Optimizing Batting Orders: Monte Carlo Game Simulation Based on Batter Swing Clustering

Team: Yonsei Blues

Participants: Juwon Lee(Captain), Yugyung Kim, Jiyong Lee

Yonsei University, Korea

I. Introduction

Swing Length and Bat Speed are relatively new concepts introduced to the baseball statistics community. However these metrics have quickly emerged as essential tools for understanding a batter's swing dynamics as they are believed to have a direct correlation with the Launch Angle, Launch Speed, and ultimately the landing point of the batted ball. This growing significance and potential of Bat Speed and Swing Length led us to question and explore whether it is possible to classify swing tendencies of batters using Bat Speed, Swing Length, and other swing-related characteristics.

If swing tendencies/characteristics of batters show a close relationship with their performance outcomes, it could provide a basis for evaluating a batting order's potential to score runs as a team. As a batting order plays a significant role in a team's scoring ability, we aimed to analyze how the construction of batting orders based on swing tendencies/characteristics of batters help to identify the team's run-scoring capacity. This analysis could also be used to suggest the most suitable batting spot for each player, as well as identify the optimal swing type for specific batting positions. Accordingly, we decided to classify batters' swings using Bat

Speed and Swing Length, analyze their tendencies through clustering results, and conduct a Batting Order Simulation to evaluate its feasibility.

II. Method

2.1 EDA

First, we treated switch hitters as two distinct types of batters depending on whether they batted left-handed or right-handed. For example, when the switch hitter “Tommy Edman” batted right-handed, his player_name was labeled as "Tommy Edman-R," and when he batted left-handed, it was labeled as "Tommy Edman-L."

Next, we extracted columns from the dataset representing batters’ characteristics and conducted a One-Way ANOVA test to identify variables that showed statistically significant differences across player_name. One-Way ANOVA is a statistical method used to compare the means of three or more groups to determine if there are any statistically significant differences exist among them. The results of the One-Way ANOVA revealed that bat speed and swing length had significantly higher F-statistics compared to other variables under a p-value threshold of < 0.05. Based on these results, we concluded that bat speed and swing length can be considered intrinsic characteristics of individual batters.

Table 1.

One-Way ANOVA Results

Variable	F-Statistic	P-value
bat_speed	54.627	< 0.05
swing_length	93.253	< 0.05
launch_speed	4.914	< 0.05

launch_angle	4.220	< 0.05

The difference in launch angle was also statistically significant, but its magnitude was not substantial enough. As a result, we decided to calculate the attack angle to better characterize batters' swing tendencies.

2.2 Attack-Angle Calculation

We used the concept of 'attack angle' instead of launch angle to more effectively classify swing characteristics of batters, alongside swing length and bat speed. While launch angle depends greatly on a batter's aim and timing, which are factors without any direct measurements [1], attack angle shows the natural swing plane when the batter makes contact with a good pitch, providing a useful metric for analyzing the mechanics and performance of swings.

