Predicting the Premier League Table Using Poisson Random Number Generation and Monte Carlo Simulation

Abstract

The Premier League is a popular soccer league founded in the United Kingdom. Using only the match results of each team for the first half of the 2021-2022 Premier League season, we predicted the final standings for 20 clubs. We used Poisson random number generation to predict the points of each match for the second half of the season. Then we totaled the points per team to calculate a final table. Applying Monte Carlo sampling, we repeated this simulation 100 times and took the mean number of points to obtain a predicted final table. Overall, this method gave a fairly accurate prediction of what the Premier League end-of-season standings would look like based on the results from the first half of the season. This offers clubs an idea of where they may end the season and can make the proper adjustments in the middle of the season.

1 Introduction

The Premier League is a soccer league founded in the United Kingdom. Twenty teams from twenty different clubs participate in the competition, and they play each other twice for a total of 38 games. In each match, one team is the "home team" and the other is the "away team." The home team hosts the away team at their stadium. When calculating the rankings for each team, a win is awarded 3 points, a tie is awarded 1 point, and a loss receives 0 points. These points are totaled for each team, and the teams are ranked after each round.

In this project, we decided to predict the Premier League final rankings of each team using data from the first half of the 2021-2022 season. We wanted to potentially predict the rankings of each team at the end of the season and even predict a winner. This analysis could drastically help domestic and international fans bet accurately on certain teams. Further, clubs may also benefit from these predictions, because this could tell them if they need to buy players, sell players, or make serious revisions to their respective clubs in the middle of the season to end based on a certain rank. We plan to use both random number generation for Poisson random variables and Monte Carlo sampling to predict this result.

These are the 20 clubs in the Premier League for this season:

2 Methods

For this project, we first retrieved all the data of the 2021-2022 Premier League season ending in May 2022 from *Fixture Download* [1] (which provides data for various sports) and then cleaned the data. The dataset contained variables such as the Match Number, Round Number, Date, Location, Home Team, Away Team, and Result. A few of the rows had the "Round Number" variable mislabeled, so we altered them. We restructured the data to separate the results for the Home and Away teams. Since our goal is to predict the second half of the season based on the first-half results, we omitted the results for all the matches after Round 19.

We wrote functions to count the total number of points per team in both the first half of the season and the final standings, and then we created another datatable based on these points to determine rankings. This datatable only included the team name, the number of points each team had midseason, and the total number of matches played. After we finished cleaning the data, we used the Poisson distribution to generate the anticipated number of goals scored for each home and away team per match. The Poisson distribution makes sense in this context because each match score is independent of each other in every round, there are 38 rounds with matches played nearly every weekend (which is a fixed interval at approximately a constant rate), and the number of goals is discrete. We calculated the lambda values for the Poisson simulation based on data from the first half of the season and the formula [2]:

$$\lambda_{\text{home team}} = \frac{\text{average goals scored by home team} + \text{average goals conceded by away team}}{2}$$

$$\lambda_{\text{away team}} = \frac{\text{average goals conceded by home team} + \text{average goals scored by away team}}{2}$$

As shown above, the home team and the away team have different lambda values. These lambda values are used to predict the number of goals scored for the home team and the away team per match. Since the score of each match is dependent on the performance of both teams (for instance, one team's defense could affect the other team's ability to score), we wanted to account for this dependence when calculating the lambda values. Therefore, both lambda values take the average of the average number of goals scored by a team and the average number of goals scored against their opposition in the first half of the season. We assumed each club would perform similarly as they did in the first half of the season for the second half, and each club will not have any sudden strings of wins or losses that were not evident in the first half season's data. If not, the average number of goals scored and conceded would change after each round in the second half of the season, and we would instead be performing Bayesian Inference.

After we calculated the lambda values for each round, we randomly generated a Poisson random variable for each home and away team per match. We could have used the Poisson random number generator in R, but we wanted to use the algorithms we learned in class. Due to the discrete properties of Poisson random variables, we could not use the accept/reject or inverse sampling methods directly. Instead, we used the idea of a homogeneous Poisson stochastic process [3]. The inter-arrival times in this process are independent exponential random variables. These random variates are then added to get the smallest number of variates with a sum not exceeding one, giving us a Poisson random variable. To generate exponential random variables, we derived the inverse function for exponential random variables, which is $E =- log(u)/\lambda$, where U follows a standard uniform distribution. The process of generating Poisson random variables in R programming is shown in Appendix B Code Block B1. This value from the Poisson random number generation represents the number of goals scored by each home and away team. If the values were the same, each team was granted 1 point. Otherwise, the winning team would be granted 3 points and the other team would receive 0 points. The points were then totaled for each team after each round. Once the total points were summed and the teams were ranked, we had one potential version of the Premier League final standing table.

Then, we implemented the Monte Carlo simulation. We ran the above process 100 times, generating 100 versions of the final standings table. Grouping the data by team, we calculated the mean number of points, rounded the point values (as points can only be integers), and calculated the standard error of the points per team, shown in Appendix B Table B1. We reordered the data by most to least number of points to predict the final rankings for the Premier League as shown in Table 1.

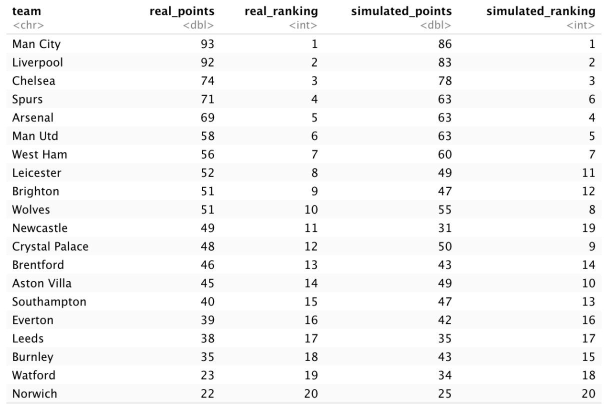
3 Algorithm:

1. Calculate lambda values for every home and away team in each match for Rounds 20-38.
2. Generate a Poisson random variable for each home and away team using random number generation to get a number of goals scored per team.
    a. Generate u from Unif(0,1),
    b. Generate E from inverse transformation, $E =- log(u)/\lambda$,
    c. Add up E's before the sum exceeds 1 and return the number of E's, which is a Poisson random variate.
3. Determine whether each team won/tied/lost based on the result from the generated Poisson random variable and give points to each team.
4. Sum up the points from the first half of the season and the predicted number of points from the second half of the season for each team.
5. Repeat steps (1-5) 100 times, which performs Monte Carlo simulation.
6. Group all of the data by team name and take the total average number of points scored and standard error per team, then round these point values.
7. Reorder the grouped data by descending the number of points to get the final Premier League predicted table.

4 Results and Discussion

The final predicted standings from our simulations and the real observed team standings, as well as the number of points for each team, are displayed in Table 1. Figure 1 shows the barplot visualization of the ranking difference between real ranking and simulated ranking. The barplot of the point differences between real points and simulated points can be found in Appendix A Figure A1.

The Wilcoxon signed-rank test applied to the real and simulated points shows that there is no significant difference between the predicted number of points of each team and their real points gained throughout the season with the p-value 0.9552. Thus, our predicted final table follows the true Premier League table well. Table 1 and Figure 1 both show that we correctly predicted the top 3 clubs and a few other club rankings. They also show there are slight variations for the other clubs in between.



team <chr>	real_points <dbl>	real_ranking <int>	simulated_points <dbl>	simulated_ranking <int>
Man City	93	1	86	1
Liverpool	92	2	83	2
Chelsea	74	3	78	3
Spurs	71	4	63	6
Arsenal	69	5	63	4
Man Utd	58	6	63	5
West Ham	56	7	60	7
Leicester	52	8	49	11
Brighton	51	9	47	12
Wolves	51	10	55	8
Newcastle	49	11	31	19
Crystal Palace	48	12	50	9
Brentford	46	13	43	14
Aston Villa	45	14	49	10
Southampton	40	15	47	13
Everton	39	16	42	16
Leeds	38	17	35	17
Burnley	35	18	43	15
Watford	23	19	34	18
Norwich	22	20	25	20

Table 1. Real and Simulated Premier League final rankings

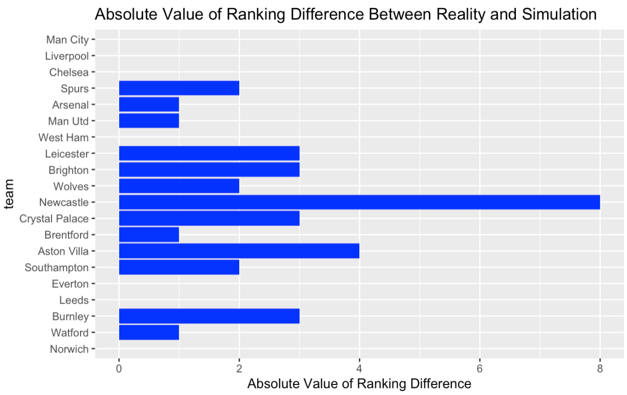


Figure 1. Ranking differences between real and simulation results

5 Limitation

There are definitely some features that we did not account for in our final model. For instance, Newcastle had a long string of wins in the second half of the Premier League season which is not reflected in our analysis. Chelsea lost or tied unexpected matches in the second half of the season that cost them points. Finally, the number of points for the top clubs seems to be slightly underestimated. Despite these drawbacks, we were able to predict most of the rankings correctly. Since Monte Carlo simulation provides an efficient way to simulate the process of prediction and enhance the accuracy of our estimation, we utilized Monte Carlo simulation to predict and assess the accuracy of our final standing table. As we were running the Monte Carlo simulations on our laptop, we ran into long runtime issues for more than 100 iterations. Thus, we settled on running only 100 iterations of the Monte Carlo simulations on which we based our final predicted table. With more iterations, we could have reduced the standard error of points for each team. Regardless, we still got a fairly accurate result and were satisfied with what our final table displayed.

Variations in our predicted table could either be random or due to outside factors. It makes sense that there is a greater variation among the teams in the middle of the table because these teams' rankings are differentiated by only a few points. In contrast, there are greater point differentials for teams at the top and bottom of the standings. However, our results show that not only is it very difficult to accurately predict the final Premier League table, but there are potentially some outside sources of variation that we did not account for. This could include: shifts within the clubs, unexpected player injuries, influential players sick with COVID-19, and unexpected fluctuations in individual players or group performances, to name a few. Another factor that could potentially influence the final table is scheduling. Due to the pandemic, matches were occasionally postponed to later times in the middle of the week between two weekends. Thus, minimal rest and intense scheduling could have a large impact on the players, as it did this season with the club Chelsea. This can also affect our lambda values in the Poisson distribution because the matches were not played at a constant rate, which was one of our assumptions. However, aside from scheduling, these factors are unpredictable and are therefore extremely difficult to account for. For these reasons, scheduling is possibly the only other factor we could have considered.

6 Conclusion

Overall, our results show where a team could rank in the Premier League based on their first half-season performance. They may either need to step up their performance, or if they are excelling, continue what was working for them. Although our process was generally accurate, further work like using Bayesian statistics and simulating from joint probability distributions can be used to generate other final tables. Then, we could compare the variances between our predicted Premier League final table using the Poisson distribution, the Bayesian approach, and the joint probability distribution approach to choose the final standing table with the least variance.

References:

1. Data Source: *Download the full English Premier League 2021/22 fixture as CSV, XLSX, ICS and JSON*. (n.d.). Fixture Download. Retrieved April 18, 2022, from https://fixturedownload.com/results/epl-2021.
2. Lahvicka, Jiri. *Using Monte Carlo Simulation to Calculate Match Importance: The Case of English Premier League*. Munich Personal RePEc Archive, 1 Sept. 2012, https://mpra.ub.uni-muenchen.de/40998/.
3. Keeler, Paul. *Simulating Poisson random variables - Direct method*. 5 Nov. 2019
4. Menyhurt, Kristof. *Predicting the Outcome of the English Premier League by Using Monte Carlo Method (in R)*. Medium, The Startup, 11 Jan. 2021

Appendix A

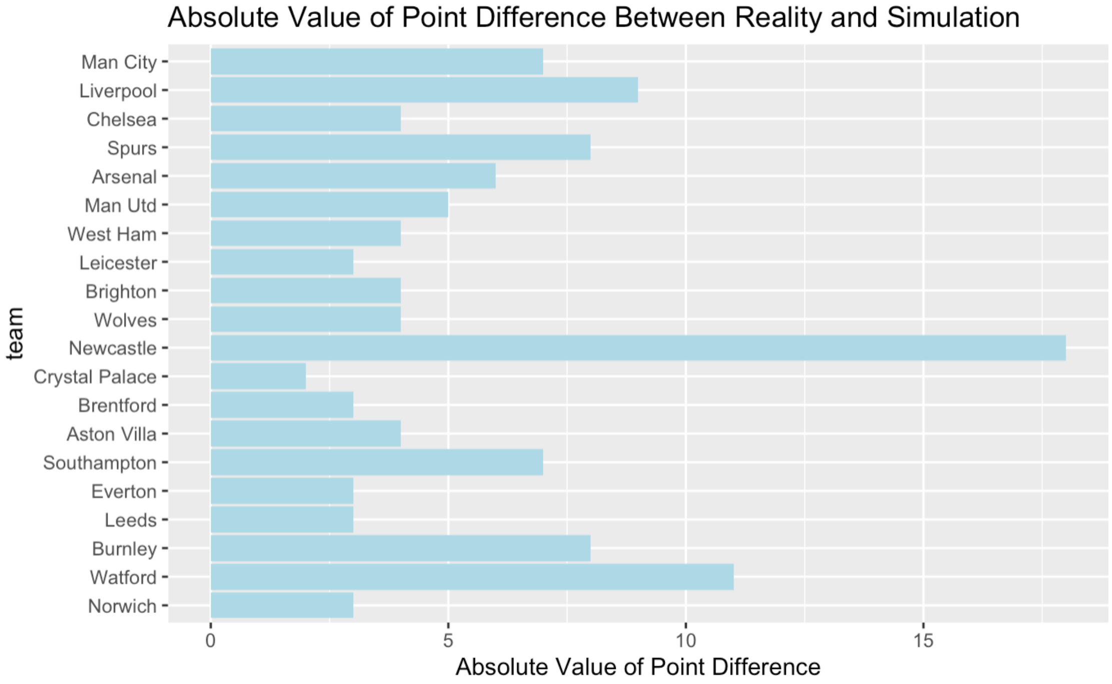


Figure A1. Barplot of the absolute value of point difference between real results and simulations for each team

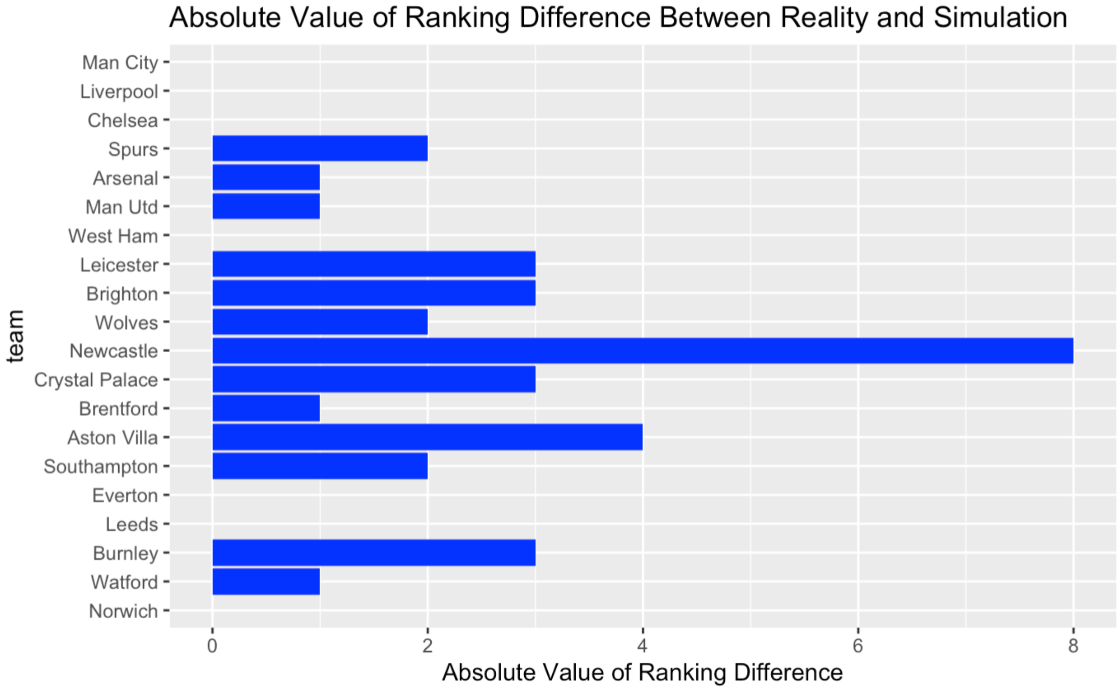
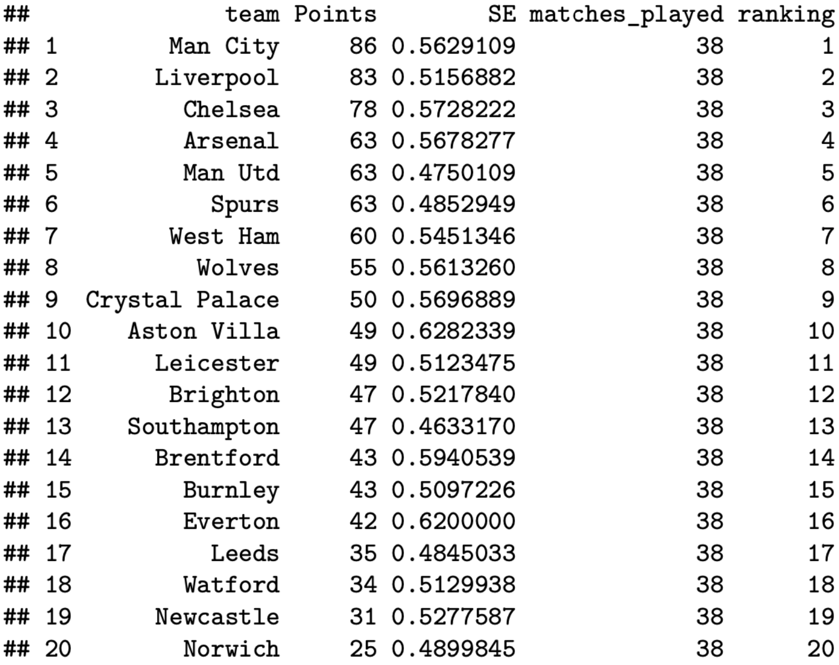


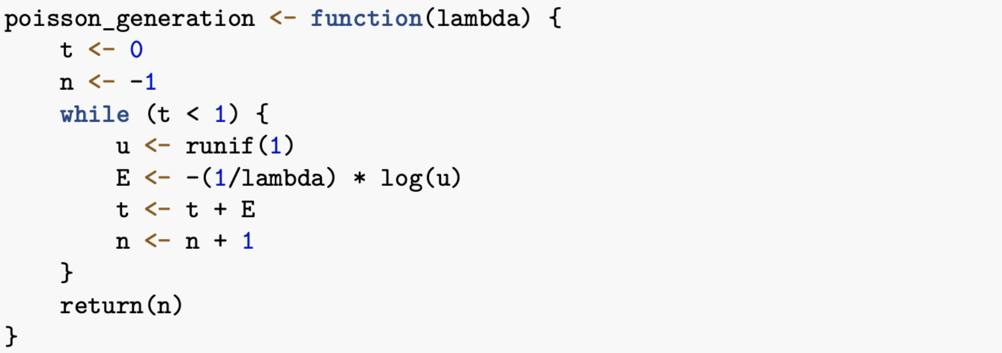
Figure A2. Barplot of the absolute value of the ranking difference between real and simulated results for each team

Appendix B



```
##                team Points        SE matches_played ranking
## 1          Man City     86 0.5629109             38       1
## 2         Liverpool     83 0.5156882             38       2
## 3           Chelsea     78 0.5728222             38       3
## 4           Arsenal     63 0.5678277             38       4
## 5           Man Utd     63 0.4750109             38       5
## 6             Spurs     63 0.4852949             38       6
## 7          West Ham     60 0.5451346             38       7
## 8            Wolves     55 0.5613260             38       8
## 9    Crystal Palace     50 0.5696889             38       9
## 10      Aston Villa     49 0.6282339             38      10
## 11        Leicester     49 0.5123475             38      11
## 12         Brighton     47 0.5217840             38      12
## 13      Southampton     47 0.4633170             38      13
## 14        Brentford     43 0.5940539             38      14
## 15          Burnley     43 0.5097226             38      15
## 16          Everton     42 0.6200000             38      16
## 17            Leeds     35 0.4845033             38      17
## 18          Watford     34 0.5129938             38      18
## 19        Newcastle     31 0.5277587             38      19
## 20          Norwich     25 0.4899845             38      20
```

Table B1. Predicted final standing table and standard error of simulations



```
poisson_generation <- function(lambda) {
    t <- 0
    n <- -1
    while (t < 1) {
        u <- runif(1)
        E <- -(1/lambda) * log(u)
        t <- t + E
        n <- n + 1
    }
    return(n)
}
```

Code Block B1. Implementation of the method for generating Poisson random variables