Using Bayesian Modeling to Predict Football Matches after the COVID-19 Pandemic *

Miguel Ángel Sánchez Cortés *SDS II, Sapienza University of Rome*

This project proposes and compares two Poisson Bayesian Hierarchical Models to predict the number of goals scored in football matches using data from the top four European leagues over ten seasons (2011-2021). The models' parameters are estimated using the Markov Chain Monte Carlo (MCMC) method via JAGS. We assess the models' predictive power, simulate match outcomes, and estimate home advantage, particularly analyzing its change post-COVID-19. Finally, the 2019-2020 season's parameters are used to simulate and evaluate the accuracy of predictions for the pandemic-affected 2020-2021 season.

Keywords: bayesian hierarchical models, poisson distribution, football, mcmc, covid-19

Introduction

Statistical modeling of sports data is a widely studied area, with a significant body of research dedicated to it, particularly in the context of football. From a statistical perspective, this field presents intriguing challenges. One key issue is the distribution pattern of the number of goals scored by each team in a single match. Understanding the distributional form of goal counts is crucial for accurately modeling football outcomes. This challenge involves determining the appropriate statistical distributions that best represent the variability and frequency of goals scored by opposing teams during a match. Although the Binomial and Negative Binomial distributions were suggested in the late 1980s (Pollard, 1985), the Poisson distribution has since become widely accepted as a suitable model for the number of goals scored in football matches. A common simplifying assumption in this context is the independence of goals scored by the home and away teams. For example, Maher (1982) utilized a model with two independent Poisson variables, where the parameters are derived from the interaction between one team's attacking strength and the opposing team's defensive weakness.

In this project, we propose and compare two different Bayesian Hierarchical Models for the number of goals scored by two teams in a given match along with a frequentist model based on Poisson regression. We use data from the top four European football leagues (English Premier League, Spanish La Liga, Italian Serie A, and German Bundesliga) for the 10 seasons spanning from 2011 to 2021. The Bayesian models are based on the Poisson distribution, and we estimate their parameters using the Markov Chain Monte Carlo (MCMC) method and the JAGS tool. We use the estimated parameters to predict the number of goals scored by a given team in a match and simulate the outcome of the match to calculate the probability of each team winning, losing, or drawing. Moreover, we use one of these models to estimate the home advantage in a given football match and analyze how this variable changed after the COVID-19 pandemic. Finally, we use

*Replication files are available on the author's Github account (http://github.com/msancor). **Current version**: September 11, 2024; **Corresponding author**: sanchezcortes.2049495@studenti.uniroma1.it.

the estimated parameters for the 2019-2020 season to simulate the outcome of the matches during the 2020-2021 season and compare the results with the actual outcomes to analyze the predictive power of our Poisson model for football seasons affected by the COVID-19 pandemic and by the lack of fans in the stadiums.

The Dataset

The dataset used in this project contains information on the number of goals scored by each team in each match of the top four European football leagues (English Premier League, Spanish La Liga, Italian Serie A, and German Bundesliga) over ten seasons (2011-2021). The dataset was obtained by scraping the WhoScored website, which provides detailed statistics on football matches. An example of the some of the entries within the dataset is shown in Table 1.

Table 1: First 6 entries of the dataset including results from games of the English Premier League in the 2011-2012 season.

League	Season	Home Team	Away Team	Home Goals	Away Goals	Points Home	GD Home
ENG-Premier League	1112	Blackburn Rovers	Wolverhampton	2	3	0	-1
ENG-Premier League	1112	Fulham	Aston Villa	1	0	3	1
ENG-Premier League	1112	Liverpool	Sunderland	2	2	1	0
ENG-Premier League	1112	Newcastle United	Arsenal	0	0	1	0
ENG-Premier League	1112	Queens Park Rangers	Bolton Wanderers	1	4	0	-3
ENG-Premier League	1112	Wigan Athletic	Norwich City	2	2	1	0

The dataset includes in total 14532 matches with 130 different teams. In Table 2 we present the number of matches and teams by league along with some statistics on the average number of goals scored per match and the average number of points obtained by the home and away teams. As we can observe, the number of matches per league during the selected time period is around 3798, with an average of 35 teams per league and an average of 3.7 goals scored per match.

Table 2: Number of matches and teams by league along with some statistics of the dataset.

League	Number of Matches	Number of Teams	Average Goals per Match	Average Points Home	Average Points Away
ENG-Premier League	3798	35	3.735124	1.575829	1.424171
ESP-La Liga	3787	33	3.706892	1.629258	1.370742

League	Number of Matches	Number of Teams	Average Goals per Match	Average Points Home	Average Points Away
GER-Bundesliga	3149	28	3.953954	1.589076	1.410924
ITA-Serie A	3798	34	3.750658	1.597946	1.402054

Moreover, the average number of points obtained by the home team is around 1.6 points per match, whereas the average number of points obtained by the away team is around 1.4 points per match, suggesting that there is indeed a home advantage in football matches as hypothesized by some experts (McGrath, 2020). This also suggests that inferring the home advantage in football matches could potentially be a useful application in order to predict the outcome of football matches.

To motivate the use of the Poisson distribution to model the number of goals scored by each team in a football match, we present in Figure 1 the distribution of the number of goals scored by the home teams for all matches in the dataset alongside with the distribution of random Poisson samples with the same mean as the home goals. As we can observe, the distribution of the number of goals scored by the home teams is well approximated by the Poisson distribution, with a mean of around 2.06 goals per match.

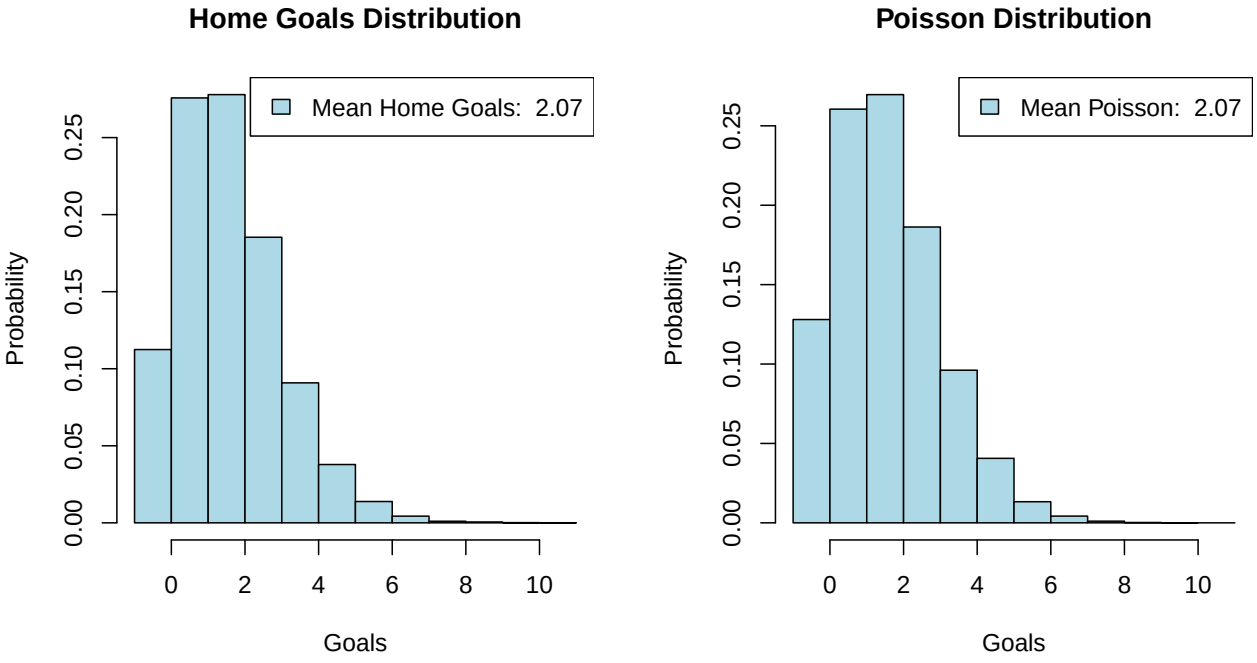

Figure 1: Probability distribution of the number of goals scored by the home teams for all matches in the dataset along with random samples from a Poisson distribution with the same mean.

Analogously, if we plot the distribution of the number of goals scored by the away teams for all matches in the dataset, we observe that this distribution is also well approximated by the Poisson distribution, with a mean of around 1.71 goals per match, which is consistent with the hypothesis that the home team has an advantage in football matches and tends to score more goals

than the away team. In general, the plots above and below suggest that the Poisson distribution is a suitable model for the number of goals scored by each team in a football match, where the mean of the Poisson distribution is the average number of goals scored by a team in a match and could depend on the teams playing, the league, the season, and other factors that could be entered within a model.

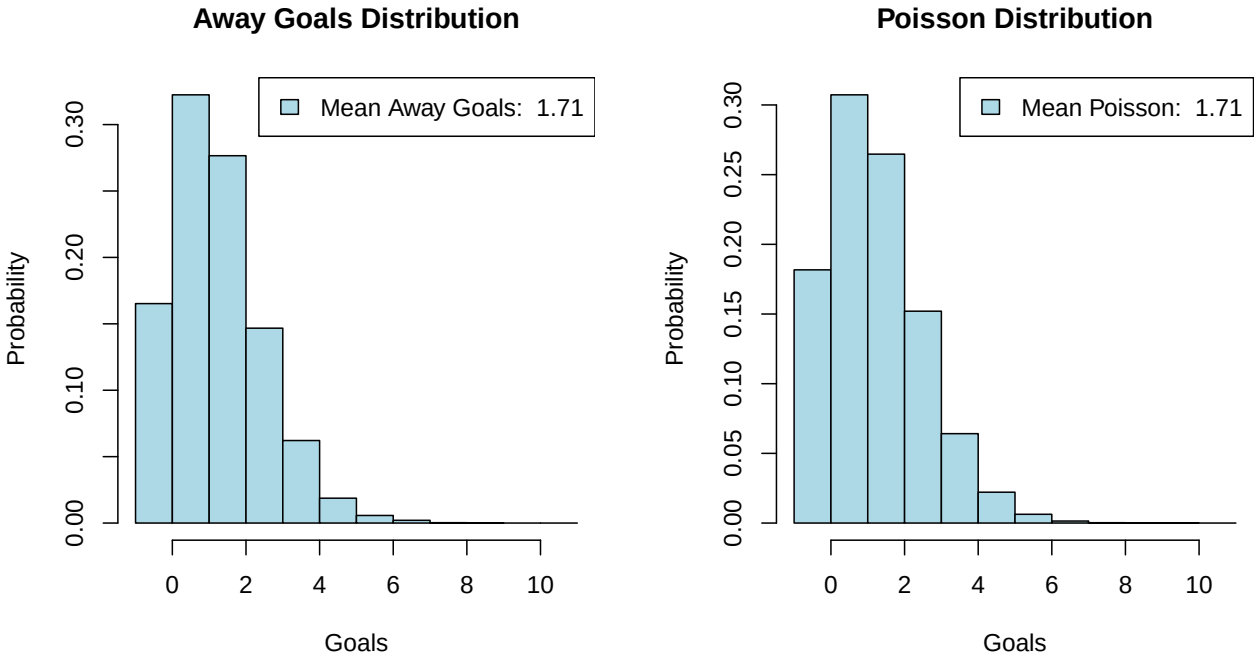

Figure 2: Probability distribution of the number of goals scored by the away teams for all matches in the dataset along with random samples from a Poisson distribution with the same mean.

