MLBDecideR: A SHINY App for Baseball

Jason A. Osborne, Melody Wen, and Neil Dey

Many of the strategic decisions that managers have to make in Major League Baseball (MLB) can be informed by thoroughly quantitative arguments. When deciding whether to put a baserunner in motion, attempt a steal, or sacrifice bunt, a good decision requires consideration not just of the situation, but of the personnel. Virtually every such decision regarding a hitter and a baserunner should be made in light of not only (estimates of) their abilities, but of the hitters who are “on deck,” “in the hole,” and even all of the hitters who follow in the lineup, particularly if there are no outs or it is early in a game. We present a web application that carries out Markov chain calculations to assist with several classes of decisions. The app, available online at the URL go.ncsu.edu/mlbdecider, reports in tabular and graphical form an estimated probability distribution for the number of runs scored during the remainder of a Major League Baseball game. Users can select any of the thirty MLB teams from any of the 2014–2019 seasons. Summary statistics for players are displayed in a table called *Player Database*, and these are the primary source of information for estimating the transition probability matrix (TPM) of the chain. We focus on three tools that can assist a decision-maker before or during a game as well as any fans who wish to be Monday-morning quarterbacks after a game. The *Lineup Tool* enables users to select two lineups (players and the order in which they will hit) from any

single team for comparative purposes. The *Steal a Base Tool* enables users to investigate how the probability distribution of runs scored changes with the probability of a successful steal of any base except home. The tool calculates a theoretical threshold beyond which it makes sense to attempt the steal, using either estimated run expectancy or the estimated chance of scoring at least one run as criteria. The *Sacrifice Bunt Tool* makes a similar calculation to assist with the decision of whether it makes sense quantitatively to ask a weak, nonpitching hitter to try to bunt a baserunner into scoring position (second or third base), or whether they should be permitted to swing away, with a focus on the four most common bunting situations. These decisions require the user to input the situation in the game: inning, base occupation, number of outs, and the current batting order, as well as the choices of using summary statistics against a right- or left-handed pitcher and whether to use a hitter’s data from a given year or over an entire career.

1. Background

Many authors have used Markov chains to model the progress of a baseball game from one hitter to the next. Sokol [Sok04] focuses on runs expectancy and suggests using annual summary statistics for an individual in the lineup to estimate his corresponding TPM, even providing Matlab code for readers interested in their own implementation. Bukiet–Harold–Palacios [BHP97] develop an algorithm to carry Markov calculations for one inning forward across a full game, keeping track of the probability distribution of runs scored and place in the order at the start of innings subsequent to the first.

This algorithm enables us to work out the entire probability mass function for the random variable R representing the number of runs scored either for an entire game or for the remainder of a game when the user specifies another initial state. Ursin [Urs14] uses this algorithm with

Jason A. Osborne is a professor of statistics at North Carolina State University. His email address is jaosborn@ncsu.edu.

Melody Wen is a senior at the North Carolina School of Science and Mathematics. Her email address is mwen@mit.edu.

Neil Dey is a senior majoring in mathematics and computer science at North Carolina State University. His email address is ndey3@ncsu.edu.

Communicated by Notices Associate Editor Richard Levine.

For permission to reprint this article, please contact:
reprint-permission@ams.org.

DOI: <https://doi.org/10.1090/noti2153>

Figure 1. State space of the Markov chain for one inning. States are combinations of outs and base occupations, and are identified by a single digit for the number of outs followed by a three-binary-digit sequence for occupation of first, second, and third base. Several transitions between states are illustrated with arrows and labels.

2013 data and some important simplifying assumptions to simulate runs scored for the entire season and finds generally good agreement with the observed totals for all thirty teams. These assumptions include exclusion of certain events that have strategic importance for some of the decisions we consider. In particular, we combine league-wide play-by-play data and an individual's summary statistics to estimate the important probabilities of baserunners taking the extra base on hits, advancing on outs (either by sacrifice or when hitters are swinging away), grounding into double plays, and scoring by sacrifice fly.