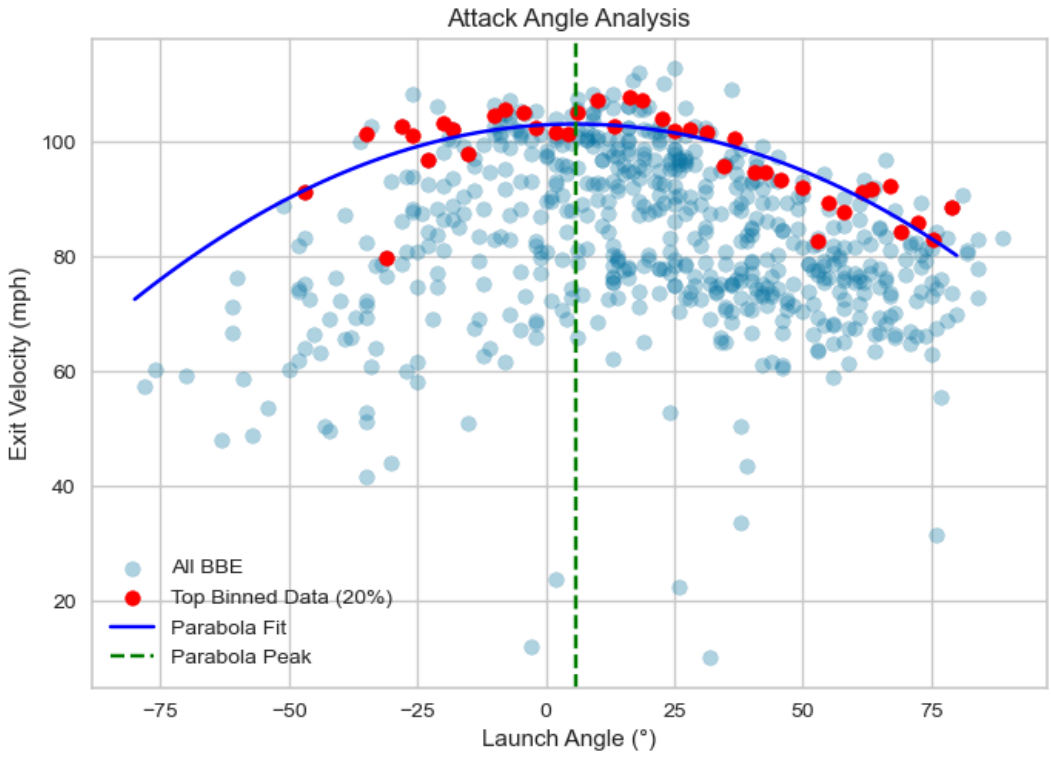
The calculation of attack angle is based on exit velocity (EV) and launch angle (LA) measured for each batted ball event (BBE), proposed by Marshall, D. (2017). We determined the attack angle by analyzing the launch angle that occurs when the batter has maximum exit velocity. The process begins by dividing LA data into bins and selecting the top 20% of EVs within each bin to focus on the most impactful events. A parabola is then fitted to these high-EV points, and attack angle is found at the peak of the parabola where EV is maximized [1].

There are two ways to estimate the value of attack angle, and the first is by fitting the parabola based on the LA with the highest EV. The second method is to average the LAs of the top 15 batted balls with high-EVs. To improve accuracy, the final attack angle is calculated by

averaging the results from both methods, offering a reliable representation of the batter's natural swing plane.

Figure 1.

Attack Angle Plot



Note. We grouped data based on LA with size 3º and extracted the top 20% batted balls. Then we fitted them to a parabolic curve in the form of a quadratic function.

For this analysis, we calculated attack angles of players who produced at least 30 batted balls, ensuring sufficient data for reliable estimation. Calculated attack angles were then used as a third major variable to characterize and classify the swing types of the batters for clustering.

2.3 Plate Discipline Stats

Up until this point, we had variables such as swing length, bat speed, and attack angles. While these variables were sufficient to describe the physical characteristics of a batter's swing, we thought they were not enough to fully capture the batter's characteristics. Therefore, we decided to incorporate plate discipline statistics, which reflects a batter's approach like batting eye and aggressiveness at the plate. We obtained the relevant data from Fangraphs' plate discipline dashboard. Among the available data, we focused on swing-related stats that could provide further insight into the batters' swing tendencies. Ultimately, we decided to add the following four features: Z-Swing%, OZ-Swing%, Z-Swing%/OZ-Swing%, and Swing%. For switch hitters, there were no specific stats to differentiate plate discipline between their left-handed and right-handed batting, so we assumed that a switch hitter's plate discipline remains consistent regardless of which side of the plate they bat from.

2.4 K-Means Clustering

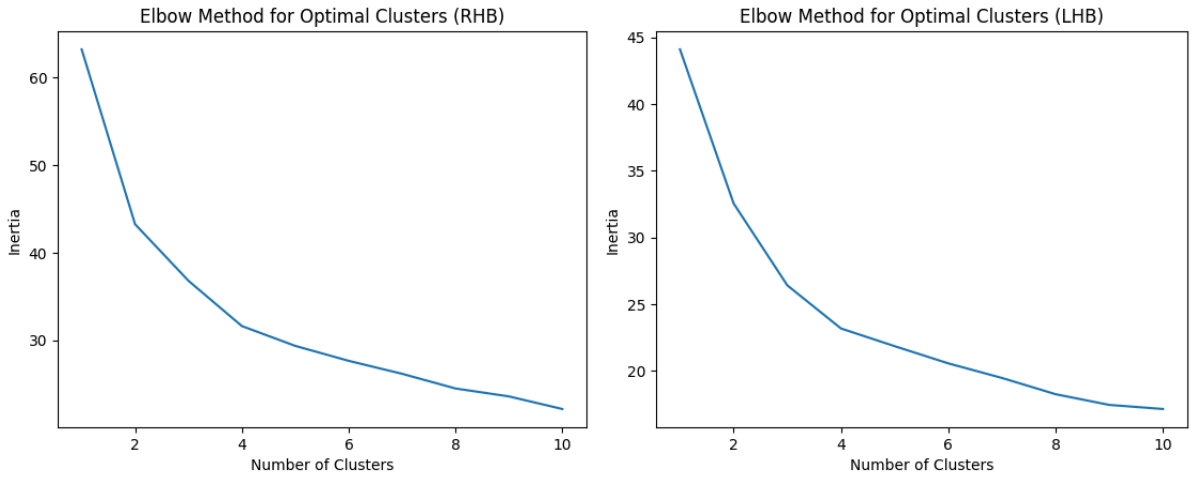
We conducted K-Means Clustering based on features that could distinguish players' swings, including attack angle, bat speed, swing length, Z-Swing%, OZ-Swing%, Z-Swing%/OZ-Swing%, Swing%. K-Means clustering is an unsupervised machine learning algorithm that groups data into K-clusters. It is used to identify patterns or structures in unlabeled data by grouping similar data points into clusters. Although we considered other clustering methods like Kohonen's SOM, K-Means clustering produced the most reasonable and clear results.

During the calculation of attack angles, 621 batters met the criteria for calculability, and clustering was performed on these players. First, we separated left-handed and right-handed

batters and performed clustering for each group. The number of clusters was determined by using the Elbow Method. The Elbow Method examines how the Sum of Squared Errors (SSE) or inertia decreases as the number of clusters increases. The idea is to identify a point where the decrease in SSE slows down significantly, forming an "elbow" in the graph. This point is considered to be the optimal number of clusters.

Figure 2.

Elbow Method for Optimal Clusters (RHB/LHB)



Using the Elbow Method, we determined that four clusters were optimal for classifying the swings of both RHBs and LHBs, resulting in a total of eight clusters. The clusters for RHBs were labeled as 1, 2, 3, and 4, while those for LHB were labeled as 5, 6, 7, and 8.

Table 2.

Right-Handed Batters' Mean Batting Summary Statistics

Cluster	**Bat Speed**	**Swing Length**	**Attack Angle**	**AVG**	**OBP**	**SLG**	**OPS**	**wRC+**
1(RHB-1)	69.464	6.973	13.358	0.225	0.312	0.346	0.658	88.889
2(RHB-2)	69.169	7.000	11.414	0.228	0.286	0.340	0.626	76.775

3(RHB-3)	71.301	7.298	13.237	0.233	0.276	0.369	0.645	78.968
4(RHB-4)	72.901	7.514	15.713	0.234	0.307	0.404	0.711	99.823

Note. We included only the key statistics here. For other advanced stats like Z-Swing%, O-Swing%, please refer to the code.

Table 3.

Left-Handed Batters' Mean Batting Summary Statistics

Cluster	**Bat Speed**	**Swing Length**	**Attack Angle**	**AVG**	**OBP**	**SLG**	**OPS**	**wRC+**
5(LHB-1)	71.011	7.201	15.923	0.234	0.290	0.384	0.674	88.121
6(LHB-2)	72.270	7.343	15.443	0.236	0.318	0.409	0.727	104.757
7(LHB-3)	68.767	6.954	18.003	0.218	0.324	0.347	0.671	95.122
8(LHB-4)	67.018	6.652	12.368	0.233	0.298	0.334	0.632	79.421

Note. We included only the key statistics here. For other advanced stats like Z-Swing%, O-Swing%, please refer to the code.

Clustering analysis revealed an intriguing finding: there is a certain degree of correlation between the classification of batters' swings and their performance. Let us first examine right-handed batters. The four clusters of RHBs could be broadly divided into two groups. Clusters 1 and 2 consisted of players with relatively slower bat speed and shorter swing length, while Clusters 3 and 4 included players with faster bat speed and longer swing length.

Cluster 1 had the lowest batting average but the highest on-base percentage, indicating excellent plate discipline. These batters were defined as "Disciplined Line Drive Hitters (R)." Cluster 2 exhibited a high Swing% and Contact%, showing aggressive tendencies and produced a large number of in-play balls. However, due to a high proportion of ground balls, their BABIP was relatively low, leading to lower productivity. These players were categorized as

"Contact-Oriented Ground Ball Hitters (R)." Cluster 3 batters showed little selectivity, swinging at a wide range of pitches and showing limited plate discipline. These very aggressive batters were defined as "Aggressive All-Field Hitters (R)." Cluster 4, among RHBs, exhibited the largest attack angle, longest swing length, and fastest bat speed. Their OPS and wRC+ were also excellent, making this the most productive group of batters. These players were defined as "Powerful Pull-Hitters (R)."

Next, similar to RHBs, LHBs could also be roughly divided into two groups. Clusters 5 and 6 were characterized by relatively fast bat speeds and long swing lengths, while Clusters 7 and 8 were defined by slower bat speeds and shorter swing lengths.

Cluster 5 batters were aggressive hitters with good slugging power. However, they had the worst plate discipline among the LHB clusters. While their fly balls were very likely to become home runs, their high ground ball rate resulted in low overall productivity. These batters were classified as "Aggressive Power Hitters (L). Cluster 6 showed decent plate discipline and excelled in most performance metrics. They produced many fly balls and hard-hit balls, increasing the likelihood of hits when the batted ball was in play. These batters were defined as "Elite All-Around Sluggers (L)." Cluster 7 showed great plate discipline overall and was generally very patient and cautious at the plate. While they had a low batting average, their high on-base percentage made them quite productive despite the low BA. These players were classified as "Patient Line-Drive Hitters (L)." Cluster 8, on the other hand, had an aggressive swing but lacked plate discipline. Despite their aggressive tendencies, they struggled to generate extra-base hits due to the small attack angle, short swing length, and slow bat speed, leading to overall low performance. These batters were categorized as "Contact-Oriented Ground Ball Hitters (L)."

Both LHB and RHB achieved a silhouette score of approximately 0.21, and considering the diverse distribution of baseball data, it was determined that the clustering performed well for the data.

2.5 Monte Carlo Simulation

Baseball is fundamentally a sport based on statistical probabilities, involving a multitude of variables such as batting average, on-base percentage, home-run probabilities, and strikeout-rates. Monte Carlo Simulation simplifies these intricate probabilistic calculations by repeatedly simulating events like at-bats, innings, or entire games using random sampling. We aimed to simulate the run production per game of certain batting orders based on our batting cluster results. Additionally, we decided to explore which combination of batting lineup and optimal swing cluster configurations would result in the highest number of runs.

First, we defined the possible events that could occur in baseball, which consisted of 17 categories. Since the provided Statcast data did not include event indices like Long Single or Short Double, we split the Single, Double, and Fly Out events based on statistical facts. According to [4], 30% of singles are classified as long singles, 50% as medium singles, and 20% as short singles. Additionally, 80% of doubles are short doubles, while 20% are long doubles. As for fly outs, 20% are classified as long fly outs, 50% as medium fly outs, and 30% as short fly balls.

We also added certain assumptions for the events to construct the state transition matrix, runs matrix, and outs matrix. Using domain knowledge, such as the high likelihood of playing the infield-in shift with bases loaded and no outs, we assumed the most plausible scenarios for increasing out counts and runner movement.

Table 4.

Events List

Event Number	Event Name
1	DoublePlay
2	Error
3	GroundOut
4	HBP/CatInt
5	HR
6	K
7	LineOut/InfFly
8	Triple
9	Walk
10	LongSingle
11	MediumSingle
12	ShortSingle
13	ShortDouble
14	LongDouble
15	LongFly
16	MediumFly
17	ShortFly

Note: HBP/CatInt includes Hit-by-Pitch, and Catcher's Interference. We assumed that these two events produce the same outcome in most cases and combined them into a single event.

We needed four matrices to simulate a baseball game. First, we constructed the "batting statistic matrix", which presents the average probabilities of batting events occurring for 16

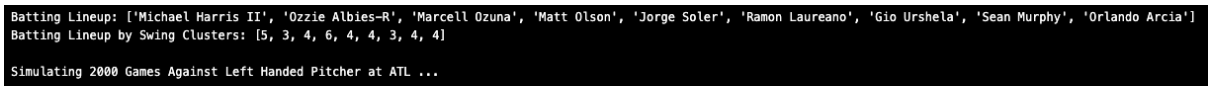
different player types (8 batter clusters × LHP/RHP). Next, we created the "state transition matrix," which shows how the game state (coded by outs and base runners) changes after a particular batting event. For example, when a ground out event occurs, the state *1001* (1 out, runner on third) transitions to *2000* (2 outs, no runners) as the runner scores and the batter is out, leaving no runners on base. The third matrix is the "runs matrix" which records the number of runs increased after each batting event. Specifically, if a ground out occurs in the state *1001*, the corresponding value in the runs matrix would be 1. Similarly, the "outs matrix" tracks how the number of outs increases with each batting event: in the same example, it would record 1 additional out.

Then, we performed a Monte Carlo Simulation. First, we input the names of nine batters, separated by commas, mapped the input to their respective batter clusters, and conducted the simulation based on the event probabilities for each cluster (as described in the batting static matrix). During this process, park factors were also considered. The park factor in MLB is a statistical metric used to measure how a specific ballpark influences scoring and offensive production. We used park factor data based on the last three years (2022–2024). We multiplied the park factor by the 'runs matrix' to ensure that the run occurrence weights for each ballpark were reflected in the simulation. Additionally, we defined two additional scenarios for the simulation. The first scenario is to test which batting order position would be most appropriate for a new player, such as a recruit, when the lineup of nine players already exists. The second scenario focuses on determining which cluster of batters should be placed in a specific batting order position to maximize the team's run production, given the lineup of nine players too.

III. Result

Figure 3.

Input data



note. We simulated the Atlanta Braves' batting order against a left-handed pitcher (LHP). This lineup was taken from the game on September 30th against the New York Mets (3-0).

Figure 4.

Atlanta Braves' Batting Order Simulation

Figure 5.

Best Batting Position for New Player(Ha-Seong Kim) at Atlanta Braves vs. LHP

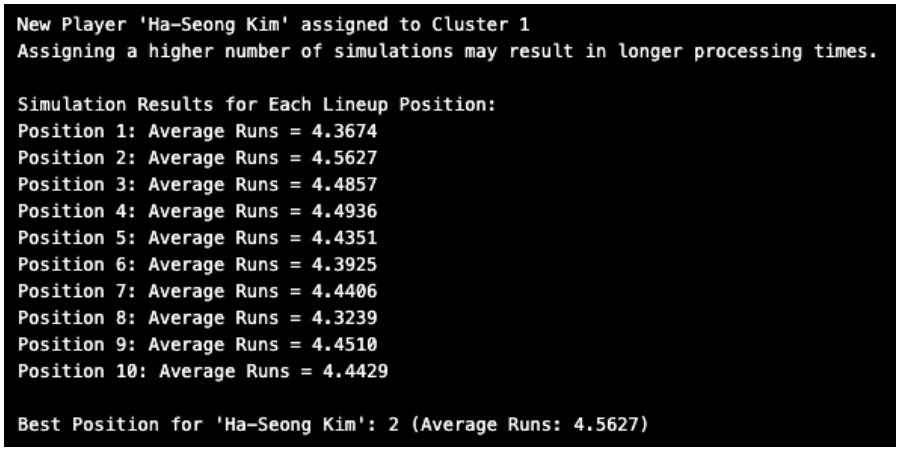
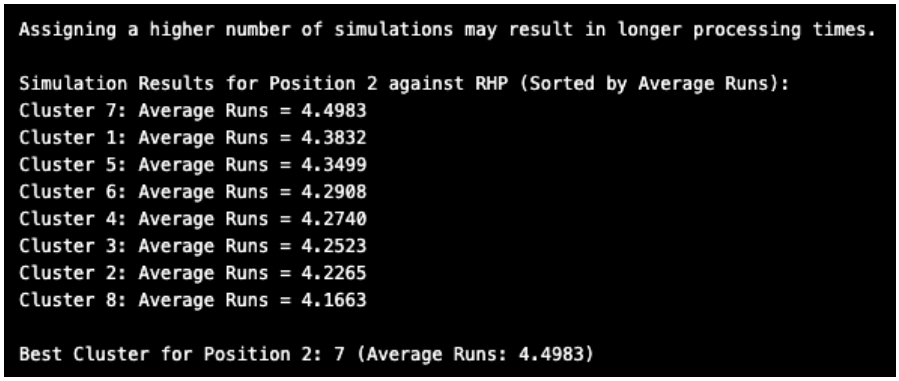


