# Analyzing Baseball Team Performance in Softball Using Markov Chains

Bryan Steven Monge Serrano
MATH 104 Winter 2024
Stanford University
bryan101@stanford.edu

**Abstract**

This study utilizes Markov Chains to simulate baseball games between the San Francisco Giants and the New York Yankees. Transition matrices derived from Retrosheet data, from the years 2015 to 2023, are used to model the probability of different game states. Various statistical comparisons are made between teams and individual players to determine key performance indicators. The results provide insight into how Markov Chains can model and predict baseball outcomes effectively as well as their limitations, i.e. Burn-in and the nature of the stationary distributions as n games approach infinity.

## 1 Introduction

Baseball is a game of probabilities with several states, making it a reasonable choice for Markov chain modeling. In this study, we build and analyze transition matrices from historical game data to evaluate the relative performance of the San Francisco Giants and New York Yankees. Using a Markov Chain representation, we track player-specific and team-level statistics to gain deeper insights into game dynamics.

### 1.1 Inspiration

To get a better understanding of how the Markov Chain works and my inspiration, I want to turn to this graph and explain my previous work.

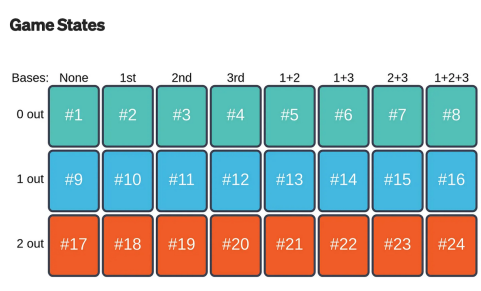


Figure 1: A graphic from the article explaining the concept of using Markov Chains to simulate baseball broadly. Credit: Lucas Calestini.

Originally, this project began as a project related to simulating real-world baseball teams playing against each other in Wii Sports Baseball. This project idea came from Lucas Calestini's article (on Medium) discussing "The Elegance of Markov Chains in Baseball." I adapted what I learned from this article to my original project to simulate baseball teams facing each other in Wii Sports.

For this project, I use this knowledge and adapt what I have to encode what baseball teams would be like under the conditions of softball, and to go beyond what I had done previously. For my Wii Sports project, all I was able to complete making to draw a comparison of the performance of different teams.

You can see the results of the previous project in the following:

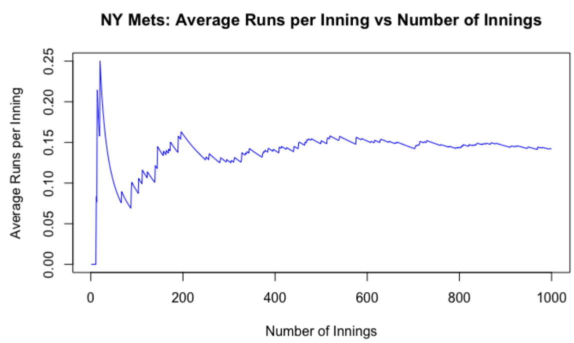


Figure 2: Comparison of Mets' Average Runs per Inning.

*The Average Runs Per Inning for the New York Mets is 0.45; however, our model converges to 0.13 Average Runs Per Inning.*

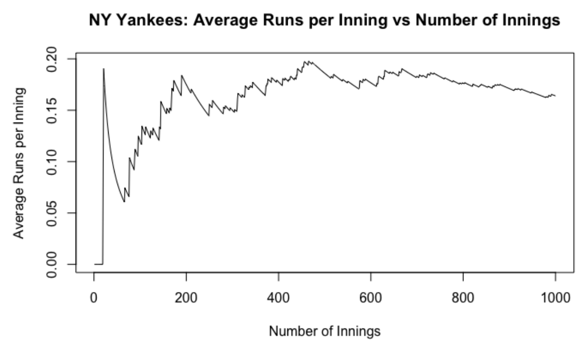


Figure 3: Comparison of Yankees' Average Runs per Inning.