As showed in the plots above, home advantage exists in football matches and it affects the way we can predict the goals made by a football team in a given match. Given this information, it is interesting to observe what happened to home advantage for the seasons after the COVID-19 pandemic (2020-2021 and 2021-2022) since it could be hypothesized that the absence of fans in the stadiums could have affected the home advantage in football matches and made it less significant. To build on this hypothesis, First, we performed a t-test to see whether the home team's match results changed during the COVID-19 break.

Table 3: T-test results comparing the average points_home and goal_difference_home before and after the COVID-19 pandemic.

Test Statistic	Expected Points	Goal Difference
t	3.6052	3.4652
df	8	8
p-value	0.0069	0.0085
95% CI	[0.0366, 0.1666]	[0.0526, 0.2621]
Mean Before COVID-19	1.6188	0.3897
Mean After COVID-19	1.52	0.2323

Table 3 shows the t-test results performed on the expected points (average points a team is expected to earn on their home field) and the goal difference (goals scored by the home team minus goals scored by the away team), both averaged per league and per season. The test results show that mean values of the expected points and goal difference changed over the COVID-19 break.

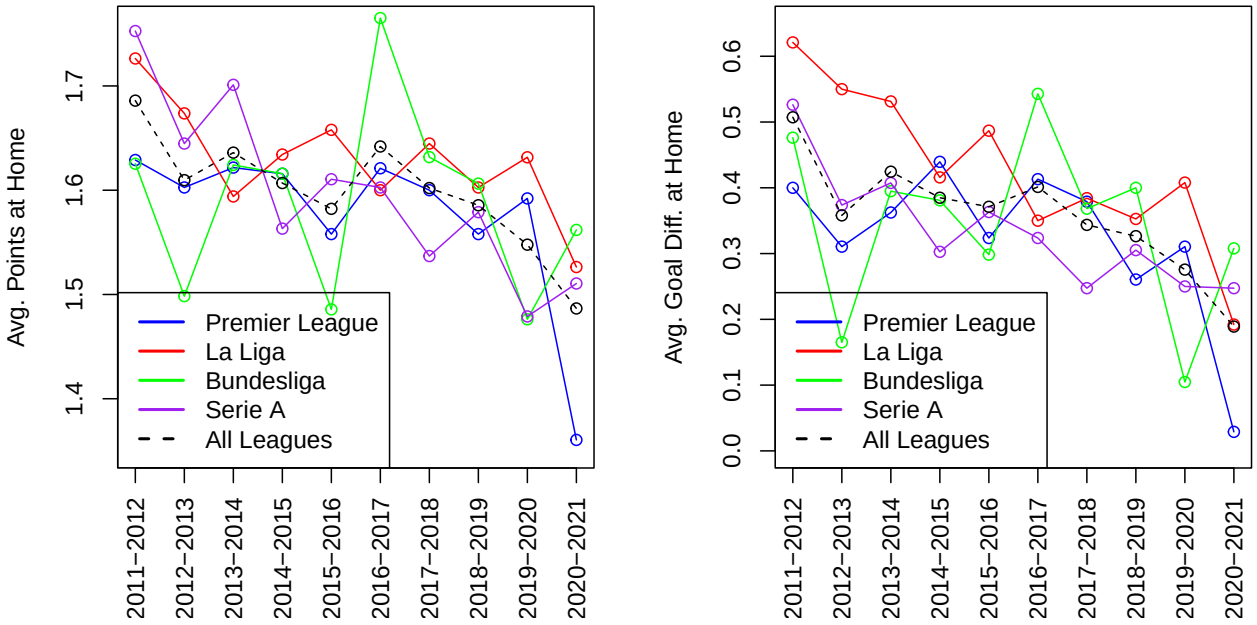

Figure 3: Average points at home per season for all leagues in the dataset along with the average goal difference at home per season.

We can also visualize these quantified measures (expected points and goal difference) to determine how an unattended home match affects the match result of the home team. In Figure 3 we can observe the trends of the expected points and goal difference of the home team for each season in four major European football leagues. We can observe that the expected points and the goal difference of the home team have dropped noticeably since the 2019–2020 season on average, as indicated by the black dashed line. We can argue that the effect of limited spectator attendance is reflected in these two quantified measures in some months after the COVID-19 break.

First Modeling Approach: Poisson Regression

As a first approach to modeling the number of goals scored by each team in a football match, and taking into account the home advantage variable we observed earlier, we propose a Poisson regression model. The Poisson regression model is a type of Generalized Linear Model (GLM)[1] that is used to model count data. In a nutshell, the Poisson regression assumes that a response variable $Y \sim Pois(\lambda)$, and assumes the logarithm of its expected value can be modeled by a linear combination of unknown parameters. Mathematically, given $\mathbf{x}$ as the vector of predictors, the Poisson regression model can be written as:

[1] A GLM generalizes linear regression by allowing the linear model to be related to the response variable via a link function and by allowing the magnitude of the variance of each measurement to be a function of its predicted value.

$$\log(E(Y|X)) = \theta\mathbf{x}, \tag{1}$$

where θ is the vector of unknown parameters to be estimated. Therefore, given a Poisson regression model θ and an input vector $\mathbf{x}$, the predicted mean of the associated Poisson distribution is given by:

$$E(Y|X) = e^{\theta\mathbf{x}}. \tag{2}$$

Now, considering we have Y_i independent observations of the response variable Y and the corresponding input vectors $\mathbf{x}_i$, we can estimate the parameters θ by the principle of maximum likelihood. To do this, we need to find the set of parameters θ that maximize the likelihood function of the data given by:

$$p(y_1, \ldots, y_n|x_1, \ldots, x_n; \theta) = L(\theta) = \prod_{i=1}^{n} \frac{y_i e^{\theta x_i} e^{-e^{\theta x_i}}}{y_i!}. \tag{3}$$

In R this is very simple to do using the glm function. As an example, we can fit a Poisson regression model to the number of goals scored by each team in the Serie A during the 2011-2021 period. To do this, we can first create a data frame with the predictors of our model: the home team, the away team, the number of goals (by the home or away team), and the home advantage variable, that takes as value 1 if the goals are made by the home team and 0 otherwise. In Table 4 we show the first rows of the data frame we created for this purpose.

Table 4: First rows of the data frame used to fit the Serie A Poisson regression model.

Home	Away	At Home	Goals
Milan	Lazio	1	3
Cesena	Napoli	1	2
Catania	ACR Siena 1904	1	0
ChievoVerona	Novara	1	3
Fiorentina	Bologna	1	2
Genoa	Atalanta	1	3

Then, we can fit our model using the glm function in R. The code below shows how to fit a Poisson regression model to the number of goals scored by each team in the Serie A during the 2011-2021 period:

```
pois_model <- glm(goals ~ home + team + opponent, family=poisson(link=log),
                  data=df_seriea)
```

In Table 5 we show the significant coefficients of the Poisson regression model fitted to the Serie A data. We can observe that the home variable has a significant effect on the number of goals scored by a team, as indicated by its p-value, and therefore adding more confidence to our hypothesis that a team playing at home are more likely to score goals. It is also interesting to see that Juventus had the highest and more significant coefficient both as a home team and as an away

team, indicating that Juventus consistently scored more goals than the other teams in the Serie A during the 2011-2021 period, confirmed by the fact that this team won the Serie A title in 9 out of the 10 seasons analyzed. Other teams considered as part of the "Big Teams" in the Serie A, such as Inter Milan, AC Milan, and AS Roma, also had significant coefficients, indicating that these teams also scored more goals than the other teams in the Serie A during this period. On the other side, teams like Crotone, Benevento, and Frosinone had also significant coefficients playing as an opponent, indicating that these teams conceded more goals than the other teams in the Serie A during this period. This is consistent with the fact that these teams were relegated to the Serie B in the seasons they played in the Serie A.

Table 5: Significant Results of the Poisson regression model fitted to the Serie A data.

	Estimate	Std. Error	Z-Value	P-Value
(Intercept)	0.4344540	0.1230875	3.529637	0.0004161
home	0.1790107	0.0168242	10.640057	0.0000000
teamAtalanta	0.2000130	0.0962737	2.077546	0.0377512
teamInter	0.2849382	0.0956959	2.977539	0.0029057
teamJuventus	0.3723597	0.0951246	3.914442	0.0000906
teamLazio	0.2634448	0.0958686	2.747978	0.0059964
teamMilan	0.2151863	0.0961365	2.238341	0.0251988
teamNapoli	0.3657100	0.0952079	3.841174	0.0001224
teamRoma	0.3406349	0.0953742	3.571563	0.0003549
opponentBenevento	0.3113252	0.1096333	2.839697	0.0045156
opponentBrescia	0.3015168	0.1305987	2.308728	0.0209587
opponentCrotone	0.2566472	0.1028132	2.496246	0.0125515
opponentFrosinone	0.2197162	0.1120245	1.961322	0.0498414
opponentJuventus	-0.4646195	0.0962864	-4.825389	0.0000014
opponentNapoli	-0.1959643	0.0934789	-2.096349	0.0360512
opponentPescara	0.3160480	0.1096779	2.881601	0.0039566

In Appendix A we can observe the results given for the Poisson regression model fitted to data from other leagues such as the Premier League, La Liga, and Bundesliga. We can see that the home variable has a significant effect on the number of goals scored by a team in all leagues, consistent with the home advantage hypothesis. Moreover, we observe similar trends than in the Serie A, with "big" teams such as Manchester City, Barcelona and Bayern Munich having significant coefficients both as a home and away team, indicating that these teams scored more goals than the other teams in their respective leagues.

Using this model, we can also predict the number of goals scored by each team in a football match. To do this, we can use the predict function in R. As an example, we can try to predict the result of the roman derby between AS Roma and Lazio for the 2021-2022 season. To do this, we can use the code below:

```
roma<-predict(pois_model,
              newdata=data.frame(home=1, team="Roma",opponent="Lazio"),
              type="response")
lazio<-predict(pois_model,
               newdata=data.frame(home=0, team="Lazio",opponent="Roma"),
               type="response")
print(paste("AS Roma - Lazio: ", round(roma), " - ", round(lazio)))
```

```
## [1] "AS Roma - Lazio:  2  -  2"
```

As we can observe, we predicted a 2-2 draw between AS Roma and Lazio in the roman derby for the 2021-2022 season. This is consistent with the fact that the roman derby is one of the most balanced matches in the Serie A, with both teams having a similar number of wins and draws in the last 10 years. Surprisingly, the real result was a staggering 3-0 win for AS Roma. This suggests that our model clearly doesn't capture all the information needed to predict the result of a football match, and that considering other factors may improve the accuracy of our predictions. Moreover, predictions given by the model are deterministic, since same input will always give the same output. This is not ideal for a model that aims to predict the result of a football match, since the outcome of a football match is inherently uncertain. In the next section we propose a Bayesian Hierarchical Model to model the number of goals scored by each team in a football match, that will allow us to capture this uncertainty and make probabilistic predictions by sampling from the posterior distribution of the model.