Figure 6.

Best Cluster for Batting Position 2



note. This indicates that Cluster 7 is the ideal cluster for the second batting position.

The average runs scored by all MLB teams in the 2024 season was 4.39 according to Baseball Reference. After performing the 9-inning simulations multiple times, we confirmed that the outputs closely matched this average.

IV. Discussion

4.1 Conclusion

We aimed to determine whether classifying batters based on swing tendencies, including their physical characteristics and decision-making ability to swing, is appropriate. Additionally, we sought to verify whether bat speed and swing length are indicative of a batter's performance. Initially, we observed a relatively high correlation between bat speed and swing length. Players with high bat speed and long swing length are more likely to produce fast and powerful hits, while players with low bat speed and short swing length may not generate ideal hits but often exhibit high plate discipline due to their excellent adaptability. However, there are also "elite" players with high bat speed and long swing length who also possess great plate discipline.

Therefore, we concluded that clustering batters' swing tendencies based on both their physical characteristics and plate discipline features is an appropriate approach.

Based on the results of the Monte Carlo simulation model using batter swing clusters, which closely resemble actual scoring data, we could conclude that our batter clustering approach has shown promise. This model can further be leveraged to develop new player signing strategies in the baseball industry and offer valuable insights into potential adjustments to existing batting orders. For example, recent news reports suggest that Ha-Seong Kim could potentially sign with the Atlanta Braves. If so, as shown in Figure 4, this model could help the Braves analyze and find the ideal spot in the batting order for Kim if he were to join the team.

4.1 Limitations

Due to our computational limitations, we were unable to calculate all possible cluster combinations for each batting order position (8^9 combinations in total). However, with sufficient computational resources, our model could be extended to evaluate the optimal cluster lineup by simulating the full range of combinations. Additionally, as batters were classified into eight distinct types in this paper, some discrepancies between the model and real-world data would be inevitable.

V. Application

In the R Shiny app, we incorporated the Monte Carlo simulation into the batting lineup, allowing users to run simulations based on their needs and goals. Designed especially for practical usability, the app aims to serve as an effective tool that could also be utilized by MLB teams. The app is divided into four main sections, starting with an overview of player characteristics in each cluster. Following that, three types of simulations are available: The first

allows users to input their own batting order (1–9) and calculates the average expected runs for that lineup. The second simulation suggests the optimal batting spot for a new player joining the team based on their fit in the existing lineup; and the third helps determine which clusters of batters would be best suited for each batting position.

VI. Reference

[1] Marshall, D. (2017, October 12). *Reverse engineering swing mechanics from Statcast data.* https://community.fangraphs.com/reverse-engineering-swing-mechanics-from-statcast-data/

[2] Freeze, R. A. (1974). *An Analysis of Baseball Batting Order by Monte Carlo Simulation. Operations Research*, *22*(4), 728–735. http://www.jstor.org/stable/169949

[3] Mongerson, K. (2023). *Explorations in baseball analytics: Simulations, predictions, and evaluations for games and players* (Publication No. 3192) [Doctoral dissertation, University of Wisconsin-Milwaukee]. UWM Digital Commons. https://dc.uwm.edu/etd/3192

[4] Winston, Wayne L, et al. *Mathletics: How Gamblers, Managers, and Fans Use Mathematics in Sports, Second Edition.* PRINCETION UNIVERSITY PRESS, 2022