*The Average Runs Per Inning for the New York Yankees is 0.50; however, our model converges to 0.16 Average Runs Per Inning.*

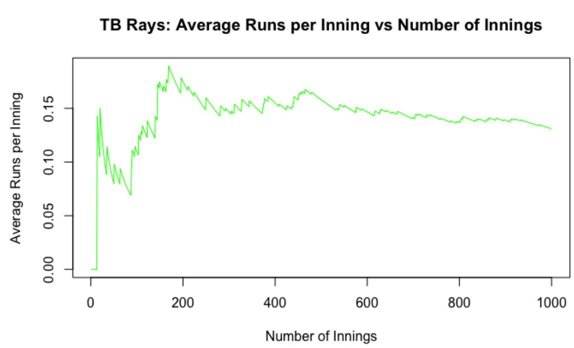


Figure 4: Comparison of Rays' Average Runs per Inning.

*The Average Runs Per Inning for the Tampa Bay Rays is 0.47; however, our model converges to 0.14 Average Runs Per Inning.*

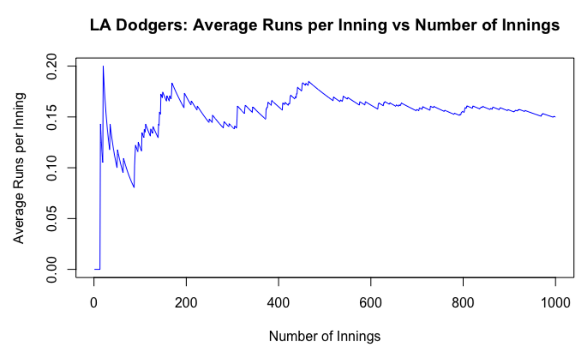


Figure 5: Comparison of Dodgers' Average Runs per Inning.

*The Average Runs Per Inning for the Los Angeles Dodgers is 0.52; however, our model converges to 0.15 Average Runs Per Inning.*

As you can see, the previous limitations for my project were that despite having a Markov chain mapped to differing states, I was unable to make different teams simulate games against each other. As such, the best metric that I could obtain to compare the performance of different teams was "The Average Runs Per Inning" to assess which teams would have more runs, thus a higher chance of winning.

What I hadn't realized about the Markov Chains was the property of the long-term distribution, which led to results where the differences were minuscule. To control for this, I opted to focus on the short-term average, given that most baseball teams do not play for 100+ games in a season and that doing so in the real world is unrealistic and would lead to fatigue. By opting to do this, I was able to see more stark differences and draw better conclusions on performance, as seen above.

For this project, I will build upon this limitation and find other statistics using the Markov Chain as well as determine which are the best players.

# 2 Methods

## 2.1 Transition Matrices

Each game is simulated using transition matrices based on historical data. The matrices capture the probability of moving from one game state (e.g., no runners, runners on base, outs) to another based on past events. However, we leverage the memoryless property of Markov Chains as we only need to know what the previous play and positions of players are on bases are to make the next move. This helps to make the project relatively light-weight in terms of computing power and is crucial to the functionality of jumping from state to state.

