



A Whole New Ballgame: Evaluating the Effects of Major League Baseball's 2023 New Rules Using Statistical Modeling

Citation

Sands, Bridget Theresa. 2024. A Whole New Ballgame: Evaluating the Effects of Major League Baseball's 2023 New Rules Using Statistical Modeling. Bachelor's thesis, Harvard University Engineering and Applied Sciences.

Link

<https://nrs.harvard.edu/URN-3:HUL.INSTREPOS:37379995>

Terms of use

This article was downloaded from Harvard University's DASH repository, and is made available under the terms and conditions applicable to Other Posted Material (LAA), as set forth at

<https://harvardwiki.atlassian.net/wiki/external/NGY5NDE4ZjgzNTc5NDQzMGIzZWZhMGFIOWI2M2EwYTg>

Accessibility

<https://accessibility.huit.harvard.edu/digital-accessibility-policy>

Share Your Story

The Harvard community has made this article openly available.
Please share how this access benefits you. [Submit a story](#)

A Whole New Ballgame:

Evaluating the Effects of
Major League Baseball's 2023 New Rules
Using Statistical Modeling

Bridget Theresa Sands

Presented to the Department of
Applied Mathematics in partial
fulfillment of the requirements
for a Bachelor of Arts degree with Honors

Harvard College
Cambridge, Massachusetts

April 1st, 2024

Abstract

In 2023, three new rules were introduced to the game of professional baseball at the Major League level. The pitch timer, defensive shift limits, and increase in base size were years in the making, all having been individually tested in the Minor Leagues prior to their call up to “The Show”. While many different aspects of the game of baseball have seemingly been affected by said new rules, I investigate their affects on plate appearance outcomes and stolen base attempts by creating models of both situations on a by-pitch basis. I am able to use indicators to distinguish the impact of the different rules by utilizing a control year, 2023, and their respective testing seasons from Double-A and Triple-A in addition to MLB data.

Acknowledgements

I am so thankful for so many people who have given me love and support throughout this project. Many thanks to Mark Glickman, for being an incredible mentor and advisor. Thank you to the HappySleep Youtube Channel for making 24 hour Mozart videos which kept my code running overnight. Thank you to Harvard, for taking a chance on a New York girl who wrote her application essay about her love of Boston baseball. Thank you to my wonderful friends- from all corners of Harvard, particularly Eliot House- as well as home. Specifically, thanks to Clio Dakolias for her amazing help in editing this monster. Thank you to the creators of baseballR, without whom this project wouldn't be possible.

And finally, thank you to my family, for putting up with my little baseball obsession, especially to my Mom, for introducing me to it.

And with that... Play Ball!

Contents

1	Introduction	6
2	Background	8
2.1	Rules	8
2.1.1	Minor League Baseball System	10
2.1.2	Pitch Timer	12
2.1.3	Defensive Shift Limits	14
2.1.4	Bigger Bases	18
2.2	In-Game Situations	20
2.2.1	Plate-Appearances & their Outcomes	20
2.2.2	Stolen Base Attempts	22
3	Analysis Framework	26
3.1	Leagues of Focus	26
3.2	Years of Focus	27
3.2.1	2014	27
3.2.2	2021	27
3.2.3	2022	28
3.2.4	2023	29
3.3	Analysis Intentions	30
3.4	Treatment Features	31
3.5	Count Sub-Models	36
4	Data	38
4.1	Data Source and Baseline Assumptions	38
4.2	Shared Data	41
4.2.1	Features	41

4.2.2	Feature Engineering	43
4.3	Unique to PA Model	46
4.3.1	Additional Features	47
4.3.2	Feature Engineering	49
4.4	Unique to SBA Model	50
4.4.1	Additional Features	51
4.4.2	Feature Engineering	52
5	PA Model	57
5.1	Previous Work	57
5.1.1	Direct PA Outcome Prediction	58
5.1.2	Indirect Modeling of PA Outcomes Used for Further Analysis	61
5.2	Model Definition	62
5.2.1	Model Intentions	63
5.2.2	Model Formula	63
5.2.3	Model Assumptions	67
5.2.4	Model R Specifics	68
5.3	Model Results	69
5.3.1	Overarching Model	69
5.3.2	Sub-Model Comparisons	76
5.3.3	Final PA Outcome Model Thoughts	79
6	SBA Model	81
6.1	Previous Work	81
6.2	Model Definition	83
6.2.1	Model Intentions	83
6.2.2	Model Formula	83
6.2.3	Model Assumptions	85
6.2.4	Model R Specifics	86

6.3	Results	87
6.3.1	Overarching Model	87
6.3.2	Sub-Model Comparisons	95
6.4	Final SBA Model Thoughts	99
7	Conclusion	102
7.1	MLB's Intentions	102
7.2	Analysis Technique Adjustments	103
7.3	MLB: What's Next	104

1 Introduction

The 2023 Major League Baseball (MLB) season brought monumental change to the game of professional baseball. On March 30th, 2023, for the first time since 1968, all 30 Major League clubs began their season on MLB Opening Day (Major League Baseball, 2023). This Opening Day was also unique in that it played host to the first 15 games in which the pitch timer, defensive shift limits, and bigger base sizes were enforced as formal rules at the Major League level ((Major League Baseball, 2022).

Defined briefly (further in Sections 2.1.2-2.1.4), the pitch timer limits the time between pitches, the defensive shift limits restrict defensive positioning adjustments, and the bigger base sizes require first, second, and third base to be 18” squares, a 3” increase on all sides from the former standard.

The implementation of these rules came after “thorough and ongoing research with our fans,” MLB Commissioner Robert D. Manfred Jr. said in a press conference on January 27th, 2023, during which the details of the new game policies were formally defined (Anthony Castrovince, 2023a). Manfred revealed the research demonstrated the following: “Number 1, fans want games with better pace. Two, fans want more action, more balls in play. And three, fans want to see more of the athleticism of our great players,” (Anthony Castrovince, 2023a). Thus, in an attempt to evolve the game in line with fan feedback, MLB developed the aforementioned rules. Specifically, it claimed to design the pitch timer to create a faster pace of play with less waiting and more action, the defensive shift limits to increase the number of hits and highlight players’ athleticism (both offensively and defensively), and the bigger base sizes to increase safety of the baserunner’s path to incentivize more aggressive baserunning (Major League Baseball, n.d.).

MLB recognized the potential significant impact the rules could have on the game and accordingly leveraged its Minor League farm system to test them in lower-stakes settings (further discussed in Section 2.1). Before their Major League debuts, each

rule was tested independently across various Minor League levels over multiple seasons, and were subject to constant observation, re-evaluation, and revision in an effort to perfect them before they got their call up to “The Show”. The first version of the pitch timer—the rule that took the longest to finalize—was first rolled out in 2015, while defensive shift limits and bigger bases were each initially tested in 2021 (Joe Trezza, [2022](#)).

While perhaps the most obvious difference in the 2023 MLB season was the pace of play, as the average regular season 9-inning game dropped almost 24 whole minutes from the 2022 average— from 3:03:44 to 2:39:49 (David Adler, [2023](#)). There are, however, the questions about what other aspects of the game have been affected and to what extent. Specifically, two such intriguing in-game situations are plate appearance outcomes and stolen base attempts. Not only are they both quintessential parts of the game, but MLB explicitly mentioned its expectations regarding the impact of the rules on these scenarios. Therefore, comparisons of plate appearance outcomes and stolen base attempts before and after the rules’ enforcement can help evaluate the impact the rules are having relative to their intended goals. Due to the concurrent implementation of the rules in the Major League, to properly isolate the impact of each rule, I utilize data from their respective Minor League testing seasons.

Therefore, my thesis constitutes an attempt to understand the causal impact of each new rule on the in-game situations of plate-appearance outcomes and stolen base attempts by utilizing not only Major League Baseball data but Minor League seasons in which the rules were tested. I perform my analysis on a by-pitch basis.

2 Background

Before proceeding to modeling and analysis, I first present background information and definitions, all of which are imperative to understand in order for proper interpretation of this project.

This chapter begins with an explanation of MLB’s Minor League structure, followed by a general history and definition of each rule, and introductions to the modeled in-game situations.

2.1 Rules

Each season prior to MLB Opening Day, the Official Baseball Rules document is released by the Office of the Commissioner of Baseball (Committee, 2023). This compilation houses everything from the most basic game rules (e.g., Rule 5.05(b)(1), which defines balls and strikes relative to the strike zone), to equipment regulations (e.g., Rule 3.02, which specifies bat size and materials), to guidelines for the most niche in-game situations (e.g., Rule 5.09(b)(7), which is a partial definition of the Infield Fly rule), and more (Committee, 2023). It contains the answers to any and every question about how the game of baseball is played at the Major League level.

Prior to the 2023 season, a group known as the “Official Playing Rules Committee” was responsible for the document’s composition. However, this collective did not include player representatives, and was thus restructured into the “Joint Competition Committee” under the 2022 Collective Bargaining Agreement between MLB and the Major League Baseball Players Association (MLBPA) ¹. The new board has since authored Official Baseball Rules: 2023 Edition (Committee, 2023).

Comprised of four active MLB players, six representatives from the Office of the Commissioner, and one official MLB umpire, the duties of the Joint Competition Committee also extend to being “responsible for recommending and adopting any new or

¹The agreement holds through the 2025 MLB Season

amended playing or scoring rules that significantly affect terms and conditions of Players' employment" (Major League Baseball and Major League Baseball Players Union, 2022). Therefore, working closely with MLB Commissioner of Baseball Robert D. Manfred Jr. and others at MLB Headquarters, the group was directly responsible for the integration and rollout of the new rules.

On September 9th, 2022, MLB officially announced the committee's approval of the pitch timer, defensive shift limits, and increase in base size for the following 2023 MLB season. "These steps are designed to improve pace of play, increase action, and reduce injuries, all of which are goals that have overwhelming support among our fans," Manfred explained in the press release (Major League Baseball, 2022). As the amendments represented the most significant change to the MLB game rules in recent history, the decision to enforce them at the Major League level was not made blindly, and came after each rule was tested thoroughly in the Minor Leagues (see rest of subsection for more). "Throughout the extensive testing of recent years, Minor League personnel and a wide range of fans – from the most loyal to casual observers – have recognized the collective impact of these changes in making the game even better and more enjoyable," Manfred continued (Major League Baseball, 2022).

Though players who spent time in the Minor Leagues in recent seasons had exposure to the new rules, it was during Spring Training 2023 where many Major-Leaguers first played with their enforcement. The six week warmup acted as an adjustment period, giving players time to properly adapt their game prior to the start of the regular season. This practice proved necessary, as the rules came with a learning curve: games averaged 2.03 pitch timer violations during the first week of Spring Training. However, by the preseason's end, not only did infractions decrease substantially, but compared to Spring Training 2022, the average game time was 26 minutes quicker and there was a significant increase in both stolen base attempts and ground balls resulting in hits (Anthony Castrovince, 2023b).

Just a few weeks later on March 30th, MLB Opening Day, the pitch timer, defensive

shift limits, and bigger bases were actively enforced for the first games of the regular season, beginning a new era of Major League Baseball.

2.1.1 Minor League Baseball System

Also known as the “farm” system, MLB’s Minor League Baseball (MiLB)’s structure of leveled competition allows organizations to develop and “grow” their players, analogous to farmers growing their crops. As of the 2024 season, there are four levels of affiliated MiLB leagues, as well as a Rookie League. In order of talent and competition level, the four full season MiLB leagues are: Low Class A (Single-A), High Class A (High-A), Double-A (Double-A), and Triple-A (Triple-A) (MiLB, [2024](#)). Before a player is called up to the Majors, they often play for the organization’s Double-A or Triple-A team.

When a player inks his first contract with an MLB organization, his rights are owned by the club for six years before he can become a free agent, regardless of the length of his initial contract or the way in which he signed (e.g., after being drafted in the Rule-4 Draft, as an International Free Agent, etc.). By signing, he gives up his right to play competitive baseball in other capacities, such as collegiately or in independent leagues (Major League Baseball and Major League Baseball Players Union, [2022](#)). The player is added to his franchise’s prospect bank, and will begin his path through their Minor League affiliates, with the ultimate goal of eventually making the MLB roster. This elongated development timeline contrasts those of other professional sports leagues, such as the National Football League or the National Basketball League, whose draftees often join their teams directly, as well as the National Hockey League (NHL), whose teams retain rights over their prospects but allow them to play collegiately or in independent leagues before they are ready to join their NHL or affiliate rosters.

The pace of a prospect’s movement through the Minors is determined solely by the organization that owns his rights. Each club’s coaching staff manage their respective prospects, promoting players through leagues as they show improvement, in order to continue to increase challenge and spur further development. They can also demote

prospects, sending players back to lower levels of the Minors as they see fit. No two players' paths through the Minors are alike. A player might progress through the leagues at a relatively consistent pace, skip leagues entirely, or become "stuck" in one for an extended period of time, even permanently. Decisions are made completely on a player-by-player basis. This trickle-up system ensures that as a player's game matures, he continues to play with others that match his competition and skill, and is able to learn and be continuously challenged by the players around him. The most coveted call up, naturally, is when a player is called up to the Majors. Though a player might hope his call to "The Show" is his final league transition, making his Major debut does not make him ineligible to be sent back down to the Minors. Unless detailed specifically within their contract, any player, even the most seasoned Major-Leaguer, can be demoted. A team's intention for the downgrade could be anything from hope for further development, to breaking the player out of a slump, an injury recovery assignment, and more.

MLB's Minor League system not only allows for proper preparation of its players before reaching the highest level of professional ball, but it also serves as a unique experiment field. If MLB wants to test a new game rule before implementing it in the Major level, it can utilize its MiLB leagues as test cases so as to better understand the rule's consequences and how it might shape the game (Committee, [2023](#)), (MiLB, [2024](#)). MLB can implement a rule at any MiLB level, as well as alter it throughout the season as it observes its impact. Often rules that are first tested in the Minors go on to be implemented at the Major level, though they sometimes can be scrapped completely. For instance, in 2021, the "Step-Off Rule," which required pitchers to step off the rubber completely in order to attempt a pick-off, was piloted in High-A only to be abandoned the following season (Anthony Castrovine, [2021](#)). This MiLB testing environment was utilized and proven to be imperative in the development and refinement of the pitch timer, defensive shift limits, and bigger base sizes prior to their 2023 MLB enforcement.

2.1.2 Pitch Timer

In practice, a baseball game does not have a regulated time constraint. This again contrasts other sports, such as hockey's three 20 minute periods or soccer's two 45 minute halves. With no minimum or maximum duration, a baseball game extends until one team has more runs than their opponent at the end of a minimum of 9 innings. Another unpredictable component of a baseball game is the quality of baseball (the practice of the sport, not the actual ball). Until each play happens, it is impossible to predict the result of the play with full certainty- the human dependency adds an unconstrainable element of stochasticity. Therefore, there is a significant range in the entertainment value of a baseball game- a game could be chock-full of incredible offensive and defensive plays, or it could be a calmer game, with the winning team scoring the sole run of the game. Even the most talented teams can and do put up performances like the latter. Frustration over unpredictability in both game duration and quality was found in fan research, which raised concern as MLB must continue to grow the game and retain fans to remain profitable (Anthony Castrovine, [2023a](#)).

The way to properly address these concerns is complicated. There is no way to impose full-game time regulations without deviating from the sport almost entirely. It is also impossible to regulate the outcome of plays in the capacity of artificially increasing entertainment ("rigging it"), as the existence of multiple potential outcomes contributes to the heart and purpose of the game. The rule of the pitch timer, however, represents an attempt to standardize what MLB believes can be controlled- the time in between pitches- and to increase the probability of entertaining plays by decreasing player decision time. Prior to the pitch timer, there were no limitations on when the pitcher or batter needed to be ready for the next pitch. A pitcher could take as long as he wanted to reset and ready himself between throws, and a batter had boundless time to re-adjust his stance before signaling attentiveness. Though previously accepted as simply a part of the game, when trying to find solutions for the aforementioned issues, MLB flagged this unconstrained time as a potential place for improvement and began

developing the pitch timer. MLB's hope was that by putting a limit on the time between pitches, the pace of play would increase (MLB, [2023d](#)). It also saw the potential benefit in terms of game quality, its hypothesis being that when both pitchers and batters have less time in between pitches, both parties are forced into faster in-game decisions, likely leading to more entertaining plays.

The first iteration of the pitch timer was created and enforced for some Double-A and Triple-A teams in 2015 (Associated Press, [2015](#)). This initial model was a simple 20-second limit between pitches, in which if the pitcher defaulted, the batter was awarded an automatic ball, and if the batter defaulted, he was charged with an automatic strike. In its pilot season, the 20-second limit had a huge initial impact, decreasing games by about 12 minutes, and was consequently expanded for use by all Double-A and Triple-A teams in the following season (Anthony Castrovine, [2021](#)). The restriction was only altered once in its first six seasons of use, in 2018 when the window was cut from 20 to 15 seconds for empty-base situations in Double-A and Triple-A. Eventually players began to adapt and learned to get around the rule: pitchers would reset the clock by stepping off the rubber or attempting a pickoff, and batters would step out of the box or call time. MLB therefore decided to test a more restrictive variant in the West divisions of Low-A in 2021 (Anthony Castrovine, [2021](#)). In this iteration, the time window shrunk from 20 to 15 seconds between pitches and pitchers were limited to only two pick-off attempts per plate appearance. The rule was again revised in 2022, and its testing was expanded to all four full-season Minor Leagues. Following a few week grace period to start the season, by April 15th, 2022 the rule was in full force (Minda Haas Kuhlman, [2022](#)). This version had a 14-second limit between pitches with empty bases, and a slightly longer limit if there was at least one man on base (19 seconds for Triple-A, 18 seconds all others). The batter had to be in the box and alert to the pitcher with at least nine seconds remaining on the clock (Joe Trezza, [2022](#)). Further, the pitcher could still restart the clock by stepping off the rubber or attempting a pickoff, but to prevent excessive use of this work-around, they were limited to two step-offs and/or pickoffs

per plate appearance. The improved rule brought about significant results, enough for MLB to be encouraged to implement a similar variant at the Major level in 2023.

The official MLB rule added in 2023 includes a 15-second limit between pitches if the bases are clear, 20 seconds if there are runners on (contrasting MiLB’s 2022 14 and 18—19 second limits, respectively) (Major League Baseball, 2023). In order to not violate the time constraint, a pitcher must start his motion to deliver the pitch before time expires, else he is charged with an automatic ball. A batter must be alert to the pitcher and in the box with eight or more seconds left, and is charged with an automatic strike if not. Just as in the 2022 Minor League version, the pitcher can restart the clock by stepping off the rubber or attempting a pickoff, and again, pitchers are limited to two step-offs and pickoffs per plate appearance. However, if the runner steals successfully, or the pickoff is successful, this count of pickoff/step-offs is reset. Furthermore, if a third pickoff attempt is made and unsuccessful, the runner will be granted the base. The batter can restart the clock by calling time, but they are limited to this once per plate appearance, and must call time while there are at least eight seconds left. Mound visits, injury timeouts, and offensive team timeouts do not count to the limit (Major League Baseball, 2023). Almost all aspects of the 2023 Major League version of the rule were held for the Minors. Like 2022, in 2023, all Minor levels had 14 seconds in between pitches if the bases were clear, extended time if players were on base, 19 seconds for Triple-A and 18 seconds otherwise (different from MLB’s 15 and 20 second limits, respectively) (Josh Norris, 2023). Also, unlike the Majors, if a runner successfully stole a base, the pickoff/step-off count is not reset.

2.1.3 Defensive Shift Limits

Following the 2003 publication of Michael Lewis’ *Moneyball*, which detailed the Oakland Athletics’ use of statistics to optimize their roster relative to the team’s low budget in the early 2000s, every MLB organization has added a baseball analytics department to their staff. Each respective group attempts to harness the abundance of available data

to gain insight that might give their team an upper edge. Everything from biomechanical training data, to static batting and pitching data, game logs, plate appearances, pitch-by-pitch data (like what I use in this project!), and more is utilized. The use of sabermetrics (the formal term for the application of statistical methods to baseball) has proved greatly successful, inspiring many adjustments to the game, including the use of the defensive shift. Subsequently, there have been complaints that some analytic advantages have gone too far and disrupted the game. MLB's most direct response to appease such protests has come in the form of the defensive shift limits rule.

The original defensive shift can be traced back to the 1920s, when teams would shift their outfielders back and towards right field to combat the offensive power of Cy Williams (Matt Kelly, [2019](#)). After its initial use, the shift re-appeared infrequently throughout the rest of the twentieth century for a sample of the League's top hitters, including an extreme version used by the Cleveland Indians in the 1940s against the other heavy-hitter with the last name Williams: Ted (Matt Kelly, [2019](#)).

The analytics-encouraged shift did not come into play until the late 2000s and escalated throughout the 2010s. Behind the support of then-skipper Joe Maddon, the Tampa Bay Rays first experimented with the shift, trying it against Boston Red Sox slugger David Ortiz in 2006. By analyzing ball location tracking data, their sabermetric analysts were able to predict where Ortiz might put balls into play relative to the pitching matchup. Finding obvious trends, they subsequently suggested shifts of their fielders into the forecasted locations, to turn the spots of historic hits into easy outs. The Rays found the strategy effective in curbing Ortiz and escalated its use concurrent with its success in the following years. Before long, other teams took notice and added the trick to their own defensive auxiliaries- by the mid 2010s it was normalized across the entirety of The League. There was not one sole version of the shift, as the suggested adjustments dynamically changed by batter; teams would shift their infielders and/or outfielders relative to the hitter that was at the plate to where they believed they would have the highest probability in recording an out. The use of the shift more than doubled

from 2016 to 2022, jumping from being enacted for 17.1% of all plate appearances to a whopping 37.5% (MLB Baseball Savant, [n.d.](#)). Its success was also apparent, as from 2006 to 2022, there was a ten point decrease in league-wide batting average. While as it became more normalized players learned to try to hit around it, from 2012 to 2022 there was still a significant six point decrease (Anthony Castrovine, [2023a](#)).

Concurrent with the increase in the shift's use and results, batters became both more frustrated and outspoken. A February 2022 article published in *The Athletic* by acclaimed baseball analyst Jayson Stark included quotes from front-office officials, coaches, and players—such as big names Max Muncy, Freddie Freeman, and Joey Gallo—sounding off on what the shift had taken from them. “I get the defensive strategies, I do. I’m 100 percent not against that. But I think at some point, you have to fix the game a little bit. I mean, I don’t understand how I’m supposed to hit a double or triple when I have six guys standing in the outfield,” Gallo claimed (Jayson Stark, [2022](#)). He was vocal again in a June 2022 article by ESPN’s Jesse Rogers, claiming that a rule limiting the shift “Opens up a whole new f—ing world for hitters,” (Jesse Rogers, [2022](#)). Another common target of the shift, leftie Kole Calhoun, was quoted in the same article, commenting on the shift’s harm to a player’s offensive momentum, “[The game without the shift] doesn’t let you fall into as deep a slump as what could possibly happen in today’s game, your 0-for-20s and 1-for-30s ... You can kind of salvage your day with something that’s kind of easy for every hitter. That’s not there anymore,” (Jesse Rogers, [2022](#)). The irates’ frustrations were soon calmed, as just a couple months later in September 2022, MLB released its intention to implement defensive shift limits for the 2023 season.

The defensive shift limits rule is a direct restriction to the extent teams can shift their defenders. The rule does not eliminate the entirety of the shift, but rather prevents it from being used in extreme. It calls for the placement of all four infielders within the boundary of the infield when the pitcher is on the rubber, with two on either side of second (MLB, [2023b](#)). This prevents teams from stacking a side of second that a

player is known to hit to, as well as the four outfielder situation. In addition, infielders cannot switch sides of second, therefore a team cannot relocate their best defender to the opposite side if a batter is known to pull the ball that way. There are no restrictions regarding bringing an outfielder shallow or fully into the infield, but teams were already unlikely to do this, as it leaves only two beyond the diamond. If a defensive team violates the rule at the time of a pitch, the offensive team can subsequently choose the result of the pitch or an automatic ball added to the batter's count. The restrictions serve as a happy-medium solution, as teams can still partially shift for batters, but to a less extreme extent as before.

Shift restrictions were first tested in Double-A in 2021 (Anthony Castrovine, [2021](#)). This original version only encapsulated half of what the rule is today, simply requiring that there be four infielders within the infield dirt at the time of the pitch. While in the official announcement of the 2021 Minor League rules testing MLB claimed that it might add the requirement of having two infielders on either side of second base midseason, and numerous blogs and news feeds reported that this adjustment was made, MLB never went through with the modification, and this additional lateral condition was not officially enforced until the following year. "The goal here is to increase batting average on balls in play, and to restore more traditional aesthetics and outcomes on batted balls. [The lateral shift restriction condition] is a regulation MLB experimented with in the independent Atlantic League in 2019 but had not fully implemented in the Minors until now," MLB writer Joe Trezza explained in a 2022 article (Joe Trezza, [2022](#)). In 2022, the full version of the rule was again tested at the Double-A level, but was also further expanded to High-A and Single-A. For the following 2023 season, the rule was not only enforced for all full-season Minor League seasons, but it also got its call up to the Majors.

2.1.4 Bigger Bases

The increase of base size has the simplest background of the three new 2023 rules. Just as the name suggests, all the rule brings is an increase in base size. Prior to the 2023 MLB season, first, second, and third base were 15.0" by 15.0" squares, each having a surface area of 225 inches squared. In 2023, three inches were added to each side of the three bases, making each a 18.0" by 18.0" square with 324 inches squared of surface area (MLB, 2023a). The home base plate dimensions were left unchanged, and it remains an irregular pentagon with two 12.0" sides forming a point parallel to a 17.0" side with 8.5" connecting segments (MLB, 2023c).

MLB's explanation for this change was again twofold. In an official announcement of the 2023 new rules, The League justified the bigger bases with an aim of "Increased safety + aggressive baserunning" (Major League Baseball, n.d.). Additionally, in the rule's MLB glossary definition, it claims that the change "will likely encourage more stolen-base attempts. The bigger bases could also have the effect of reducing over-sliding, whereby a player loses contact with the bag after sliding through it," (MLB, 2023a).

This is not the first time MLB has introduced a rule that specifically targets baserunning in the hopes of increased player safety. In 2014, MLB added Rule 7.13 to try to reduce the amount of home plate collisions between the baserunner and catcher (MLB, 2014). Colloquially known as the "Buster Posey rule", the amendment came after serious injuries the All-Star catcher (season-ending broken leg, May 2011) and other players suffered during attempting tag-outs at home plate. The initiative was further supported by research that used data from the 2002-2011 MLB seasons and found that the situation represented the highest risk of injury for players (D. A. Rosenbaum, 2014). The rule proved to be quite effective, resulting in a significant reduction of both concussions and other injuries for attempted tag-outs at home plate (Green et al., 2019). MLB also created Rule 6.01(j), the "Chase Utley rule", for the 2016 season, to regulate sliding on double play attempts following a collision between baserunning Utley and shortstop

Ruben Tejada in 2015 wherein Tejada broke his leg. A study published in April 2022 juxtaposed infield collision injuries from 2010-2015 and 2016-2019, before and after creation of the rule, and found a statistically significant decrease in collisions in the period after (Tanenbaum et al., [2022](#)). While both Rule 7.13 and Rule 6.01(j) have proved successful, they regulate niche plays, and baserunning injuries are still a prevalent issue in other in-game situations. Even during the most simple plays baserunners can over-slide and collide with fielders, resulting in significant injury for either party. For instance, in April 2017, while trying to advance from first to second, runner Manny Machado overslid second base and his foot collided with second baseman Dustin Pedroia's knee, causing damage that eventually led to the end of Pedroia's career (Jeff Seidel, [2017](#)).

With the increase in base size, baserunners have less ground to cover when running from base to base, and have a bigger target to reach to be safe. In total, the distance between first and second base, as well as second and third base is decreased by 4.5", the gap between home and first or third is reduced by 3.0", and the surface area of each square base is expanded by 99 square inches. The biggest benefit of the change perhaps goes to fielders, as they can be closer to the play while still tagging the base (i.e., a first baseman can be 3.0" closer to the throw coming from the shortstop while still stepping on the base for the out). The extra space between baserunners and fielders should make both sides more confident that there is a lower chance of collision, in turn incentivizing more stolen bases. MLB sees this potential steal increase as a secondary benefit, claiming "Though the base-size change may have a modest impact on stolen-base success rate, the modification's primary goal is to give players more room to operate and to avoid collisions" (MLB, [2023a](#)).

Bigger bases were first tested in 2021 in Triple-A and the Arizona Fall Rookie League. For the following 2022 season, the change extended to all other full-season Minor Leagues (Double-A, High-A, and Low-A) before reaching the Majors in 2023 (Joe Trezza, [2022](#)). That season, there was a decline in base-running injuries at every level of the Minors, with a 13.5% decrease overall. MLB hoped to continue these MiLB

trends into the Majors when they brought the rule to The Show in 2023 (Major League Baseball, [2022](#)).

2.2 In-Game Situations

The following subsection contextualizes each in-game situation of focus.

2.2.1 Plate-Appearances & their Outcomes

In the game of baseball, a plate appearance (PA) is considered to be the completed event of a batter stepping up to the plate to face a pitcher for a single batting turn (MLB, [n.d.-c](#)). The results of a PA describe the outcome of the batting turn- the exact way in which a batter recorded an out or got on base.

PAs are a vital aspect of baseball- a game's result is determined by the layered outcomes of its PAs. In each half inning, the batting team is given PAs until three result in outs. For a team to score a single run, one of its players must cycle through all four bases. While this can occur in a single PA with a home run, it is relatively rare and difficult to execute (otherwise all players would do is hit home runs). Instead, the majority of runs are scored by players who are already on base by advancing home via hits by their teammates. For this to happen, there must be multiple PAs that do not result in outs within a single half inning. Therefore, the more PAs a team has, the more opportunities they have to get runners on base, thereby increasing their chances of scoring. Consequently, teams aim to maximize their own batting PAs while minimizing those of their opponents.

A player's PA results over the course of a season can be used as a measure of player talent. A batter's success can be measured by how many of his PAs end with him reaching base, driving in runs, or scoring. This is reflected in such statistics as batting average, slugging percentage, on-base percentage, runs batted in count, and more. For pitchers, evaluation is dependent on how many PAs end by getting batters out- the higher the proportion of batters out to the total PAs pitched, the better the pitcher.

Statistics that mirror this include, but are not limited to, earned run average and batting average against. Both parties can also be evaluated relative to the subevent frequency of their PA results. For instance, with respect to batters, the more extra bases from a hit, the more valuable (singles ; doubles ; triples ; home runs), and for pitchers, the higher the strikeout and whiff rates the better. Depending on the relative frequency of their respective subevents, both batters and pitchers can be considered further valuable or potentially liable to their team- in regards to offensive and defensive production, respectively.

A PA is often confused with an at-bat (AB). Contrasting a PA, which encompasses all batting turn outcomes, an AB only considers those that result in a hit, error, non-sacrifice out, or when the batter reaches on a fielder's choice (i.e., all ABs are also PAs, but not all PAs are also ABs) (MLB, [n.d.-a](#)). The distinction between the two exists for the purpose of batting statistic calculations, so situations in which a batter reaches base without putting the ball in play- when they are walked, hit by a pitch, etc- are not misreflected as batting talent. Therefore, a player's batting average and slugging percentage are calculated considering ABs rather than all PAs. In this project, while I do exclude specific batting turn outcomes, I do not limit the breadth to the extent that an AB does and thus will refer to the completed events as PAs.

Often the result of a PA has consequences beyond the scope of the batter, and cascades into other plays in the game. For instance, a batter singles, and a baserunner that was previously on first advances to second. Alternatively, a batter singles, and a baserunner that was previously on first advances to third. Or, a batter singles, and a baserunner that was previously on first is tagged out while attempting to advance further (not a force out, the single holds). In this sense, there are hundreds of possible results deriving from a single PA. However, for simplicity, I will only consider PA outcomes in the context of the batter. This means that for the examples described above, the result for all would simply be a single. Not to mention, while often represented by the overarching outcome, the full span of a PA includes all pitches thrown to the batter

during the batting turn. In this regard, PA results can be extended to be considered on a by-pitch basis.

PA outcomes are clearly important to the game, and the new rules directly adjusted their structure. This renders PA results an intriguing corner of baseball to investigate relative to the impacts of the new rules. In an attempt to explore these impacts, I fit a model relative to PA outcomes on a by pitch basis, taking into consideration the time before and after each of the rules' implementation. Though the rest of the model's details are discussed in later chapters of this project, in its fitting I considered the unique PA outcomes of ["Strike", "InPlay_Out", "Foul", "Ball", "Single", "Double", "Triple", "HR"].

2.2.2 Stolen Base Attempts

If a PA results in the batter getting on base, he can further progress the bases relative to the results of his teammates' subsequent PAs (e.g., advancing from first to second following a teammate's single, or advancing from first to second following a teammate's walk), or he can attempt to steal a base.

A stolen base (SB) occurs when a baserunner advances to the following base at a time when the ball is not actively put into play by a batter. Baserunners often attempt to steal as the pitcher is preparing to pitch, concurrent with the pitch delivery, when the catcher is returning the ball to the pitcher, or when the pitcher is attempting a pick off (MLB, [n.d.-e](#)). In all instances, the baserunner is only given steal credit if he previously displayed intent to steal, determined by discretion of the official scorekeeper.

There are many instances that might appear to be stealing, but are not officially ruled as such. If the baserunner advances to the next base on a passed ball, error, or wild pitch without showing intent to steal prior it is not considered a steal (Committee, [2023](#)). If a defending team leaves a base open and undefended (unlikely, but possible) and the defensive team shows no attempt to prevent the baserunner from advancing, the baserunner is not given stolen base credit but rather the play is ruled as "defensive

indifference” (Rule 9.07(g)) (Committee, [2023](#)).

A baserunner can steal any base in front of him that is open. The most common base stolen is second, when a base runner advances from first. This is partially attributed to the fact that a baserunner at second is less of a scoring threat than a baserunner at third, and therefore a defending team is more likely to prevent stealing from second to third or the rare cases of third to home. Singles are also more common outcomes than doubles and triples, thus baserunners have more opportunities to steal from first than from second or third. Additionally, if there are multiple men on base they can attempt to steal at the same time. This is most commonly tried with two baserunners, usually when they are on first and second, and if pulled off successfully is called a “double steal”. A “triple steal” is also possible, but is extremely rare, risky, and hard to pull off- the most recent one was in 2008 by the Cleveland Indians (New York Times, [2005](#)).

The benefit and incentive to stealing is to get the baserunner(s) closer to scoring, as advancing one base in any situation advances the baserunner closer to crossing home. They can be decisive in close games (see: ALDS 2004 Game 4 BOS vs NYY). While clearly a desirable feat, the actual execution of stealing proves very difficult. If a pitcher believes that a baserunner is going to steal, perhaps made uneasy by the magnitude of lead he has off the base, the pitcher can attempt to pick him off. When a baserunner is not actively touching the base they occupy they are in risk of being tagged out by a fielder who has possession of the ball. Therefore, a pickoff is defined as an attempt by the pitcher to get the ball to the occupied base’s defender, so the defender can then tag the runner out before the runner re-reaches the base (and safety). In order to prevent successful pickoffs, baserunners often dive back to their bases when threatened, sometimes resulting in injury. However, pickoff attempts do not indicate official stolen base attempts, nor is the baserunner considered to be caught stealing in successful pickoffs. The reasoning for this is because by just leading off of the base, “[the baserunner] never attempted to steal in the first place” (MLB, [n.d.-b](#)).

If the baserunner has left his original base with intent to steal and is tagged before

reaching the next base, he is considered to be caught stealing (MLB, [n.d.-b](#)). For instance, if a baserunner attempts a steal while a pitch is being thrown, following his control of the ball, the catcher can “gun down” the runner by getting the ball to the defender of the base the runner is trying to steal, who then attempts to tag the runner out. Catchers are often evaluated on how fast they can get command of a pitch, transition, and complete a throw to any of the other bases. The faster and more accurate the throws, the more defensively talented the catcher is considered. A baserunner can also be caught stealing before a pitch, which often occurs when a pitcher attempts a pickoff but the baserunner is already en route to second, requiring the first baseman to send the ball to the second baseman or shortstop to get the tag. Additionally, if the baserunner successfully steals but loses contact with the base before he gains his balance and is tagged, he is considered to have been caught stealing, because he was “never established on the base,” (MLB, [n.d.-b](#)). A player’s total stolen base attempts (SBAs) is the sum of their successful stolen bases and number of times caught.

As described in 2.1, part of MLB’s intention in implementing the new rules was to increase the frequency of stealing. In this last MLB season, the regular season averaged .72 stolen bases and .90 stolen base attempts per game. This was a .21 and .22 increase from 2022, and the highest of the last ten seasons by margins of .15 and .12, respectively (Chris Gilligan, [2023](#)). However, as more than one of the new rules sought to increase the number of stolen bases, it is impossible to isolate the specific impact of each rule using data solely from 2023. Thus, stolen base attempts are an interesting in-game situation to explore relative to the implementation timeline of the new rules, and are investigated in this project. To do so, I used pitch-by-pitch data from PAs in which there was a single baserunner on first base, fitting the overall model to understand situations in which runners were more likely to attempt to steal- i.e., record a SBA.

The reasoning for why I narrowed my focus to only consider the situations in which there was a sole base runner on first is three fold. First, relative to steals, stealing second base from first is the most common occurrence. Second, there was no information in

the data that reflected if there was only one runner on the bases if not only on first- there was no way to know if third base was open for the baserunner on second to steal. Finally, stealing home from third is a completely different in-game scenario than when a baserunner steals second or third- not only is it such a rarity, but one could argue a player uses a different skill set as well (heavily relying on the element of surprise).

3 Analysis Framework

The following chapter outlines my project’s analysis framework. It first explains the reasoning behind the choices of the leagues and years of data used, as well as maps out the rules that were enforced in each unique season. It proceeds by discussing how using said specific league/year combinations allows for the independent evaluation of each rule. In such explanation, the treatment variables that were added to the data are introduced, as well as the assumptions the project makes in using them. The chapter closes by describing the additional subset evaluation methods considered.

3.1 Leagues of Focus

The intention of this project is to understand the independent impacts of the pitch timer, defensive shift limits, and increase in base size on the game of professional baseball. In order to do so properly, I am unable to use data solely from the Major level, because of the rules’ concurrent implementation. Therefore, I integrate seasons of data from the Minors, specifically from the Double-A and Triple-A levels.

As explained in 2.1.1, Double-A and Triple-A are the top two tiers of the Minors, hosting the highest competition level of players and prospects. Therefore, relative to the other Minors, the talent and gameplay in Double-A and Triple-A is most similar to the Majors. Consistent with this, Double-A and Triple-A also have higher quality data collection and amount of available resources. Also, as is evident from the detailed histories of the rules dissected in the previous chapter, prior to their Major League debuts, each rule of focus was tested thoroughly in the Minor Leagues- including Double-A and Triple-A.

When considering the rollout timelines of the pitch timer, defensive shift limits, and bigger bases across the Major, Triple-A, and Double-A levels, windows of time exist wherein each of the rule’s impacts can be isolated. With the added fact of relatively comparable competition among the leagues, this solidified the decision to utilize Double-

A and Triple-A data in addition to MLB for this project.

3.2 Years of Focus

This project considers different years, or rather baseball seasons, of data: 2014, 2021, 2022, and 2023. As detailed in the rules respective backgrounds, each of the pitch timer, defensive shift limits, and bigger base rule amendments were tested and implemented in different Minor Leagues across different seasons before all three of their MLB debuts in 2023. In order to attempt to encapsulate full impact of the rules, we must consider a control year, the years of MiLB experimenting, and 2023, in respect to all three leagues of focus.

3.2.1 2014

The pitch timer, the rule with the longest experiment history, was first tested in 2015 in some Double-A and Triple-A games. Therefore, the most recent year in which NONE of the rules were implemented in any capacity among all three leagues was 2014 (Associated Press, [2015](#)). The 2014 season therefore acts as the control year in this study. Something of note is that the aforementioned Rule 7.13: “Buster Posey Rule” was first enforced in 2014, however it was added to all leagues and has been an active rule since, therefore it is insignificant to flag in my analysis (MLB, [2014](#)).

3.2.2 2021

The first season of treated data I will consider is 2021. In 2021, the first variant of the defensive shift limits were enforced at the Double-A level and the bigger bases were tested in Triple-A. Both Double-A and Triple-A were also enforcing an earlier version of the pitch timer for said season (Anthony Castrovince, [2021](#)).

Though there were two new rules added to the Major level in 2021, neither were one of the rules of focus nor should be flagged for potential confounding impact. The first rule altered the game’s doubleheader structure, limiting games on the double-day

to be seven innings rather than nine, and the second placed a baserunner on second base at the beginning of every extra-innings half inning (MLB, 2021). The former rule is insignificant to my research, as even though this was a change at the Major level, all MiLB teams already had (and still have) the option to make doubleheaders seven innings, and because of the inconsistencies among leagues it does not make sense to control for this, as it is a minor impact at best (Committee, 2023). To handle the latter rule, though the “ghostrunner” had been tested in all levels of MiLB since 2018, I am simply going to only consider pitches that occur within the first nine innings of games (MiLB, 2018).

3.2.3 2022

In the following 2022 season, Double-A saw a more aggressive version of the defensive shift limits, and both Double-A and Triple-A used an enhanced revision of the pitch timer. Double-A also used the bigger bases for its first season. The amplified pitch timer was enforced for all games after April 14th (Minda Haas Kuhlman, 2022), while all other rules were held for the entirety of the season. In addition to the rules of focus, “Robot Umps” were also tested in a limited capacity in Triple-A (Joe Trezza, 2022). With the modification, instead of the traditional home plate umpires determining if pitches were balls or strikes, the “automated ball and strike system” (ABS) took charge. Created by the tech company Hawk-Eye, the system automatically adjusts the strike zone dependent on the height of each batter and classifies pitches thrown to be within the calculated zone’s bounds (strikes) or out of it (balls). In 2022, the system was used for all games throughout the season hosted by the International League (East Coast League)’s Charlotte Knights, as well as all games played in the Pacific Coast League after May 17th. In these games, home plate umpires directly relayed the results that the tech classified (Joe Trezza, 2022). The rule posed enough chance of potential impact to be controlled for- the specific method for how is discussed in the coming subsections.

At the Major level the only formal amendment was the universal designated hitter

(DH) rule, which allowed teams in both the American League and National League to use a DH (Major League Baseball and Major League Baseball Players Union, [2022](#)). The rule had a quick stint during the partial 2020 season during the COVID-19 pandemic, and could be elected for use in the Minors on a by-league basis prior to 2022. The rule will not be significant enough to control for in my analysis, as even when National League pitchers had to hit, they often only recorded a single PA per game, and attempted to avoid stepping up to the plate whenever possible.

Though not a formal rule change, the PitchCom system was also approved for Major League (but not Double-A or Triple-A) use in 2022 (Anthony Castrovine, [2022](#)). The PitchCom system allows catchers to signal pitches without using their fingers. With PitchCom's use, the old-fashion signals are replaced by a catcher pressing a button on a wearable tech sleeve. The sleeve has multiple buttons corresponding to different pitch types- once the catcher has selected his preference, the decision is sent to the pitcher, and he audibly hears the choice from a small speaker/receiver that he wears in his hat. The language of the receiver can be adjusted to each player- breaking previously imposing language barriers between teammates. The intention of PitchCom is to prevent sign stealing, to make communication between pitchers and catchers easier, and to potentially increase game speed. Use of the system is optional and thus the choice is left to each pitcher's discretion. My project does not control for the use of PitchCom, as not only is there no record of what pitchers use it and which do not, but its use should have no effect on my situations of focus- it is illegal for teams to sign-steal in the first place, therefore this project operates under the assumption that they do not, ease of communication is impossible to measure, and pace of play is relatively irrelevant.

3.2.4 2023

The final year of data I will use is the most recent complete full-season at the time of writing, 2023. The season brought no significant changes to the rules at the Double-A level. Triple-A added the full defensive shift limits rule, expanded its use of ABS, and

got access to PitchCom (Buster Olney, [2023](#)). Half of the games were played in “Full-ABS”, with umpires relaying the pitch calls determined by the tech, and half with a challenge system, where the umpires would make the call but if a player disagreed his team could challenge and it would be checked by the system- teams got up to three incorrect challenges a game (Josh Norris, [2023](#)). Again explained further later, I do indeed control for the use of ABS. Consistent with its introduction to the Majors in 2022, I do not control for the use of PitchCom in Triple-A in 2023.

2023 brought the biggest change to the Majors in recent years- enough to inspire the creation of this project. As previously explained in detail, the pitch timer, defensive shift limits, and bigger bases were all implemented for the first time at the Major League level in 2023. The use of PitchCom also expanded, allowing pitchers to wear the same sleeve as catchers and contribute to pitch-type calls (Manny Randhawa, [2023](#)).

3.3 Analysis Intentions

Up to this point, I have mentioned the years and leagues of specific data I will use to evaluate the impact of each rule, but have not thoroughly explained how or why I am able to do this.

Consider a simple experiment, in which there is a control group and a treatment group. In order to evaluate the imposed changes, one compares observations of the treated group before and after they are given the treatment, while also juxtaposing the observations of the control group. If I was considering the 2023 MLB new rules to be a combined treatment, I might attempt to evaluate the changes invoked by comparing the 2022 and 2023 seasons. However, this methodology lacks a control group, thus any observed differences cannot be justified by the new rules.

In order to incorporate control groups to evaluate the new rules, we incorporate the Minors. Returning to the aforementioned example, consider the theoretical situation in which the rules of focus were tested in a minor league in 2022 before they made their MLB debut. Further, assume that 2022 was the first time in which these rules

were implemented for said minor league (meaning no new rules were active in its 2021 season). In this case, the minor league team can be considered the treatment group, with baseline observations from 2021 and treated observations in 2022. Further, MLB is considered the control group, as it does not have any treatment applied in either the 2021 or 2022 season.

To understand the rules' impact on a specific in-game situation, a model of it can be created and its coefficients interpreted. The data the model is fit on must have enough entries for both the treatment and control groups from before and after the time of treatment. The data must also include indicators that describe the league, year, and treatment status of each of its entries. Such indicators control for the differences their respective values contribute to the situation, the magnitude of which is reflected in their fitted coefficients. Back to the theoretical example- if a model is fit for a specific situation with the necessary indicators included in its data, though differences caused by the rules can now be formally observed, they still cannot be individualized among them because of the rules' concurrent implementation.

While the real rules I am considering - the pitch timer, defensive shift limits, and bigger base sizes- were all enforced for the first time at the Major level in 2023, they were not all simultaneously tested across the Minors. Because of this staggered implementation, windows of time can be isolated in which each rule was the only new treatment of a specific league(s). Therefore, by properly indicating league, year, and each individual rule's implementation level in the data, I can fit models and interpret the rule's impact on individual levels by interpreting their coefficients.

3.4 Treatment Features

In my project, I use pitch-by-pitch data. Though in the next chapter I discuss the source of my data, as well as other features I include in my models, below I continue the definition of my analysis framework by introducing the specific indicators that allowed for its execution.

The subset of indicators include:

- **Year**, type factor with levels [“2014”, “2021”, “2022”, “2023”]: The season in which the pitch was thrown.
- **League**, type factor with levels [“AA”, “AAA”, “MLB”]: The league in which the pitch was thrown.
 - “AA”: Games played at the Double-A level.
 - “AAA”: Games played at the Triple-A level.
- **ABS**, type logical with values [TRUE, FALSE]: Indicates if the current pitch was thrown in a game in which ABS was used in any capacity.
- **Pitch_timer**, type factor with levels [“Control”, “v1”, “v2”]: Indicates variation of pitch timer rule implemented at time of pitch.
 - “Control”: No version of the rule implemented.
 - “v1”: Earlier version of the rule implemented.
 - “v2”: Current version of rule respective to league implemented.
- **Defensive_shift_limits**, type factor with levels [“Control”, “v1”, “v2”]: Indicates variation of pitch timer rule implemented at time of pitch.
 - “Control”: No version of the rule implemented.
 - “v1”: Earlier version of the rule implemented - Double-A 2021.
 - “v2”: Current version of rule implemented.
- **Bigger_bases**, type logical with values [TRUE, FALSE]: Indicates if bigger bases were used in the game of pitch.

Condensing the previous chapter’s history of each rule respective to league, year, and version, the added features pertaining to the rules were present in the data in the following way:

*: TRUE for all Triple-A West division games after May 17th, and games played at Triple-A Charlotte, FALSE otherwise

Rule/Year	League	2014	2021	2022
ABS	MLB	FALSE	FALSE	FALSE
	AAA	FALSE	FALSE	*
	AA	FALSE	FALSE	FALSE
Pitch.timer	MLB	Control	Control	Control
	AAA	Control	v1	v2
	AA	Control	v1	v2
Defensive_shift_limits	MLB	Control	Control	Control
	AAA	Control	Control	Control
	AA	Control	v1	v2
Bigger_bases	MLB	FALSE	FALSE	FALSE
	AAA	FALSE	TRUE	TRUE
	AA	FALSE	FALSE	TRUE

Table 1: Rules and Conditions by League and Year

3.4.0.1 Associated Assumptions

The inclusion of such variables introduces assumptions relative to what each truly represents.

By controlling for years with the “Year” value, I expect any significant changes in the way that the game of baseball is played to be encapsulated by the coefficient. This belief holds even though there is a nine year difference between the first (2014) and last season (2023) I use. In the “League” feature, I am assuming any differences in talent level or outcomes among the Leagues will be accounted for. I do not, however, indicate or control for taxi players (those that frequently get called up to and sent down from the Majors) or Major Leaguers who are playing in the Minors on injury recovery assignments.

As detailed previously in this chapter, ABS, or the automatic ball and strike system, was first tested in some capacity in Triple-A for the 2022 season. Specifically, the system was used for all games hosted by the Charlotte Knights, as well as in all games hosted by members of the Pacific Coast League after May 17th. Subsequently, the ABS feature was “TRUE” for all these specific Triple-A 2022 games, and “FALSE”

otherwise. Additionally, the feature was expanded to all games in 2023. Half of the games were played with umpires relaying the calls by the system, as they had in all games that ABS was active for previously. The other half were played with a challenge system, in which the umpire made the call and the batter had the chance to challenge it and reference the system. There was no indication of the strategy used in each game within the returned data nor in other available sources. As a result, the ABS feature blankets both cases. Though not ideal, this was the only way to control for its use in any capacity, and thus the feature is used assuming that its direct use and the challenge system have the same effect.

When the advanced version of the pitch timer made its debut in MiLB in 2022, players were given a brief grace period at the beginning of the season to give them time to adjust properly. Though the rule was used, it wasn't actively enforced until April 15th, meaning that players in violation prior to that point were simply told when they had faulted. To account for this gray-area period in which the pitch timer was somewhat half-implemented, I only consider Double-A and Triple-A games from the 2022 season after April 15th. Being that it was only two weeks (MiLB opening day was March 29th, 2022), taking out this stretch had a miniscule impact relative to the amount of games omitted, and it was thus unnecessary to filter out of the Major season.

Additionally, though the pitch timer implemented in the Minors was a significant enhancement of the rule previously enforced, it varied slightly among the leagues. Recall from before that in empty base situations in all leagues there were 14 seconds between pitches, but if at least one man was on base, Triple-A had 19 seconds between pitches, while all other milb levels had 18. Variation in the rule among leagues increased when the rule was added to the Majors in 2023. All Minor Leagues used the respective version of the rule they had in the season prior, with Triple-A retaining the second difference in man-on situations. In MLB, the rule that was enforced had a 15 second limit between pitches in empty base situations and increased to 20 seconds when there was at least one man on base. MLB pitchers were also restricted to two pickoff attempts per PA (as they

were in the Minors in 2022 and 2023), but, if the batter advanced, this limit was reset, unlike in the Minors. The 2023 rule for all required the batter to be alert to the pitcher with at least eight seconds remaining on the clock, while it was nine seconds in the Minors in 2022. In the scope of my project, I ignore these slight differences, and blanket all “advanced” versions of the rule to be represented by “v2” in the “Pitch_timer” factor. The variations, though they exist with intention, if acknowledged, would take away the ability to use my defined analysis technique relative to the pitch timer. Therefore, my project assumes that the impact of the slight discrepancies among the advanced version of the pitch timer implemented across leagues is miniscule.

Perhaps the biggest assumption made in using the aforementioned indicators is the independence among them- I do not use any interaction terms among the predictors. The choice to do this is not because I don’t believe there are relationships between them (as will be further discussed in the conclusion), but rather because of the immense number of terms this would have added for my models to consider, which already have significant computation time relative to the data and other features. I justify the independence among leagues and years by the fact that the relative talent level of the leagues should not change over the years- MLB still houses the best talent, Triple-A the second, and Double-A the most amateur- thus the league coefficients should adequately encapsulate the differences. In the same line of thinking, there should not be any change in how the respective leagues respond to the rule changes. As explained earlier in this chapter for why I chose them as additional leagues of focus, Double-A and Triple-A have the most similar play level to MLB relative to other leagues, and any differences in play should already be reflected in the league feature, leaving the rules to be evaluated in their own respect. I consider the rules to be independent of each other, as they each refine parts of the game in different ways, and the intention of the project was to understand them independently. Finally, the impact of each rule should not change over time- MLB created the rules for lasting rather than temporary change, thus I work under the assumption that the change in years do not directly affect the

impact of the rules. Not to mention, MLB gave its players grace periods to adjust to the rules (beginning of 2022 MiLB season and 2023 Spring Training)- therefore it expected any changes outside of said periods to be reflective of the rules directly rather than players attempting to adjust to them- further supporting evaluation of the rules independent to time.

3.5 Count Sub-Models

Within both the PA and SBA models, in addition to values that control for the implementation status of each of the three rules, features that describe that game-state are considered. This is both to control for and analyze the effects that game-states might have on the respective situations. Though all of the features are further discussed in the next chapter, two of the included predictors are the number of balls and the number of strikes in the PA at the time of the pitch.

The two measures come together to describe “the count” at the time of each pitch in a PA. The aforementioned order holds, the first number representing the total balls thrown prior, the second the total strikes. Different states of the count have different perceptions relative to which player has the advantage in the PA. Though regardless of its associated count a batter could put a pitch in play (the PA thus resulting in an in-play out or hit), the advantage of a count is determined relative to how close a pitcher is to walking a batter compared to how close the batter is to striking out.

The most neutral state is the initial count, prior to the first pitch of the PA when there are no balls or strikes. While the current PA is not necessarily independent from each parties’ previous, at the time of a 0-0 count, no additional information has been learned from the current PA, and any and every PA result is still possible. Both similar and different is a “full count”, when there are three balls and two strikes. Full counts and initial counts share the fact that on paper, the batter is no closer to walking than striking out, therefore the advantage between the players is neutral. However, to get to a full count, at least five pitches must have been thrown, and in making their subsequent

decisions, both the batter and pitcher use information gained about each other from the previous pitches. Thus, the pitch at the time of a full count is seen as a pivotal moment, and is highly likely to determine the result of the PA. The only instance in which a 3-2 pitch is not terminal is when the batter fouls, and as he already has two strikes, the PA continues and the next pitch is thrown again with the count full.

A hitter's count is when the pitcher is closer to walking the batter than the batter is to striking out. It follows that the more extreme the difference, the better the position for the batter, and that the 3-0 count is the most advantageous. In such a situation, it is debated whether a batter should swing, because the pitcher is more likely throwing an easier strike in order to prevent a walk, or if they should again take the pitch, as the probability of the pitcher being able to pitch consecutive three strikes after three balls is perceived to be miniscule (Adam Berry, [n.d.](#)). Regardless, the batter has the advantage as he has the flexibility to choose between multiple strategies, while it is imperative for the pitcher to throw a strike. Oftentimes in this case, if there is a baserunner, he will try to advance the bases and steal. A pitcher's count is when the opposite is true, when the batter is closer to striking out than a pitcher is to walking them. Likewise, the 0-2 scenario, colloquially called the "0-2 hole", gives the pitcher the highest margin of advantage.

The difference of player strategy in each case encourages me to take a closer look at the four situations. Therefore, in addition to the overarching models of each situation (PA outcomes and SB attempts) I also train four submodels that respectively only consider pitches with 0-0, 3-2, 3-0, and 0-2 counts. This allows for deeper understanding and analysis of the new rules' effects on each specific game-state.

4 Data

Though I model two separate in-game situations (PA outcomes and SB attempts (SBAs)), they share many of the same data considerations- to the extent that it did not make sense to individualize their data descriptions within their respective chapters.

Therefore, this chapter first describes the overall data acquisition, but follows with subsections unique to aspects shared, to the PA model, and to the SBA model relative to features used, feature engineering, data assumptions, and overall choices made.

4.1 Data Source and Baseline Assumptions

All of the data that I use was sourced using the “baseballR” package in R (Petti and Gilani, 2024). Accessible through R’s CRAN (R Core Team, 2024), the package’s defined intention is to be used for “Acquiring and Analyzing Baseball Data”. I chose to use this source, rather than scraping MLB’s Stats API myself, because of the extent of information it centralizes from not only the Major level but from MiLB as well. The package, published by developers Bill Petti and Saiem Gilani, bolsters a set of functions that scrape multiple different sources, including FanGraphs, Baseball Reference, and MLB’s Baseball Savant, in order to provide team schedules, rosters, game scores, stat leaders, and more. The only limiting factor to what baseball data the package provides is the extent of what exists.

By calling baseballR’s “mlb_pbp()” function, I accessed pitch-by-pitch data for the Double-A, Triple-A, and MLB seasons from 2014, 2021, 2022, and 2023, calling the function once for each of the twelve league/year combinations. When sourcing the data, I added a parameter that ensured all pitches returned would only be from regular season games, in order to be both consistent and to avoid any potential confounders brought in by Spring Training and/or playoff baseball. Many prospects play in teams’ Spring Training games, as they are trying to make the Big League roster, and star players only play sporadically. Playoff baseball does not include the participation of all

teams, and is played with a win-or-go-home mentality. Playoff games are statistically proven to have different characteristics than regular season games, and October Baseball is widely considered to be more unpredictable (Mike Petriello, 2019).

When sourcing the data, I handled each league/year combination one by one. I iterated through each game of each season, calling the function that returned its pitch-by-pitch data, subsequently adding the information to its associated combination’s dataframe. While the great majority of games had pitch-by-pitch data available, most seasons had at least a few missing. If a game had missing data, then their specific “mlb_pbb()” function call threw an error and did not return anything. The total games for which data was successfully pulled compared to the total games played relative to league/year are reflected below:

Year	MLB		Triple-A		Double-A	
	# Games with PBP Data	% Of Total Games	# Games with PBP Data	% Of Total Games	# Games with PBP Data	% Of Total Games
2014	2430	100.00%	3029	99.15%	2105	99.67%
2021	2198	90.45%	1925	98.72%	1741	96.72%
2022	2430	100.00%	2237	99.42%	2050	99.03%
2023	2430	100.00%	2225	98.89%	2054	99.23%

Table 2: Number of Games by Year with Available PBP Data across MLB, Triple-A, and Double-A

Though the function returned datasets that provided an immense amount of columns for each pitch, detailing everything from “entertainment quality” to projected location, such specifics were not available for all league/year pairings, leading to many columns being full of “NA”s. This is a result of the improvement in baseball technology and subsequent increase in the amount of available data over the years, as well as the difference in data allocation resources across the three leagues, as the Majors have significantly higher budgets than MiLB. Therefore, when choosing the features of my models, I was limited to a base selection of the columns available for the most limited season/league combination, which was Double-A 2014.

Following their acquisition, I continued to engineer features and clean data for each of the twelve seasons, rather than combining them into one. I did this because each

season was initially composed of nearly a million rows, and the computation complexity was faster to handle each independently. Working on this smaller scale had the added benefit of allowing me to test and ensure the validity of my methods across each sample. Therefore, continuing the ideology above, I used the Double-A 2014 season to initially develop the engineering and cleaning steps (Kluyver et al., 2016). I added points within each of the notebooks that displayed the progress and results of each step, to both confirm that the data format was uniform across the different combinations, and ensure consistency of the methods' execution relative to their intention. I used the same notebooks to clean each unique combination, and because of the steps taken, there should not be any major deviations in their final form. Prior to their model fittings, the seasons were combined and the final variable types were confirmed.

I decided to use pitch-by-pitch data for this project, as compared to by-PA, by-half inning, by-game, or others, because it allowed for evaluation on the smallest incremental aspect of the game- all others are intrinsically built off of it. However, by using pitch-by-pitch data, I am assuming that each pitch in the dataset is somewhat independent of the others. While I control for pitcher and batter as random effects, as well as account for the count and gamestate of the pitch, I do not denote if two pitches are from the same PA, nor if the batter and pitcher have faced each other before (within the season or historically). In reality, the pitch a pitcher delivers is dependent on what they know about a batter and how they have reacted to other pitches in a PA- e.g., pitchers will attempt to avoid certain spots of players' strike zones known to be their sweet spots, or if the pitcher has had success getting the batter to whiff with a certain pitch during the PA, he might try this same pitch again. However, beyond the aforementioned features, I do not account for this, as to do so properly would immensely increase the number of predictors and complexity of evaluation. Therefore, in relation to others, this project considers pitches to be independent.

All of the pitches I consider were thrown within the first nine innings of games and were from instances of complete PAs. If there is no deterministic winner after

nine innings of a baseball game, it continues into extra innings (the longest one ever recording 26 innings!) (Dave Adler, 2020). However, as mentioned in the previous chapter, in 2020, MLB introduced a rule that added a ghostrunner on second base at the start of each half-inning (MLB, 2021). Though the rule had previously been tested in and remains in MiLB, it is not a rule of focus for this project, and adds unnecessary layers of complication- the easiest way to avoid its effects was to simply use the first nine innings of games. Using complete PAs ensures that each PA has an actual definitive result. Incomplete PAs could be the result of injury or another factor that interrupted the game.

4.2 Shared Data

While the PA and SBA models differ in the capacity of the situation they intend to represent, they share many similarities. Both are attempting to quantify the impacts of the same rules over the same years and leagues. Not to mention, in order to properly model either, the game-state at the time of the pitch must be considered. Therefore, the models incorporate many of the same features (some of which were engineered) and their corresponding assumptions. As the treatment features were discussed already, the models' shared game-state features are introduced below.

4.2.1 Features

The objective of my project is to understand the impact of the pitch timer, defensive shift limits, and the increase in base size relative to plate-appearance outcomes and stolen base attempts. In order for the models to properly reflect the potential impacts, I needed to properly control for different in-game scenarios. Accordingly, in addition to the aforementioned pitch context features, I chose to include variables that characterize the game state at the time of each pitch.

While the two studied scenarios are different corners of the game (and have additional specific predictors), their models share the following game-state features:

- **Strikes**, type integer in the range $[0, 2]$: Represents the number of strikes in the count prior to the current pitch.
- **Balls**, type integer in the range $[0, 3]$: Represents the number of balls in the count prior to the current pitch.
- **Outs**, type integer in the range $[0, 2]$: Represents the number of outs in the half-inning prior to the result of the current plate appearance.
- **Batter_home**, type logical with values $[TRUE, FALSE]$. Indicates if the current batter was home or away. TRUE if the batter was home, FALSE if away.
- **Inning**, type factor with levels $["1", "2", "3", "4", "5", "6", "7", "8", "9"]$: Represents the inning of the game at the time of the current pitch.
- **Batter_id**, type factor, controlled for random effects: Represents the unique identification for each batter.
- **Pitcher_id**, type factor controlled for random effects: Represents the unique identification for each pitcher.

4.2.1.1 Associated Assumptions

The game-state features attempt to control for effects that the specific state of the game might have on the models. As described in Baseline Assumptions, by choosing to use pitch-by-pitch data in an attempt to make the pitches as independent from each other as possible, I must control for game-state. Thus, the “Strikes”, “Balls”, “Outs”, “Batter_home”, and “Inning” features are assumed to account for any variation their unique potential values might bring relative to the PA outcomes and SB attempts (SBAs).

I additionally consider “Batter_id” and “Pitcher_id” as random effects. When defining the player sub-group analysis in the previous chapter, I validate this consideration by claiming it allows for the respective ids to bring variation into the data without over-compensating and fitting the models relative to them. It makes sense to consider them as random effects, as their variation, while not the primary focus of the project, will

inherently affect outcomes. Because the pitches are thrown across such different leagues and years, they consider an immense amount of different batter/pitcher matchups, the magnitude of which is impractical to fit in addition to being extremely computationally expensive. Both “Batter_id” and “Pitcher_id” are factor variable types- unordered, as the value a player is assigned has no correspondence to his talent or ability level. By defining these variables as random effects, both of my models are assuming that, relative to other features involved, both “Batter_id” and “Pitcher_id” contribute secondary impact relative to the effects of the new rules on PA outcomes and SB attempts (SBAs).

4.2.2 Feature Engineering

Some of the shared features were added to the data while some were taken or adjusted from the function-returned datasets. All pitch context attributes were added relative to what was true to the year and league of each pitch- this did not need advanced engineering. The returned datasets, on the other hand, while they provided an immense amount of valuable information, required significant adjustments in order to finalize the game-state features.

Many of the columns were faulty with respect to how they described the game-state that resulted from the current pitch or PA. However, in order to be functional to my models, they needed to describe the state prior. The “Inning”, “Outs”, “Batter_id”, and “Pitcher_id” features were taken straight from the dataset and had the sole alterations of their feature names. “Batter_home” and “Balls” were simple deviations of other given features, while the construction of “Strikes” and “Men_ON” employed the need for deeper logic chains.

4.2.2.1 Batter_home

The simplest adjustment of the shared engineered features was the “Batter_home” indicator. To determine if “Batter_home”, I simply checked a provided column that indicated if it was the top or bottom of the inning. If the given value was true, the

batter was a member of the visiting team, and the “Batter_home” was set to FALSE, else they were a member of the home team and it was set to TRUE.

4.2.2.2 Balls, Strikes

The counts of balls and strikes at the time of the pitch was misrecorded within the returned pitch-by-pitch datasets. If a throw was deemed a ball or strike, the respective count associated with the pitch would be one too many than the true number of balls or strikes at the time of the pitch. For instance, in the row entry of a pitch that was the fourth ball of a PA, where the batter was walked, the ball count in the row would reflect that four balls at the time of the pitch. Similarly, if a pitch was the first of the PA and called a strike, though no other pitches had been thrown in the PA, its row entry would indicate that it was thrown with a strike count of one. Such issues held for all ball or strike instances, and thus needed to be adjusted on a by-pitch basis in order to make both respective counts into predictors rather than post-result descriptions.

For the ball count, the fix was a simple subtraction- I utilized an indicator column of the original data that distinguished if a pitch was a variant of a ball (ball, wild pitch, pitch timer default, etc.), and if it was true, subtracted one from the original ball count.

Because of fouls, correcting the strike count proved a bit more complicated. In any theoretical PA, if a batter’s strike total at the time of the pitch is less than two and he subsequently fouls, the foul is added to the count as a strike. However, if the batter already has two strikes and fouls off the pitch, the result does not affect the strike count further, it remains at two, and the PA continues. If I assumed the strike count could be handled in the same way as the ball count, the subtraction would have been made in all situations, misvaluing a two-strike foul to have a single strike at the time of the pitch. To account for this, I created an additional indicator that signaled if a pitch was the last in the PA (done by finding the pitch in each PA that was associated with the highest pitch count). Using another column that described the results of each distinct pitch’s results with a string, I filtered out two-strike foul situations by isolating the pitches

that were indicated to be a strike, had specific results of a foul, a given strike count of two, and were not the last pitch of the PA. After isolating the problematic pitches, I proceeded with subtraction (same as I did to correct the ball count) for all other cases. I then returned to the unicorn cases, iterated through each of them, checking if the pitch prior also had the two strikes- if it did, the corrected count would also have two strikes, else it would be decreased to one.

4.2.2.3 Men_OB

In order to right the “Men_OB” feature, significant computational power was required. Though not used as a predictor in the stolen base attempts model, the feature was still imperative in filtering the data the model uses. The original column from the dataset described the state of the men on base at the end of the PA that the pitch was from- for instance, if a batter singled when the bases were empty, the column would claim “Men_On”, even though no one was on base prior to the batter and while he was up to bat. The misassignment was associated with all of the pitches in a PA- continuing with the previous example, if the batter faced pitches in the same plate appearance prior to his single, they would also be mislabeled with “Men_On”. If left untouched, this misrepresentation would cause a huge leak in the data, as the state of the “Men_OB” variable would reflect the outcome, while it was intended to be used to help predict it.

In order to fix the column to describe the state of the bases at the time of each pitch, I went through each unique half inning of each season and league, and iterated through all of the PAs. I assigned the pitches of the first PA of each half inning to have empty bases and the pitches of the PAs following to have the original labels of the PA before. Because each season had tens of thousands of unique half-innings, the degree of necessary iterations forced the correction to take an elongated computation time. Unfortunately, the way in which the data was recorded did not dynamically adjust the base state in the middle of a PA. There was therefore no simple way to include changes of base state that occurred in the middle of PAs (successful stolen base, failed stolen

base) in the adjusted feature either.

There were some error cases that had to be handled, as there were times in the data in which pitches were recorded to hit a player yet the overall result of the PA did not end with the player taking a base. I skipped all half innings that had a PA that suffered from this and did not give any of their respective PAs “Men_OB” values, because the issue had a cascading effect that altered all PAs after it. After the overarching iterations were complete, there remained some PAs in which pickoff attempts or variations of stolen base attempts occurred which were not assigned a “Men_OB” value (it still had the initial temporary value). I manually corrected these rows, setting them to “Men_On”. There were only a few of these instances in each season (averaging less than 10 total PAs in which this was the case, relative to tens of thousands unique PAs each season). Its small frequency, along with the fact that many of them came from the skipped innings, made me confident that there were no issues in my logic, and I rather attributed the errors to potentially being from when the data was collected. I set the instances from the skipped innings because of the relatively small number of SBAs and pickoffs considering the rest of the data, and believed it was important to retain them for the SBA model. After the “Men_OB” feature engineering was complete, I filtered out any remaining rows that still had the initial temporary “Men_OB” value (many of them from the problem half innings).

4.3 Unique to PA Model

The supplementary data used for the PA model was specific to batter/pitcher matchup, and was used to create its response variable.

In its final cleaning, any PAs that resulted in variations of baserunners caught stealing, successful pickoffs, intentional walks, catcher’s interference, or a balk (illegal pitcher motion ending with the batter being granted a base) were excluded, as those were results that were out of control of the batter. Such instances of caught stealing and pickoffs often ended innings, leaving the PA at the plate unfinished, and therefore as it

was not a full PA, I ignored it in my data. The way in which catcher’s interferences and balks were entered in the data made it easier to blanket all pitches resulting in either rather than excluding the terminal pitches specifically. Pitches that were intentional balls were also excluded, on account of the batter having no effect. Finally, last pitches of PAs that resulted in field errors, wild pitches, or the batter being hit were additionally excluded, though pitches in their respective PA prior to the occurrences were retained.

4.3.1 Additional Features

For the plate appearance model, extra features I used included:

- **Men_OB**, type ordinal factor with levels [“Empty”, “Men_On”, “RISP”, “Loaded”]: Represents the state of bases prior to the result of the current plate appearance.
 - “Empty” if the bases were empty
 - “Men_On” if there was a man on first
 - “RISP” if there was at least one runner in scoring position (at least one man on second or third base)
 - “Loaded” if the bases were filled
 - **Note:** For the SBA Model, all pitches were filtered such that Men_OB == “Men_On”, therefore it was unnecessary to use it as a formal SBA model feature.
- **Matchup_handed**, type factor with levels [“RR”, “LR”, “RL”, “LL”]: Represents the handedness matchup of the batter and pitcher. The first character describes the batter, and the second character describes the pitcher. “R” if the player is right-handed, “L” if they are left-handed.

While I will expand on the definitions of the resulting outcomes as well as on my reasoning for choice in the model’s own chapter, in the plate appearance outcome model I used the following response:

- **Pitch_outcome**: type ordinal factor with levels [“Strike”, “InPlay_Out”, “Foul”, “Ball”, “Single”, “Double”, “Triple”, “HR”]: Describes the outcome of the pitch.

- “Strike” if the pitch resulted in a strike
- “InPlay_Out” if the pitch resulted in an in-play out
- “Foul” if the pitch resulted in a foul
- “Foul” if the pitch resulted in a foul
- “Ball” if the pitch resulted in a ball
- “Single” if the pitch resulted in a single
- “Double” if the pitch resulted in a double
- “Triple” if the pitch resulted in a triple
- “HR” if the pitch resulted in a home run

4.3.1.1 Associated Assumptions

Even prior to the era of sabermetrics, trends were found in batter and pitcher success relative to handedness matchup. The “Matchup_handed” feature therefore assumes to encompass all significant differences in outcomes attributed to the handedness matchup between batters and pitchers.

By using the “Pitch_outcome” feature in the way I defined it above, I bucket the potential outcomes of a pitch into eight discrete cases, while in reality there are many more- especially if you expand the scope past the sole consideration of the batter. Additionally, while the feature has a distinct “InPlay_Out” outcome, it does not distinguish between a “Strike” and strikeout, or a “Ball” and walk. This is attributed to the fact that the data pitch-by-pitch, and though a pitch that results in a strike and is the third to complete a strikeout might hold more importance relative to the game than a pitch that results in the first strike of a PA, the two have the same end result when just considering a pitch’s outcome. Rather, the previously discussed game-state variables attempt to account for the situational importance.

4.3.2 Feature Engineering

All three of the unique features used in the plate appearance outcome model required modification to some extent. The obstacles faced with the “Men_OB” predictor were detailed above, and were much more complicated than what was done to create the remaining “Matchup_handed” and “Pitch_outcome”.

4.3.2.1 Matchup_handed

In order to create the “Matchup_handed” feature, I utilized two columns from the original dataset that provided information regarding the batter and pitcher’s handedness, respectively. Each column had the same unique possible values, “L” to denote that its describing player was batting or throwing left handed, and “R” to indicate he was playing as rightie. The finalized feature is a simple concatenation of the columns together, the first character representing the batter.

4.3.2.2 Pitch_outcome

To bucket the result of each pitch into one of the outcomes, [“Strike”, “InPlay_Out”, “Foul”, “Ball”, “Single”, “Double”, “Triple”, “HR”], I used columns that corresponded to its individual result, as well as of the overarching PA it was a part of. There were indicator features that showed if the pitch was a strike, ball, or put into play, and if it caused an out. While I used them to build base conditional filters, the set of indicators acted as umbrellas for all the variants of their respective situations- e.g., the in-play indicator for all in-play outs, singles, doubles, triples, and home runs was TRUE- and I wanted to consider these subevents individually. In addition, some rows representing the final pitch of a PA had inconsistent markings relative to their true result -e.g., the fourth ball in a PA that resulted in a walk would not be properly marked as a ball. To handle this, I utilized the indicator I created in the construction of “Men_OB” which distinguished if the pitch was the last in the at-bat.

I started the filtration with the “Ball” label, assigning it to any non-final pitch

with a TRUE value from the given ball indicator, as well as any final pitch that's PA outcome was a walk. It is important to note that in the context of my project, pitches that resulted in the end of a PA in the context of a walk or strikeout were reduced to the context of a "Ball" or "Strike"- this is further discussed in the data assumptions sub-section. My intention with starting with "Ball" was because it is the majority label and does not have any sub-possibilities worthy of individualizing. I then handled strikes, properly dividing them into true strikes or fouls. Pitches that were tagged as true fouls, foul bunts, pitchouts, or tips that also were not the last pitch of the PA were considered "Foul". The remaining non-last pitches that were also noted to be strikes by the overarching indicator were deemed "Strike", as well as last pitches with PA results of strikeouts or strikeout double plays.

The remaining levels were easier to catch. If a pitch was indicated to have been put in play, was the last of its PA, and the PA was indicated to end in an out, it was deemed an "InPlay_Out". Types of outs in this bucket included field outs, sacrifice bunts, sacrifice flies, ground outs, force outs, fielder's choices, as well as instances of any of the previous that resulted in a double or (the rare) triple play. While some of these outs could be argued to also provide positive offensive value, they also contributed an out, and for simplicity were reduced to this. For a pitch to be deemed a variation of a hit ("Single", "Double", "Triple", "HR"), it was the final of its PA and its PA result was labeled as a single, double, triple, or home run, respectively. Following completed construction of "Pitch_outcome", each season had about 20 random rows of data that didn't fit into any categories (random niche instances), and these were filtered out and not further considered.

4.4 Unique to SBA Model

In addition to the shared game-state features, the SBA model uses data that describes the number of pickoff attempts as well as identifies the runner at first.

Though in theory, a baserunner can steal second, third, or home base, this project

only looks at situations in which they have the possibility to steal second. For this to be the case, prior to being fitted, the data is finalized to only include situations in which the “Men_OB” value equals “Men_On”. As previously discussed, though the label might imply there being multiple men on base, it represents the case with a single baserunner on first. Cases in which there are multiple baserunners, or a baserunner on further bases, are labeled to be “RISP” (to indicate there is a runner in scoring position (second or third base)), or “Loaded”.

4.4.1 Additional Features

For the stolen base attempts model, I added the following features:

- **AB_pickoffs**, type factor with levels [“1”, “2”, “3”, “4” , “5”, “6”, “7”, “8”, “10”, “11”, “12”]: Represents the number of pickoffs during the plate appearance prior to the next pitch.
- **firstID**, factor, controlled for random effects: Represents the unique identification for the baserunner on first base.

The dependent variable I used was:

- **SBA**, type logical with values [TRUE OR FALSE]: Indicates if there was a stolen base attempt on the pitch (1), or not (0).

4.4.1.1 Associated Assumptions

Similar to “Batter_id” and “Pitcher_id”, “firstID” is a feature that is controlled for as a random effect. It didn’t make sense to include it in the PA model, as we are assuming irrelevance of outcome relative to the specific baserunners, but in the capacity of the SB model, as the man on first is the one that has the potential to record an SBA, and must be accounted for accordingly. Consistent with “Batter_id” and “Pitcher_id”, “firstID” is incorporated as a random effect because it is not a primary feature of focus but will still impact the fitting of models. Also like the others, it does not have an order, as it is simply the batter ID of the baserunner on first, and there is no importance in its

numeric values. This, along with its many unique values, suggests the predictor should not be incorporated as a regular predictor, in regards to both model and computational complexity. “firstID” is therefore controlled as a random effect within the SB model, assuming it has relatively small impact on the model’s outcomes.

The “AB_pickoffs” variable counts the number of pickoff attempts tried in the PA prior to the next pitch. While many baserunners that end up stealing bases tempt the pitcher into attempting a pickoff prior to their burglary, there are instances in which both baserunners are too quick or sly and advance without any attempted pickoffs beforehand, or on the contrary, act as if they might try a SBA just incentivize a pickoff to wear out or distract the pitcher. Therefore, this feature attempts to quantify how relevant pickoff attempts are. I acknowledge concern that it might have significant correlation to the response variable, thus, as detailed in the SB model’s respective chapter, I fit variations with and without the attribute.

4.4.2 Feature Engineering

The “firstID”, “AB_pickoffs”, and “SBA” predictors used uniquely in the stolen base attempts model all took tedious processes to concoct.

4.4.2.1 firstID

While the plate appearance model did not incorporate the identities of the specific men on base (if there were any), as discussed, it was important to consider and control for the runner on first in the stolen base attempts model, as they were the one that was attempting to steal. Though the returned datasets provided columns that claimed to provide the unique batter IDs of the runner on each base if they existed, pitches from all twelve seasons in which there were men on base at the beginning of the PA had this information missing. In order to have consistency among all of the rows, even when the values weren’t NAs, I employed the same correction technique.

After the terminal play of a PA, first base could be occupied by the batter of the

same PA, the man on first during the start of the (same) PA, or no one. It then extends that by using the result of the previous PA, as well as by knowing who was on first base for it, you can identify the first baserunner occupying the base at the start of the PA of focus, if he exists. The first PA of every half inning has empty bases, thus the information that is needed to find the information of the potential first baserunner of the second PA is known. Finding the information regarding the second PA allows you to find the third PA's information, and so on. Further, if any PA is known to begin with empty bases, there is no first baserunner, and baserunners of the subsequent PAs can only come from the PA with empty bases or those after.

To properly identify the player on first base at the start of a PA, I utilized the aforementioned logic and iterated through the results of previous PAs of the same half inning. I only iterated through PAs that had men on the base- i.e., PAs that had a “Men_On”, “RISP”, or “Loaded” label- to reduce computation time, and because it was unnecessary. For each PA ID in the list, I found the index of its first pitch and subtracted one to get the index of the last pitch of the previous PA and subsequently gain access to its information. There was no concern that this method would improperly bleed into a different half inning, because as previously mentioned, the bases were always empty for the first PA of a half inning ².

I isolated a column that provided a single string that denoted the previous PA's terminal play and another that was its final play's elongated description. Though I previously discussed issues regarding using the results of a PA as a predictor and how it would create leaky data, that was in respect to the PA model, and rather than be used as variables, in the scope of the SBA model, these columns were simply used to help engineer “firstID”. Therefore, depending on combinations of the previous PA's single string terminal play code and its more detailed description, I was able to identify the player on first base for the PA of focus and record his information. To derive the handling of the situations that are described below, I first experimented with the 2023

²Note that it has already been clarified that I am only considering the first nine innings of a game, making the extra-innings ghostrunner rule irrelevant

MLB data, and found the other eleven seasons to be consistent with the assumptions.

Some of the single string codes had a sole option for who was on first after the previous PA was completed, while others had to be broken down further by identifying substrings in its description. Plays that always ended with the batter of the previous PA on first for the PA of focus included but were not limited to walks, singles, hit by pitch situations, fielder's choice outs, and catcher's interference. A fielder's choice out occurred when the batter reached base yet another baserunner recorded an out (all situations of fielder's choices were confirmed to result in the batter on first). In addition, if the previous batter doubled or tripled, there would be no one on first for the PA of focus (note that following home runs the bases are always empty and were thus already accounted for).

Further filtration was needed for the remaining PA outcomes, of which all were variations of outs. Although there were some very niche cases handled, the sub-filtering of many of the outstanding cases followed similar patterns. None of them were considered to be fielder's choice outs (they were rather line outs, pop outs, ground outs, etc), therefore the batter of the previous PA was always assumed to be out and not the first baserunner for the PA of focus, unless it was otherwise indicated in the play's description that he had somehow made it to first. If such mention was absent, the "new" first baserunner was assumed to be the same as the one at the start of the previous PA, unless the description claimed he had gotten out or further advanced the bases- in which case first base was then open.

I iterated through the PAs in order that they happened, so when a PA had to refer back to a previous PA of the same half inning, it had already been processed and had the proper information needed for reference. Though there is the chance that there have been changes to or variation in the format of a PA's terminal play description over the years and leagues, I attempted to impose error catching, assessing the number of PAs designated to have "Men_On" and no information regarding the man on first. It indicated very little issue in applying the logic to different year/league combos, the

majority of years having less than 10 mislabeled PAs (out of about 40,000). While this error could be attributed to missed cases in my logic, its minuscule magnitude indicates it could also be the result of improper data records/data entry.

4.4.2.2 AB_pickoffs, SBA

The final features considered in the SBA model were “AB_pickoffs” and “SBA”. “AB_pickoffs”, the number of pickoff attempts in the PA prior to the current pitch, was used as a predictor, while “SBA”, an acronym for “Stolen Base Attempt”, was the response variable. A similar process was used to derive both values. First, I isolated the unique PA ids of row entries that claimed to result in a stolen 2B, runner caught stealing 2B, or pickoff caught stealing 2B, as well as those that included pickoff attempts to 1B. As I iterated through each, I created a temporary data frame which encompassed all pitches of the PA of focus. I then cycled through each of the rows of the temporary data. For each row, I counted the prior number of pickoff attempts to 1B in the PA, setting the value to “AB_pickoffs”.

Identifying SBAs was not as easy. In the dataset, stolen bases, pickoffs, instances where the baserunner was caught stealing, and anything else of that sort were labeled to have type “action”, contrasting regular pitches that had the type label “pitch”. While these “action” entries had some of the same values available as “pitch”s, they represented duplicates relative to the pitch they occurred during. Though I tried finding a way to simply drop the regular pitches and retain their corresponding action entries, the dataset limited this action from being successful, and instead linked each instance with the pitch that best described its game-state, so within the same iterations I set the “AB_pickoffs”. In most cases the SBA was associated with the pitch right before it happened in a PA, as the order of entries in the dataset was consistent with where it occurred, and it was a simple match. Just as was described in the above in the feature engineering for the shared columns, strikes and balls were also deducted properly. For instance, consider the case where a player steals in the middle of the first pitch of a PA

that results in a strike. The pitch would be entered in the dataset before the steal, and both would be entered into the data with the original faulty strike count of 1, when in reality there were no strikes prior to the pitch or the steal. With the logic I employed, the steal would be associated with this pitch, and the strike count would be properly corrected to 0.

I had to specifically consider when the SBA happened before the first pitch of a PA. In this scenario, the SBA is entered before the first pitch of the PA, and it does not have values in the faulty ball and strike count columns. This pitch is associated with the pitch after it, which will also have a strike and ball count of 0-0 after proper alteration, therefore reflecting the same game state as the SBA.

Finally, if the SBA was unsuccessful and ended the inning (i.e., two out caught stealing), it was not recorded an “action” type, but rather a pickoff with a PA result that indicated being caught stealing. This third unicorn case had to be specifically identified in order to be corrected, and followed the same levels as above- and was either associated with the pitch right before or behind it. When an SBA was linked to a pitch, there was simply a classifier added that indicated it occurred- all non SBA pitches (the great majority) received a zero instead.

5 PA Model

This chapter pertains to my specific model implementation and results of plate appearance outcomes. As defined in the background chapter, a plate appearance (PA) is the event of a completed batting turn in which a batter faces a pitcher. While its results can be interpreted on a by-PA basis, I drill down further and consider outcomes by-pitch in order to understand the situational impacts of the pitch timer, defensive shift limits, and bigger bases.

Subsequently, this chapter first discusses previous PA outcome modeling attempts, then moves into my own model's derivation, followed by model summary statistics, and concludes with a discussion of overarching and sub-model results and analysis.

5.1 Previous Work

As described briefly before, the publication of Michael Lewis' Moneyball revolutionized the use of sabermetrics, both in the capacity of team-driven insights and general academic research. Also, as outlined in its in-game situation definition, PAs are a very important component of the game of baseball, acting as the backbone structure in which games are built upon. Combining the two concepts, it naturally extends that there has been comprehensive work done in attempt to mathematically consider PAs. Unfortunately, the majority of models and/or predictions derived by teams' internal baseball analytics departments are kept internal for the purpose of maintaining a competitive advantage. Therefore, the publicly available research is not truly reflective of the depth of the work done, and has significant variance in the level of sophistication of techniques used. Also, much of the work, including all instances analyzed below, consider the outcomes on a by-PA basis rather than by-pitch as I do.

5.1.1 Direct PA Outcome Prediction

The most obvious end goal when modeling PAs is to model potential outcomes. Doing so further allows for analysis and predictions relative to game outcomes as well as team and player evaluation.

FanGraphs.com is a popular website that centralizes a myriad of baseball statistics, and dynamically updates its information concurrent with games (baseballR, the package I used to source my data, has many functions that returns data scraped from FanGraphs) (FanGraphs, [n.d.](#)). Besides being an information hub, FanGraphs employs statistical experts to create projections, utilizing and further innovating self-developed metrics. Outside of its paid analysts, it also boasts not only a very active but a relatively advanced community research page, where curious baseball nerds post mathematical answers to any question the sport might need answered.

In 2014, Jonah Pemstein published a series of articles to FanGraphs in which he derived a tool he dubbed the “Outcome Machine” (Jonah Pemstein, [2014](#)). When given the subevent rates for a batter and pitcher, the model incorporated league averages to estimate the chances of specific outcomes of a PA matchup between the specified players. Subevents it considered include a batter striking out, walking, hitting a home run, singling, doubling, tripling, and being hit by a pitch individually. The model did not directly compute the probability of each subevent, but rather relied on the fact that the expected value of the rates could be extended to represent the probability of each subevent. Further, it considered each outcome independent of others, and relative to the method that fit each subevent best, used either linear regression or logarithmic equations to calculate each expected subevent rate. The coefficients used in the respective calculations were taken from training models fit and tested on players’ individual season data from 2003-2013 (as it was 2014, the input was intended to be for the most recent season). The ensemble further found the probability of the batter recording an in-play out by subtracting the sum of the previous event probabilities from one.

From Pemstein's base, another FanGraphs user, Davis Weir, extended the concept, adapting the ideology to use logistic regression in 2020 (Davis Weir, [2020](#)). Logistic regression is one of the most obvious models to use to predict PA outcomes, as a PA subevent result can be simply classified in terms of a batter's success (the event occurring) or failure (it not). By using logistic regression, Weir was able to directly calculate the probability of subevents occurring, contrasting Pemstein's expected rate method. Also, rather than using players' complete season rates when training on PAs results of the season prior, Weir used their projected Steamer rates instead (Steamer is a highly-regarded projection system for batting and pitching stats), which also accounted for matchup handedness and ballpark impact (MLB, [n.d.-d](#)). By doing this he prevented data leakage, not letting players' season complete rates influence the prediction of events that they reflect the results of. Weir's approach was similar to Pemstein, as rather than invoking multinomial logistic regression, he modeled subevents independently and found the probability of an in-play out as the remainder of the sum of the rest from one.

Both Pemstein and Weir's models lean towards the simplest of modeling attempts, and trade the use of interaction terms and situational features for lower complexity and easier interpretation. They both produced results that slightly beat random selections, which though might seem only relatively impressive, baseball (or any sport) will always be impossible to definitively model, and their endeavors should be considered relatively successful. However, one obvious potential improvement to their approaches pertains to their respective independence beliefs, as Pemstein and Weir both assumed the result of each subevent was independent to the rest. Each article provided links to spreadsheet compositions of their respective models, so the user could investigate different batter/pitcher matchups. Though neither specified if they used the full dataset to fit each subevent model (and perhaps they did account for what I am about to critique), the way each composition calculated and output each subevent probability implied both Pemstein and Weir failed to condition the outcome of each subevent on others. Rather than fitting each subevent model on the entire dataset, cascading subevents should be

taken out of the pool as they are fit- e.g., if the model predicting the terminal event of a strikeout is calculated first, all PAs resulting in strikeouts should be removed from the data in fitting the next event, and so on. Extending this, instead of direct subevent probability outputs, the cascading events probabilities need to be adjusted relative to the percentage they truly represent- e.g., if an ensemble theoretically determined the probability of a strikeout to be 50%, modeled the probability of a walk next and also determined it to have 50% probability, the true walk probability would be 25%, or 50% of the remaining 50% not determined to be strikeouts.

Powers et al. (2017) took a more advanced approach, implementing a nuclear penalized multinomial regression (NPMR) to predict PA outcomes (Powers et al., 2017). Their model predicts the probability of PA outcomes by leveraging PA outcome frequency relative to the batter and pitcher involved, as well as the stadium the PA occurs in and who had home field advantage. An additional term was added to consider the handedness matchup. The group attempted to use nuclear penalized regularization rather than ridge regression to regulate their model, only adding penalties to the batter, pitcher, and stadium sub-matrices rather than all coefficients. The group provided results relative to the model's mean test deviance compared to a null model as well as one regularized with ridge regression, that proved significantly better performance of the non-null models. The visualization showed how as the percentage of training data used increased (from an original 5% to maximum 75% of PAs), the mean test variance continued to decrease. There was further comparison and discussion of the model's coefficients, which the group also used to divide batters and pitchers into tiers relative to different skills, including patience, trajectory, and speed, and power, trajectory, and command, respectively. No further evaluation statistics were provided, leaving the evaluation of the model's overall performance slightly ambiguous.

5.1.2 Indirect Modeling of PA Outcomes Used for Further Analysis

Given the instrumental role of PAs in the overarching game of baseball, they are often used in modeling capacities other than to just predict their own results- often in attempts to understand how additional variables might impact their outcomes.

In 2014, Tzeel et al. attempted to investigate the relationship duration of PAs and hitter success (Benjamin Tzeel, [2014](#)). In addition to the length in seconds of each PA, which they collected themselves by timing on a stopwatch, the group considered additional variables including but not limited to pitch velocity, the handedness of PA matchup, and pitch types thrown. They considered a successful PA to be one that resulted in an increase in a batter's on base percentage, and in addition examined the added value of extra-base hits. Using simple ANOVA analysis methods, the group found the biggest covariates that impact PA success to be total duration, pitcher and batter handedness, and the use of breaking balls in a PA. Overarchingly, their research suggested that with a basis of success and failure, ignoring subevents, there are no strong indicators within the pool of variables they considered. However, the pool of variables investigated, as well as the way in which they collected the durations of PAs, was questionable, forcing the integrity of the study to be debated.

In 2016, Casey used PA outcomes as a vehicle to model and evaluate pitcher decision making (Nathaniel Casey, [2016](#)). Casey modeled the progression of a PA as a Markov chain, basing his logic off of former research that suggested the quintessential memory-less Markov property holds in a PA (Sal Baxamusa, [2006](#)). To engineer the probabilities of PA outcomes, he utilized pitcher subevent percentages, distinct count percentages, as well as expected on-base percentage. However, as Casey's focus was relative to pitcher evaluation, he did not incorporate any batter specific data. By choosing to do so, his method of prediction overlooks batters completely, and though can be used in terms of understanding pitcher trends, is limited to interpretation in that scope.

Brill et al. (2023) used a Bayesian multinomial logistic model to evaluate how cycles through batting order affects pitcher fatigue (Brill et al., [2022](#)). Their model

was fit to predict the probability of different PA outcomes (not independently, like the aforementioned Pemstein and Weir!). The model incorporated features that described how many batters the pitcher had faced in the game prior to the current, if they were in their first, second, or third time through the batting order, as well as running average estimators for batter and pitcher quality, and indicators for matchup handedness and home field advantage. The running average estimators of both batter and pitcher ability were used to account for data bleeding, in place of their end-of-season averages which would reflect results of PAs that the model was trying to predict. The estimators were built using normal-normal conjugate distribution set ups, and their respective posteriors updated for each player after completion and prediction of each of their PAs. Wyner et al. interpreted their results relative to individual pitchers, in an attempt to understand how fatigue affects individuals of the population differently. The results ultimately suggested that in each subsequent PA a batter has against a pitcher in the same game, the batter gains advantages relative to PA outcomes. Such results make intuitive sense, because as the game continues and the batter is exposed to the pitcher's arsenal of pitches, he can anticipate more accurately what he might be thrown.

5.2 Model Definition

My model relative to PA outcomes incorporates bits and pieces of multiple of the previously discussed works, but I wouldn't directly compare or relate it to any single one. Rather than considering overarching PA outcomes, I consider them on a by-pitch basis. Also, while I do use the model as a vehicle to analyze an additional component of the game, rather than continue the investigation intentions of any of the discussed papers, my model attempts to isolate the impact of the pitch timer, defensive shift limits, and bigger bases.

5.2.1 Model Intentions

As discussed in the analysis framework chapter, the intention of my model is not precision. Rather, I use model fitting as a tool in order to attempt to understand the impact of the aforementioned features on plate appearance outcomes on a by-pitch basis. Applying the logic built out from both the treatment and game-state features, I am able to do this by evaluating features respectively fit coefficients. When fitting my models, rather than splitting the data into training and validation sets, I fit on the entirety of the available data, in an attempt to get the most comprehensive and descriptive coefficients possible.

5.2.2 Model Formula

In my critiques of the previous literature review, I slightly hint at a deviation of a model I use. Within the examples that employed variations of logistic regression, I questioned their respective reasoning for using binary classification rather than multinomial, as it is a natural extension of the tool that considers multiple outcomes. However, because of the close nature of the results of plate appearance outcomes, I hesitated to derive a direct multinomial logistic regression model, and instead chose to utilize the closely related proportional odds model. The proportional odds allows for the modeling of multinomial response types that involve ordered categories. When considering the different types of PA outcomes on a by-pitch basis, I focused on those that really only depended on the interaction of the batter and pitcher (filtering out errors, catcher interference, balks (only the pitcher error), and more), and narrowed down the options to consider strikes, balls, in-play outs, fouls, singles, doubles, triples, and home runs.

Keeping in mind that the results were significantly nested in each other (especially because I was considering the data on a by-pitch basis rather than on a by-PA or other), I attempted to order the outcomes relative to worse case outcomes relative to the batter to best. Relative to a batter's personal success, the worst outcome of a PA is a strikeout. However, because I was working on a by-pitch scope, I decided the bottom of the basin

was the only type of pitch-outcome that could result in a strikeout, a strike. I decided that the penultimate worst case was an in-play out, further justifying that it was higher in the order than a strike because although it resulted in an out, the batter was still able to make contact and put the ball in play. With most League pitchers throwing pitches in the upper 90s (mph), making contact is quite impressive- much more than either taking a strike or whiffing. Foul was the next in line, as though in some situations it added a strike to the count against the batter, it didn't terminate the PA- it could perhaps be considered the most neutral result of a PA pitch outcome. It was followed by ball, the first PA pitch outcome that actively impacted the batter of a PA in a positive way. The remaining four outcomes were the easiest to order, single followed ball, then double, triple, and home run, as the more bases the batter advanced on the outcome, the better the result. Therefore, I decided the final ordinal order of PA pitch outcomes to be (denoted by their official factor level names): "Strike" < "InPlay_Out" < "Foul" < "Ball" < "Single" < "Double" < "Triple" < "HR".

Using the above order of response variables, as well as both the treatment and game-state features specified previously, I derived the following proportional odds model: (Glickman, [2023](#))

First, consider each ordered PA pitch outcome to be associated with the following values:

Strike = 1, InPlay_Out = 2, Foul = 3, Ball = 4, Single = 5, Double = 6,

Triple = 7, HR = 8

Allow an observed PA pitch outcome Y to be furthered represented across the 8 levels as such:

$$Y = \begin{cases} 1 : \text{Strike} & \text{if } Y^* \leq \theta_1 \\ 2 : \text{InPlay_Out} & \text{if } \theta_1 < Y^* \leq \theta_2 \\ 3 : \text{Foul} & \text{if } \theta_2 < Y^* \leq \theta_3 \\ 4 : \text{Ball} & \text{if } \theta_3 < Y^* \leq \theta_4 \\ 5 : \text{Single} & \text{if } \theta_4 < Y^* \leq \theta_5 \\ 6 : \text{Double} & \text{if } \theta_5 < Y^* \leq \theta_6 \\ 7 : \text{Triple} & \text{if } \theta_6 < Y^* \leq \theta_7 \\ 8 : \text{HR} & \text{if } \theta_7 < Y^* \end{cases}$$

Where $\theta_1, \theta_2, \dots, \theta_7$ are fixed values.

And Y^* is an unobserved numerical response score, that can be represented:

$$Y^* = \mathbf{x}'\boldsymbol{\beta} + \epsilon,$$

where β_0 is not included in $\mathbf{x}'\boldsymbol{\beta}$ and ϵ has a standard logistic distribution

Then it extends that: $\gamma_k(\mathbf{x}) = P(Y \leq k|\mathbf{x})$, for $k = 1, \dots, 7$

And we can calculate both $\theta_1, \theta_2, \dots, \theta_7$ and β by maximizing their likelihoods to find their maximum likelihood estimates:

$$\text{logit } \{\gamma_k(\mathbf{x})\} = \log \frac{\gamma_k(\mathbf{x})}{1 - \gamma_k(\mathbf{x})} = \theta_k - \mathbf{x}'\boldsymbol{\beta}$$

Substituting in the context of PA pitch outcomes:

$$\begin{aligned} \mathbf{x}'\boldsymbol{\beta} = & \beta_0 + \beta_S \text{Strikes}_i + \beta_B \text{Balls}_i + \beta_O \text{Outs}_i + \beta_{BH} \text{Batter_home}_i \\ & + \sum_{\text{Innj} \in \text{Inns}} \beta_{\text{Innj}} \mathbb{I}_{\text{Innj},i} + \sum_{\text{Yrj} \in \text{Yrs}} \beta_{\text{Yrj}} \mathbb{I}_{\text{Yrj},i} + \sum_{\text{Men_OBj} \in \text{Men_OBs}} \beta_{\text{Men_OBj}} \mathbb{I}_{\text{Men_OBj},i} \\ & + \sum_{\text{MHj} \in \text{MHs}} \beta_{\text{MHj}} \mathbb{I}_{\text{MHj},i} + \beta_{\text{ABS}} \text{ABS}_i + \sum_{\text{PTj} \in \text{PTs}} \beta_{\text{PTj}} \mathbb{I}_{\text{PTj},i} + \sum_{\text{DFj} \in \text{DFS}} \beta_{\text{DFj}} \mathbb{I}_{\text{DFj},i} \\ & + \beta_{BB} \text{BB}_i + \alpha_B + \alpha_P \end{aligned}$$

Where:

- $\text{logit } \{\gamma_k(\mathbf{x})\} = \text{logit } \{P(Y \leq k|\mathbf{x})\}$ represents the log odds of ordered response Y to be k given $\mathbf{x}$, which has a linear relationship to the difference of the threshold of k , θ_k and the predictors, $\mathbf{x}'\beta$
- β_i s are the calculated model coefficients for the non-random features, corresponding to following key:

{S: Strike Count, B: Ball Count, O: Out Count, BH: Batter Home, Inn_j: Inning j in Inns, Yr_j: Year j in Yrs, Men_OB_j: Status of baserunners j in the PA prior in Men_OBs, MH_j: Handedness of the matchup j in MHs, ABS: ABS indicator, PT: Pitch Timer implementation version j in PTS, DF: Defensive shift limits implementation version j in DFS, BB: Bigger bases indicator}

- α_i s are the calculated model coefficients for the random features distributed $\alpha_i \sim \mathcal{N}(0, \sigma_i^2)$ and correspond to the following key:

{B: Batter IDs, P: Pitcher IDs}

- FEATURE_{ji} represents the value of feature FEATURE_j for pitch i , $\text{FEATURE}_j \in \{\text{ Strikes, Balls, Outs, Batter_home, ABS, BB}\}$
- $\mathbb{I}_{\text{FACTOR},j,i}$ acts as an indication if pitch i has factor level j for feature **FACTOR**, equals 1 if TRUE, 0 otherwise; $\text{FACTOR} \in \{\text{Inn, Yr, Men_OB, MH, PT, DF}\}$ and further corresponds with features and factor levels:

{Inn: Inning, Inn_j $\in$ Inns, Inns = {2, 3, 4, 5, 6, 7, 8, 9}; Yr: Year, Yr_j $\in$ Yrs, Yrs = {2021, 2022, 2023}; Men_OB: Men_OB, Men_OB_j $\in$ Men_OBs, Men_OB = {"Men_On", "RISP", "Loaded"}; MH: Matchup_handed_j $\in$ MHs, MH = {"LR", "RL", "RR"}; PT: Pitch_timer, PT_j $\in$ PTs, PTs = {v1, v2}; DF: Defensive_shift_limits, DF_j $\in$ DFS, DFS = {v1, v2}.

Note: In the Data Chapter I defined each of the aforementioned features to have an additional level. However, because of the way it was encoded by the model, the unrepresented level is actually incorporated as the base case and doesn't need a specific indicator. E.g., for the Pitch_timer or Defensive_shift_limits feature, "Control" is the base layer.

Note: In the model's fitting, both $\theta_1, \theta_2, \dots, \theta_7$ and β are fit relative to the data.

Note: Models were derived utilizing reference resources (Glickman, 2023) (University

of California Los Angeles Advanced Research Computing, [2021b](#)).

5.2.3 Model Assumptions

By using the above derived model, I am forced to work under further assumptions, in addition to the many associated with the data and derived features (defined and specified in Chapters 3 and 4). Some of the model-specific assumptions have already been partially described above, but will subsequently be further expanded on. First and foremost, by using a proportional odds model, I assume that the categorical response variable is ordinal. Because the actual model is not able to produce discrete values, the predicted probability of a value of x (in this specific case a pitch within the dataset) being equal to a specific outcome, is calculated relative to estimates being within thresholds. Specifically, the log-odds of a pitch being under an outcome is the difference between the threshold's calculated coefficient and the pitch's features multiplied by their respective calculated feature coefficients.

The model is able to calculate different threshold coefficients by assuming the parallel regression assumption holds, meaning that even with the different levels of response variable, the relationships between each pair of outcome groups is the same. Rather than each level having its own coefficients that predict it directly, the only difference between levels is the threshold coefficients- the entire model's validity rests on this assumption, and calculations would not be possible if it didn't hold. This is also a key difference between a proportional odds model and a multinomial logistic regression model, whose coefficients can vary depending on different response levels. An intercept term is not included with the rest of the feature predictors, as the different threshold coefficients are used in its place. Additionally, in order to build the layered relationship we assume that the errors of the model have a logistic distribution.

In order to interpret the model's feature coefficients (rather than threshold coefficients), we consider that for any k , the odds of $Y > k$ with a unit increase in x_j while holding everything else constant, is multiplied by a constant of $\exp(\beta_j)$ (Glick-

man, 2023). Therefore, any β_j values < 0 decrease the odds, as $\exp(< 0) < 1$, while any positive β_j values positively scale the odds of $Y > k$, as $\exp(> 0) > 1$. A β_j equal to 0 would not affect the odds, as $\exp(0) = 1$. Because we calculate logit $\{\gamma_k(\mathbf{x})\}$ as $\theta_k - \mathbf{x}'\boldsymbol{\beta}$, it extends that the greater the $\mathbf{x}'\boldsymbol{\beta}$, the greater probability Y is a higher number.

The thresholds can be interpreted in the same capacity a intercept might be in a regular logistic regression model, as if you substitute the base case in which $\mathbf{x}' = 0$ it into logit $\{\gamma_k(\mathbf{x})\} = \theta_k - \mathbf{x}'\boldsymbol{\beta}$, the function returns probability of each event relative to the base case. The differences between the calculated probabilities of the base case can only be the differences between the event probabilities in the base case, as in non-base case situations, other feature coefficients become active and thus adjust it.

Additionally, my adaption of a proportional odds model incorporates random effects. As previously described, the “Batter_id” and “Pitcher_id” are fit as random effects. They assume to have normal distributions with zero means, and while the model fits the $\theta_1, \theta_2, \dots, \theta_7$ and $\boldsymbol{\beta}$ values, it also finds the variance that best describe each random effect term’s normal distribution. Greater variances indicate that the feature being controlled for as a random effect has a greater impact on the value.

5.2.4 Model R Specifics

Once the data of each specific was cleaned and ready for model fitting, I imported all of the seasons into an R file. Here, I combined all twelve into one, and did a final check of variable type and unique values. I double checked the order of the response variable as well.

To fit the model, I used the “clmm”, cumulative link mixed function, from R’s “ordinal” package (Christensen, 2023a). This let me fit the proportional odds model while concurrently considering the intended random effects. I additionally specified the algorithm used to maximize the likelihood- I choose “nlminb”, which is an acronym for “nonlinear minimization subject to box constraints”, and after thorough research,

was indicated to be the best method to use relative to the complexity of my model and the amount of data I was attempting to consider (Christensen, 2023b). I ensured the calculation of the Laplace matrix for the random effects was disabled at every iteration, to save computation power and increase fitting efficiency. Continuing these intentions, I also increased the absolute tolerance value (increasing it to 1) and maximum absolute gradient of the inner optimization (to .1 from 1e-4), decreased the max number of Newton updates of the inner optimization (from 50 to 20), as well as the maximum number of Newton number of overcorrections. Though my models still indicated failure of “perfect” convergence, the aforementioned alterations were settled on following a significant amount of trial and error, in which the majority of models didn’t converge at all and quite literally ran for days.

5.3 Model Results

5.3.1 Overarching Model

I fit my first model on the entirety of the data, which had a total of 7,627,127 rows. The eight different PA pitch outcomes were represented in the data as follows:

	Strike	InPlay_Out	Foul	Ball	Single	Double	Triple	HR
Count	2137480	874110	1361755	2808388	290744	89197	9380	56073

Table 3: PA Pitch Outcomes in Full Data

There are many more strikes, balls, and fouls relative to the number of doubles, triples, and home runs. However, this was expected, as most PAs that end in any version of hit have pitches thrown prior in the PA. In fact, only 3% of the empty counts in the data have the next pitch- which is the first of the PA- result in a hit. I considered sampling from the data in an attempt to balance the classes, however ultimately decided that I would rather stick to the original plan of fitting the model on the entire dataset to see how the model would account for the disparities.

5.3.1.1 General Takeaways

After approximately three days of running, my “clmm” model fit on the entire dataset did not converge. Rather, it stopped iterating once it hit its default maximum number of iterations. Though it could be considered somewhat disappointing, I fully expected this result. I ran a trace on the model as it iterated, in order to track its progress. At each of the 150 iterations that it ran, contrary to the model’s internal threshold, I observed convergence in respect to the coefficient values as well as log likelihood (the final value of which was -1,141,258, but, as I am not fitting another version of this model, doesn’t really hold further significance relative to this analysis).

The model failed to converge relative to its gradient value, as it was too high to make its convergence. While I lowered the gradient tolerance to be .1, after the model’s final iteration, the gradient was 9,149 (no wonder it didn’t converge!). This number is quite high, but it is not cause for concern. The model was attempting to navigate over seven million rows of data that could generally be considered quite sparse relative to the classes. Not to mention, the model I was trying to fit was quite complex, ending up with 35 unique coefficients between the expanded factor features and thresholds, I also I didn’t initialize any starting values (on account of not having any significant prior information to condition on besides domain knowledge), and I did not impose any regularization. Regardless of the model’s convergence, it proved to provide valuable information relative to fitted coefficients, the role of the random effects, and threshold coefficients.

In addition, when fitting a “clmm” model, though there is access to a “fitted.values” feature that returns the probability of the conditional mode subevent relative to each data entry, there is no “predict” function to extend the model. Therefore, even if I had intended to explore further accuracy evaluations of the model, there was no way to predict on data aside from the data that was considered in training. Such limitations further solidify the decision to use the fit of the model’s feature coefficients, random effects, and threshold coefficients in terms of situation analysis rather than prediction.

5.3.1.2 Coefficient Analysis

Predictor	Estimate	Std. Error	$\mathbb{P}(> z)$
Strikes	0.1823590	0.0008153	< 2e-16
Balls	-0.0749543	0.0007217	< 2e-16
Outs	-0.0077747	0.0008312	< 2e-16
Men_OB.L	0.0237092	0.0028038	< 2e-16
Men_OB.Q	-0.0383186	0.0022819	< 2e-16
Men_OB.C	0.0022978	0.0016370	0.16041
Batter_homeTRUE	0.0199665	0.0013238	< 2e-16
Matchup_handedLR	0.1289105	0.0045153	< 2e-16
Matchup_handedRL	0.1018302	0.0041064	< 2e-16
Matchup_handedRR	0.0266532	0.0055717	1.72e-06
Inning2	0.0139925	0.0027928	5.44e-07
Inning3	0.0364002	0.0027555	< 2e-16
Inning4	0.0494518	0.0027740	< 2e-16
Inning5	0.0486130	0.0027918	< 2e-16
Inning6	0.0378294	0.0028499	< 2e-16
Inning7	0.0245124	0.0029781	< 2e-16
Inning8	0.0088495	0.0030917	0.00421
Inning9	-0.0143351	0.0034037	2.54e-05
Year2021	0.0061245	0.0044004	0.16399
Year2022	-0.0007298	0.0044194	0.86885
Year2023	0.0052515	0.0067336	0.43545
LeagueAAA	0.0295880	0.0047440	4.46e-10
LeagueMLB	-0.0593270	0.0046426	< 2e-16
ABSTRUE	0.0258381	0.0042642	1.37e-09
Pitch_timerv1	-0.0486716	0.0059964	4.79e-16
Pitch_timerv2	-0.0365360	0.0073163	5.92e-07
Bigger_basesTRUE	0.0116842	0.0067259	0.08236
Defensive_shift_limitsv2	0.0282687	0.0057829	1.02e-06

Table 4: PA Outcome Model Coefficients

In the above table, the first column is the algorithm’s final estimate for each respective coefficient, the second value is its associated standard error the model produced during fitting, and the final column is each coefficient’s p-value. The p-values were calculated directly from another column the model returned that gave z-scores, however I decided against including it. I did this because z-scores can easily be misinterpreted across coefficients, as they are not standardized among the coefficients, while the p-value is calculated from it in an attempt to standardize significance.

As discussed in the model definition above, the coefficients of the proportional odds can be interpreted as factors relative to the odds of $Y > k$ for any k . If we assume to hold everything else in the model constant, a one unit change in x_j can be interpreted as a

multiplicative factor of $\exp(\beta_j)$ to the odds of $Y > k$ for any k . By the characteristics of the $\exp()$ function, negative β_j values act as fractional scales, therefore decreasing the odds, while positive β_j increase the odds.

The p-values the model returned were calculated from a Ward hypothesis test that considered the test statistic to be the z-score, which was found for each coefficient by dividing its calculated estimate by its respective standard error. The resulting quotient therefore reflected how many standard errors the estimate was from zero. The higher the z-score, the more standard errors the estimate is away from zero, and the more significant the fit of the coefficient. In general, we evaluate coefficients relative to how many standard errors they are away from zero because if a coefficient is close to zero, it might indicate inclusion of zero its 95% (or another threshold) confidence interval. If the interval includes zero, the analysis does not have enough evidence to suggest that the feature's coefficient could not be zero and can therefore not reject the null hypothesis which suggests indifference of the feature to the response variable.

Upon initial inspection of the table, it can immediately be seen that there are many coefficients that are considered significant. While a standard p-value threshold is .05 (therefore suggesting rejection of the null within a 95% confidence interval), as shown above, there are many p-values that are considerably more significant. Specifically, "Strikes", "Balls", "Outs", "Men.OB.Men.On", "Men.OB.RISP", "Batter_homeTRUE", all levels of "Matchup_handed", all levels of "Inning" (besides the 8th, which is only significant relative to a still solid .01 threshold), both "League" levels, "ABSTrue", both levels of the "Pitch_timer" and finally "Defensive_shift_limitsv2" have p-values that are practically zero. Each of their respective p-values indicate that they as features have somewhat of an effect on the PA pitch outcome situations.

In terms of magnitude and sign, "Strikes" and the mismatched "Match_handed" levels have the most positive values, indicating 1.2, 1.14, and 1.11 multiplicative factors to the odds of any event, when the feature equals one ("Strikes") or is indicated ("Matchup_handed") respectively- meaning they shift the odds to favor the probability

of higher ordered events when enacted. Relative to the game of baseball, this makes sense, as there is the wide belief that in mismatched batter/pitching matchups, advantage goes to the batter, which is what the model indicates. The “Strikes” coefficient is intriguing, and could perhaps be interpreted in the capacity that in order to get the pitch a batter favors he must take a couple strikes first.

The features associated with negative coefficients with the greatest absolute value are those for “Balls”, “LeagueMLB”, and “Pitch_timerv1”. They are thus all associated with the factors that decrease the odds of $Y > k$ for any k , with the respective factors of .929, .942, and .952. While such fractional factors don’t seem that significant, those are only their interpretations when holding all other features constant. Though the relationship between the terms in a proportional odds model is more complicated, when there are multiple changes in features, their effects cumulate quickly. The decrease in balls is somewhat counterintuitive, as I would have expected as the number of balls increased the odds of a higher numbered outcome would too, as pitcher might continue to throw balls and walk the batter or be forced to give up a strike that leads to a hit. However, instead it could indicate that pitchers throw more strikes with higher ball counts, or batters tend to swing when they are ahead in the count. To know decisively more investigation would need to be done. It makes sense that of the three leagues (Double-A accounted for in the base-case/thresholds), the factor level associated with MLB has the most negative coefficient, implying it is the hardest to get high number outcomes in MLB relative to the other two, consistent with the increase in talent level. Associated interpretations of “Pitch_timerv1” will be discussed with the remainder of the rules.

The most insignificant features were indicated to be the coefficient associated with loaded bases, as well as the differences between years. Loaded bases makes sense, as there are only 207,724 instances of pitches in which the bases are loaded in the data, which makes up roughly 2.7% of all entries. Therefore, there was not enough information for the model to adequately define the effect that stolen bases had on PA

pitch outcomes. The lack of significance relative to the years' coefficients is most likely explained by how closely related they are to the rules indicators, as change between years might have just been encapsulated by those coefficients instead.

Finally, the coefficients of the rules all prove to be pretty significant- besides the one that indicates if "Bigger_bases" were implemented or not. When the ABS system was used, it proved to slightly favor the outcome of the batter, perhaps suggesting that traditional umpire systems favor the pitcher. Both versions of the pitch timer indicated to favor the pitcher, having negative coefficients, therefore suggesting when they are enacted a batter's odds decrease, lower numbered outcomes become more likely. A different explanation for this, as mentioned above, could be that some of the year effects got mixed into the pitch timer coefficients, thus reflecting the fact that pitchers have continued to up their velocities over the years, making it much harder to make contact and get on base- there have been decreasing trends relative to batting average (which, as discussed, the rules attempted to switch in the opposite direction). Lastly, the defensive shift limits have a significant positive coefficient, indicating that when the second version of the rule was in use it ever so slightly amplified the chances of a favorable outcome for the batter. The coefficient distinguishing the first version of the rule, however, was dropped from the model for making the model rank deficit, i.e., it was too similar to other coefficients. My inference is that it was too similar to the control version of the rule, as the first version of the rule was only implemented in Double-A for a single season, so there was a relatively low sample size of the data, and therefore any impact it had on PA pitch outcomes were negligible. It follows that the rule was made more aggressive following the first version, and that the more aggressive version had enough of an impact to be considered significant.

5.3.1.3 Threshold Coefficient Analysis

The threshold coefficients of the PA Outcome model fit on the entire dataset are:

First, all of the associated z-value (aka z-score) are large in absolute value. While I

Transition	Estimate	Std. Error	z value
Strike $\rightarrow$ InPlay_Out	-0.739470	0.006619	-111.71
InPlay_Out $\rightarrow$ Foul	-0.215475	0.006617	-32.56
Foul $\rightarrow$ Ball	0.511638	0.006618	77.31
Ball $\rightarrow$ Single	3.000412	0.006772	443.03
Single $\rightarrow$ Double	4.098503	0.007100	577.27
Double $\rightarrow$ Triple	4.969607	0.007729	643.01
Triple $\rightarrow$ HR	5.125482	0.007896	649.10

Table 5: Threshold coefficients

am eating my words from before by considering them without their respective p-values, it doesn't really make sense to consider extensive p-value assessment of the threshold coefficients. This is both because of their miniscule standard errors, as well as the fact that it is unnecessary in the capacity in which I am interpreting the model, for evaluation rather than prediction.

In the base case, the results have the following probabilities: (Strike: .323, InPlay_Out: .123, Foul: .179, Ball: .325, Single: .03, Double: .01, Triple: .004, HR: .006).

While some of the probabilities are quite low, they do seem to be reflective of the data. Like the originally given sub-classes of all of the data, the thresholds suggest higher probabilities of strike, balls, and fouls relative to the rest of the groups. Just as the entire data has more home runs than recorded triples, the home run outcome has a slightly higher probability than a triple. Given, the probabilities only describe the sub event probabilities of the base case- in which all coefficients are zero, meaning its 2014, Double-A, first inning, zero strikes, balls, outs, no men on base, control versions of all the rules- but, alas, they make relative sense. Recall how I mentioned that in empty count situations, only 3% of the next pitch resulted in any type of hit. Adding up the threshold probabilities for all hits from the base case, it totals to .05, and the relative difference could be explained other features not enacted in the base.

Groups	Variance	Std.Dev.
Batter_id	0.013476	0.11609
Pitcher_id	0.009014	0.09494
Number of groups: Batter_id 5104, Pitcher_id 4894		

Table 6: Random Effects of the PA Model

5.3.1.4 Random Effect Analysis

The fitted variance and standard deviation of both the "Batter_id" and "Pitcher_id" random effects in the overarching PA pitch outcome model suggest minimal effect of the batter and pitcher independently on the outcome of the PA pitch. Given the magnitude of the data, as well as the unique number of ids- 5,104 and 4,8934, respectively- it makes sense that the variation has too many dimensions to consistently impact situations. Not to mention, this is reflective of the real game, in which players vary in their own production.

5.3.2 Sub-Model Comparisons

While the thresholds are significant to some extent, when considering sub-model analysis more is gained from understanding the deviations in feature fits among the models. Therefore, I proceed with the analysis of the count sub-models coefficient estimates and p-values.

The table containing said coefficient estimates and their corresponding p-values provides a heaping amount of information and can be perceived to be overwhelming. Because of said abundance, I am only going to highlight a few things.

First, consider the "Outs" estimate across the four different models. While indicated in the original model to be quite significant, when the scope is lowered, such significance doesn't necessarily hold. Relative to its p-value in the original model, the p-values from both the Batter's and Pitcher's count sub-models are much bigger. Note that I am not saying that the models don't consider "Outs" to be completely indifferent in the respect of predicting PA pitch outcomes, as the values still pass the .01 and .1 p-value thresholds

	Batter's		Pitcher's		Empty		Full	
	Estimate	Pr(> z)	Estimate	Pr(> z)	Estimate	Pr(> z)	Estimate	Pr(> z)
Outs	0.021509	0.009169	-0.0049707	0.09232	-0.007333	1.03E-05	-0.025522	2.25E-12
Men_OB.L	0.136181	2.93E-07	-0.0835937	< 2e-16	0.097076	< 2e-16	0.00723	0.56257
Men_OB.Q	-0.105256	1.09E-06	-0.0002599	0.974272	-0.068544	< 2e-16	-0.00669	0.50992
Men_OB.C	-0.037799	0.01415	-0.023669	4.60E-05	0.013281	4.25E-05	0.013687	0.05856
Batter_homeTRUE	0.020812	0.112671	0.0150524	0.001317	0.017266	7.34E-11	0.025409	1.08E-05
Matchup_handedLR	0.094751	0.000408	0.0367871	0.000584	0.142402	< 2e-16	0.086312	5.76E-13
Matchup_handedRL	0.054158	0.066617	-0.0001158	0.991547	0.106873	< 2e-16	0.08759	1.63E-11
Matchup_handedRR	0.044598	0.113047	-0.0086811	0.449367	0.022193	0.002218	0.028232	0.02504
Inning2	-0.018417	0.511623	-0.0220144	0.023351	0.015127	0.006678	-0.036382	0.00241
Inning3	0.049156	0.074134	-0.0128113	0.186106	0.05979	< 2e-16	-0.021874	0.06699
Inning4	0.099746	0.000283	0.0009962	0.918285	0.089722	< 2e-16	-0.031156	0.00932
Inning5	0.050066	0.067432	-0.0126102	0.197499	0.090648	< 2e-16	-0.01428	0.2327
Inning6	0.054442	0.044664	-0.0170881	0.082586	0.094871	< 2e-16	-0.029347	0.01398
Inning7	0.132913	1.04E-06	-0.0340642	0.000728	0.078473	< 2e-16	-0.022163	0.06556
Inning8	0.031195	0.259221	-0.0507851	7.84E-07	0.070684	< 2e-16	-0.038722	0.00151
Inning9	0.029344	0.333326	-0.0979722	< 2e-16	0.040056	7.75E-10	-0.060338	6.01E-06
Year2021	0.083633	0.011673	-0.035449	0.003377	0.042776	7.50E-09	-0.01789	0.20547
Year2022	0.06403	0.052589	-0.0062071	0.601425	0.024332	0.000894	-0.033597	0.01529
Year2023	0.099334	0.039386	0.0255305	0.174801	0.031274	0.005901	-0.038811	0.07093
LeagueAAA	-0.125341	9.43E-06	0.0861381	3.39E-13	0.001969	0.78415	0.022192	0.09048
LeagueMLB	-0.264624	< 2e-16	0.0415226	0.000115	-0.116339	< 2e-16	-0.072073	1.04E-09
ABSTRUE	0.197094	1.06E-10	-0.0430813	0.000648	0.060544	4.41E-16	0.035973	0.01156
Pitch_timerv1	-0.097442	0.029916	-0.0048656	0.774244	-0.072289	1.45E-12	-0.068275	0.0005
Pitch_timerv2	-0.140437	0.013462	-0.0075972	0.724834	-0.048694	0.000145	-0.035895	0.15107
Bigger_basesTRUE	0.173505	0.0002	-0.04189	0.02507	0.015918	0.154094	0.021159	0.31684
Defensive_shift_limitsv2	-0.052819	0.196741	0.0331487	0.050582	0.021399	0.032818	0.015301	0.42525

Table 7: Coefficients and p-values of the PA sub-models

respectively, they are just much greater relative to the p-value of the original model. Also, in the first model, the “Outs” coefficient estimate was quite close to zero (but not relative to its standard error!), but negative. In the batter’s count model, though the value is still quite close to zero (still not that close relative to its standard error!), it is a positive coefficient rather than a negative one, and given all other constants hold, would increase the odds of any of the outcomes by roughly 2% when it is incremented by one unit. In context of the game, its contrast to not only the overarching model, but the other count sub-model’s negative “Outs” coefficient estimate, supports a claim that if the batter is up in the count, the number of outs do not negatively impact their probable outcome as it might if they were neutral in the count or behind.

Next, the “Men_OB” coefficient. In the first model the first two ordinal levels with coefficients, “Men_On” and “RISP”, both had very low p-values, the former’s estimate was positive, suggesting that situations in which “Men_On” favored outcomes for the batter, while the latter’s was negative, suggesting favor for the opposition. Of note

is the fact that neither of them are that significant in absolute magnitude, yet both indicate coefficient significance. In the context of the count sub-models, the “Men_On” factor is decisively not significant for full counts, whereas “RISP” is not only decisively insignificant for the full count model, but the pitcher’s count model as well, its respective p-value .97.

Of the four models, the only one to indicate consistent significance across multiple consecutive innings is the empty count model. Relative to the game of baseball, this is somewhat surprising, as theoretically outcomes among different counts shouldn’t be so drastically swayed by innings. But alas, I controlled for such game-state variables to identify potential unexpected relationships like this.

Just as there was variability in other feature coefficients, those pertaining to the rules changed relative to the sub-model. The “ABSTRUE” had very low p-values for the batter’s and empty count models, both with positive coefficients. Similar to the overarching model, this suggests a potential favoring of pitchers by traditional umpire systems. However, such significance did not hold to the same degree for the pitcher’s or full count models, indicating that the ABS system didn’t make a big difference in some of the most tension heavy situations of the game- full counts.

The pitch timer coefficients did not prove as significant in any of the sub-models as it did in the original. In the sub-models, the first version of the rule was somewhat relevant in the empty and full count models, with negative coefficients of similar magnitude. The absolute value of the coefficients were actually greater than in the original, suggesting it decreases the odds when activated to a greater extent.

Additionally, the bigger bases did not seem to have much of an impact across the board. Same goes for the defensive shift limits- which had a significantly low p-value in the original model, therefore most likely effected pitches of counts not represented by the model- middle count situations rather than initial or border terminal state counts. Also of note should be the sub-model’s handling of the random effects. Note that I only included the fitted variance of each term rather than both the standard deviation and

variance to both save space and make it easier to compare among the models:

Variance	Batter's	Pitcher's	Empty	Full
Batter_id	0.04233	0.01511	0.009012	0.009424
Pitcher_id	0.01419	0.01362	0.008458	0.006322

Table 8: Variance Estimates

As was consistent with the overarching model, the random effects of “Batter_id” and “Pitcher_id” didn’t make that much of a difference relative to the outcomes of the models. Because of the many different id combinations, as well as the other features, it makes sense that the “Batter_id” and “Pitcher_id” take a backseat in respect to their effects on the models. However, it was still important to control for them as random effects, because to overlook this fact would be to invoke the wrong assumption that they make absolutely no difference.

5.3.3 Final PA Outcome Model Thoughts

I believe that my models produced results to the capacity I expected. I executed the modeling and analysis with the underlying understanding that the model was attempted to fit on over seven million rows of data, which in addition to the model’s layered complexity, was practically impossible for it to properly do. Despite this, however, I believe the models fit coefficients that fit the data relatively consistent to what they should have been expected to do.

In the overarching model, the base-case probabilities were reflective of the overarching population. In all five models, in the context of their respective situations, their respective feature coefficients could be interpreted relative to the game in a way that made sense.

The overarching model fit multiple rule indicators with significance, suggesting that batters had a harder time adapting to the pitch timer while the second version of the defensive shift limits worked in their favor. Across the sub-models, the ABS indicator proved relatively significant, and the first version of the pitch timer for empty and full counts.

If I were to extend this part of my project, or perhaps rework some of it, I would change more than a few things. I would fit the model with random samples of the data to be used as representation. Doing so would decrease the model variable size, and allow me to take advantage of other statistical tests that when I initially tried to run did not converge and threw errors.

Early on in my project, I had fit an iteration of the model without the use of random effects. I ended up abandoning this model, as it was fit on accident, and I ended up changing some of the features. Rather than keeping that specific iteration of model fit, I wish I had fit another version of the final model without the random effects, which would have subsequently given me the ability to further analysis differences among it and the one I did discuss.

Finally, relative to the results of the non random effects model, I would have tried to either pursue use of the “clm” (non random effects cumulative link model) function or something similar with the intention of being able to predict specific features. While “clmm” did return the aforementioned “fitted.values” list, it only returned the probability relative to one of the thresholds, whereas I would have wanted to explore further analysis relative to all eight different defined sub-events. This also prevented me from extending the analysis in additional directions, such as evaluation on a by-player or by-player group basis, as I was unable to distinguish the outcome I wanted to calculate probabilities for.

6 SBA Model

This is the second of my two model-specific chapters, and houses details of the derivation and results of my stolen base attempts model. A stolen base attempt (SBA) is an umbrella term that describes instances where a baserunner records a successful stolen base (SB) or is caught stealing (CS). In order to understand the independent impacts of the pitch timer, defensive shift limits, and bigger bases relative to stolen base attempts, I attempt to model the situation on a by-pitch basis.

This chapter begins with a brief overview of preceding work that tried to model SBAs, followed by an introduction to my own methods, corresponding relevant summary statistics, and finishes by discussing and interpreting overarching and sub-model results.

6.1 Previous Work

Compared to the other in-game situation of focus, PAs, SBAs have a less extensive portfolio of research that is publicly accessible. After an extensive search, there were only a couple articles I believed to be relevant and worthy discussing as they relate to my own attempt.

Stanley (2023), attempted to predict the probability of a successful SBA- i.e., a SB (Cade Stanley, 2023) He used by-pitch data from the 2018 MLB season, which he acquired by merging Retrosheet (a well-regarded MLB play-by-play source) records with an open source found on Kaggle. The basis of his project was to predict the success of a SBA, therefore the only pitches he considered were SBAs- he filtered out all other pitches from the 2018 season (the great majority). Stanley trained both a logistic regression model and a random forest model in an attempt to answer his question, with the models producing very similar results. He considered additional features including but not limited to the number of outs at the pitch of the SBA, if the count favored the batter or pitcher, the matchup's handedness, and both the batter and pitcher's SBA success rate from the previous season, in 2017. While both of his models produced a

higher accuracy than a random model, some questions remain about his methodology. Stanley did not specify if his data only included regular season games, the number of total data points that he used in his training, nor how many of them resulted in a successful SBA. His models prove to be successful in classifying successful SBAs but fail to distinguish ones that failed. Further, the accuracies are around 75%, and as the 2018 successful stolen base rate was 72%, he might have just blanketed most attempts as successful. Without understanding the additional parameters the models were trained on it is hard to derive further conclusions from the project. Not to mention, because the scope of the project is limited to predicting only the probability of a SBA's success, any interpretations relative to his results only pertain to situations in which players successfully steal, rather than in the context for which any SBA might occur.

Houghtaling (2022) attempted to understand the impact of rules tested in the minor leagues in 2021 on stolen bases (Zachary Houghtaling, 2022). The project's intentions were the publicly published source I found to be most similar to my intentions, though my methods and execution significantly differed from Houghtaling's. Houghtaling considered the bigger bases in Triple-A, the aforementioned "Step-Off" rule that was tested and subsequently retired in High-A, and a rule in Low-A that limited pitchers to two pickoff attempts per PA (which would be incorporated into the pitch timer the next year). Double-A and MLB were used as control leagues, where no rules were implemented. Because of the pandemic, Houghtaling quantified the effects of the rules by comparing differences in stolen base trends from the 2019 and 2021 seasons. Houghtaling investigated different questions in response to a survey he conducted in an attempt to gauge the interest impact of stolen bases. He evaluated the success rate of stolen bases as well as the frequency of stolen base attempts, but the latter scope only considered pitches that resulted in SBAs or home runs. Many different analysis techniques were implemented, including basic frequentist analysis, calculating bayesian posterior probability, and ANOVA tests. Houghtaling's results indicated that the rules had causal relationships with both the number of successfully stolen bases and the number of stolen

base attempts. However, some of the assumptions he used were questionable, such as using different leagues to calculate posteriors for another's prior without controlling for difference in league, not controlling for different situations in which the SB or SBA occurred (gamestate, etc), and ignoring all other pitches that were not SBAs or home runs.

6.2 Model Definition

Contradicting the previously discussed projects, rather than attempting to predict which SBAs would be successful and result in SBs, I expand my scope to consider all pitches, fitting my model in an attempt to evaluate game-state characteristics that might incentivize any kind of SBA.

6.2.1 Model Intentions

It should also be noted that while the aforementioned work attempted to maximize the accuracy of their respective models, I do not make that a focus of my own. Instead, because of the magnitude of data considered over time, as well as the intention of isolating the impact of the pitch timer, defensive shift limits, and bigger bases relative to SBAs, I simply use model fitting as an analysis technique. I do not even separate testing and training data, and use all available data to fit the model. Fitting the model structure (defined in the next subsection) to the overarching data and sub-groups allows me to interpret their coefficients and characteristics in the scope of how the model weighed and accounted for the 2023 MLB new rules.

6.2.2 Model Formula

In my attempt, I decided to use a logistic regression model that controlled for random effects. Logistic regression, as briefly discussed, is a natural choice to use when modeling binary outcomes. It makes sense to use in the context of my project, as each pitch in my data can be indicated to have an SBA or not.

Therefore, I used the following model derivation:

$$\mathbb{P}(\text{SBA}_i = 1) = \frac{1}{1 + e^{-x}}$$

$$\begin{aligned} \ln\left(\frac{\mathbb{P}(\text{SBA}_i = 1)}{1 - \mathbb{P}(\text{SBA}_i = 1)}\right) &= x \\ &= \beta_0 + \beta_S \text{Strikes}_i + \beta_B \text{Balls}_i + \beta_O \text{Outs}_i + \beta_{\text{BH}} \text{Batter_home}_i \\ &\quad + \sum_{\text{Innj} \in \text{Inns}} \beta_{\text{Innj}} \mathbb{I}_{\text{Innj},i} + \sum_{\text{Yrj} \in \text{Yrs}} \beta_{\text{Yrj}} \mathbb{I}_{\text{Yrj},i} + \sum_{\text{AB_Pj} \in \text{AB_Ps}} \beta_{\text{AB_Pj}} \mathbb{I}_{\text{AB_Pj},i} \\ &\quad + \beta_{\text{ABS}} \text{ABS}_i + \sum_{\text{PTj} \in \text{PTs}} \beta_{\text{PTj}} \mathbb{I}_{\text{PTj},i} + \sum_{\text{DFj} \in \text{DFS}} \beta_{\text{DFj}} \mathbb{I}_{\text{DFj},i} + \beta_{\text{BB}} \text{BB}_i \\ &\quad + \alpha_B + \alpha_P + \alpha_F, \end{aligned}$$

Inns = {2, 3, 4, 5, 6, 7, 8, 9}, Yrs = {2021, 2022, 2023}, AB_Ps = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10},

PTs = {v1, v2}, DFS = {v1, v2}

Where:

- **SBA_i is the response variable of pitch i**, 1 if it had an associated SBA, 0 otherwise
- **$\ln\left(\frac{\mathbb{P}(\text{SBA}_i=1)}{1-\mathbb{P}(\text{SBA}_i=1)}\right)$ represents the log odds of an SBA**, which has a linear relationship with the predictors
- **β_0 is the intercept**, the log-odds of $\mathbb{P}(\text{SBA}_i = 1)$ when all the features/feature indicators are equal to zero
- **β_i s are the calculated model coefficients for the non-random features**, corresponding to following key:

{S: Strike Count, B: Ball Count, O: Out Count, BH: Batter Home, Inn_j: Inning *j* in Inns, Yr_j: Year *j* in Yrs, AB_P_j: Number of Pickoffs *j* in the PA prior in AB_Ps, ABS: ABS indicator, PT: Pitch Timer implementation version *j* in PTS, DF: Defensive shift limits implementation version *j* in DFS, BB: Bigger bases indicator}

- **α_i s are the calculated model coefficients for the random features** distributed $\alpha_i \sim \mathcal{N}(0, \sigma_i^2)$ and correspond to the following key:

{B: Batter IDs, P: Pitcher IDs, F: First baserunner}

- **FEATURE_{ji}** represents the value of feature **FEATURE_j** for pitch **i**, $\text{FEATURE}_j \in \{\text{Strikes, Balls, Outs, Batter_home, ABS, BB}\}$
- **$\mathbb{I}_{\text{FACTOR},j,i}$** acts as an indication if pitch **i** has factor level **j** for feature **FACTOR**, equals 1 if TRUE, 0 otherwise; $\text{FACTOR} \in \{\text{Inn, Yr, AB_P, PT, DF}\}$ and further corresponds with features and factor levels:

{Inn: Inning, $\text{Inn}_j \in \text{Inns}$, $\text{Inns} = \{2, 3, 4, 5, 6, 7, 8, 9\}$; Yr: Year, $\text{Yr}_j \in \text{Yrs}$, $\text{Yrs} = \{2021, 2022, 2023\}$; AB_P: AB_pickoffs, $\text{AB_P}_j \in \text{AB_Ps}$, $\text{AB_Ps} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$; PT: Pitch_timer, $\text{PT}_j \in \text{PTs}$, $\text{PTs} = \{v1, v2\}$; DF: Defensive_shift_limits, $\text{DF}_j \in \text{DFS}$, $\text{DFS} = \{v1, v2\}$.

Note: In the [Data Chapter](#) I defined each of the aforementioned features to have an additional level. However, because of the way it was encoded by the model, the unrepresented level is actually incorporated as the base case and doesn't need a specific indicator. E.g., for the Pitch_timer or Defensive_shift_limits feature, "Control" is the base layer.

6.2.3 Model Assumptions

Many of the assumptions I consider as they pertain to the data and features chosen were discussed in the Data chapter. However, when I chose to use a mixed-effects logistic regression model, many more were introduced. First, by definition of logistic regression, I assume the relationship between predictor variables and the log-odds of the response variable is linear on the logit scale. Thus, the coefficients are interpreted in the respect that they lead to changes in the log-odds of an SBA. I assume the model's errors are independent from each other, and that all of the features that I include do not have issues with collinearity- none of them perfectly correlate.

Though I thought it was important to initially include, I have concerns about the "AB_pickoffs" feature being too strong of a feature relative to the response variable. In addition to the fact that it is not a typically available stat for a PA, I was motivated to fit versions of each sub-model that do not consider "AB_pickoffs," and compare results.

6.2.4 Model R Specifics

Following the cleaning and final preparation of each league/year combination- completed individually and in python s (methods discussed specifically in the Data chapter)- I combined the twelve seasons in R, confirmed final variable types and unique values, and proceeded with my modeling. In order to fit my logistic regression mixed-effects model, I used the “glmer” function from the “lme4” package which provides tools for fitting a wide array of linear mixed-effects models that have been implemented using R’s S4 class level (Bates et al., 2015).

I passed in “NLOPT_LN_BOBYQA” to the algorithm parameter in an attempt to optimize the model. The first part of the parameter specifies the method’s implementation source, while “BOBYQA” is the name of the actual solving method, “Bound Optimization by Quadratic Approximation”. I choose to use this specific optimization algorithm because of the mass amount of data I trained the model on (specified in the next section), as BOBYQA does not need to calculate the derivatives to find the optimal values (*lme4 performance tips*, n.d.). To ensure “glmer” did not try to override this, I specifically turned off derivative calculations. I did not pass in specific bounds to BOBYQA, as though the deviance values after each iteration was quite high, there was relatively good convergence for all ten of the variations of the SBA model I ended up fitting. The deviance represents the quality of fit of the model, and while I did want to minimize this to the best of my ability, I focused on getting convergence from the model, because as discussed the intention of my model was not to predict but rather to analyze. Convergence came after much trial and error, in which I continued to adjust algorithm parameters for the model fit on the entirety of the dataset until I was content with the convergence. To do so, I set a numeric tolerance for declaring convergence relative for “glmer” to .1 (rather than its 1e-7 default), and if there was a change in deviance of 1 or less (up from the default of 0) for BOBYQA.

6.3 Results

6.3.1 Overarching Model

The overarching model, in which I considered all available pitches from the four seasons across the three leagues that had a singular baserunner on first base, was fit on a total of 1,577,820 rows. Of the rows, 32,078 were instances in which a SBA occurred, leaving 1,545,742 in which one was not. I want to recognize from the start that this proportion is not ideal, as $32,078/1,577,820$ reduces to reveal that SBAs were recorded for just about 2% of the data. While I considered sampling from the data in an attempt to rebalance the classes (and will discuss this further at the end of this chapter), I wanted the model to fit a dataset that was as reflective of situations in which SBAs might occur as possible, in an attempt to find any true indicators that might differentiate them from the rest of the population.

6.3.1.1 General Takeaways

Statistic	Value
AIC	287919.8
BIC	288361.5
Log-Likelihood	-143923.9
Deviance	287847.8
df.resid	1577784

Table 9: Evaluation Statistics of Overarching SBA Model

As I hinted before, initial indicators suggested that my model did not do a great job of fitting the data. It produced immensely high AIC (Akaike information criterion), BIC (Bayesian information criterion), and deviance values, as well as a very negative log-likelihood, as shown above. In an ideal world, the three former values should be minimized while the latter maximized, thus this model does not indicate great fit. The residual degrees of freedom is also high but that was expected, as it is just the number of observations minus the number of coefficients being estimated, and I have over 1.5 million rows of data- this high number also doesn't suggest much relative to the data,

it rather indicates the number of independent pieces of information available during the model fitting (University of California Los Angeles Advanced Research Computing, 2021a).

Actual	Predicted	
	0	1
	FALSE	TRUE
FALSE	1545657	85
TRUE	32034	44

Table 10: Conf. Mat for .25 Threshold

Actual	Predicted	
	0	1
	FALSE	TRUE
FALSE	1526760	18982
TRUE	28854	3224

Table 11: Conf. Mat for .1 Threshold

Further, when working under the default .5 threshold to consider SBAs, my model blankets every single pitch as a non-SBA. Therefore, its accuracy is the same as a native model, roughly 98%! This would be great if it actually identified SBAs. Alas, the highest probability of an SBA my data returns is .36, with only 22,206 of the 1,577,820 (1.4%!) rows returned with 10% or greater probability. It extends that the density distribution of the returned probabilities is highly right skewed. When I decrease the threshold of classification to .25 and .1, respectively, though the model is able to properly classify some SBAs, it comes with the tradeoff of many false-positives (as shown in the confusion matrices above).

Although initial reaction to the descriptive statistics might not be favorable, its intention should be kept in mind. Rather than be used for accurate predictions, its intention was to gain understanding of the importance of the features included, which, by producing non-zero coefficients with significant p-values (as discussed in the next section) it accomplished. It is also to be used as a baseline in the comparison of other submodels.

6.3.1.2 Coefficient Analysis

The coefficients the algorithm fit to the data are:

The first column represents the algorithm's final estimate for each coefficient, the second the standard error of said estimate, and finally p-value of the coefficient calculated relative to its z-score. The algorithm also returned the z-score associated with each coefficient, though I did not include it, as it is not standardized across the coefficients

Variable	Estimate	Std. