**Markov Chain Simulation of MLB Run Distributions for Batting Lineup Optimization**

**Ryan Presnell**

Introduction

Baseball is a game of strategy. Small decisions, such as what type of pitch to throw, can have a massive effect on a game. Batting lineups are one example of decisions that can play a big role in the outcome of a game. Markov Chain simulation can be used to model the transitions between different game states (e.g. number of outs, base runners) and estimate the resulting run distributions for different teams. If the model is accurate, teams can use simulations to determine which batting lineup(s) to use in games.

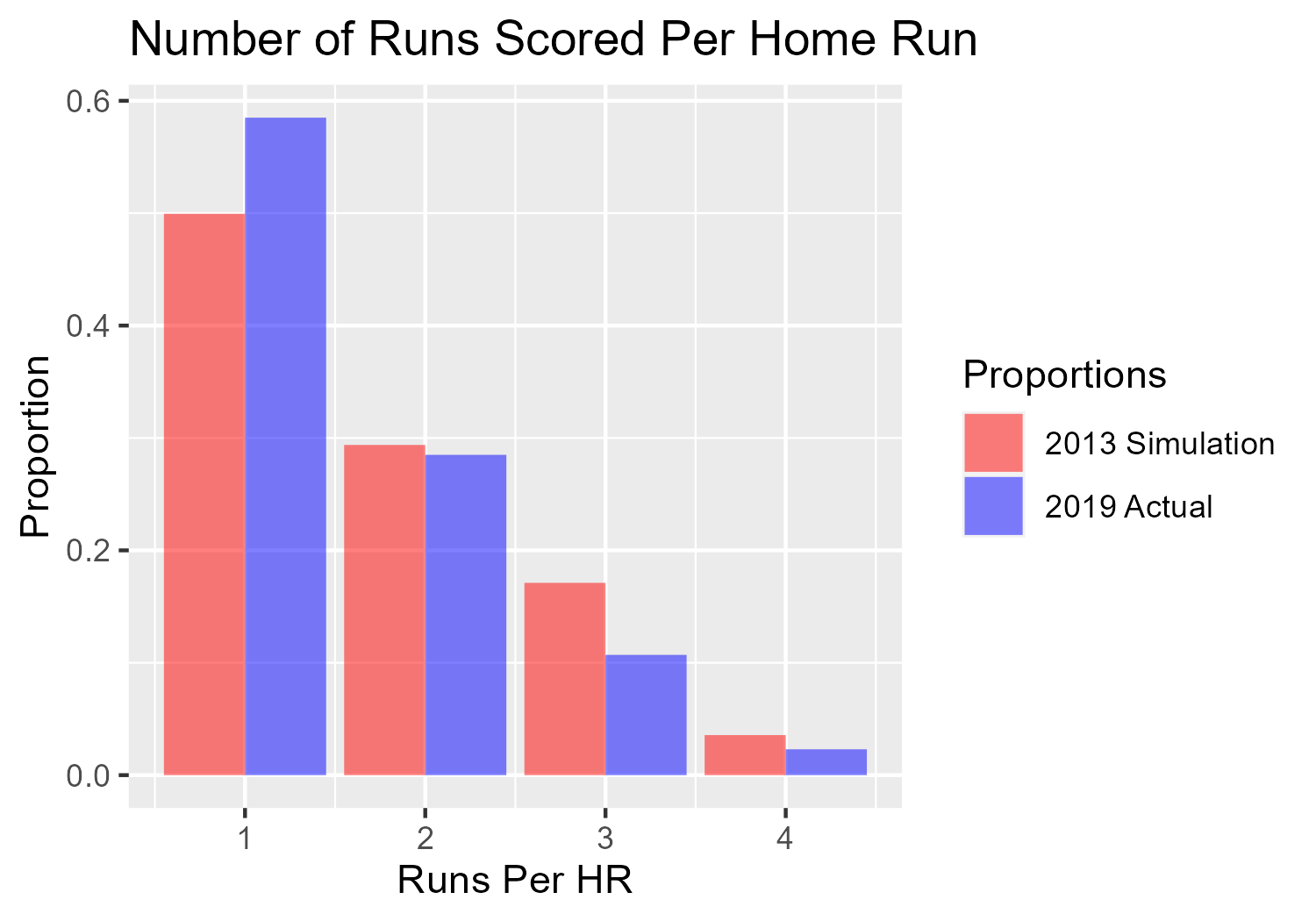
Methods

In order to calculate the run distribution for each baseball team, I used a Markov chain to model an inning of batting and then simulated through nine innings (for the sake of simplicity) to represent an entire game, while keeping track of the batting order. Each batter has a transition probability matrix using their respective proportions of at-bats resulting in singles, doubles, triples, etc. to calculate estimators for the respective probabilities of each change of state. I initially began with the transition matrix proposed by Bukiet, Harold, and Palacios (1997)[[1]](#endnote-1) as well as Ursin (2014)[[2]](#endnote-2). Ursin’s model makes certain assumptions about baseball games, including:

* When one or more outs occur on a single play, the arrangement of players on the bases does not change from the previous state.
* Runners in “scoring position”, i.e., runners on second or third base, score after a hit.
* Bases cannot be stolen— runners only advance when forced.

Baseball, in reality, is much more complex than the model assumes. However, these assumptions result in a model with a desirable level of simplicity; the only question is whether simplifying the model in such a manner will negatively impact its goodness-of-fit. These assumptions allow decent predictions to be made without making it necessary to have access to and implement statistics such as the probability of a runner on second base scoring after flyout. The model of an inning consists of 25 states—one for each possible arrangement of runners on bases with zero, one, or two outs (24 in total) and one more for the three-out state. Ideally, in order to create a representative batting lineup for each team, I would want to select the nine batters with the most at-bats for each position. This method was employed by Bukiet, Harold, and Palacios (1997). However, this data is not readily available; consequently, I went with Ursin’s practice of selecting the nine batters with the most at-bats and put them in order of descending at-bats.

Simulating [n=5000 games each, 30 teams] through 9 innings using the transition matrix offered by Ursin with 2022 MLB data led to consistent overpredictions in runs per game by about 0.5 runs. This is due in part to the assumption that a runner on second base will score after a single without fail. In reality, however, this only occurs about two-thirds of the time.[[3]](#endnote-3) After adjusting the transition probability matrices to account for this, there was still an average error of about 0.34 runs per game. In order to ensure that this was not caused by differences in the data from 2013 to 2022, I re-ran the simulation using the 2013 data. The simulation using data from the 2013 season had an average overprediction of about 0.48 runs per game. This was surprising since Ursin reported *under*predictions between 0.2-0.3 runs per game on average, and the sample size was large enough to discount randomness being responsible for this difference. I hypothesized that the reason I was getting overpredictions was at least partially due to the fact that the model assumes that home runs are equally likely to occur no matter the arrangement of baserunners—thus, the simulation may have been producing a higher number of runs due to a higher proportion of home runs with runners on base than is true to reality. In order to test this hypothesis, I ran the game simulation (n=5000) for all 30 teams using the 2013 MLB data and kept track of each change of state. After some searching, I located the proportions for number of runs scored per home run for the 2019 season on a Reddit post[[4]](#endnote-4). Since the data came from Retrosheet, which I do not frequently use, I decided to just use the 2019 numbers since they were readily available (ideally, I would want proportions for each individual player or at least each team), and it intuitively seems that the proportions would stay relatively constant throughout the years. I found that, after adjusting the number of simulated runs scored per home run to reflect 2019 proportions more accurately, the total simulated run count for the 2013 season was reduced by 29,248. This reduced the average runs per game count by about 0.194 runs per game. The graphic below illustrates the differences in proportions of runs scored from home runs between the simulation using 2013 data and the actual 2019 proportions. In light of Ursin’s assumption that a runner on second base scores after a single and lack of distinction between a home run with bases loaded and a home run with no base runners, it is difficult to understand why Ursin recorded underpredictions since both of these lead to a large increase in runs per game. Further analysis should examine why this is the case.