| | [,1] | [,2] | [,3] | [,4] | [,5] | [,6] | [,7] | [,8] | [,9] | [,10] | [,11] | [,12] | [,13] | [,14] | [,15] | [,16] | [,17] | [,18] | [,19] | [,20] | [,21] | [,22] | [,23] | [,24] | [,25] |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| [1,] | 323 | 2791 | 555 | 98 | 0 | 0 | 0 | 0 | 8633 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| [2,] | 77 | 0 | 0 | 18 | 612 | 0 | 141 | 0 | 0 | 1597 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| [3,] | 20 | 0 | 39 | 1 | 199 | 0 | 0 | 0 | 0 | 0 | 573 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| [4,] | 3 | 39 | 6 | 1 | 0 | 11 | 0 | 0 | 0 | 0 | 0 | 111 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| [5,] | 12 | 0 | 0 | 6 | 0 | 0 | 34 | 162 | 0 | 0 | 0 | 0 | 402 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| [6,] | 5 | 0 | 0 | 5 | 33 | 0 | 9 | 12 | 0 | 0 | 0 | 0 | 0 | 126 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| [7,] | 2 | 0 | 8 | 1 | 0 | 43 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 118 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| [8,] | 2 | 0 | 0 | 1 | 0 | 0 | 9 | 42 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 109 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| [9,] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 200 | 2012 | 394 | 48 | 0 | 0 | 0 | 0 | 6328 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| [10,] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 82 | 0 | 0 | 16 | 766 | 0 | 176 | 0 | 0 | 1893 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| [11,] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 37 | 0 | 60 | 7 | 366 | 0 | 0 | 0 | 0 | 0 | 880 | 0 | 0 | 0 | 0 | 0 | 0 |
| [12,] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 10 | 79 | 26 | 4 | 0 | 49 | 0 | 0 | 0 | 0 | 0 | 295 | 0 | 0 | 0 | 0 | 0 |
| [13,] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 27 | 0 | 0 | 4 | 0 | 0 | 39 | 279 | 0 | 0 | 0 | 0 | 684 | 0 | 0 | 0 | 0 |
| [14,] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 15 | 0 | 0 | 6 | 97 | 0 | 21 | 29 | 0 | 0 | 0 | 0 | 0 | 281 | 0 | 0 | 0 |
| [15,] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 7 | 0 | 22 | 5 | 0 | 104 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 234 | 0 | 0 |
| [16,] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13 | 0 | 0 | 5 | 0 | 0 | 25 | 100 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 232 | 0 |
| [17,] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 150 | 1583 | 312 | 44 | 0 | 0 | 0 | 0 | 5007 |
| [18,] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 81 | 0 | 0 | 19 | 738 | 0 | 159 | 0 | 673 |
| [19,] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 33 | 0 | 75 | 20 | 437 | 0 | 0 | 0 | 411 |
| [20,] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 12 | 93 | 37 | 4 | 0 | 91 | 0 | 0 | 195 |
| [21,] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 37 | 0 | 0 | 12 | 0 | 0 | 65 | 321 | 1051 |
| [22,] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 16 | 0 | 0 | 4 | 111 | 0 | 29 | 41 | 462 |
| [23,] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 12 | 0 | 16 | 2 | 0 | 127 | 0 | 0 | 353 |
| [24,] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 6 | 0 | 0 | 3 | 0 | 0 | 14 | 125 | 382 |
| [25,] | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Figure 6: The loaded transition states without normalization.