The states of the chain are simply the combinations of the $2^3 = 8$ possible occupancies of the three bases other than home and the number of outs during the inning (0, 1, or 2), plus the additional absorbing state of three outs, for a total of $(8 \times 3) + 1 = 25$ states in the state space. We let states be denoted with four digits, the first of which indicates the number of outs, followed by three consecutive binary digits to indicate the occupancy of first, second, and third base, respectively. For example, a runner on second with nobody out, a prime candidate for the sacrifice bunt, is denoted $\boxed{0\ 010}$. The 24 nonabsorbing states are displayed using this notation in Figure 1. Several transitions among states are shown. For example, at the beginning of an inning, with no outs and no baserunners, the most probable transition, even for Babe Ruth, is to make an out and move to the state of one out and no runners on base. Alternatively, Babe is not unlikely to hit a home run, leading to a transition back to the state of no outs and no runners on base with a run scored. Many of the elements of

the transition probability matrix when the Babe is up can be estimated using only the empirical relative frequencies implied by either his career summary statistics or those for a given year. Other transitions that involve nondeterministic baserunner advancement on outs and some types of hits are obtained by combining individual summary statistics and league-wide information obtained by parsing play-by-play information freely available from the incredible historical archive at www.retrosheet.org. This type of estimation for certain situations is sketched out in the first of the next three sections, which correspond to the three tools in the app.

2. Lineup Tool

The default order of the players in our shiny app table entitled *Player Database* is descending by number of plate appearances and should be changed by the user for specific applications. Using these plate appearance ranks for the 2019 World Series Champion Washington Nationals, the opening day lineup of Eaton, Turner, Rendon, . . . , Scherzer, Robles can be specified by entering (8 11 7 13 1 10 6 17 15) in the field entitled "Batting Order A." An astute reader might note that the manager, Dave Martinez, has gone against convention by placing the pitcher in the eighth spot instead of ninth, a habit he may have learned from his mentor, Joe Madden, when they were together with the Chicago Cubs. The *Lineup Tool* enables the user to quickly consider the order obtained by switching the last-place hitter, Victor Robles, with the starting pitcher, Max Scherzer. This is accomplished by entering (8 11 7 13 1 10 6 15 17)

in the field entitled “Batting Order B.” The app will sequentially estimate transition probability matrices using batters in these two sequences provided by the user, then in turn work out the estimated probability mass function for the remainder of the game. This distribution can be seen in tabular form by clicking the “Table of run probabilities” tab or as a plot by clicking “Probability distribution of runs scored.” The right tail of this distribution can be collapsed by selecting a number smaller than 10 in the “Max runs in plot:” selector at the bottom left. Inspection of the reported means indicates that, from a run expectancy point of view, there is no obvious justification for batting Scherzer eighth instead of ninth, since the means are 4.192 and 4.204, respectively. Interestingly, if Robles is moved further up in the order to sixth, the estimated mean goes up to $\hat{E}(R) = 4.215$. This is because he is a better hitter in the sense of more bases per plate appearances and better on-base percentage than the players in the sixth, seventh, and eighth positions. Obviously, there are other things to consider, such as alternating the handedness of hitters in the lineup to cause problems for the opposing manager’s possible decision to change pitchers later in the game, but this tool provides a way to quantify and compare lineups.

For the most part, the summary statistics shown in the *Player Database* are used to estimate transition probabilities. Let us order the states so that the elements in the top row of the diagram in Figure 1 are “situations” 1 through 8, moving left to right. Similarly, the elements in the second and third rows are situations 9–16 and 17–24, respectively. The 25th situation is the absorbing 3-out state. For the lead-off hitter Adam Eaton, the first element of the TPM, $P_{1,1}$, is estimated using the empirical relative frequency of home runs in his career, $\hat{P}_{1,1} = 52/2614 = 0.0199$, since the only event in an at-bat from state $\boxed{0\ 000}$ with no outs and no runners on base that could result in situation #1 again is a home run. Similarly, the next element in the TPM, the probability of moving from $\boxed{0\ 000}$ to $\boxed{0\ 100}$, can be estimated by summing the relative frequencies of walks and singles, $\hat{P}_{1,2} = (248 + 454)/2614 = 0.2686$. (In our first of many very small departures from reality, we neglected to consider the possibility of a four-base error in $P_{1,1}$ or of reaching base by error or by hit-by-pitch in $P_{1,2}$.) While this simple transition probability is estimated using only data for Adam Eaton, other transition probabilities with runners on base involve borrowing information from plate appearances from other players. For example, suppose Eaton comes to bat with no outs and a baserunner on first base, state $\boxed{0\ 100}$. Suppose he hits a single. How do we estimate the chance that the baserunner advances to second or third, or even scores all the way from first?