After adjusting player home run probabilities in the transition matrices, I re-ran the simulation using the 2022 MLB data and found overpredictions by 0.14 runs per game on average. This is relatively consistent with the reduction in runs per game after adjusting simulation results to 2019 proportions. However, this is still a fairly large mean difference, especially when considering the fact that the simulation only goes through nine innings. Historically, about 10 percent of games have gone to extra innings.[[5]](#endnote-5) The MLB implemented a rule a few seasons ago to place a runner on second base to start each regular-season extra inning,[[6]](#endnote-6) so there are likely a significant number of runs being scored in extra innings. This means that, given that the simulation is a perfect fit except for the number of innings, it should be producing underpredictions. There are a few factors to consider in trying to explain why the model may still be over-predicting even after accounting for things like runs scored per home run. First, the Markov model assumes that, on an out, the arrangement of runners on base does not change. This means that there are extra opportunities to score runs in the simulation that would not be available in a real game— in a real game of baseball, there would be one less base runner after a double play, for example, in comparison to the model. Also, teams do not use the same batting lineups every game due to injuries, fatigue, etc. An inability to use an optimal lineup under these circumstances would reduce the number of runs scored by a given team.

Despite some overpredictions, the model's performance was relatively strong. Specifically, when comparing the predicted run distributions of 27 out of the 30 MLB teams to the actual 2022 MLB data using a Kolmogorov-Smirnov test with a significance level of 0.05, the null hypothesis of identical distribution could not be rejected. This suggests that the model's predictions were consistent with the actual data for the majority (90%) of teams analyzed. The distributions of simulated and actual run distributions for the 2022 MLB season are pictured at the end of this document.

Batting Lineup Optimization

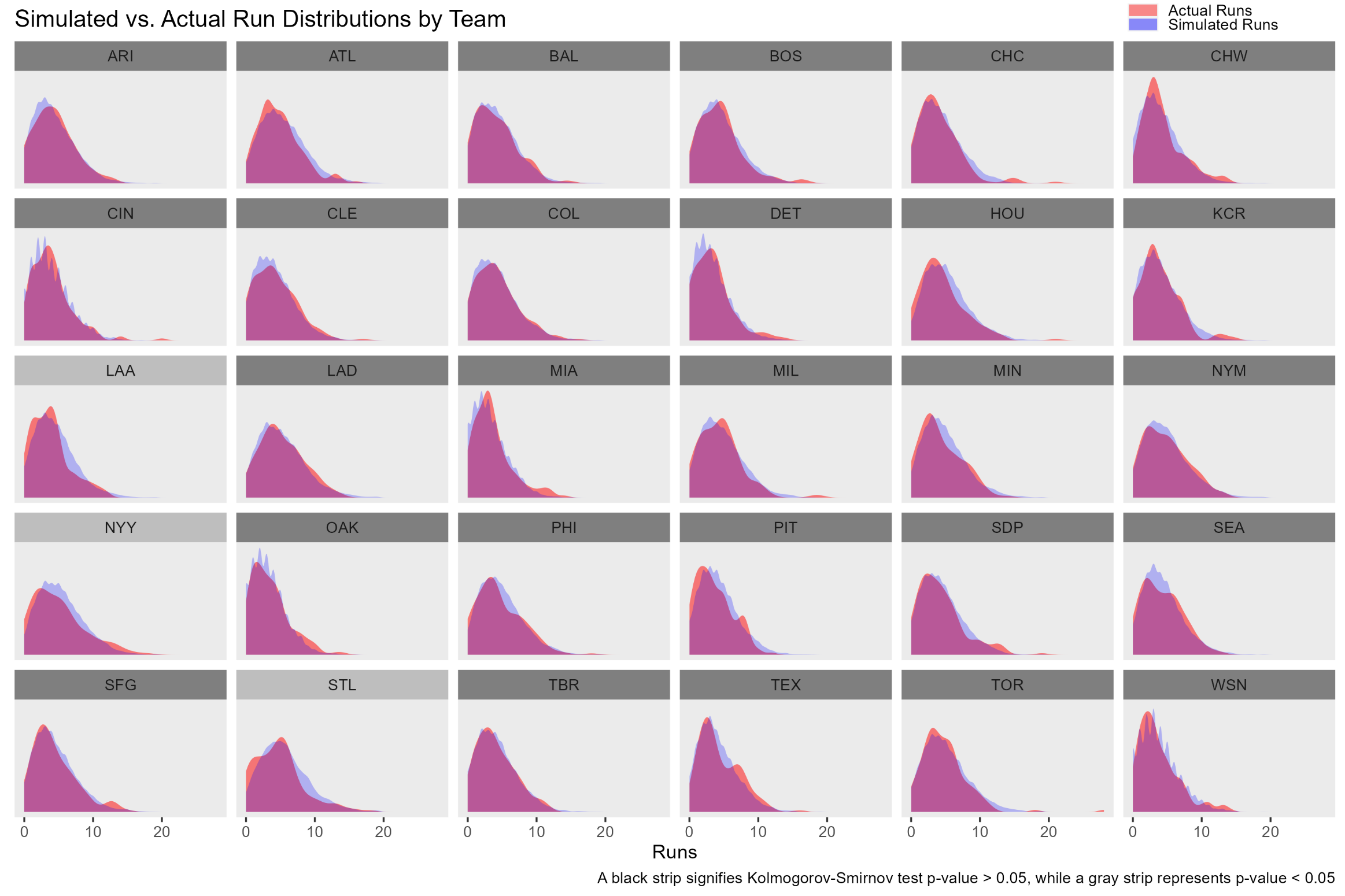
Since the quantities of runs scored in each simulated game are independent of one another and identically distributed, we can construct confidence intervals for the expected number of runs scored per game for a given lineup thanks to the central limit theorem (CLT). Additionally, the CLT allows us to conduct hypothesis tests to determine whether the expected number of runs per game for one batting lineup is significantly greater than that of another lineup. I arbitrarily chose to illustrate this with two lineups from the Washington Nationals. The first lineup I chose is the lineup suggested by Ursin as a representative lineup, namely, the nine batters with the most at-bats in decreasing order. The second lineup, which I arbitrarily chose, is the first lineup backwards. I simulated 250 replications of 100 games for each lineup. The grand sample means for the lineups were 3.850 and 4.052 runs per game, respectively. Conducting a two-sample t-test yielded a p-value of 6.841e-14, meaning that the expected runs per game of the second lineup was significantly more than that of the first lineup. This may be due to the lineup chosen to be representative for the team’s season not truly being a representative lineup—unfortunately, the number of at-bats at each lineup position for each player is not a statistic that is readily available. Another possibility is that the model is simply incorrect in predicting that the expected runs per game of the reversed lineup is higher than that of the original; it is hard to imagine that, given that the original lineup is a truly representative lineup, the same lineup in reverse order would be better. Baseball teams use certain heuristics when composing lineups that typically result in lineups with good performance.

Chart, histogram

Description automatically generated

Conclusion & Discussion

Markov chain simulation is an effective tool for modeling the transitions between different game states and estimating the resulting run distributions for different baseball teams. While the assumptions made by the model resulted in a desirable level of simplicity, they negatively impacted the model's goodness-of-fit; the simulation results consistently overpredicted runs per game. Further analysis should incorporate more complexity into the model by using different estimators for change-of-state probabilities that take the current state into account. Also, it would likely be beneficial to not have the model rely on the assumption that the arrangement of runners on bases does not change after outs. The combination of the lack of model complexity as well as the lack of readily available data make it difficult to determine how accurate the results from hypothesis tests would be using this particular model.



1. Bukiet, B., Harold, E. R., & Palacios, J. L. (1997). A Markov Chain Approach to Baseball. *Operations Research*, *45*(1), 14–23. http://www.jstor.org/stable/171922 [↑](#endnote-ref-1)
2. Ursin, Daniel Joseph. (2014). A Markov Model for Baseball with Applications. *Theses and Dissertations*. 964. https://dc.uwm.edu/etd/964 [↑](#endnote-ref-2)
3. Tango, T. (2014). The Book--Playing the Percentages in Baseball, 154. [↑](#endnote-ref-3)
4. https://www.reddit.com/r/baseball/comments/lwdh0r/stats\_question\_average\_runs\_scored\_per\_home\_run/ [↑](#endnote-ref-4)
5. https://www.cbssports.com/mlb/news/mlb-ghost-runner-rule-league-makes-extra-innings-change-permanent-for-regular-season-games-per-report/#:~:text=The%20rule%20remained%20on%20a,extra%20innings%2C%20or%209.2%20percent. [↑](#endnote-ref-5)
6. https://www.espn.com/mlb/story/\_/id/35652660/source-extra-inning-extra-runner-rule-remain­­ [↑](#endnote-ref-6)