| | [,1] | [,2] | [,3] | [,4] | [,5] | [,6] | [,7] | [,8] | [,9] | [,10] |
|---|---|---|---|---|---|---|---|---|---|---|
| [1,] | 0.02604839 | 0.2250806 | 0.04475806 | 0.007903226 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.69620968 | 0.0000000 |
| [2,] | 0.03149284 | 0.0000000 | 0.00000000 | 0.007361963 | 0.2503067 | 0.00000000 | 0.05766871 | 0.00000000 | 0.00000000 | 0.6531697 |
| [3,] | 0.02403846 | 0.0000000 | 0.04687500 | 0.001201923 | 0.2391827 | 0.00000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 |
| [4,] | 0.01754386 | 0.2280702 | 0.03508772 | 0.005847953 | 0.0000000 | 0.06432749 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 |
| [5,] | 0.01948052 | 0.0000000 | 0.00000000 | 0.009740260 | 0.0000000 | 0.00000000 | 0.05519481 | 0.26298701 | 0.00000000 | 0.0000000 |
| [6,] | 0.02631579 | 0.0000000 | 0.00000000 | 0.026315789 | 0.1736842 | 0.00000000 | 0.04736842 | 0.06315789 | 0.00000000 | 0.0000000 |
| [7,] | 0.01162791 | 0.0000000 | 0.04651163 | 0.005813953 | 0.0000000 | 0.25000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 |
| [8,] | 0.01226994 | 0.0000000 | 0.00000000 | 0.006134969 | 0.0000000 | 0.00000000 | 0.05521472 | 0.25766871 | 0.00000000 | 0.0000000 |
| [9,] | 0.00000000 | 0.0000000 | 0.00000000 | 0.000000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.02226676 | 0.2240036 |
| [10,] | 0.00000000 | 0.0000000 | 0.00000000 | 0.000000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.02795772 | 0.0000000 |
| [11,] | 0.00000000 | 0.0000000 | 0.00000000 | 0.000000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.02740741 | 0.0000000 |
| [12,] | 0.00000000 | 0.0000000 | 0.00000000 | 0.000000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.02159827 | 0.1706263 |
| [13,] | 0.00000000 | 0.0000000 | 0.00000000 | 0.000000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.02613746 | 0.0000000 |
| [14,] | 0.00000000 | 0.0000000 | 0.00000000 | 0.000000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.03340757 | 0.0000000 |
| [15,] | 0.00000000 | 0.0000000 | 0.00000000 | 0.000000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.01881720 | 0.0000000 |
| [16,] | 0.00000000 | 0.0000000 | 0.00000000 | 0.000000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.03466667 | 0.0000000 |
| [17,] | 0.00000000 | 0.0000000 | 0.00000000 | 0.000000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 |
| [18,] | 0.00000000 | 0.0000000 | 0.00000000 | 0.000000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 |
| [19,] | 0.00000000 | 0.0000000 | 0.00000000 | 0.000000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 |
| [20,] | 0.00000000 | 0.0000000 | 0.00000000 | 0.000000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 |
| [21,] | 0.00000000 | 0.0000000 | 0.00000000 | 0.000000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 |
| [22,] | 0.00000000 | 0.0000000 | 0.00000000 | 0.000000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 |
| [23,] | 0.00000000 | 0.0000000 | 0.00000000 | 0.000000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 |
| [24,] | 0.00000000 | 0.0000000 | 0.00000000 | 0.000000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 |
| [25,] | 0.00000000 | 0.0000000 | 0.00000000 | 0.000000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 |

| | [,11] | [,12] | [,13] | [,14] | [,15] | [,16] | [,17] | [,18] | [,19] |
|---|---|---|---|---|---|---|---|---|---|
| [1,] | 0.00000000 | 0.000000000 | 0.0000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 | 0.00000000 |
| [2,] | 0.00000000 | 0.000000000 | 0.0000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 | 0.00000000 |
| [3,] | 0.68870192 | 0.000000000 | 0.0000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 | 0.00000000 |
| [4,] | 0.00000000 | 0.649122807 | 0.0000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 | 0.00000000 |
| [5,] | 0.00000000 | 0.000000000 | 0.6525974 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 | 0.00000000 |
| [6,] | 0.00000000 | 0.000000000 | 0.0000000 | 0.6631579 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 | 0.00000000 |
| [7,] | 0.00000000 | 0.000000000 | 0.0000000 | 0.0000000 | 0.68604651 | 0.00000000 | 0.00000000 | 0.0000000 | 0.00000000 |
| [8,] | 0.00000000 | 0.000000000 | 0.0000000 | 0.0000000 | 0.00000000 | 0.66871166 | 0.00000000 | 0.0000000 | 0.00000000 |
| [9,] | 0.04386551 | 0.005344021 | 0.0000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.70452015 | 0.0000000 | 0.00000000 |
| [10,] | 0.00000000 | 0.005455165 | 0.2611660 | 0.0000000 | 0.06000682 | 0.00000000 | 0.00000000 | 0.6454143 | 0.00000000 |
| [11,] | 0.04444444 | 0.005185185 | 0.2711111 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 | 0.65185185 |
| [12,] | 0.05615551 | 0.008639309 | 0.0000000 | 0.1058315 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 | 0.00000000 |
| [13,] | 0.00000000 | 0.003872217 | 0.0000000 | 0.0000000 | 0.03775411 | 0.27008712 | 0.00000000 | 0.0000000 | 0.00000000 |
| [14,] | 0.00000000 | 0.013363029 | 0.2160356 | 0.0000000 | 0.04677060 | 0.06458797 | 0.00000000 | 0.0000000 | 0.00000000 |
| [15,] | 0.05913978 | 0.013440860 | 0.0000000 | 0.2795699 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 | 0.00000000 |
| [16,] | 0.00000000 | 0.013333333 | 0.0000000 | 0.0000000 | 0.06666667 | 0.26666667 | 0.00000000 | 0.0000000 | 0.00000000 |
| [17,] | 0.00000000 | 0.000000000 | 0.0000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.02113867 | 0.2230834 | 0.04396843 |
| [18,] | 0.00000000 | 0.000000000 | 0.0000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.04850299 | 0.0000000 | 0.00000000 |
| [19,] | 0.00000000 | 0.000000000 | 0.0000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.03381148 | 0.0000000 | 0.07684426 |
| [20,] | 0.00000000 | 0.000000000 | 0.0000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.02777778 | 0.2152778 | 0.08564815 |
| [21,] | 0.00000000 | 0.000000000 | 0.0000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.02489906 | 0.0000000 | 0.00000000 |
| [22,] | 0.00000000 | 0.000000000 | 0.0000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.02413273 | 0.0000000 | 0.00000000 |
| [23,] | 0.00000000 | 0.000000000 | 0.0000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.02352941 | 0.0000000 | 0.03137255 |
| [24,] | 0.00000000 | 0.000000000 | 0.0000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.01132075 | 0.0000000 | 0.00000000 |
| [25,] | 0.00000000 | 0.000000000 | 0.0000000 | 0.0000000 | 0.00000000 | 0.00000000 | 0.00000000 | 0.0000000 | 0.00000000 |