Second Modeling Approach: Bayesian Hierarchical Model

As a second approach to modeling the number of goals scored by each team in a football match, we propose a Bayesian Hierarchical Model that assumes that the number of goals scored by each team in a match follows a Poisson distribution and that the distribution of goals for home and away teams are not the same, since we observe this in the data. The model is called hierarchical because it has multiple levels of parameters that are estimated using Bayesian statistics. Mathematically, we write that the goal outcome of a match between team i and team j is modeled as:

$$\text{GOAL}_{\text{home}} \sim Pois(\lambda_{\text{home},i,j}), \tag{4}$$
$$\text{GOAL}_{\text{away}} \sim Pois(\lambda_{\text{away},i,j}). \tag{5}$$

We also assume that the mean of the Poisson distribution depends on the skills of the teams playing, since not all teams are equally good. At the same time, the mean can also depend on external factors for teams, such as the stadium, weather, etc. Therefore, we model the mean of the Poisson distribution as:

$$\log(\lambda_{\text{home},i,j}) = \text{OTHERS} + \text{SKILL}_i - \text{SKILL}_j, \tag{6}$$
$$\log(\lambda_{\text{away},i,j}) = \text{OTHERS} - \text{SKILL}_i + \text{SKILL}_j. \tag{7}$$

As the number of goals are assumed to be Poisson distributed it is natural that the skills of the teams are on the log scale of the mean of the distribution, since in this way we can linearize

multiplicative relationships between the teams' abilities and the expected number of goals. After defining the model, we need to specify the prior distributions for the parameters of the model in order to be able to make inferences about the skills of the teams and the means of the Poisson distributions. We propose very weakly informative priors for the parameters of the model, in order to avoid biasing the results of the model:

$$\text{OTHERS} \sim N(0, 4^2), \tag{8}$$

$$\text{SKILL}_i \sim N(\mu_{\text{teams}}, \sigma^2_{\text{teams}}), \tag{9}$$

$$\mu_{\text{teams}} \sim N(0, 4^2), \tag{10}$$

$$\sigma_{\text{teams}} \sim U(0, 3). \tag{11}$$

We can observe that the prior on the OTHERS parameter has a standard deviation of 4, and since this is on the log scale of the mean number of goals, it implies that the prior on the mean number of goals is a log-normal distribution with a mean of 1 and a standard deviation of 4. This is a very weakly informative prior, since it implies that the mean number of goals is likely to be between 0.018 and 54.6 goals, which is a very wide range. Given that the average number of goals per match by a team is fewer than two in European football leagues, we affirm that this prior is weakly informative and doesn't introduce prior knowledge to our model.

Now that we defined the model and the prior distributions, we can estimate the parameters of the model using Bayesian statistics. Specifically, we use the JAGS software to estimate the parameters of the model by performing a Markov Chain Monte Carlo (MCMC) simulation. The MCMC simulation is a method that allows us to sample from the posterior distribution of the parameters of the model, which is the distribution of the parameters of the model given the data. We can write the above model using JAGS syntax as follows:

```
model1 <- "model {
  for (i in 1:n_games) {
    home_goals[i]~dpois(lambda_home[home_team[i], away_team[i]])
    away_goals[i]~dpois(lambda_away[home_team[i], away_team[i]])
  }

  for (home_i in 1:n_teams) {
    for (away_j in 1:n_teams) {
      lambda_home[home_i, away_j]<-exp(others+skill[home_i]-skill[away_j])
      lambda_away[home_i, away_j]<-exp(others+skill[away_j]-skill[home_i])
    }
  }

  skill[1] <- 0
  for (j in 2:n_teams) {
    skill[j]~dnorm(mu_team, tau_team)
  }

  mu_team~dnorm(0, 0.0625)
  tau_team<-1/pow(sigma_team,2)
  sigma_team~dunif(0, 3)
  others~dnorm(0, 0.0625)
}"
```

It is important to notice that the model is written in JAGS syntax, which is a language that allows us to specify the model and the prior distributions in a way that JAGS can understand. In this model, the normal distribution is written as $N(\mu, \tau)$, where μ is the mean of the distribution and τ is the precision of the distribution, which is the inverse of the variance. At the same time, we can notice that we previously defined the skill of the first team as 0, which is a common practice in Bayesian statistics to avoid identifiability issues in the model. This is because the skills of the teams are relative to each other, and if we don't fix the skill of one team, the model will not be able to estimate the skills of the other teams.

After defining the model above, we can use the R2jags package to run the MCMC simulation and estimate the parameters of the model. From this moment onward, we will exclusively work with the Serie A dataset (unless it is specified otherwise) for computational reasons. We can write the code to setup and run the MCMC simulation as follows:

```
run_model1 <- function(){
  #Here we compile the first model
  m1 <- jags.model(textConnection(model1),
      data = data_seriea, n.chains = 3, n.adapt = 5000, quiet = TRUE)
  #Here we burn the first 5000 iterations
  update(m1, 5000)
  #Here we generate MCMC samples
  s1 <- coda.samples(m1,
      variable.names = c("others", "skill","mu_team","sigma_team"),
      n.iter = 10000, thin = 2)
  #Here we obtain the DIC
  dic1 <-dic.samples(m1, n.iter = 10000, thin = 2)
  return(list(samples = s1, dic = dic1))
}
```

We can make a few comments about the code above. First, to compile our model, we specified some important parameters to ensure convergence of the MCMC simulation. Specifically, we specified the number of chains as three, which is a common practice to ensure that the chains are exploring the parameter space correctly. We also specified the number of adaptation steps as 5000, which is the number of iterations that JAGS will use to adapt the proposal distribution of the MCMC algorithm. This is important to ensure that the MCMC algorithm is sampling from the posterior distribution correctly. After compiling the model, we burned the first 5000 iterations of the MCMC simulation to ensure that the chains are sampling from the stationary distribution of the model. Finally, we generated 10000 MCMC samples by using a thinning of 2, which means that we are keeping every second sample of the MCMC chain to reduce autocorrelation in the samples.

Once we have the MCMC samples, it is also good practice to check the convergence of the MCMC simulation. We can do this by checking the trace plots of the MCMC samples, which are plots that show the evolution of the MCMC samples over the iterations of the simulation. For example, in Figure 4, we show the trace plot of the OTHERS variable the teams in the Serie A dataset. We can see that the trace plot is stationary, which is a good indication that the MCMC simulation has converged. At the same time, we can observe the plot for the posterior distribution of the

OTHERS variable, as we can observe, the values are less spread than the prior distribution, which had a normal distribution with mean 0 and variance 0.0625.

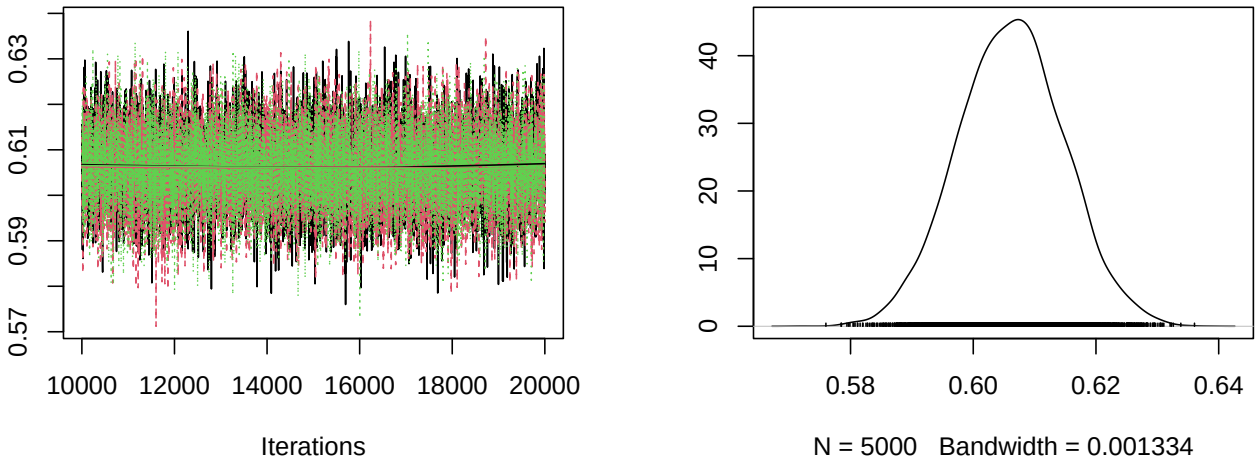

Figure 4: Trace plot of the OTHERS variable of the teams in the Serie A dataset.

We can also observe the Gelman-Rubin statistic (i.e. shrink factor) to check the convergence of the MCMC simulation. The Gelman-Rubin statistic is a measure of the convergence of the MCMC chains, and it is calculated as the ratio of the between-chain variance to the within-chain variance. A value of 1 indicates that the chains have converged, while a value greater than 1 indicates that the chains have not converged. In Figure 5, we can see that the Gelman-Rubin statistic is close to 1 for the OTHERS parameter of the model, which is a good indication that the MCMC simulation has converged.

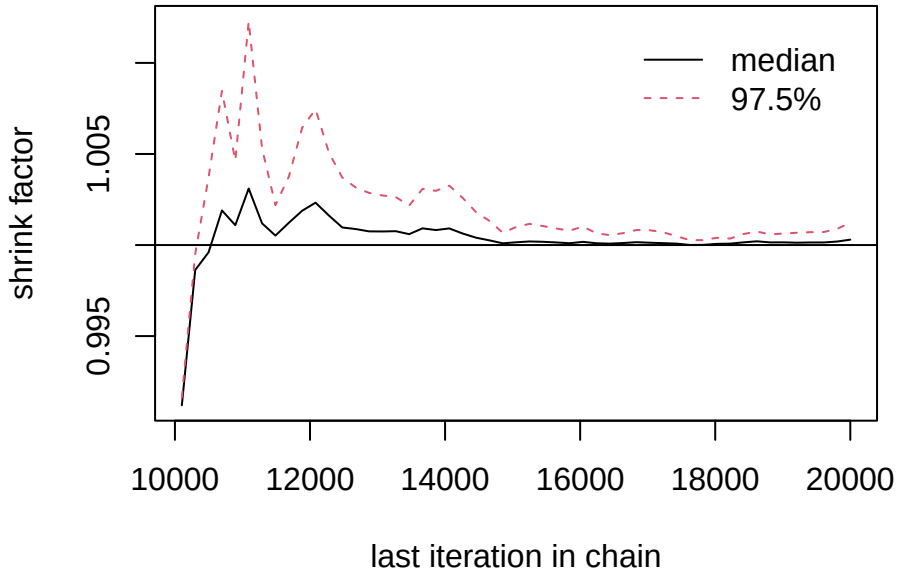

Figure 5: Gelman-Rubin statistic for the OTHERS variable of the teams in the Serie A dataset.

Finally, we can also check the autocorrelation of the MCMC samples to ensure that the samples are independent. Since we are performing a MCMC simulation, samples are not independent

given that the next sample is dependent on the previous one. However, if our simulation is working correctly, the samples should be approximately independent. In Figure 6, we show the autocorrelation plot of the OTHERS variable of the teams in the Serie A dataset. We can see that the autocorrelation is close to zero for all lags, which is a good indication that the MCMC samples are approximately independent.

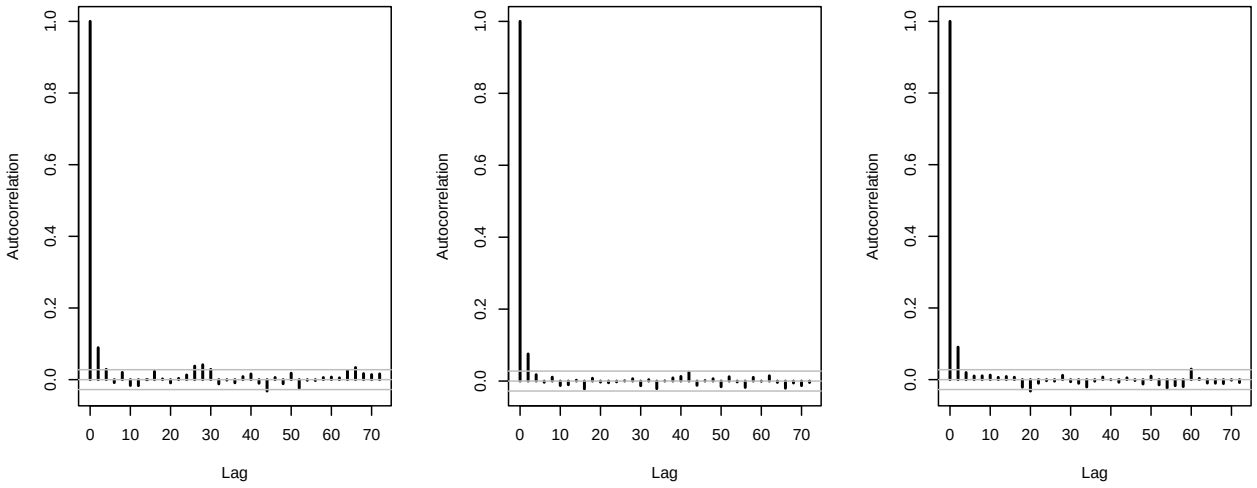

Figure 6: Autocorrelation plot of the OTHERS variable of the teams in the Serie A dataset.

