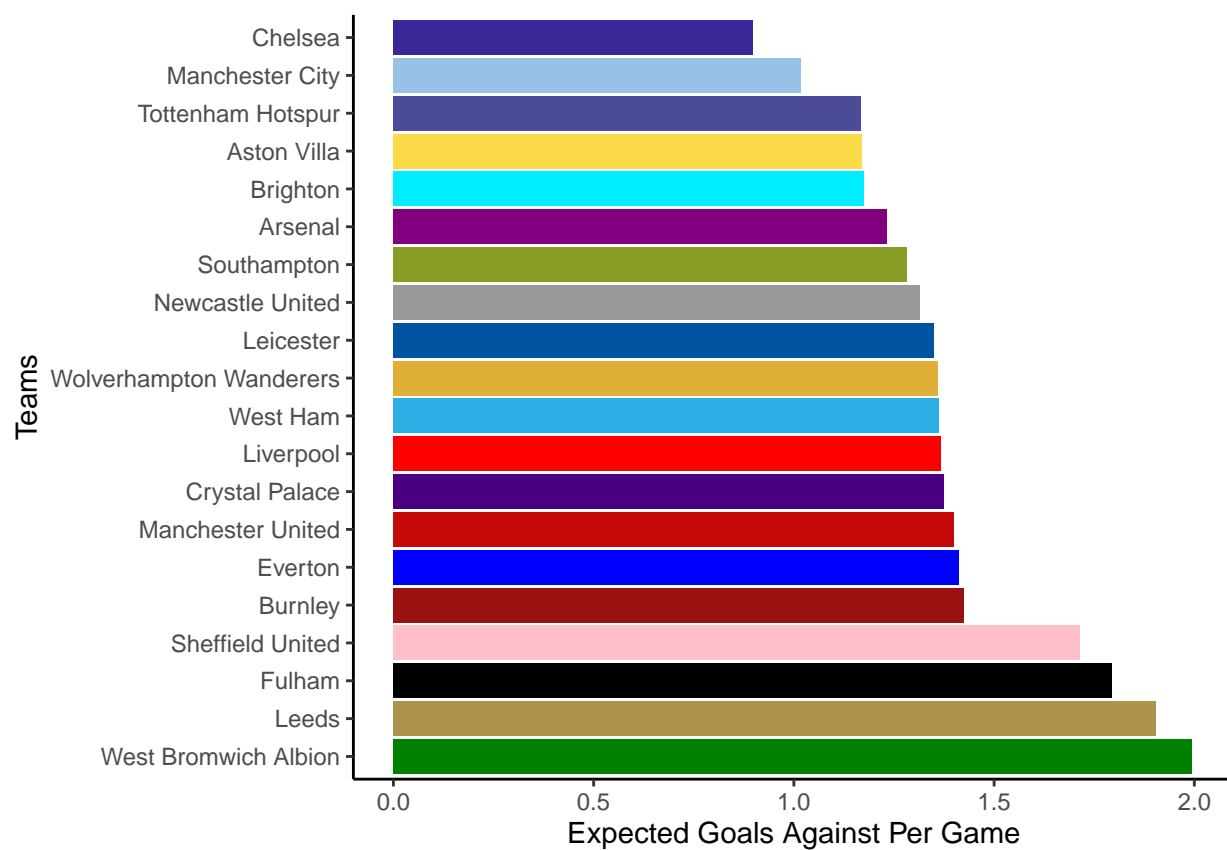


Figure 2: PL Teams Ranked by xGA Per Game

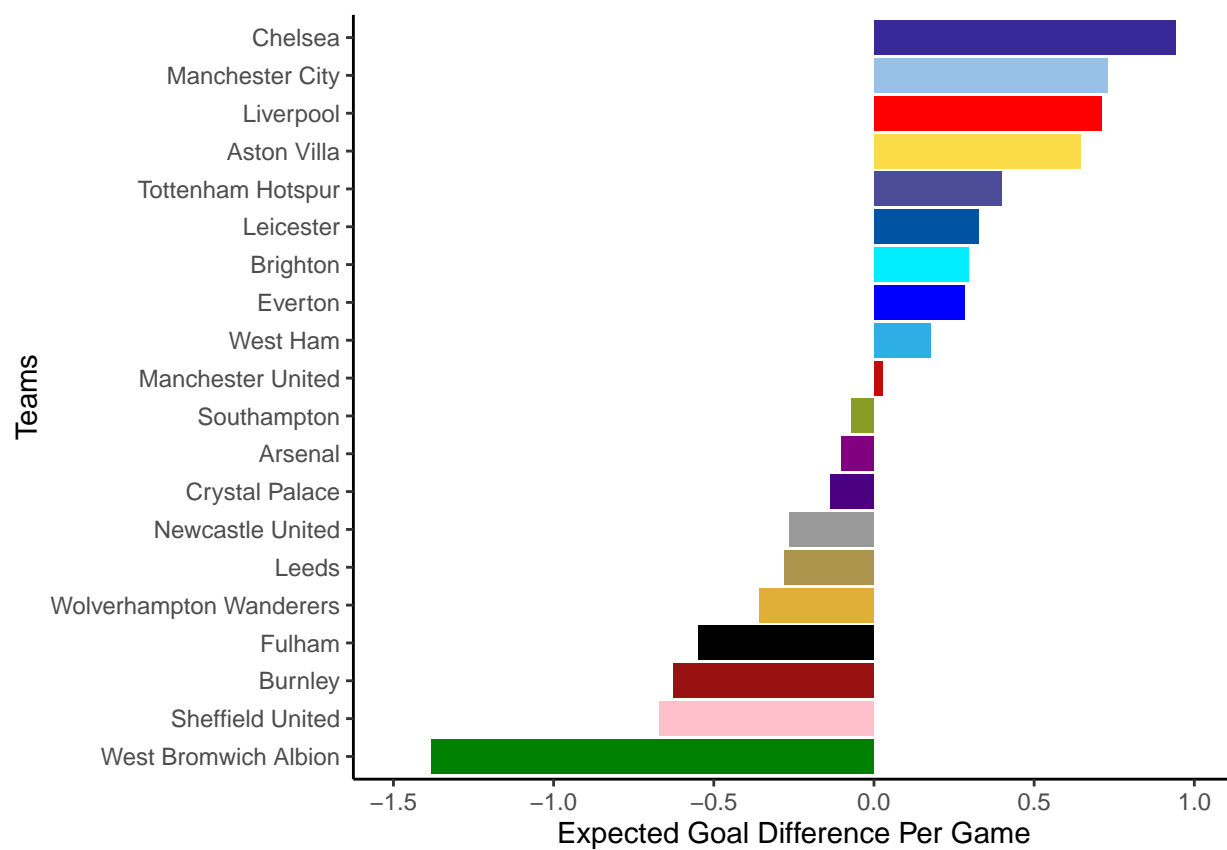


Figure 3: PL Teams Ranked by xGD Per Game

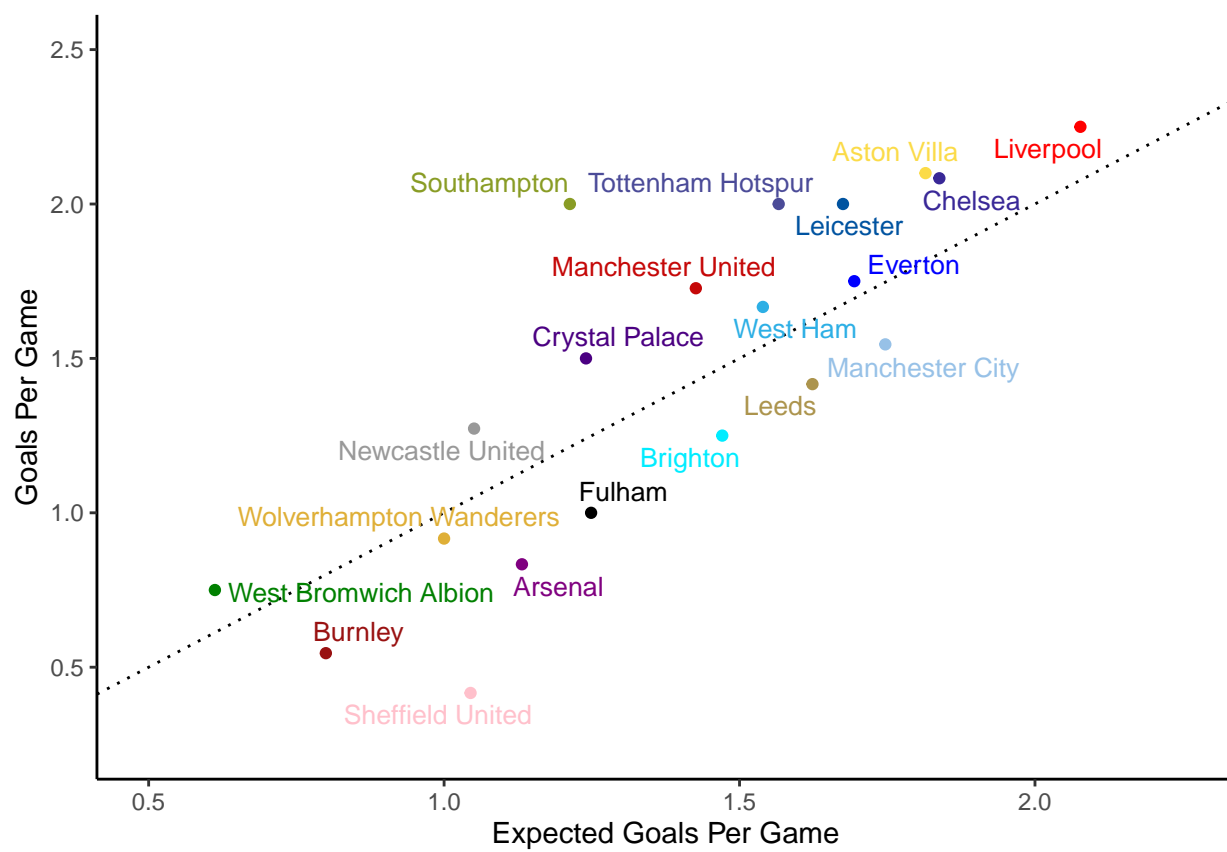


Figure 4: Expected vs. Actual Goals Per Game

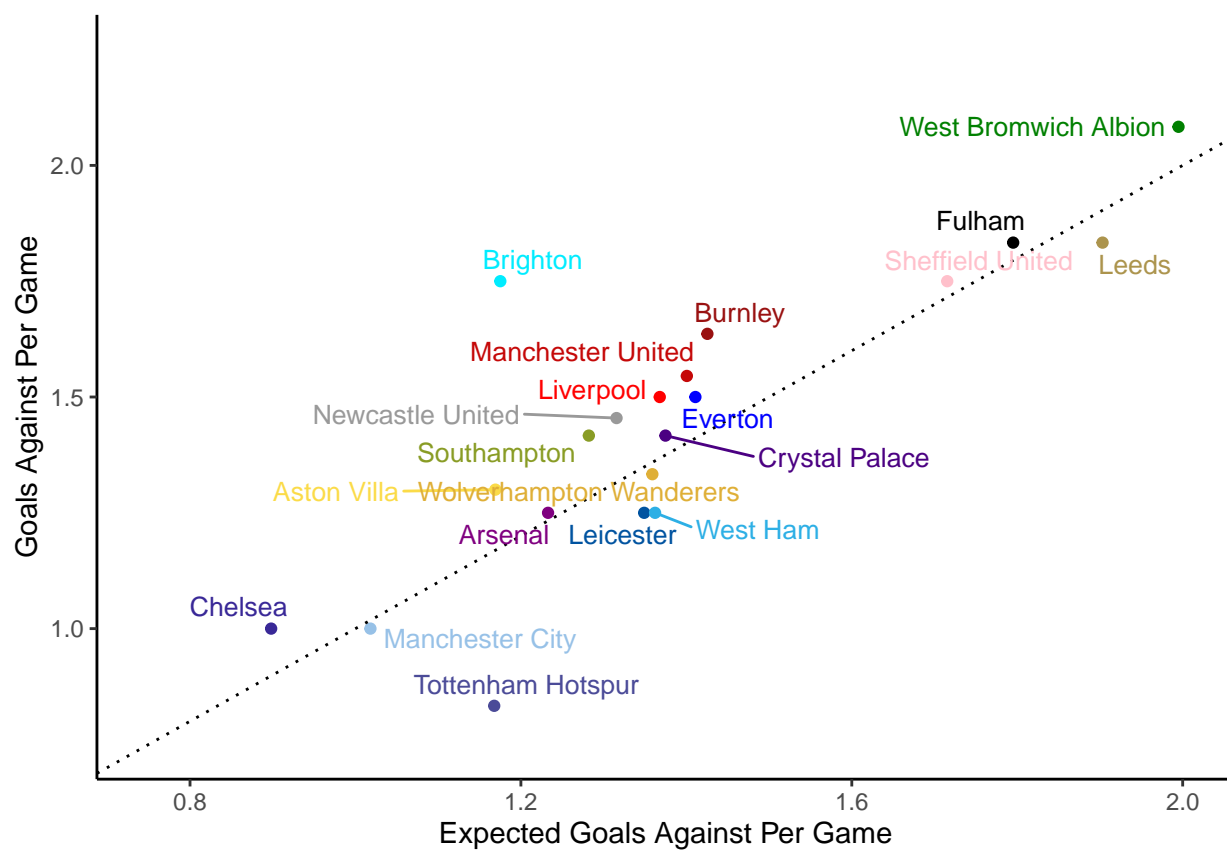


Figure 5: Expected vs. Actual Goals Against Per Game

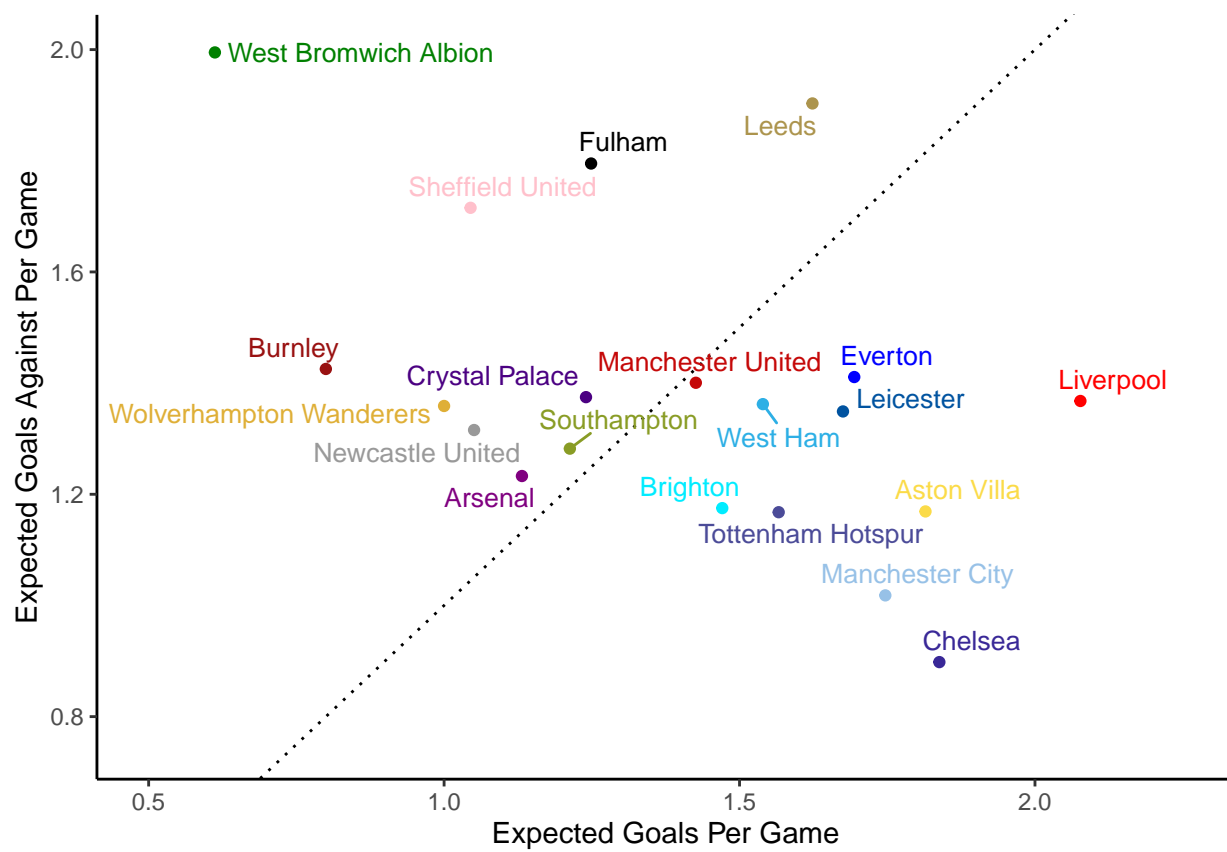


Figure 6: Expected Goals vs. Expected Goals Against



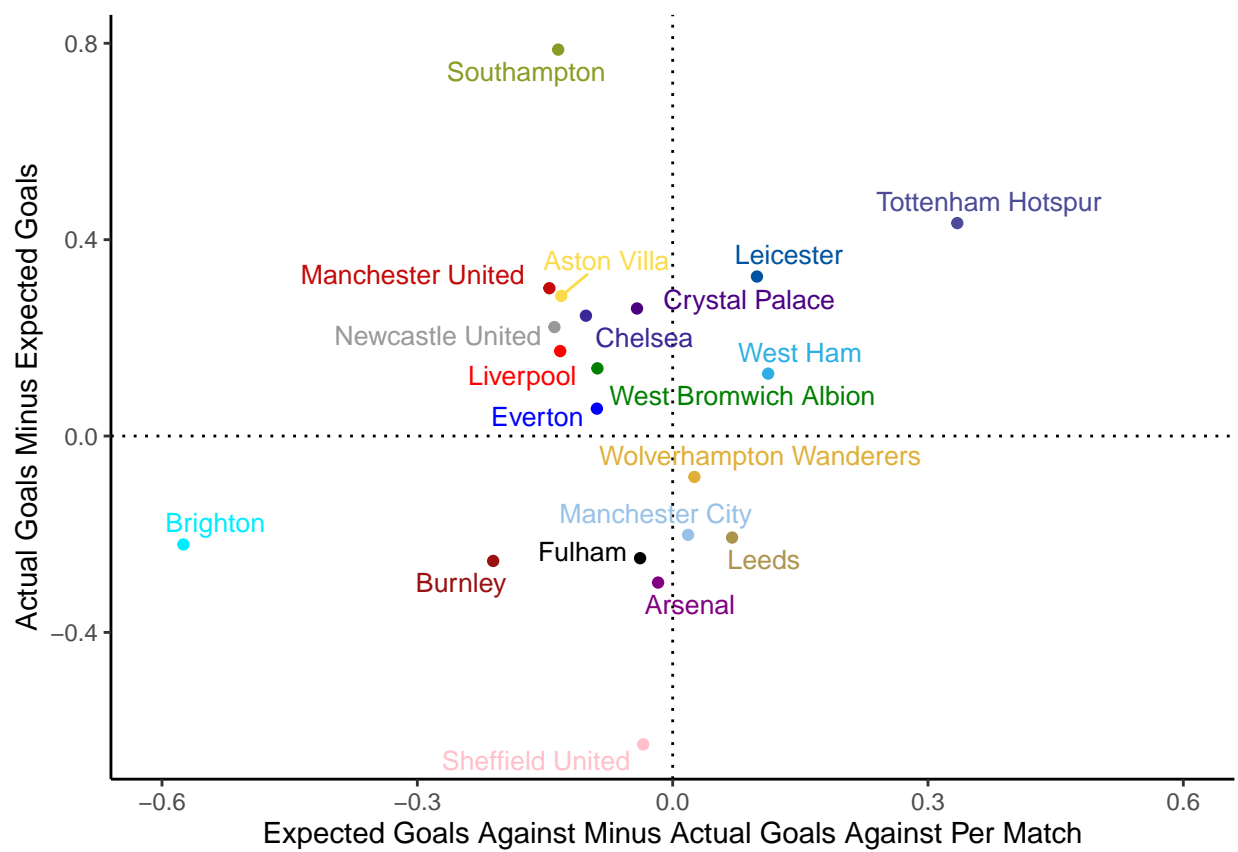


Figure 7: Under and Overperforming Teams Based on Expected and Actual Goals

## References

- Anand, V. (2020). Github Repository: Fantasy-Premier-League. Retrieved from <https://github.com/vaastav/Fantasy-Premier-League>
- Chu, S. (2003). Using Soccer Goals to Motivate the Poisson Process. Retrieved 2020, from <https://pubsonline.informs.org/doi/abs/10.1287/ited.3.2.64>
- Dyte, D., & Clarke, S. R. (2017). A ratings based Poisson model for World Cup soccer simulation. Retrieved 2020, from <https://orsociety.tandfonline.com/doi/abs/10.1057/palgrave.jors.2600997#.X-JMJdhKiUk>
- Hickman, R. (2020). An Introduction to Modelling Soccer Matches in R (part 2). Retrieved 2020, from [https://www.robert-hickman.eu/post/dixon\\_coles\\_2/](https://www.robert-hickman.eu/post/dixon_coles_2/)
- Hao Zhu (2019). kableExtra: Construct Complex Table with ‘kable’ and Pipe Syntax. R package version 1.1.0. <https://CRAN.R-project.org/package=kableExtra>
- Kamil Slowikowski (2020). ggrepel: Automatically Position Non-Overlapping Text Labels with ‘ggplot2’. R package version 0.9.0. <https://CRAN.R-project.org/package=ggrepel>
- R Core Team (2020). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL <https://www.R-project.org/>.
- Sheehan, D. (2017). Predicting Football Results With Statistical Modelling. Retrieved 2020, from <https://dashee87.github.io/data-science/football/r/predicting-football-results-with-statistical-modelling/>
- Smith, A. (2017). Sky Sports bust common football myths: Home advantage? Retrieved 2020, from <https://www.skysports.com/football/news/11096/10955089/sky-sports-bust-common-football-myths-home-advantage>
- Taylor, M. (2016). How To Frame An Individual Match Outcome. Retrieved 2020, from <http://thepowerofgoals.blogspot.com/2016/02/how-to-frame-individual-match-outcome.html>
- Understat. (2017). EPL xG Table and Scorers for the 2020/2021 season. Retrieved 2020, from <https://understat.com/league/EPL>
- Wickham et al., (2019). Welcome to the tidyverse. Journal of Open Source Software, 4(43), 1686, <https://doi.org/10.21105/joss.01686>