Figure 7: The loaded transition states with normalization (*cutoff states).

## 2.2 Simulation Process

To simulate a game, we:

1. Construct transition matrices from historical play-by-play data for each team.
2. Model each half-inning as a Markov process, transitioning between different states based on probabilities.
3. Track key statistics such as runs, hits, strikeouts, and batting averages, as well as player statistics.

## 2.3 Player and Team Performance Metrics

Performance is measured using:

- Total Game Outcomes (with Extra Innings)
- Run Distribution
- Average Runs per Inning
- Average Runs per Team
- Percentage of Shutouts
- Standard deviation of runs
- Total of Close (winning by exactly 1 run) and Big Wins (winning by more than 5 runs)
- Top Run Scorers
- Top Batters on Average
- Top Players who strikeouts
- OPS (On Base + Slugging) Comparison

# 3 Results

Below are the results of all of my calculations

## 3.1 Win Probability Comparison

Win probability is a crucial metric in evaluating team performance, as it reflects the overall likelihood of victory in simulated matchups. By comparing win percentages and average runs per game, we can identify which team performs better over a large number of simulated games and assess the impact of scoring efficiency on game outcomes.

| Team | Win Percentage | Average Runs/Game |
|---|---|---|
| Giants | 51.10% | 13.6791 |
| Yankees | 48.86% | 13.0853 |
| Extra Inning Ties | 00.04% | - |

Table 1: Win probabilities and average runs per game across simulations, highlighting team performance.

### 3.2 Percentage of Shutouts

Shutout games provide insight into team offensive performance and consistency. Below is the percentage of games where each team was held scoreless during the simulations.

| Team | Percentage (%) |
|---|---|
| Giants | 1.19 |
| Yankees | 2.85 |

Table 2: Percentage of shutout games where each team was held scoreless in the simulations.

### 3.3 Additional Game Statistics

The table below presents additional game statistics, including the standard deviation of runs, close wins, and big wins for both teams.

| Team | Std. Dev. |
|---|---|
| Giants | 5.654 |
| Yankees | 6.654 |

Table 3: Standard deviation of runs per team.

| Team | Close Wins |
|---|---|
| Giants | 450 |
| Yankees | 444 |

Table 4: Number of close wins (winning by 1 run).

| Team | Big Wins |
|---|---|
| Giants | 3145 |
| Yankees | 2740 |

Table 5: Big wins (winning by 5+ runs).

### 3.4 Game Outcomes

Analyzing overall game outcomes provides insight into how often each team wins. This data helps assess team strength beyond just average runs scored.

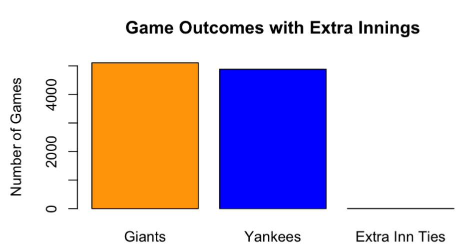


Figure 8: Total Game Outcomes visualized, showing win rates and tie frequencies.

### 3.5 Run Distributions (per team)

Understanding the distribution of runs per game is crucial for evaluating team consistency and offensive capabilities. A wider distribution suggests a higher variance in scoring, meaning the team can have both explosive performances and weak outings. A tighter distribution, on the other hand, indicates steadier offensive output.

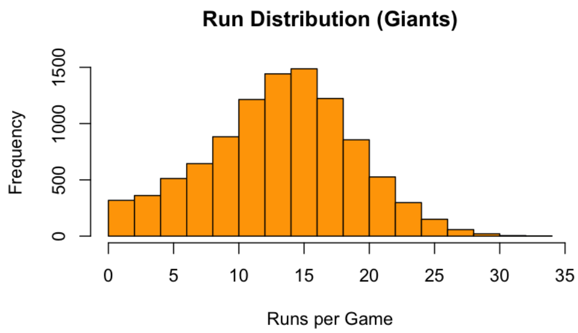


Figure 9: Overall run distribution for the Giants.

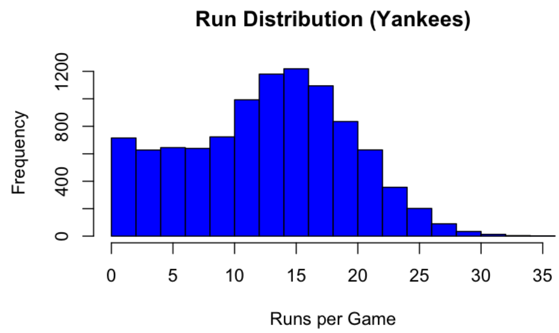


Figure 10: Overall run distribution for the Yankees.

### 3.6 Average Runs per Inning (per team)

Tracking the average runs per inning is essential for evaluating a team's offensive consistency. This metric helps determine how frequently a team scores throughout a game rather than just total runs. A team that maintains a steady scoring rate is more likely to sustain leads and recover from deficits, making it a crucial factor in overall performance assessment.

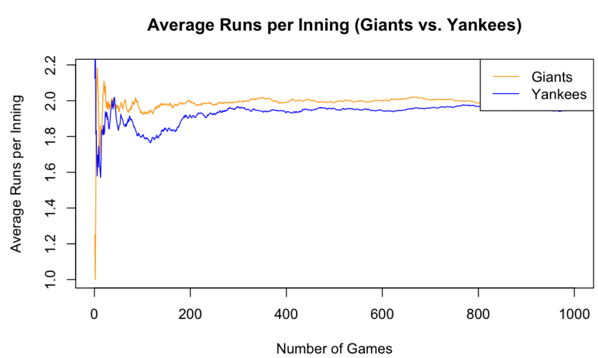


Figure 11: The Average Runs Per Inning for Both Teams, illustrating their scoring consistency.

### 3.7 Player Performance Metrics

Understanding individual player contributions is crucial for evaluating a team's overall success. Key statistics such as top run scorers, batting performance, and strikeouts provide insight into offensive efficiency and areas for improvement. By identifying high-performing players, teams can strategize better lineups and training regimens.

#### 3.7.1 Top Run Scorers

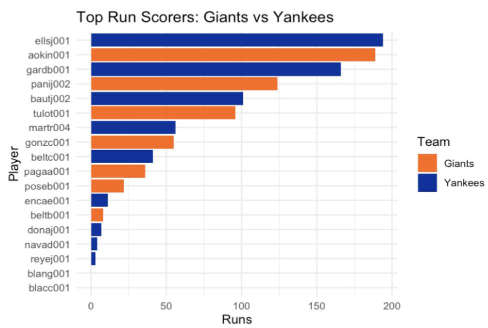


Figure 12: Top run scorers for each team based on simulation results, highlighting key offensive contributors.