As we could see, the MCMC simulation has converged, and the samples are approximately independent (this is also true for the other variables of the model which we will not show here). Therefore, we can use the MCMC samples to make inferences about the parameters of the model. For example, in Figure 7, we can obtain a summary for the SKILL parameter for AS Roma and Lazio respectively.

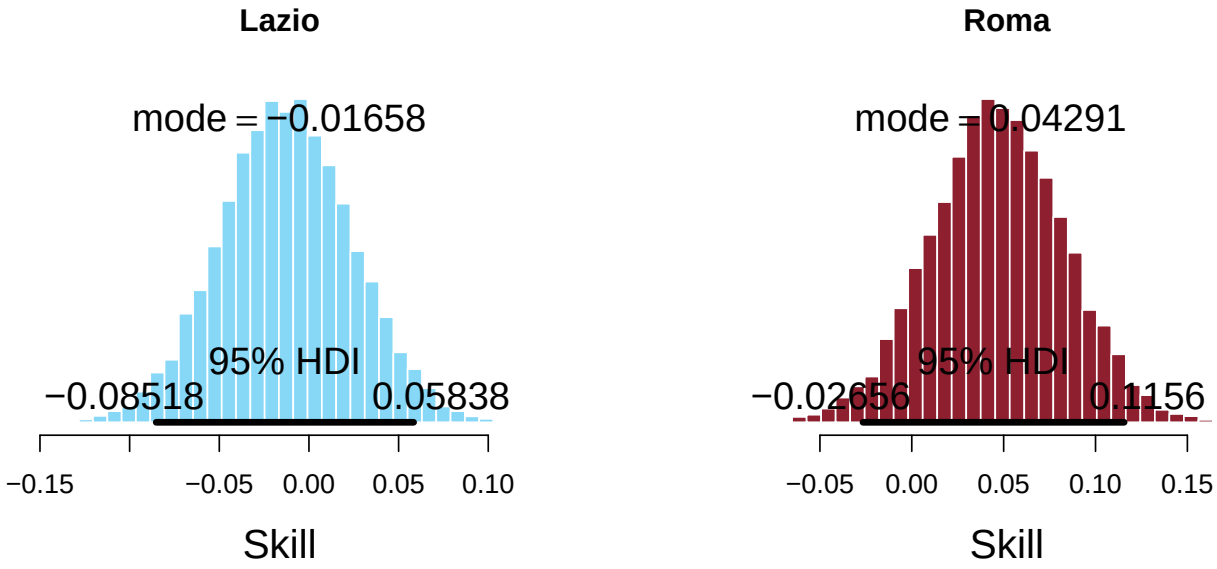

Figure 7: Summary of the skill parameter for AS Roma and Lazio.

As we can observe, the posterior distribution for the skill parameter for AS Roma has a mode of 0.042 with a HDI of [-0.026, 0.115], whereas for Lazio it takes a mode of -0.016 with a HDI of [-0.085, 0.05]. This means that AS Roma has a consistently higher skill parameter than Lazio, which could suggest that AS Roma is a better team than Lazio (at least according to the model). It is important to notice that this parameter is relative with respect to the other teams in the league, in particular, we set the 0 value to be the skill of AC Milan. Therefore, a positive value indicates a better team than AC Milan, while a negative value indicates a worse team than AC Milan.

Using the MCMC samples it is not only possible to look at the distribution of parameter values but it is also straightforward to simulate matches between teams and obtain the distribution of goals scored and the probability of a win for the home team, a win for the away team or a draw. In Figure 8, we show the simulated results for a match between AS Roma and Lazio and the distribution of the real results.

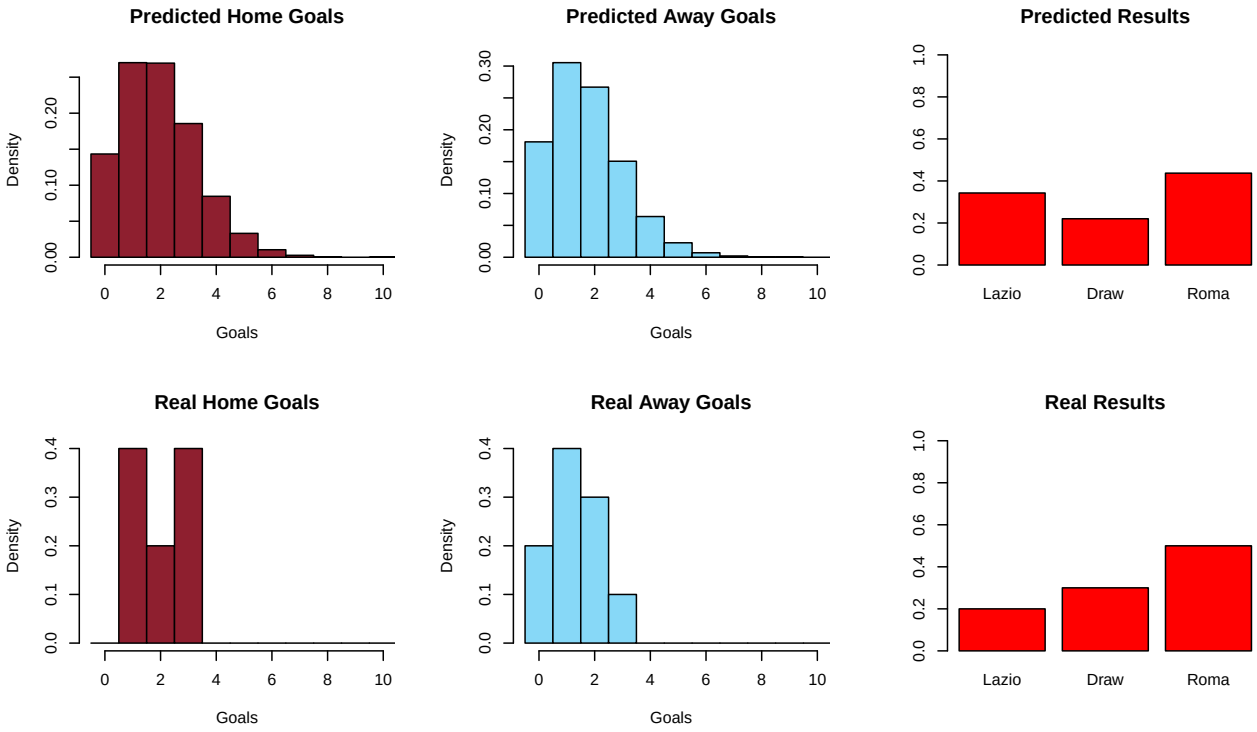

Figure 8: Histograms of the predicted goals and results for the match between AS Roma and Lazio.

As we can observe, the simulated data fits the historical data reasonably well and both the historical data and the simulation associate a higher probability of a win for AS Roma than for Lazio. Even though we can observe that the difference is not very high. It is of interest also to analyze the inverse match, that is, the match between Lazio and AS Roma. In Figure 9, we show the simulated results for this match and the distribution of the real results. In this case we can visualize in a better way a problem with our current model. While the simulated data looks the same, except that the home team and the away team swapped places, the historical data now shows that Lazio often wins against Roma when being the home team. We couldn't predict this pattern since

our model doesn't incorporate the advantage of being the home team. This is why we propose a second model that includes this advantage.

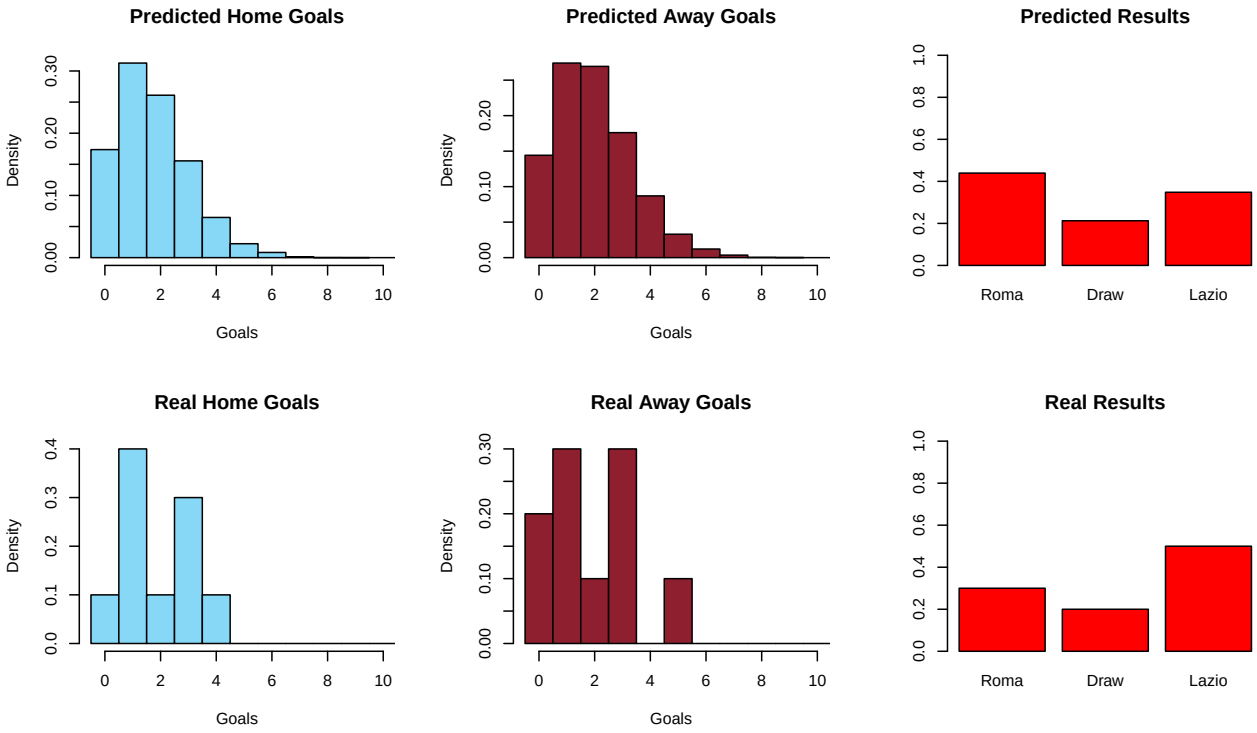

Figure 9: Histograms of the predicted goals and results for the match between Lazio and AS Roma.

Third Modeling Approach: Bayesian Hierarchical Model with Home Advantage

In our past model we didn't considered two important facts that we observe or hypothesize from football data. The first, is that home advantage exists and it is important when predicting a match. The second, is that the performance of a team is not constant over time, but it varies by different factors like new players, injuries, etc. To account for these two facts, we propose a new Bayesian Hierarchical Model that incorporates these two factors. This model was proposed in a previous study by Lee et al. (2022) and its very similar to the previous model, but incoporating the home advantage and season effects. Mathematically, for this model we write that the goal outcome of a match between team i and team j in season s is modeled as:

$$\text{GOAL}_{\text{home}} \sim Pois(\lambda_{\text{home},i,j,s}), \tag{12}$$

$$\text{GOAL}_{\text{away}} \sim Pois(\lambda_{\text{away},i,j,s}). \tag{13}$$

We also assume that the mean of the Poisson distribution depends on the skills of the teams playing, since not all teams are equally good in different seasons. At the same time, the mean can also depend on external factors for the home or away teams, such as the stadium, weather, etc.

Therefore, we model the mean of the Poisson distribution as:

$$\log(\lambda_{\text{home},i,j,s}) = \text{OTHERS}_{\text{home},s} + \text{SKILL}_{i,s} - \text{SKILL}_{j,s}, \tag{14}$$

$$\log(\lambda_{\text{away},i,j,s}) = \text{OTHERS}_{\text{away},s} - \text{SKILL}_{i,s} + \text{SKILL}_{j,s}. \tag{15}$$

Now, th parameter for team performance, $\text{SKILL}_{i,s}$, denotes the team performance of team i during season s. It is reasonable to assume that the team performance in each match is defined as a sample from the team performance distribution of the last season, since the team performance is not expected to change drastically from one season to another. Therefore, we model the team performance as:

$$\text{SKILL}_{i,s} \sim N(\text{SKILL}_{i,s-1}, \sigma^2_{\text{seasons}}). \tag{16}$$

After defining the model, we need to specify the prior distributions for the parameters of the model in order to be able to make inferences about the skills of the teams and the means of the Poisson distributions. In this model we also propose very weakly informative priors for the parameters of the model, in order to avoid biasing the results of the model:

$$\text{OTHERS} \sim N(0, 4^2), \tag{17}$$

$$\text{SKILL}_{i,\text{1st season}} \sim N(\mu_{\text{teams}}, \sigma^2_{\text{teams}}), \tag{18}$$

$$\mu_{\text{teams}} \sim N(0, 4^2), \tag{19}$$

$$\sigma_{\text{teams}} \sim U(0, 3), \tag{20}$$

$$\sigma_{\text{seasons}} \sim U(0, 3). \tag{21}$$

Now that we defined the model and the prior distributions, we can estimate the parameters of the model using Bayesian statistics. We can write the above model using JAGS syntax as follows:

```
model2 <- "model {
  for (i in 1:n_games) {
    home_goals[i]~dpois(lambda_home[season[i],home_team[i], away_team[i]])
    away_goals[i]~dpois(lambda_away[season[i],home_team[i], away_team[i]])
  }

  for (season_i in 1:n_seasons){
    for (home_i in 1:n_teams) {
      for (away_j in 1:n_teams) {
        lambda_home[season_i, home_i, away_j]<-exp(others_home[season_i]
        +skill[season_i,home_i]-skill[season_i,away_j])
        lambda_away[season_i,home_i, away_j]<-exp(others_away[season_i]
        +skill[season_i,away_j]-skill[season_i,home_i])
      }
    }
  }

  skill[1,1] <- 0
  for (j in 2:n_teams) {
    skill[1,j]~dnorm(mu_team, tau_team)
  }

  mu_team~dnorm(0, 0.0625)
```

```
  tau_team<-1/pow(sigma_team,2)
  sigma_team~dunif(0, 3)

  others_home[1]~dnorm(0, 0.0625)
  others_away[1]~dnorm(0, 0.0625)

  for (season_i in 2:n_seasons) {
    skill[season_i,1]<-0
    for (j in 2:n_teams) {
      skill[season_i,j]~dnorm(skill[season_i-1,j], tau_season)
    }
    others_home[season_i]~dnorm(others_home[season_i-1], tau_season)
    others_away[season_i]~dnorm(others_away[season_i-1], tau_season)
  }

  tau_season<-1/pow(sigma_season,2)
  sigma_season~dunif(0, 3)
}"
```

After defining the model above, we can use the R2jags package to run the MCMC simulation andestimate the parameters of the model. We can write the code to setup and run the MCMC simulation as follows:

```
run_model2 <- function(){
  #Here we compile the first model
  m2 <- jags.model(textConnection(model2),
      data = data_seriea, n.chains = 3, n.adapt = 10000, quiet = TRUE)
  #Here we burn the first 5000 iterations
  update(m2, 10000)
  #Here we generate MCMC samples
  s2 <- coda.samples(m2,
      variable.names = c("others_home", "others_away", "skill",
                         "sigma_season","sigma_team", "mu_team"),
      n.iter = 40000, thin = 8)
  #Here we obtain the dic
  dic2 <- dic.samples(m2, n.iter = 40000, thin = 8)
  return(list(samples = s2, dic = dic2))
}
```

One important observation is that given the changes in the model above, we had a lot of autocorrelation when sampling from the posterior distribution. This is a problem because it means that the samples are not independent and thus the MCMC sampler is not working properly. To solve this problem, we increased the number of samples and the amount of thinning, we also increased the number of iterations in the burn-in phase. This is not ideal, but it is a common problem when working with hierarchical models and it is a trade-off between computational time and accuracy.

Now that we have the samples, we can start analyzing the results. As a first observation, we can compare the DIC (Deviance Information Criterion) of the two Bayesian models proposed. The DIC is a measure of the goodness of fit of a model that takes into account the complexity of the model. The DIC is calculated as follows:

$$DIC = D(\bar{\theta}) + 2p_D, \tag{22}$$

where $D(\bar{\theta})$ is the deviance of the posterior mean of the parameters and p_D is the effective number of parameters of the model. The larger the effective number of parameters is, the easier it is for the model to fit the data, and so the deviance needs to be penalized, this is why the lower the DIC, the better the model. However, the DIC is not an absolute measure, it is only useful for comparing models. We can calculate the DIC for the two models as follows:

```
## DIC Model 1 DIC Model 2
##    24290.12    24045.27
```

As we can observe, the DIC of the second model is lower than the DIC of the first model, which means that the second model is a better fit for the data. This is expected since the second model is more complex and can fit the data better. At the same time, the second model incorporates crucial information like seasonality and home advantage, which are important factors in soccer matches.

Now we can analyze the results of the second model. Once we have the MCMC samples, it is also good practice to check the convergence of the MCMC simulation. We can do this by checking the trace plots of the MCMC samples, which are plots that show the evolution of the MCMC samples over the iterations of the simulation. For example,in Figure 10, we show the trace plot of the σ_{seasons} variable the teams in the Serie A dataset.

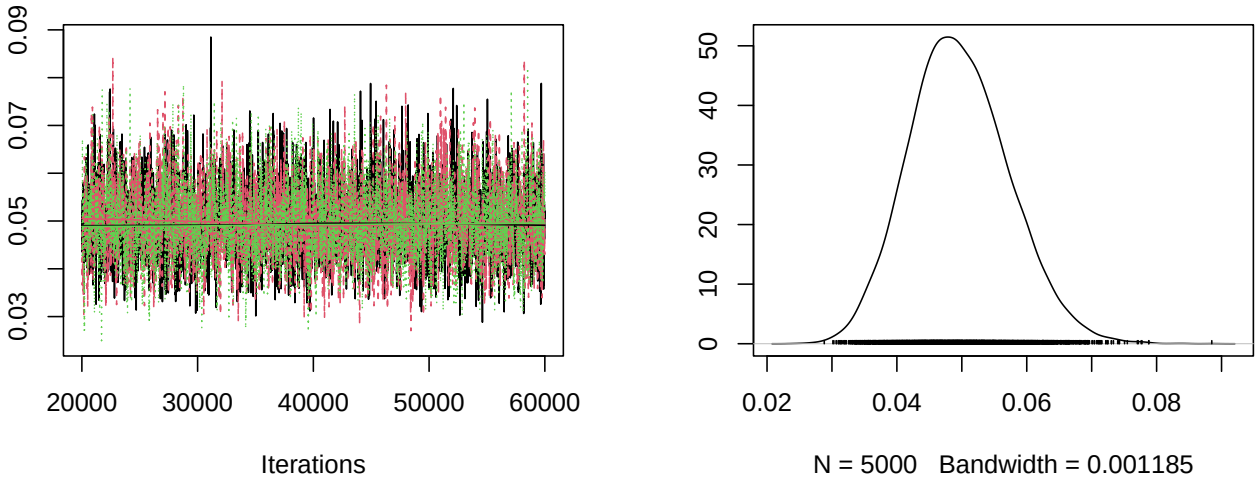

Figure 10: Trace plot of the sigma_season variable of the teams in the Serie A dataset.

We can see that the trace plot is stationary, which is a good indication that the MCMC simulation has converged. At the same time, we can observe that the posterior distribution has a peak around 0.6 in comparison with the prior distribution, which was uniform. This means that the model has learned that the variance of the season effect on skill is around 0.6, which is a good result. We can also observe the Gelman-Rubin statistic (i.e. shrink factor) to check the convergence of the MCMC simulation. In Figure 11, we can see that the Gelman-Rubin statistic is close to 1 for the σ_{seasons} parameter of the model, which is a good indication that the MCMC simulation has converged. We can also check the trace plot and the Gelman-Rubin statistic for the other parameters of the model, but we will not show them here for brevity.

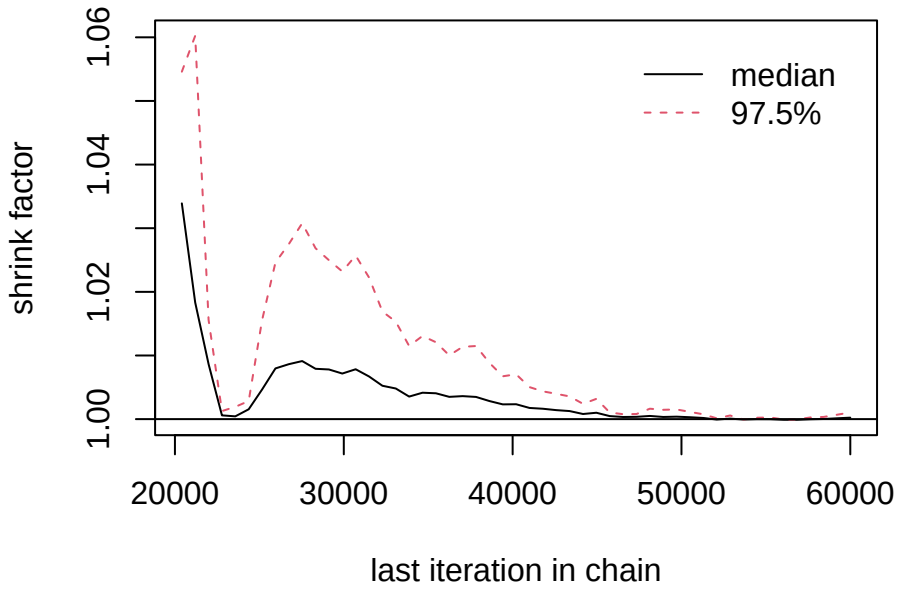

Figure 11: Gelman-Rubin statistic for the sigma_season variable of the teams in the Serie A dataset.

Finally, in Figure 12, we show the autocorrelation plot of the σ_{seasons} variable of the teams in the Serie A dataset. As we can observe, this case is different than in the first model, with the autocorrelation being much higher at the beginning and taking longer to decay. This means that our hierarchical model is capturing more persistent season-to-season variability among the teams, indicating that the variance between teams' performances remains correlated over time. The slower decay suggests that the model accounts for more complex dependencies across seasons, aligning with the hierarchical structure's ability to model longer-term trends. Although the autocorrelation is still high, it is within acceptable limits, and the model is still valid.

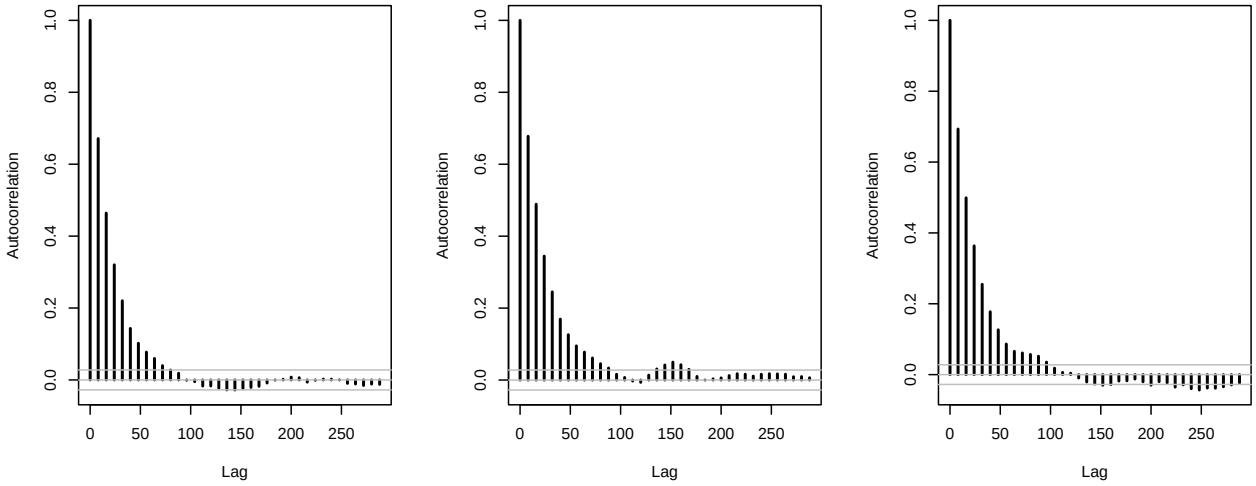

Figure 12: Autocorrelation plot of the sigma_season variable of the teams in the Serie A dataset.

As we can see, the MCMC simulation has converged, and the samples are approximately independent (not really, but close enough). Now. we can start making inferences with our model.

For example, now, in Figure 13, we can obtain a summary for the SKILL parameter for AS Roma and Lazio respectively for a particular season, in this case we chose the 2020-2021 season.

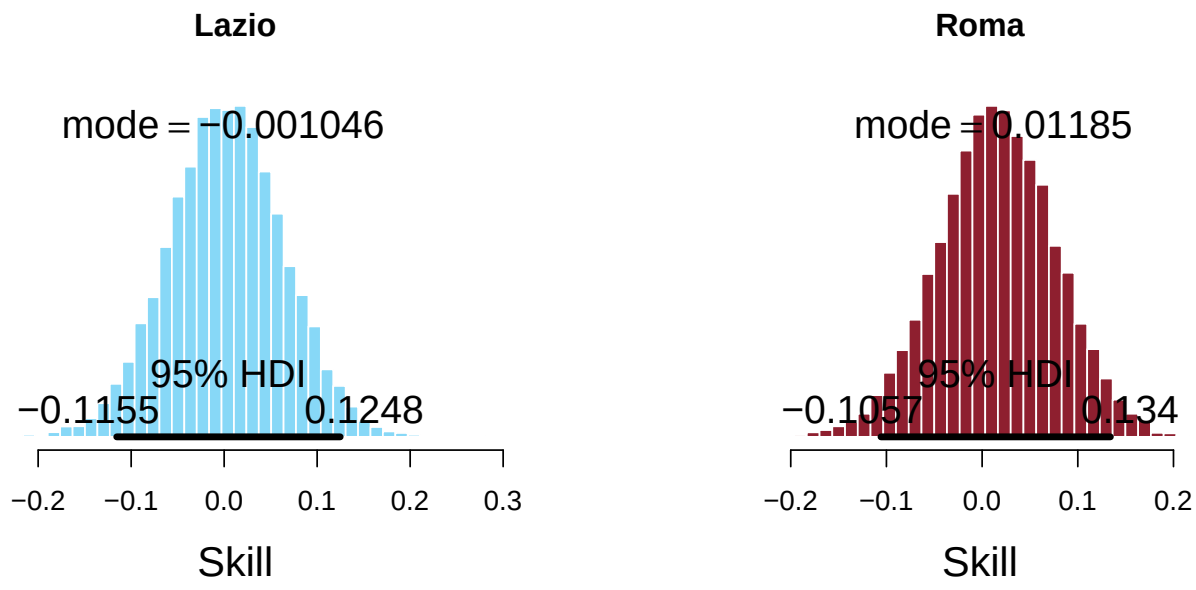

Figure 13: Summary of the skill parameter for season AS Roma and Lazio in season 2020-2021.

As we can observe, the posterior distribution for the skill parameter for AS Roma in this season has a mode of 0.011 with a HDI of [-0.105, 0.134], whereas for Lazio it takes a mode of -0.001 with a HDI of [-0.115, 0.124]. This means that AS Roma and Lazio are very similar in terms of skill for this season, with AS Roma having a slightly higher skill parameter than Lazio in contrast with what we found on the previous model. We can also observe what happens for the season before in Figure 14, where the roles invert, with Lazio having a slightly higher skill parameter than AS Roma.

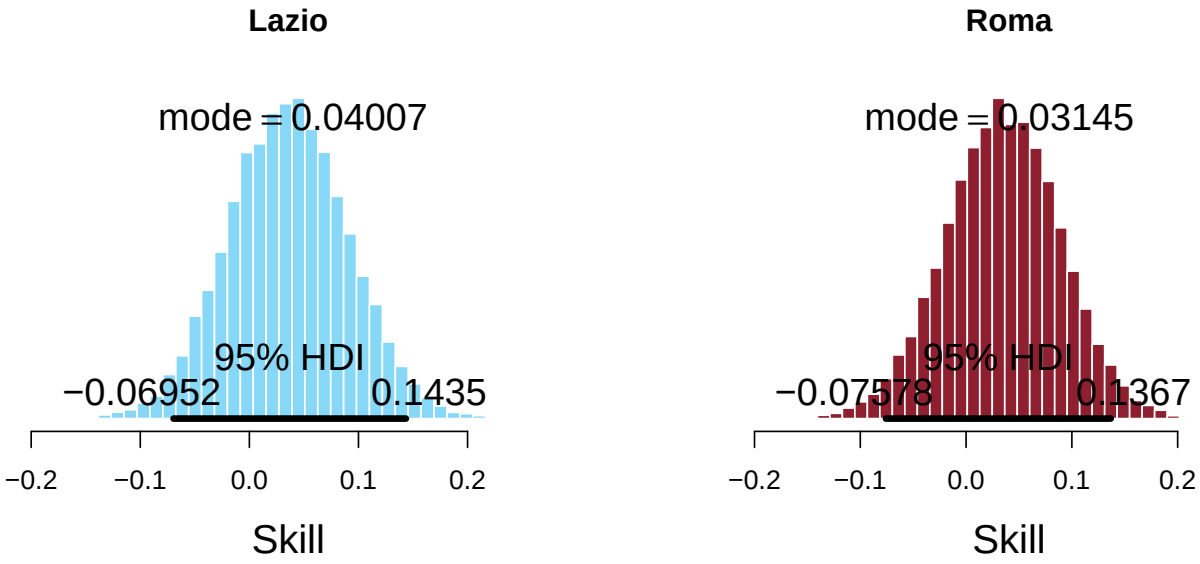

Figure 14: Summary of the skill parameter for season AS Roma and Lazio in season 2019-2020.

It is important to notice that the skill parameter is defined relative to the skill of AC Milan in season 2011-2012, which is set to zero. Nevertheless, we can re-center this parameter by substraction the mean of the skill parameter for each team. In this way, we can compare the skill of the teams relative to the average skill of the other teams in the league. In Figure 15, we show the caterpillar plot for the skill parameter of the teams in all the 4 most important European leagues for the 2020-2021 season.

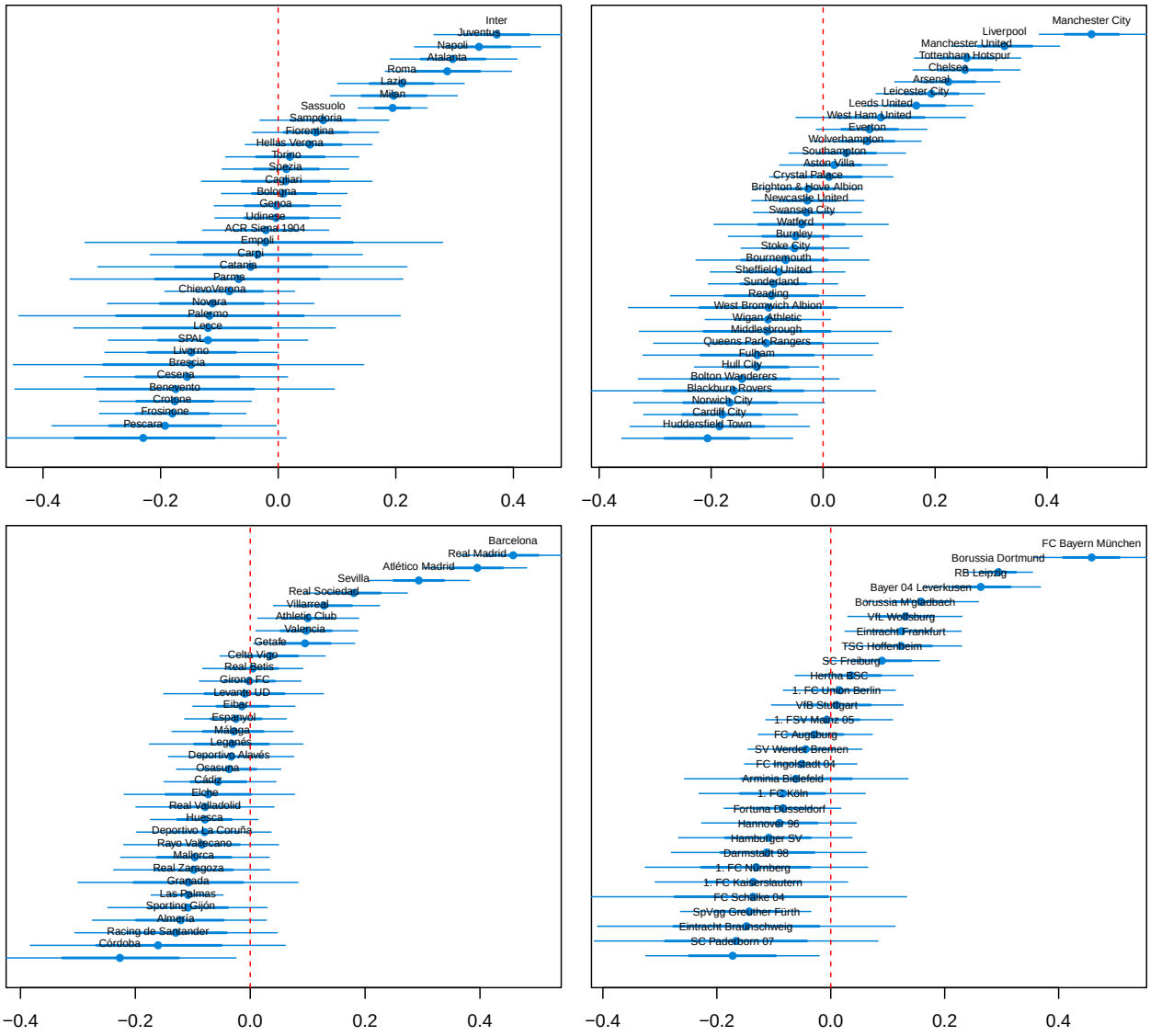

Figure 15: Caterpillar plot of the skill parameter for the teams in all European leagues for the 2020-2021 season.

As we can observe, from this plot we can distinguish the teams that are stronger than the average team in the league. It is of interest to notice that the well-considered "big" teams of the 4 most important European leagues are the ones with the highest skill parameter. In the Serie A, Juventus, Inter Milan, and Napoli are the teams with the highest skill parameter. In the Premier League, Manchester City, Liverpool, and Manchester United are the teams with the highest skill

parameter. In La Liga, Barcelona, Real Madrid, and Atlético de Madrid are the teams with the highest skill parameter. In the Bundesliga, Bayern Munich, Borussia Dortmund, and RB Leipzig are the teams with the highest skill parameter. It is of interest to see that even though there are teams that didn't participate in the 2020-2021 season, we can still estimate their skill parameter, this happens because the model is able to estimate the skill parameter for the teams that didn't participate in the season by using the information of the teams that did participate in the season by means of the prior distribution of the skill parameter.

Home Advantage

It is of interest also to infer the home advantage of a home team against an away team during a given season s and showing how the external factors of home and away affect the match result. We can define the home advantage as the difference in other external factors between the home and away teams. As the situation changes over time, external factors can differ by season. Thus, the home advantage can be quantified as the home advantage in a particular season s using the following equation:

$$\text{HA}_s = \text{OTHERS}_{\text{home},s} - \text{OTHERS}_{\text{away},s}. \tag{23}$$

We can notice that this equation holds under the assumption that the home and away teams have the same skills. Therefore, we can estimate the home advantage by the difference in the Poisson parameter of the home and away teams as follows:

$$\text{HA}_s = \lambda_{\text{home},s} - \lambda_{\text{away},s}. \tag{24}$$

This is, the difference between average goals scored by the home team and the average goals scored by the away team, that inherently depends on whether a team is playing at home or away. We can estimate the home advantage by season and plot it to see how it changes over time. In Figure 16 we present the home advantage by season for the top-4 European leagues from the 2011-2012 season to the 2020-2021 season along with its 95% credible interval and its posterior mean. Here, "After COVID-19" represents the collection of matches after the major stop of four major European leagues in March 2020 due to the COVID-19 pandemic.

As we can observe, in all four leagues, the mean values of HA in matches after the COVID-19 break were lower than those of the other nine seasons before the COVID-19 break. The mean value of HA in all matches in four European leagues from the 2011–2012 season to the season immediately before the COVID-19 break was 0.35. The mean value of HA after the COVID-19 break was 0.17. This is, the number of goals scored by the home team decreased by an average of 0.18 goals, while considering the skill difference between the teams. This shows that COVID-19 negatively affected the home advantage for all four major European football leagues. In fact, specifally for the Premier League and La Liga, we can observe that their 95% credible interval falls completely outside of the mean value of HA before the COVID-19 break. This means that the home advantage in these two leagues was significantly affected by the COVID-19 pandemic.

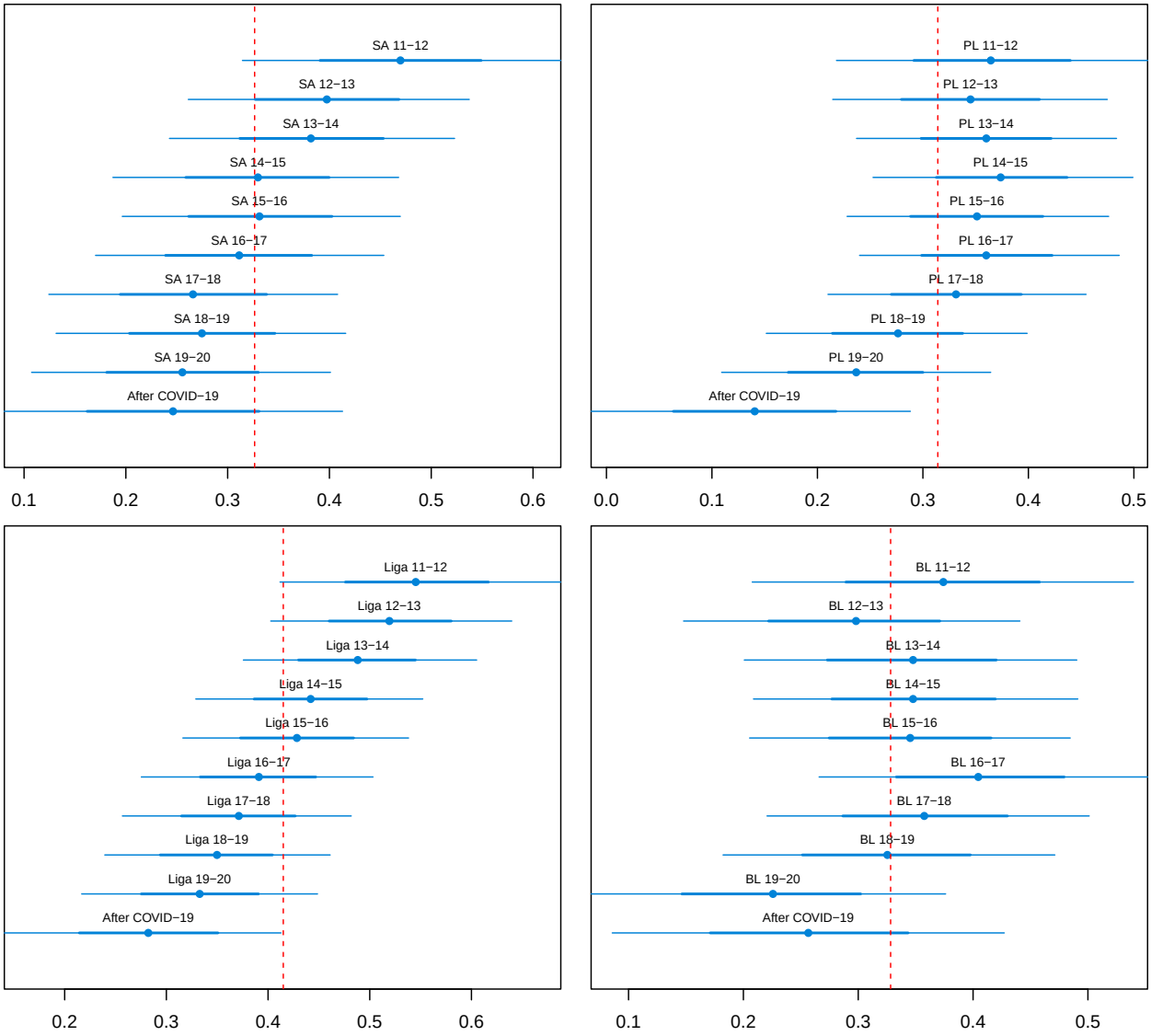

Figure 16: Home advantage by season from the 2011-2012 season to the 2020-2021 season for top-4 European leagues.

Prediction of Match Results

As in the previous model, from the MCMC samples it is not only possible to look at the distribution of parameter values, but also to make predictions about the matches by sampling from the posterior distribution of the parameters. In this case, the main distintion is that we can make predictions of matches depending on the season, given that the skill of the teams is different for each season. The main objective of this (last) part of the project is to predict the results of the matches for the 2020-2021 season from the posterior distribution of the parameters for the 2019-2020 season. In Figure 17 we show a density plot of the predictions of the number of goals scored by the home and away teams in two example matches: Roma vs Lazio and Juventus vs Torino, contrasted with the real results of the matches.

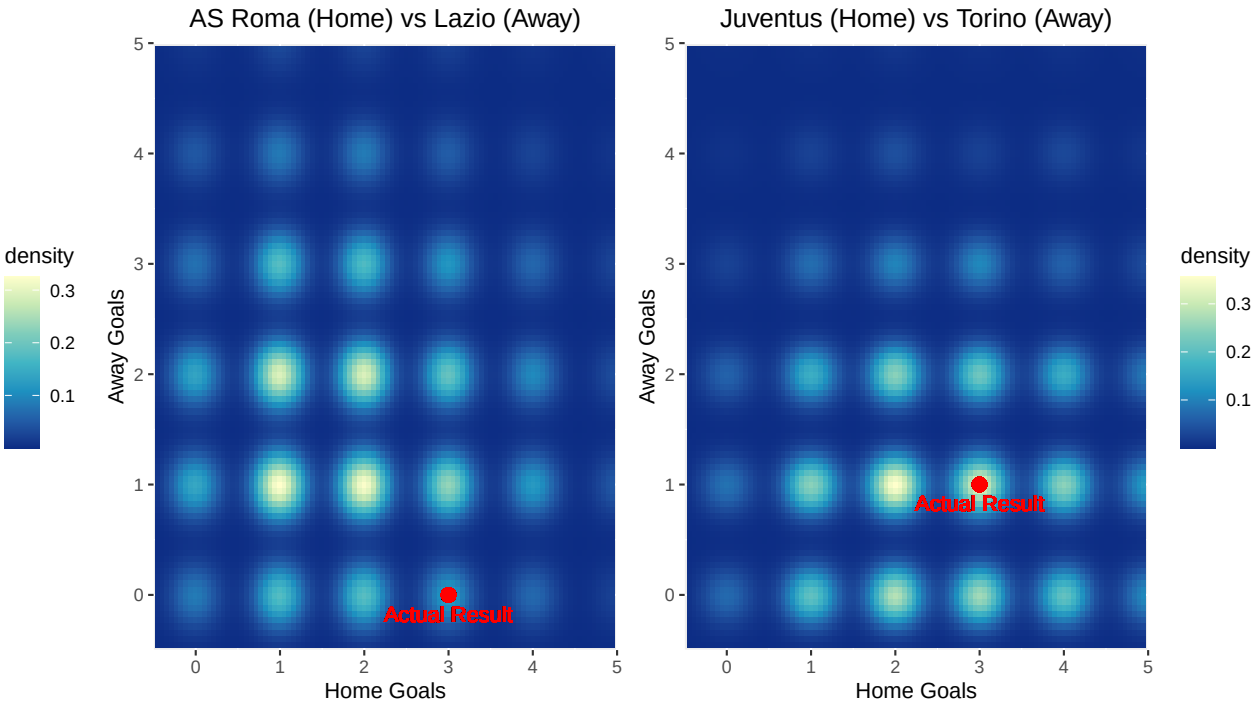

Figure 17: Density plot for the predictions of the number of goals scored by the home and away teams in a match between Roma - Lazio and Juventus - Torino in the 2020-2021 season.

As we can observe, in both cases we were able to predict AS Roma and Juventus' victories the majority of the time. However, we were not able to predict the exact number of goals scored by each team. In Table 6 we present a table with some estimations for the teams' skills, the probability of losing, drawing, and winning, the most frequent number of goals scored by the team, and the actual score. We can see more clearly from the table that we associate a higher probability of winning to the home teams given that their respective average goal scores are higher than the away teams. In particular, we present that 2-1 is the most frequent prediction for AS Roma and Juventus, which is a pretty good estimation of the actual score.

Table 6: Table with estimated parameter values and probabilities for the two matches.

Status	Team Name	Mean OTHERS	Mean SKILL	Lambda	Lose	Draw	Win	Outcome	Most Freq.
Home	Roma	0.72	0.03	2.06	0.35	0.21	0.44	3	1
Away	Lazio	0.59	0.04	1.81	0.44	0.21	0.35	0	1
Home	Juventus	0.72	0.15	2.83	0.15	0.16	0.69	3	2
Away	Torino	0.59	-0.17	1.32	0.69	0.16	0.15	1	1

As a final exercise, we can try to predict the top-4 teams that enter the Champions League for the Serie A by predicting all the match results and tracking the points of each team. We will use the same methodology as before, by simulating the matches and predicting the results. We will then calculate the points of each team and see which teams are in the top-4. We will also compare

the results with the actual Serie A table. In Table 7 we present the top-4 teams that we predicted using our model.
As we can observe, we predicted Inter as the winner of the league, with Juventus as a runner-up and Atalanta and Napoli as the other two teams that enter the Champions League. Surprisingly, this is not very far away from the actual Serie A table, where Inter won the league, Atalanta entered the Champions League, but Juventus finished in the fourth position and Napoli in the fifth. We can see that we were able to predict three out of the four teams that entered the Champions League just by using the match results and the Poisson regression model. This is an example of the power of Bayesian statistics and the Poisson regression model in predicting football match results and the potential of this model to be used in other sports as well.

Table 7: Top-4 teams of the Serie A 2020/2021 season predicted by the model.

	Team	Points
7	Juventus	108
12	Inter	108
14	Atalanta	99
17	Napoli	93

It is of interest to quantify the accuracy of our model in predicting the match results. We can do this by comparing the predicted winners with the actual winners of the matches. We can calculate the accuracy of our model by dividing the number of correct predictions by the total number of matches. The accuracy of our model is given by:

```
## [1] "Accuracy: 56.32 %"
```

This is not bad, considering is better than random guessing. It is not ideal, but predicting football match results is a very challenging task, as there are many factors that can influence the outcome of a match. This model can be further improved by including more features, such as the number of shots on target, the number of corners, the number of fouls, and other statistics that can influence the outcome of a match. This model can also be used to predict the outcomes of other sports, such as basketball, baseball, and American football, by including the relevant statistics for each sport.

Conclusion

In conclusion, this project successfully proposed and compared two Bayesian Hierarchical Models and a frequentist Poisson regression model to predict the number of goals scored in football matches. Using data from the top four European leagues over ten seasons (2011-2021), we demonstrated the effectiveness of Bayesian methods in modeling and predicting match outcomes, estimating parameters via the MCMC method using JAGS. The models provided valuable insights into key aspects of football, including home advantage and its evolution post-COVID-19, reflecting the influence of the pandemic on match dynamics. Furthermore, the simulation of the 2020-2021 season outcomes, based on the estimated parameters from 2019-2020, allowed us to assess the predictive power of our models under unusual conditions, such as matches played without spectators.

Appendix A: Poisson Regression Model

Table 8: Significant Results of the Poisson regression model fitted to the Premier League data.

	Estimate	Std. Error	Z-Value	P-Value
(Intercept)	0.6164442	0.0539988	11.415885	0.0000000
home	0.1735093	0.0168550	10.294224	0.0000000
teamAston Villa	-0.3780906	0.0598172	-6.320766	0.0000000
teamBlackburn Rovers	-0.2552702	0.1254796	-2.034356	0.0419157
teamBolton Wanderers	-0.3618748	0.1318386	-2.744832	0.0060542
teamBournemouth	-0.2510742	0.0646902	-3.881181	0.0001040
teamBrighton & Hove Albion	-0.4132809	0.0746832	-5.533789	0.0000000
teamBurnley	-0.4067691	0.0639102	-6.364694	0.0000000
teamCardiff City	-0.5032867	0.1042727	-4.826638	0.0000014
teamCrystal Palace	-0.3616559	0.0569621	-6.349064	0.0000000
teamEverton	-0.2079947	0.0506661	-4.105204	0.0000404
teamFulham	-0.3623736	0.0667815	-5.426259	0.0000001
teamHuddersfield Town	-0.6682351	0.1119940	-5.966703	0.0000000
teamHull City	-0.4337836	0.0847900	-5.115979	0.0000003
teamLeicester City	-0.1172540	0.0550263	-2.130873	0.0330996
teamManchester City	0.2026820	0.0456113	4.443682	0.0000088
teamMiddlesbrough	-0.6609566	0.1546484	-4.273931	0.0000192
teamNewcastle United	-0.3172551	0.0540120	-5.873785	0.0000000
teamNorwich City	-0.4542526	0.0691572	-6.568410	0.0000000
teamQueens Park Rangers	-0.4051415	0.0830359	-4.879115	0.0000011
teamReading	-0.3125975	0.1297213	-2.409763	0.0159629
teamSheffield United	-0.6402197	0.1108816	-5.773904	0.0000000
teamSouthampton	-0.2169814	0.0524827	-4.134344	0.0000356
teamStoke City	-0.3986079	0.0600178	-6.641498	0.0000000
teamSunderland	-0.3887909	0.0632001	-6.151750	0.0000000
teamSwansea City	-0.3321271	0.0586984	-5.658200	0.0000000
teamWatford	-0.3225293	0.0662520	-4.868216	0.0000011
teamWest Bromwich Albion	-0.3979535	0.0574889	-6.922260	0.0000000
teamWest Ham United	-0.2070782	0.0523630	-3.954670	0.0000766
teamWigan Athletic	-0.2865870	0.0934304	-3.067384	0.0021594
teamWolverhampton	-0.3125409	0.0713122	-4.382713	0.0000117
opponentAston Villa	0.2086530	0.0587330	3.552566	0.0003815
opponentBlackburn Rovers	0.4426248	0.1095467	4.040512	0.0000533
opponentBolton Wanderers	0.4477953	0.1090759	4.105353	0.0000404
opponentBournemouth	0.3134883	0.0627115	4.998894	0.0000006
opponentBrighton & Hove Albion	0.1604580	0.0704469	2.277716	0.0227435
opponentBurnley	0.1555768	0.0621135	2.504720	0.0122548
opponentCardiff City	0.3912846	0.0838364	4.667240	0.0000031

	Estimate	Std. Error	Z-Value	P-Value
opponentCrystal Palace	0.1367949	0.0577392	2.369186	0.0178273
opponentFulham	0.2984786	0.0629912	4.738419	0.0000022
opponentHuddersfield Town	0.3088134	0.0857201	3.602579	0.0003151
opponentHull City	0.2750168	0.0749942	3.667171	0.0002452
opponentManchester City	-0.1720750	0.0596380	-2.885327	0.0039101
opponentManchester United	-0.1191081	0.0585290	-2.035026	0.0418483
opponentNewcastle United	0.2100141	0.0551247	3.809802	0.0001391
opponentNorwich City	0.2827981	0.0632106	4.473904	0.0000077
opponentQueens Park Rangers	0.2914883	0.0746532	3.904566	0.0000944
opponentReading	0.3554357	0.1129815	3.145964	0.0016554
opponentSouthampton	0.1337946	0.0561975	2.380794	0.0172754
opponentStoke City	0.1315416	0.0599617	2.193762	0.0282525
opponentSunderland	0.1708745	0.0620831	2.752352	0.0059169
opponentSwansea City	0.1364731	0.0599306	2.277187	0.0227751
opponentWatford	0.2182047	0.0644580	3.385223	0.0007112
opponentWest Bromwich Albion	0.1778983	0.0571331	3.113755	0.0018472
opponentWest Ham United	0.1691354	0.0557267	3.035087	0.0024047
opponentWigan Athletic	0.3502384	0.0853088	4.105534	0.0000403
opponentWolverhampton	0.1415893	0.0710543	1.992691	0.0462953

Table 9: Significant Results of the Poisson regression model fitted to the La Liga data.

	Estimate	Std. Error	Z-Value	P-Value
(Intercept)	0.4869279	0.1222461	3.983179	0.0000680
home	0.2324899	0.0169945	13.680340	0.0000000
teamAtlético Madrid	0.3285552	0.1006242	3.265172	0.0010940
teamBarcelona	0.7302292	0.0985725	7.408043	0.0000000
teamReal Madrid	0.6915327	0.0987484	7.002978	0.0000000
teamReal Sociedad	0.2574782	0.1011546	2.545393	0.0109155
teamSevilla	0.3198573	0.1007321	3.175325	0.0014967
teamValencia	0.2693298	0.1010561	2.665151	0.0076954
teamVillarreal	0.2033518	0.1023398	1.987026	0.0469196
opponentAthletic Club	-0.2759207	0.0863014	-3.197174	0.0013878
opponentAtlético Madrid	-0.5963545	0.0897990	-6.640995	0.0000000
opponentBarcelona	-0.4559651	0.0885121	-5.151444	0.0000003
opponentDeportivo Alavés	-0.2027502	0.0935473	-2.167354	0.0302078
opponentGetafe	-0.2314870	0.0867822	-2.667449	0.0076430
opponentLeganés	-0.2239045	0.0975697	-2.294817	0.0217436
opponentMálaga	-0.2381450	0.0895899	-2.658168	0.0078567
opponentReal Madrid	-0.4125852	0.0880062	-4.688136	0.0000028
opponentReal Sociedad	-0.1981454	0.0856748	-2.312761	0.0207358
opponentSevilla	-0.2555187	0.0861969	-2.964361	0.0030331

	Estimate	Std. Error	Z-Value	P-Value
opponentValencia	-0.2613283	0.0862224	-3.030862	0.0024386
opponentVillarreal	-0.2537563	0.0871181	-2.912784	0.0035822

Table 10: Significant Results of the Poisson regression model fitted to the Bundesliga data.

	Estimate	Std. Error	Z-Value	P-Value
home	0.1710699	0.0179944	9.506814	0.0000000
team1. FC Köln	0.3875427	0.1658175	2.337165	0.0194306
team1. FC Union Berlin	0.5146317	0.1808488	2.845646	0.0044321
team1. FSV Mainz 05	0.4522702	0.1632022	2.771226	0.0055846
teamBayer 04 Leverkusen	0.6622416	0.1622427	4.081796	0.0000447
teamBorussia Dortmund	0.8595960	0.1615031	5.322474	0.0000001
teamBorussia M'gladbach	0.5979869	0.1624968	3.679991	0.0002332
teamEintracht Frankfurt	0.5731256	0.1631074	3.513792	0.0004418
teamFC Augsburg	0.4189470	0.1633604	2.564557	0.0103308
teamFC Bayern München	1.0000345	0.1610336	6.210098	0.0000000
teamFC Schalke 04	0.5162507	0.1628956	3.169212	0.0015285
teamFortuna Düsseldorf	0.4349400	0.1746298	2.490640	0.0127513
teamHannover 96	0.4194406	0.1654926	2.534498	0.0112609
teamHertha BSC	0.4647719	0.1636588	2.839883	0.0045130
teamRB Leipzig	0.7360519	0.1656821	4.442554	0.0000089
teamSC Freiburg	0.4704679	0.1636568	2.874722	0.0040438
teamSV Werder Bremen	0.4936216	0.1630474	3.027472	0.0024661
teamTSG Hoffenheim	0.6261751	0.1624180	3.855331	0.0001156
teamVfB Stuttgart	0.4904379	0.1641868	2.987072	0.0028166
teamVfL Wolfsburg	0.5794521	0.1625877	3.563935	0.0003653
opponentFC Bayern München	-0.4301248	0.1296886	-3.316598	0.0009112

References

Lee, Jaemin, Juhuhn Kim, Hyunho Kim and Jong-Seok Lee. 2022. "A Bayesian Approach to Predict Football Matches with Changed Home Advantage in Spectator-Free Matches after the COVID-19 Break." *Entropy* 24(3).

Maher, M. J. 1982. "Modelling association football scores." *Statistica Neerlandica* 36(3):109–118.

McGrath, M. 2020. "Anfield Is Only Anfield When It Is Full,' Marcelo Bielsa Says ahead of Liverpool vs. Leeds Game.". Available online: https://www.telegraph.co.uk/football/2020/09/11/anfield-anfield-full-marcelo-bielsa-says-ahead-liverpool-vs/ (accessed on 5 September 2024).

Pollard, Richard. 1985. "69.9 Goal-Scoring and the Negative Binomial Distribution." *The Mathematical Gazette* 69:45.