English Premier League Monte Carlo Analysis

Megan Dunnahoo, Jasmine DeMeyer, Macey Dodd

12/16/2021

Article

We chose the article "Using Monte Carlo Simulation to Calculate Match Importance: The Case of English Premier League" by Jiri Lahvicka. This article describes the process of using Monte Carlo simulations to predict the outcome of a match given the results of previous matches. It specifically predicts the result of the Manchester City versus the Manchester United game in 2012. It then goes further and uses Monte Carlo simulations to predict the final ranking of the teams in the English Premier League at the end of a season.

Background Information

The English Premier League is regarded as the most popular sports league in the world due to its massive audience views and impressive revenue. There are 20 teams in the English Premier League. Manchester United is considered to be the most popular football club with the Liverpool club in second. In football, a game can result in a tie as well as a win or a loss. Three points are awarded for a win, one for a draw and zero for a loss. At the end of each season, the lowest ranking three teams will be "relegated" or demoted to the lower football league, the English Football League (EFL). The highest three ranking clubs in the English Football League will be "promoted" into the Premier League.

Data and Code Setup

We got our data from football-data.co.uk. We wanted to use the specific variables FTR, FTAG, and FTHG, along with the identifier variables of Date, Away Team, and Home Team. There were no NA values in any of our selected variables.
We used four seasons ranging from years 2011-2015. Each team played 19 away games and 19 home games. We created functions to get the points scored and the outcome of the team for each team over the four years, specifying away or home games.

Monte Carlo Estimation

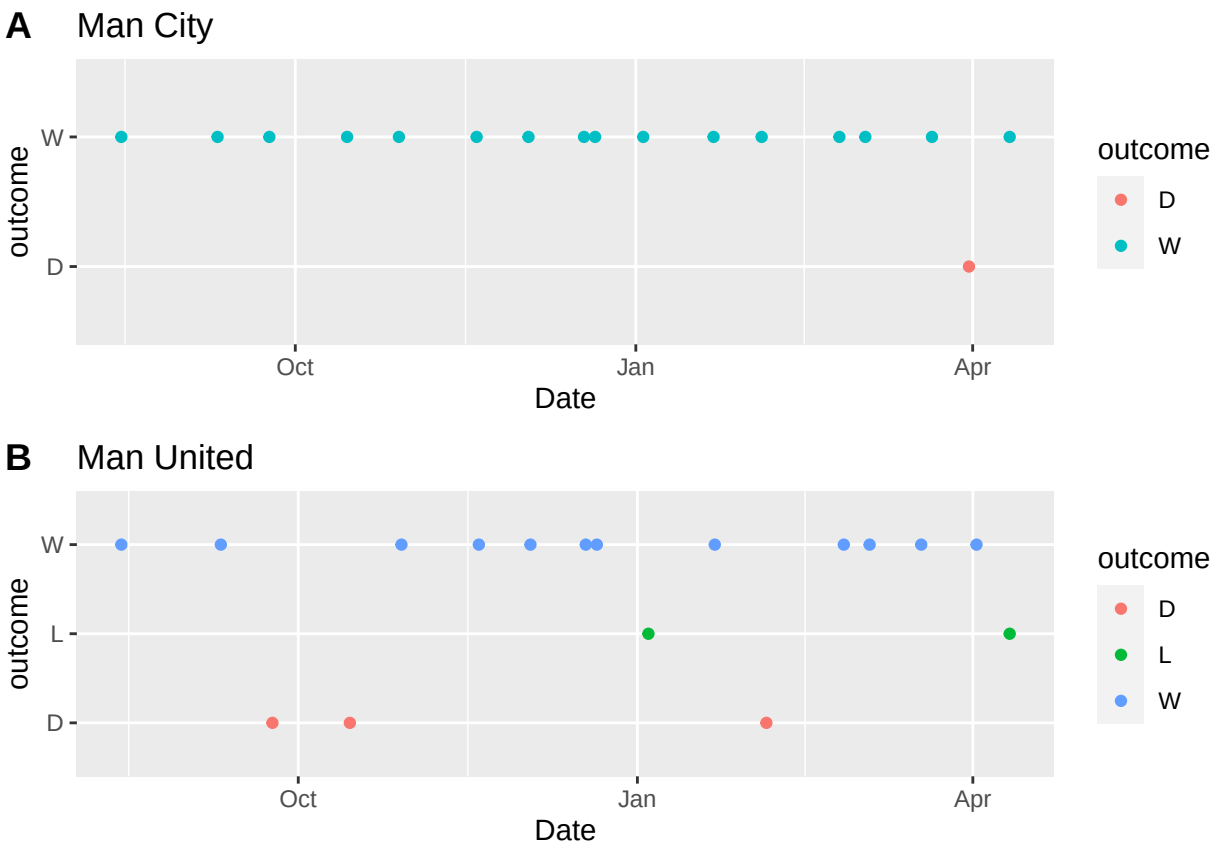
We estimated the lambda home and lambda away values, which are the expected goals scored by the home and away team respectively, using Monte Carlo. These lambda values are assumed to be independent Poisson distributed variables and are calculated using the last 19 matches for each team. The article ran 10,000,000 simulations, but due to our low computational power, we chose to run 100,000 simulations. The general purpose of Monte Carlo is to model the probability of different outcomes and reduce uncertainty. It is very useful for modeling probabilities that come from processes in which random variables intervene with each other making them difficult to predict. The purpose of the paper's Monte Carlo simulations is to devise a new way to calculate match importance which refers to the relationship between match results and the final outcome for a specific season.

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

Replication From Article

Manchester City (Home) vs Manchester United (Away) 4/30/2012

Exploratory Plot of Outcomes Before Match



The exploratory plot above, plot A, shows the outcomes of matches for Manchester City as the home team in the 2011-2012 season. Specifically, it includes all of Man City's home games leading up to the match that they played against Manchester United as the away team on April 30, 2012. The outcomes of the matches are measured as a win (three points), loss (one point) or draw (zero points). Manchester City won the vast majority of those matches and had very few losses. It doesn't appear Man City had any draws. The exploratory plot above, plot B, shows the outcomes of matches for Manchester United as the away team in the 2011-2012 season leading up to the team's match at Manchester City on April 30, 2012. Man United won most of those matches, had some losses and very few draws.

Prediction for Match

Predictions are in terms of the home team
Actual result was a home win (1-0)

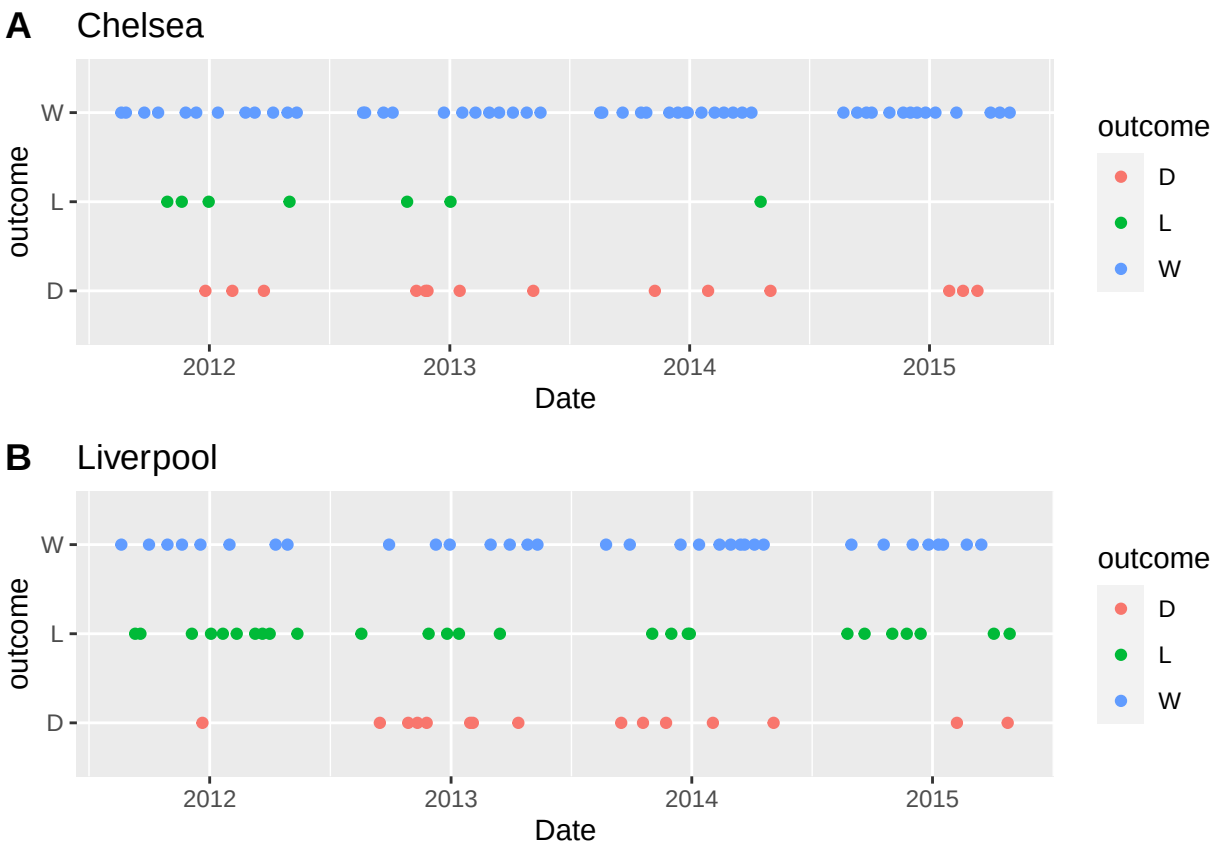
```
##   Match Result Occurances Percent Article Result
## 1           Win      49059  49.059         51.589
## 2          Loss      32671  32.671         22.779
## 3          Draw      18270  18.270         25.632
```

The results of the article differ slightly from the results we found. This is likely due to the randomness of sampling, as well as the increased number of simulations the paper ran.

Further Exploration

Chelsea (Home) vs Liverpool (Away) 5/10/2015

Exploratory Plot of Outcomes Before Match



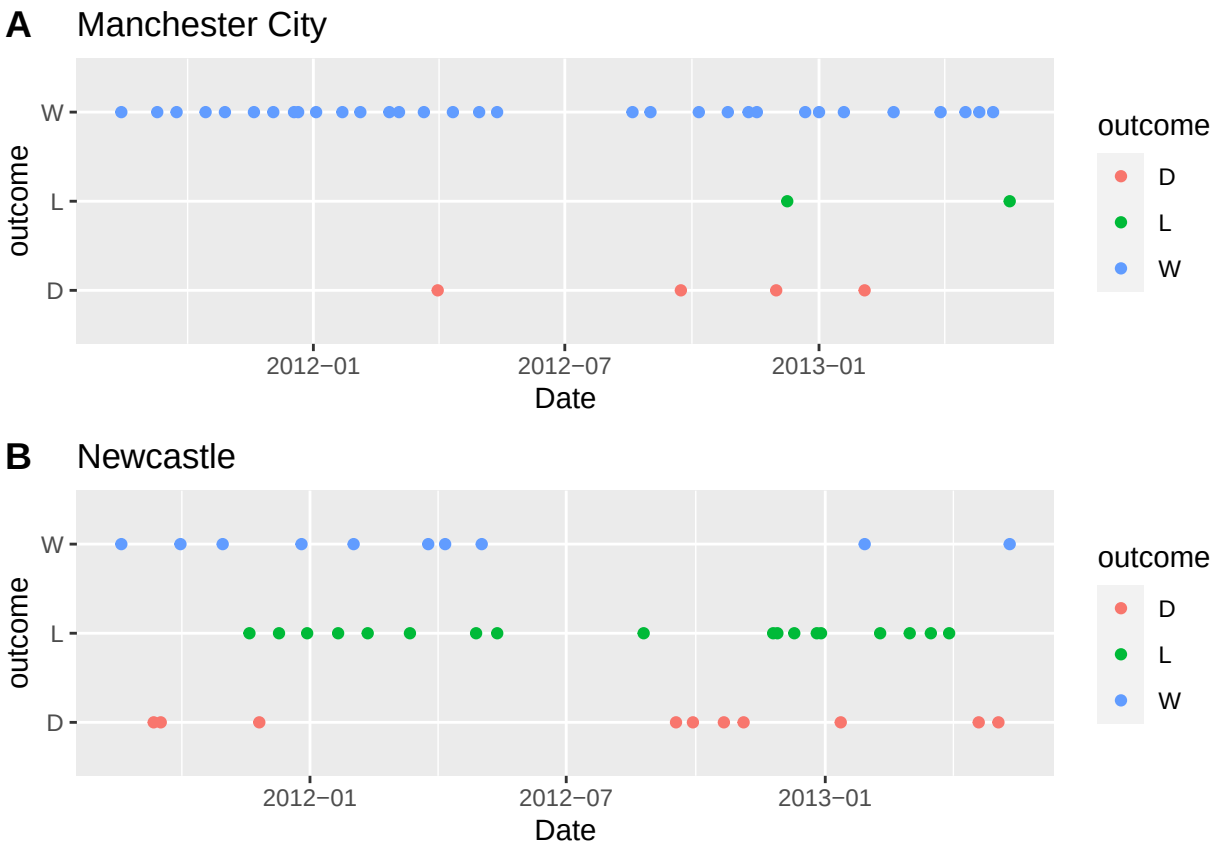
Prediction for Match

Predictions are in terms of the home team
Actual result was a draw

```
##   Match Result Occurances Percent
## 1           Win      51999  51.999
## 2          Loss      29180  29.180
## 3          Draw      18821  18.821
```

Manchester City (Home) vs Newcastle (Away) 8/19/2013

Exploratory Plot of Outcomes Before Match



The exploratory plot above, plot A, shows the outcomes of matches for Manchester City as the home team in the 2013-2014 season. Specifically, it includes all of Man City's home games leading up to the match that they played against Newcastle as the away team on August 19, 2013. Manchester City won the vast majority of those matches and had very few losses and even less draws. The exploratory plot above, plot B, shows the outcomes of matches for Newcastle as the away team in the 2011-2012 season leading up to the team's match at and against Man City on August 19, 2013. Newcastle won most of those matches, had some losses and very few draws.

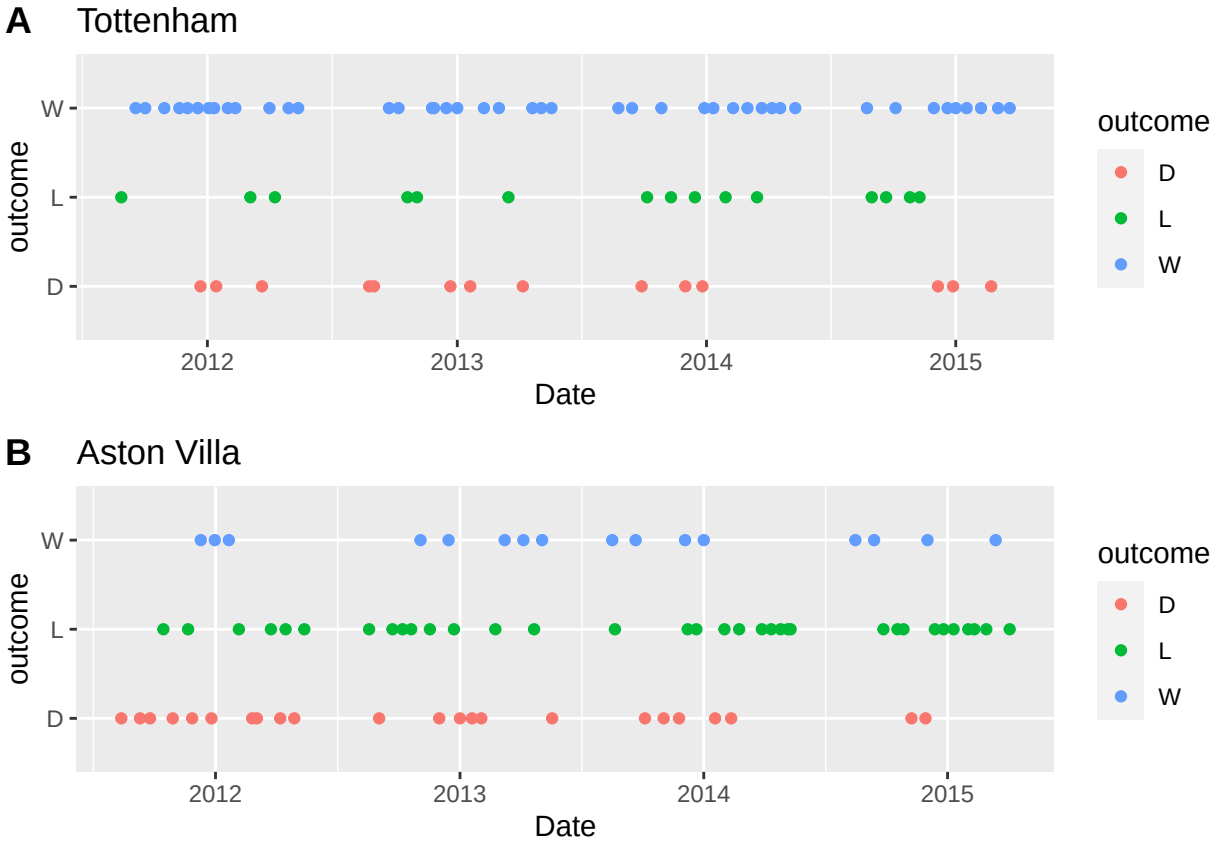
Prediction for Match

Predictions are in terms of the home team
Actual result was a home win

```
##   Match Result Occurances Percent
## 1           Win      60084  60.084
## 2          Loss      23188  23.188
## 3          Draw      16728  16.728
```

Tottenham (Home) vs Aston Villa (Away) 04/11/2015

Exploratory Plot of Outcomes Before Match



The exploratory plot above, plot A, shows the outcomes of matches for Tottenham as the home team in the 2014-2015 season. Specifically, it includes all of Tottenham's home games leading up to the match that they played against Aston Villa as the away team on April 11, 2015. Tottenham won the vast majority of those matches and had very few losses and draws. The exploratory plot above, plot B, shows the outcomes of matches for Aston Villa as the away team in the 2014-2015 season leading up to the team's match at and against Tottenham on April 11, 2015. Aston Villa won some of its matches, but its matches mainly resulted in draws and losses.

Prediction for Match

Predictions are in terms of the home team
Actual result was a home loss

```
##   Match Result Occurances Percent
## 1           Win      53301  53.301
## 2          Loss      28376  28.376
## 3          Draw      18323  18.323
```

Estimating final season ranking

```
teams1415 <- unique(S14_15$HomeTeam)
S14_15$HomePoints <- rep(NA, length(S14_15$Date))
S14_15$AwayPoints <- rep(NA, length(S14_15$Date))
for(i in 1:length(S14_15$HomePoints)){
```

```
  if(S14_15$FTR[i] == "H"){
    S14_15$HomePoints[i] <- 3
    S14_15$AwayPoints[i] <- 0
  }
  if(S14_15$FTR[i] == "A"){
    S14_15$HomePoints[i] <- 0
    S14_15$AwayPoints[i] <- 3
  }
  if(S14_15$FTR[i] == "D"){
    S14_15$HomePoints[i] <- 1
    S14_15$AwayPoints[i] <- 1
  }
}
head(S14_15)
```

```
##       Date   HomeTeam       AwayTeam FTHG FTAG FTR   Id HomePoints AwayPoints
## 1 16/08/14    Arsenal Crystal Palace    2    1   H 1141          3          0
## 2 16/08/14  Leicester        Everton    2    2   D 1142          1          1
## 3 16/08/14 Man United        Swansea    1    2   A 1143          0          3
## 4 16/08/14        QPR           Hull    0    1   A 1144          0          3
## 5 16/08/14      Stoke    Aston Villa    0    1   A 1145          0          3
## 6 16/08/14  West Brom     Sunderland    2    2   D 1146          1          1
```

```
hpts <- S14_15 %>%
  group_by(HomeTeam) %>%
  summarise(HomePoints = sum(HomePoints))

apts <- S14_15 %>%
  group_by(AwayTeam) %>%
  summarise(AwayPoints = sum(AwayPoints))

tpts <- cbind(apts, hpts)

tpts$TotalPoints <- tpts$AwayPoints + tpts$HomePoints
tpts
```

```
##          AwayTeam AwayPoints       HomeTeam HomePoints TotalPoints
## 1         Arsenal         34        Arsenal         41          75
## 2     Aston Villa         17    Aston Villa         21          38
## 3         Burnley         14        Burnley         19          33
## 4         Chelsea         38        Chelsea         49          87
## 5  Crystal Palace         27 Crystal Palace         21          48
## 6         Everton         19        Everton         28          47
## 7            Hull         15           Hull         20          35
## 8       Leicester         15      Leicester         26          41
## 9       Liverpool         27      Liverpool         35          62
## 10       Man City         34       Man City         45          79
## 11     Man United         26     Man United         44          70
## 12      Newcastle         13      Newcastle         26          39
## 13            QPR          7            QPR         23          30
## 14    Southampton         23    Southampton         37          60
## 15          Stoke         21          Stoke         33          54
## 16     Sunderland         18     Sunderland         20          38
## 17        Swansea         24        Swansea         32          56
```

```
## 18      Tottenham         31      Tottenham         33          64
## 19      West Brom         19      West Brom         25          44
## 20       West Ham         16       West Ham         31          47
```

```
tpts <- tpts[order(-tpts$TotalPoints),]
```

```
#match 1501
set.seed(400)
Match1501 <- mc_funct(match_id = 1501, home_team = "Liverpool", away_team = "Crystal Palace")
res1501<- table(t(Match1501$ResultH))
res1501 <- as.data.frame(res1501)
res1501$Percent <- res1501$Freq/100000 * 100
order <- c("Win", "Loss", "Draw")
res1501 <- res1501 %>%
  slice(match(Var1, order))
colnames(res1501) <- c("Match Result", "Occurances", "Percent")
res1501
```

```
##   Match Result Occurances Percent
## 1          Win      45705  45.705
## 2         Loss      31466  31.466
## 3         Draw      22829  22.829
```

```
#match 1520
set.seed(400)
Match1520 <- mc_funct(match_id = 1520, home_team = "Stoke", away_team = "Liverpool")
res1520<- table(t(Match1520$ResultH))
res1520 <- as.data.frame(res1520)
res1520$Percent <- res1520$Freq/100000 * 100
order <- c("Win", "Loss", "Draw")
res1520 <- res1520 %>%
  slice(match(Var1, order))
colnames(res1520) <- c("Match Result", "Occurances", "Percent")
res1520
```

```
##   Match Result Occurances Percent
## 1          Win      44566  44.566
## 2         Loss      36241  36.241
## 3         Draw      19193  19.193
```

```
# win all three
www <- .51999*.45705*.44566

# win, win, lose
wwl <- .51999*.45705*.36241

# win, win, draw
wwd <- .51999*.45705*.19193

# win, lose, win
wlw <- .51999*.31466*.44566

# win, draw, win
wdw <- .51999*.22829*.44566

# win, draw, lose
wdl <- .51999*.22829*.36241
```

```
# win, lose, draw
wld <- .51999*.31466*.19193

# win, draw, draw
wdd <- .51999*.22829*.19193

# win, lose, lose
wll <- .51999*.31466*.36241



# lose, win, win
lww <- .2918*.45705*.44566

# lose, win, lose
lwl <- .2918*.45705*.36241

# lose, win, draw
lwd <- .2918*.45705*.19193

# lose, lose, win
llw <- .2918*.31466*.44566

# lose, draw, win
ldw <- .2918*.22829*.44566

# lose, draw, lose
ldl <- .2918*.22829*.36241

# lose, lose, draw
lld <- .2918*.31466*.19193

# lose, draw, draw
ldd <- .2918*.22829*.19193

# lose, lose, lose
lll <- .2918*.31466*.36241




# draw, win, win
dww <- .18821*.45705*.44566

# draw, win, lose
dwl <- .18821*.45705*.36241

# draw, win, draw
dwd <- .18821*.45705*.19193

# draw, lose, win
dlw <- .18821*.31466*.44566
```

```r
# draw, draw, win
ddw <- .18821*.22829*.44566

# draw, draw, lose
ddl <- .18821*.22829*.36241

# draw, lose, draw
dld <- .18821*.31466*.19193

# draw, draw, draw
ddd <- .18821*.22829*.19193

# draw, lose, lose
dll <- .18821*.31466*.36241

probdf <- data.frame(ddd, ddl, ddw, dld, dll, dlw, dwd, dwl, dww, ldd, ldl, ldw, lld, lll, llw, lwd, lw

probpoints <- c(3, 2, 5, 2, 1, 4, 5, 4, 7, 2, 1, 4, 1, 0, 3, 4, 3, 6, 5, 4, 7, 4, 3, 6, 7, 6, 9)

probdf <- rbind(probdf, probpoints)

r6 <- data.frame(0)
r5 <- data.frame(0)
rt5 <- data.frame(0)
rt4 <- data.frame(0)

for(i in 1:ncol(probdf)){
  if(probdf[2, i] < 3){
    r6 <- rbind(r6, sum(probdf[1, i]))
  }
  if(probdf[2, i] > 3 & probdf[2, i] < 9){
    r5 <- rbind(r5, sum(probdf[1, i]))
  }
  if(probdf[2, i] == 3){
    rt5 <- rbind(rt5, sum(probdf[1, i]))
  }
  if(probdf[2, i] == 9){
    rt4 <- rbind(rt4, sum(probdf[1, i]))
  }
}

r5 <- na.omit(r5)
r6 <- na.omit(r6)
rt4 <- na.omit(rt4)
rt5 <- na.omit(rt5)

resdf <- data.frame("Rank 5" = sum(r5), "Rank 6" = sum(r6), "Rank T4" = sum(rt4), "Rank T5" = sum(rt5))

knitr::kable(resdf, caption = "Ranking Probabilities for Liverpool", "pipe", digits = 5)
```

Table 1: Ranking Probabilities for Liverpool

Rank.5	Rank.6	Rank.T4	Rank.T5
0.60106	0.13623	0.10592	0.1568

Using the Monte Carlo simulation method that we had used for predicting outcomes of a few matches, the next step was to try and estimate a team's final ranking. The team we chose to do this estimate for was Liverpool. When looking at the actual final rankings of the teams, it seemed uncontested that Chelsea would end up ranking first which is why we ended up choosing Liverpool. In order to get the final ranking, we first conducted the Monte Carlo simulation on the Chelsea versus Liverpool match we had picked for the last step. We then ran this simulation two more times on the matches Liverpool had left in their season. Once we had these results we could look into the probabilities of each final ranking for Liverpool. In order to figure out the other team's final rankings, we took the results of all of their home and away games and converted them to points depending on that result, 3 for a win, 1 for a draw, and 0 for a loss. Once converted to points, we were able to add up the total amount of points for each team and sort them to see each teams actual final ranking from the 2014-15 season. Since we knew all of the other team's final rankings, we were able to fit Liverpool into the results where they belonged based on the Monte Carlo results. The table above shows the probability of each possible ranking for Liverpool based on their previous matches in the season and the Monte Carlo simulations. The overall probabilities of all possible scenarios for Liverpool's final three games based on our simulations were calculated. Then based on these probabilities, the total amount of possible points for those last three games was calculated. This total was then added to Liverpool's point total from their previous games in the season. If the point value of the final three games was less than 3, Liverpool would be ranked 6th. If the point value was between 3 and 9, they would be ranked 5th. If the point value was exactly 3, they would end up tying for 5th, which would cause the rank to be determined by some other factor which we do not have the data for. Finally, if the point value was exactly 9, they would end up tying for 4th. The probabilities of each of these scenarios were calculated and the most likely result was that they would end up ranking 5th with a probability of about 68.719%. In the actual results, they ended up placing 6th, which our simulation said would only happen with a probability of 16.95%.

References

"Data Files: England". *Football-Data.co.uk*, 15 December 2021, http://www.football-data.co.uk/englandm.php
Lahvička, Jiří. "Using Monte Carlo Simulation to Calculate Match Importance: The Case of English Premier League." Journal of Sports Economics, vol. 16, no. 4, May 2015, pp. 390–409, doi:10.1177/1527002513490172.