#### 3.7.2 Top Batters

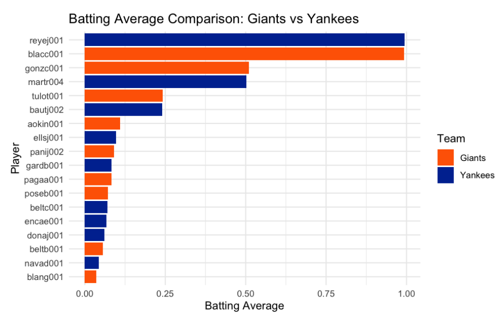


Figure 13: Top batters for each team based on simulation results, showcasing players with high batting averages and consistency.

#### 3.7.3 Top Struck Out Players

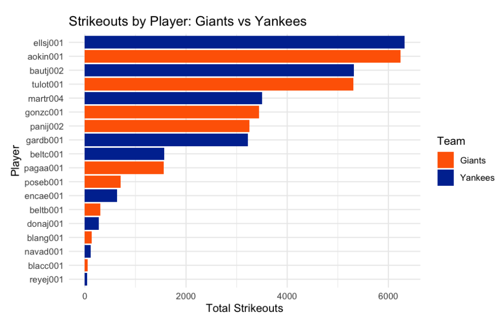


Figure 14: Top players who recorded the most strikeouts, indicating potential weaknesses in hitting performance.

# 4 Discussion

The results of this study demonstrate the effectiveness of Markov Chain modeling in simulating baseball game dynamics and player performance while simulating softball rules. By constructing transition matrices from historical data, I captured the probability of various in-game states, allowing me to analyze team and player performance comprehensively.

## 4.1 Team Performance and Win Probability

The simulated matchups between the Giants and the Yankees revealed a nearly even competition, with the Giants securing a slight edge in win probability (51.10%) over the Yankees (48.86%). Despite the small margin, the win percentages suggest that both teams perform at comparable levels over multiple simulated games. The frequency of extra-inning ties (0.04%) reinforces the idea that many games were closely contested.

## 4.2 Run Scoring Patterns and Consistency

Examining the run distributions and average runs per inning provided deeper insights into each team's offensive consistency. The Giants maintained a slightly higher average runs per game (13.6791) compared to the Yankees (13.0853), reflecting a marginal scoring advantage. However, the standard deviation of runs (Giants: 5.65, Yankees: 6.65) suggests that the Yankees exhibited greater variability in offensive output. This higher variance could imply that the Yankees were more prone to both high-scoring and low-scoring games, whereas the Giants maintained a steadier run distribution over the course of their seasons.

## 4.3 Shutouts, Close Wins, and Big Wins

The analysis of shutouts showed that the Yankees were more frequently held scoreless (2.85% of games) compared to the Giants (1.19%), suggesting that the Yankees' offense had more instances of stagnation. Additionally, close wins (games decided by a single run) were nearly identical for both teams, with the Giants winning 450 such games and the Yankees winning 444. This further supports the notion of highly competitive games. However, when considering big wins (victories by 5 or more runs), the Giants had a more dominant presence with 3,145 big wins compared to the Yankees' 2,740, indicating a stronger ability to capitalize on high-scoring opportunities on the field.

## 4.4 Individual Player Contributions

The simulation also highlighted key player performance metrics, including top run scorers, batting averages, and strikeouts. Identifying the best offensive contributors allows for a more nuanced understanding of team strengths and areas for improvement. Players with high run prove to be significant factors in their respective teams' offensive production. Conversely, analyzing the most struck-out players provides insight into potential weaknesses in batting consistency, which could be an area for further strategic adjustments, such as focusing more batting training time on said players.

Using Markov Chain simulations, we identified the top-performing players from the San Francisco Giants and New York Yankees between 2015-2023 based on runs scored, batting performance, and strikeout rates. In terms of runs scored, the top runners for the Giants were Norichika Aoki

(2015) and Joe Panik (2014-2019), while for the Yankees, Jacoby Ellsbury (2013-2014), Brett Gardner (2018-2021), and Troy Tulowitzki (2019) led in run production. In batting performance, which accounts for total hits and bases, the top batters for the Giants were again Norichika Aoki (2015) and Joe Panik (2014-2019). For the Yankees, the best batters were Troy Tulowitzki (2019), Russell Martin (2011-2012), and Jacoby Ellsbury (2013-2014). In terms of strikeouts, the Giants' most strikeout-prone players were Norichika Aoki (2015) and Joe Panik (2014-2019), while for the Yankees, Jacoby Ellsbury (2013-2014), Troy Tulowitzki (2019), and Brett Gardner (2018-2021) had the highest strikeout rates.

This analysis highlights that Norichika Aoki (2015) was the most impactful overall player for the Giants, balancing offensive contribution with run generation, while Jacoby Ellsbury (2013-2014) stood out for the Yankees despite his high strikeout rate. The key weakness observed in both teams was the tendency for top offensive contributors to have high strikeout rates, whereas a major strength was the ability to generate runs. This evaluation demonstrates how Markov Chains provide deeper insights into player performance beyond traditional statistics. .

### 4.5 Limitations and Considerations

Despite the success of this Markov Chain approach, some limitations should be acknowledged. First, while the transition matrices provide a robust probabilistic model for game states, they do not account for contextual elements such as pitcher-batter matchups, defensive strategies, or in-game decision-making that could alter outcomes. Furthermore, I faced difficulty filtering out the teams who had played against the Giants and Yankees as the statistics depended on it. This led to the player graphs correctly highlighting the stats for players after simulated games, but had the error of choosing and graphing some players who had played against the desired teams. Additionally, since real-world baseball exhibits long-term trends and streaks, our model's assumption of memorylessness may oversimplify certain dynamics. Lastly, while our analysis was based on historical data, future improvements could involve incorporating real-time player performance trends to refine predictions further. Such data could train a predictive model that has a higher accuracy as players are often traded, and thus the team's snyergy could change year to year and impact player performance.

### 4.6 Limitations

Building upon these findings, future iterations of this model could integrate additional factors such as defensive efficiency, pitcher performance, and situational batting statistics. Furthermore, expanding the analysis to include more teams and longer time spans could enhance the generalizability of the results while also analyzing short spans of time to evaluate season-by-season. Advanced statistical techniques, such as reinforcement learning or Bayesian updating, could also be further explored to dynamically adjust transition probabilities based on evolving team and player performance.

## 5 Conclusion

This study demonstrates the effectiveness of Markov Chain modeling in baseball simulations. The transition matrices provide a powerful tool for understanding game dynamics and player contributions. As a whole, I would want to go back to this project and conduct further filtering of the data to ensure I avoid the errors I encountered.

## Acknowledgments

The student researcher would like to thank Dr. Gene B. Kim and Dr. Julia Palacios for guidance on Markov Chains and baseball analytics. Additional thanks to Retrosheet for providing play-by-play data, and Lucas Calestini for highlight the novelty of pursuing this idea.


## References


1. Retrosheet, "Play-by-Play Baseball Data," `https://www.retrosheet.org/`.
2. Sports Analytics. "The Elegance of Markov Chains in Baseball." Medium, 2021. `https://medium.com/sports-analytics/the-elegance-of-markov-chains-in-baseball-f0e8e02e7ac4`.
3. Baseball-Reference, "MLB Statistics and History," `https://www.baseball-reference.com/`.
4. FanGraphs, "Advanced Baseball Analytics," `https://www.fangraphs.com/`.
5. Retrosheet Tools, "Baseball Event File Processing," `https://www.retrosheet.org/tools.htm`.
6. Ryan Frey. "Baserunning: Should You Send the Runner?" Medium, 2022. `https://rfrey22.medium.com/baserunning-should-you-send-the-runner-ead23c6229df`.
7. Julia Palacios, STATS 217 Class Notes, Stanford University, 2024.
8. Gene Kim, Math 104 Class Notes, Stanford University, 2024.
9. Data Processed using R.