We use play-by-play data for the 2019 season available from Retrosheet and functionality introduced in [MAB18]. We observe that among the 1,576 instances where a single was hit in situation #2, the baserunner advanced to second with relative frequency $1160/1576 = 0.736$. We then “distribute” the probability of Eaton getting a single in the situation to the states $\boxed{0\ 110}$ and $\boxed{0\ 101}$ with conditional probabilities 0.736 and $1 - 0.736$, leading to the TPM estimates $\hat{P}_{2,5} = (248 + 0.736 * 454)/2614$ and $\hat{P}_{2,6} = (0.264 * 454)/2614 = 0.0459$. This argument requires the unreasonable assumption that the conditional baserunner advancement probabilities we are estimating are constant over hitters and baserunners. However, to estimate specific probabilities without this type of “smoothing,” we would have to limit ourselves to instances where Eaton came to bat in situation #2. This would severely restrict information and lead to high variability in the estimation of the TPM, not to mention the amount of parsing of play-by-play data that would be necessary. We make similar adjustments by borrowing information from other hitters to estimate baserunner advancement probabilities on doubles, triples, and on generic outs (nonstrikeout).

Suppose there are fewer than two outs. If we wish to estimate a player’s laudable ability to drive a runner home from third or a player’s lamentable tendency to hit into a double play when a runner is on first, we can use the raw frequencies of these events, labeled in the *Player Database* as “SF” and “GDP,” respectively. However, we can no longer use plate appearances, “PA,” as the denominator because these events can occur only in certain subsets of all of a player’s plate appearances. While summary statistics that are “split” according to these situations are available, the number of plate appearances in each split dwindles rapidly, particularly when using only one year of data. We estimate the chance of a successful sacrifice by down-scaling the plate appearances proportionally to the relative frequency of a sacrifice situation across the entire league. For state $\boxed{1\ 001}$, the observed frequency in 2019 was $9232/186517 = 0.0495$. For Eaton to bring that player home via the sacrifice fly, we estimate the probability using an “effective” or “estimated” sample size of $0.0495 \times 2614 = 129.4$ with $\hat{P}_{12,17} = 10/129.4 = 0.077$. (The success rate across the 2019 season was about 0.12.) We use a similar approach to smooth the estimated probability of grounding into a double play.

3. Steal a Base Tool

Another important decision for the manager is whether to allow a speedy baserunner to attempt to steal a base. Consider the Nationals opening day lineup again. Suppose that Victor Robles, batting ninth, has reached first base in

Figure 2. Inputs for the *Steal a Base Tool*, including inning, out, position in the lineup, and states after successful or failed stolen base attempt.

the fourth inning with the top of the lineup coming to bat with no outs. This situation can be specified by simply clicking the *Steal a Base Tool* tab near the top of the toolbar on the left, then choosing “4” for the “What Inning?” and “Runner on 1st” and “Baserunners” input selectors in the left-hand toolbar. Now, to provide the input for the event where Robles succeeds in swiping second base, we simply choose “Runner on 2nd” in the “Baserunners” selector in the column entitled “Situation if successful” and “Bases empty” under “Situation if not successful.” We also take care to change “Outs:” to 1 and both “What Inning?” inputs to “4.” How speedy is Victor Robles? This information can be provided by dragging the “Pr(SB)” stylus left and right. All of these inputs are displayed in Figure 2. The default value 0.72 was calculated using 2019 stolen base totals across both leagues. Robles has stolen 17 bases and been caught 6 times, for a success rate of 0.74, very close to the league success rate. Though MLB does not include “picked off” as caught stealing [TLD07], this is not an unreasonable estimate. The app reports the estimated probability distribution of runs scored in the remainder of the game if (1) he does not attempt to steal, (2) he successfully steals, (3) he gets caught. Additionally, the marginal distribution is computed by averaging over (2) and (3) using the $Pr(SB)$ specified by the user. If the manager elects not to send the runner, the run expectancy for the remainder of the game is 3.252 runs. If Robles gets the green light and succeeds, it goes up a little to 3.424, but if he gets caught it goes down considerably to 2.606. Averaging over these two possibilities using weights $Pr(SB) = 0.74$

and $P(CS) = 0.26$ gives the marginal mean of 3.117, which is lower than the mean if he does not attempt to steal at all. The threshold for $Pr(SB)$ beyond which the marginal mean, $E(R) = Pr(SB)E(R|SB) + (1 - Pr(SB))E(R|CS)$, exceeds the current mean, $E(R|no\ attempt)$, is also reported along with the same threshold in terms of $P(R > 0)$. Additionally, the app provides all four probability mass functions for R both in tabular and graphical format. Lastly, interactive plots of $E(R|attempt)$ and $Pr(R > 0|attempt)$ against $Pr(SB)$ are provided for the more visually oriented decision-maker. These plots have horizontal reference lines for the successful/unsuccessful/no attempt scenarios and vertical reference for the user-inputted $Pr(SB)$. All of the four visualizations produced by the *Steal a Base Tool* are shown in Figure 3. In the app, the plots are interactive, so that when the user hovers over any point a floating box appears with appropriate numerical values and conditions.

To consider the possibility that Mr. Robles has a better chance to succeed than is indicated by his current rate of success, maybe because the particular catcher they are facing has a weak arm, or the pitcher has a slow delivery to the plate, the user can either optimistically drag the stylus to the right, or look to the right of the vertical reference on these plots. The threshold is the point on the horizontal axis where the slanted line for the marginal mean crosses the horizontal line for the mean given no attempt, which is $Pr(SB) = 0.789$ for Robles using the opening day lineup, with Eaton and Turner coming up. The next several hitters that follow the baserunner in the lineup are crucial in the decision about whether to steal. As an experiment,

Figure 3. Visuals from the *Steal a Base Tool*: (a) $P(R = r)$ conditionally on no attempt, successful steal, caught stealing, or picked off; (b) $P(R = r)$ with and without attempted steal; (c) $E(R)$ vs. $P(SB)$; (d) $P(R > 0)$ vs. $P(SB)$. All are calculated for a theoretical game involving the Nationals opening day roster, assuming the runner Robles, batting from the sixth spot against a right-handed hitter, reached base with no outs.

suppose we move Robles from the last spot in the order to the fifth spot, where if he reaches base, he will be followed by the weaker hitting Gomes, Dozier, and Scherzer. We can conduct this experiment just by changing the input to the “Batting order” field from (8 11 7 13 1 10 6 17 15) to (10 6 17 8 11 7 13 1 15). Note that the threshold for recommending a stolen base attempt drops all the way from 0.789 to 0.667, as shown in Figure 3. This type of quantification provides a rationale for putting faster but weaker hitters lower in the lineup, or giving the occasional green light to middle-order hitters of modest base-stealing ability should they reach. The app is quite flexible and allows users to evaluate this probabilistic utility of the stolen base for almost any situation. (It does not yet accommodate attempts to steal home.)

4. Sacrifice Bunt Tool

The sacrifice bunt seems to be falling out of fashion in MLB even more rapidly than the stolen base attempt. Consider situation #2, state $\boxed{0\ 100}$: a runner on first base and nobody out. The idea behind the sacrifice bunt is to exchange an out for a base. If executed successfully, the runner moves to second and becomes more likely to score. The chain goes from state #2 to #11 (one out, runner on second). If the attempt fails it goes to #10, $\boxed{1\ 100}$. The quantitative argument against the sacrifice bunt is that while the chance of scoring one additional run may have increased, $P(R > 0 | 1\ 010) > P(R > 0 | 0\ 100)$, the chance for a big inning has been squashed, $E(R | 1\ 010) < P(R > 0 | 0\ 100)$, though of course these probabilities all depend on personnel. If we let p denote the chance of a successful bunt attempt, the marginal mean rarely exceeds the

Situation	Year		
	2009	2014	2019
0 110	$\frac{419}{2580} = 0.162$	$\frac{374}{2488} = 0.150$	$\frac{239}{2500} = 0.096$
0 010	$\frac{499}{3274} = 0.152$	$\frac{360}{3104} = 0.116$	$\frac{200}{3106} = 0.064$
0 100	$\frac{997}{10027} = 0.099$	$\frac{846}{9662} = 0.088$	$\frac{405}{9697} = 0.042$
1 100	$\frac{155}{11584} = 0.013$	$\frac{153}{11334} = 0.014$	$\frac{132}{11798} = 0.011$
Total	$\frac{2070}{27465} = 0.075$	$\frac{1733}{26588} = 0.065$	$\frac{976}{27101} = 0.036$

Table 1. Declining use of the sacrifice bunt for each of the four major sacrifice situations.

mean with no attempt, even for p as large as 1: $E(R) = pE(R|1\ 010) + (1 - p)E(R|1\ 100) < E(R|0\ 100)$. Even if the attempt can be assured of success, run expectancy has been decreased by the out. This argument may have taken hold in recent years. By parsing recent play-by-play data, it can be seen that among plate appearances in the four most frequent bunting situations, the proportion where the bunt was attempted is now half what it was ten years ago, as shown in Table 1.

However, this thinking is confined to two possible outcomes from an attempt: success or failure. In reality, many things can and often do happen when a bunt is attempted. In a mid-2019 game between the Baltimore Orioles and Oakland A's the chain went from state #2, $\boxed{0\ 100}$, to state #4, $\boxed{0\ 001}$, on a failed sacrifice bunt attempt to the pitcher, who fielded the bunt but threw it into right field, allowing the baserunner to come around and score all the way from first, with the bunter reaching third. This type of event happens with nonnegligible frequency. Tango-Lichtman-Dolphin [TLD07] study bunting strategy in considerable detail, focusing on nonpitching hitters, and declare a sacrifice to have been attempted if a bunt was attempted on any of the pitches in the plate appearance. The *Sacrifice Bunt Tool* in our app is based on this approach, treating the entire plate appearance as the sampling unit, rather than investigating outcomes pitch-by-pitch. The number of bunt attempts for most batters is very small, making estimation of all the different possible outcome probabilities difficult. Instead, we adopt the notion of a universal bunter, whose transition probabilities we estimate separately for each of the four major bunting situations. These estimates use 2019 data and pool over all

non-hitting pitchers and all instances where a bunt was attempted during the plate appearance.

For state #2, $\boxed{0\ 100}$, we observe that in 3% of bunt attempts, the bunter reaches base safely and the baserunner reaches second base and in an additional 3% of attempts, the runner even reaches third! Fielding sacrifice bunts is not easy. To compare the two strategies of bunting and swinging away, we replace the appropriate row of the hitter's TPM by that of the universal bunter (and then restore it for the remainder of the game). The tool then reports again the entire probability distribution under the two strategies, along with separate boxes to compare $E(R|\text{swing away})$ vs. $E(R|\text{bunt attempt})$ and similarly for $P(R > 0)$. Note that the replacement row of the TPM has many nonzero entries, which is a different approach than the *Stolen Base Tool* which considers only two possible outcomes from an attempt. If we consider the aforementioned game with the Orioles facing the right-handed Mike Fiers, the game was in the second inning, Pedro Severino had reached first base, and young Rio Ruiz was at the plate. To input this information, we first select the *Sacrifice Bunt Tool* from the dashboard at the top left of the page, then choose the Orioles 2019 team, with the default "career" statistics against "RHP." To supply that spot in the batting order, ranks from the *Player Database* are used with the string (9 11 1 16 3 13 5 15 12) supplied because Ruiz is up with Severino now batting last. With career statistics well below those of an average player, the young Ruiz is estimated to be such a poor hitter that his impact on the probability distribution of runs is worse than that of the universal bunter, as evidenced by the lower run expectancy ($E(R|\text{swing}) = 3.478$ and $P(R > 0|\text{swing}) = 0.893$) compared to when he is bunting ($E(R|\text{sac.att.}) = 3.487$ and $P(R > 0|\text{sac.att.}) = 0.896$). These are small differences, but it is not difficult to find cases where the evidence in support of bunting is more pronounced. Consider the lineup of the weak-hitting 2019 Miami Marlins used in a late season game against right-handed pitcher Zach Eflin. They had Jon Berti leading off, followed by light-hitting shortstop Miguel Rojas. Suppose Berti reached to lead off the ninth with the score tied, bringing up Rojas, for whom the app reports that $P(R > 0|\text{swing}) = 0.349$ and $P(R > 0|\text{sac.att.}) = 0.378$. This is an improvement of almost 10%, and $E(R)$ improves as well. This information is displayed in Figure 4, which is a screen capture from the *Sacrifice Bunt Tool*. It was obtained by supplying the ranks for the 2019 Marlins against right-handers in the lineup field: (5 4 2 7 19 12 16 29 14).

Figure 4. Using the *Sacrifice Bunt Tool* to assist the 2019 Miami Marlins in deciding whether to bunt using their lineup from a game against the Phillies on September 28, 2019. The situation uses their actual batting order and assumes the lead-off runner Jon Berti has reached first base with no outs in the ninth inning and light-hitting Miguel Rojas at the plate.

5. Conclusion

Optimal decisions in baseball should be made in light of personnel. If we can imagine cloning a base-stealer and putting him on two different teams in the same batting situation, the decision whether he should attempt a stolen base should depend on who follows the runner in the lineup. In general, if the lineup is weak, it makes more sense to attempt a stolen base than if the lineup is strong. Our app enables users to specify different lineups and quantify the different outcomes, thus assisting in decisions. We have relaxed many common assumptions, such as allowing baserunners to occasionally advance an extra base on balls in play, but other simplifications remain and could be improved. An example is to allow these conditional probabilities for extra base advancement to depend on outs and even pitcher handedness, since left-handed pitchers are believed to be more effective at holding runners on first base because of the physical advantage of being able to face the runner prior to and during pitch delivery. A more substantial improvement could be brought about by incorporating a more principled approach to estimation of transition probabilities by pooling information from individuals and across the leagues. Decision-making in baseball is not easy. No manager wants to commit an error. Markov chains and visualizations of their application might be able to provide an assist.

References

- [BHP97] Bruce Bukiet, Elliott Rusty Harold, and José Luis Palacios, *A Markov chain approach to baseball*, Operations Research 45 (1997), no. 1, 14–23.
- [MAB18] M. Marchi, J. Albert, and B. Baumer, *Analyzing Baseball Data with R, Second Edition*, CRC Press, 2018.
- [Sok04] Joel S. Sokol, *An intuitive Markov chain lesson from*

baseball, INFORMS Transactions on Education 5 (2004), no. 1, 47–55.

[TLD07] Tom M. Tango, Mitchel G. Lichtman, and Andrew E. Dolphin, *The book: Playing the percentages in baseball*, Potomac Books, Inc., 2007.

[Urs14] Daniel Joseph Ursin, *A Markov model for baseball with applications*, Master's Thesis, 2014.

Jason A. Osborne

Melody Wen

Neil Dey

Credits

All figures are courtesy of the authors.

Photo of Jason A. Osborne is courtesy of Becky Kirkland.

Photo of Melody Wen is courtesy of Melody Wen.

Photo of Neil Dey is courtesy of Neil Dey.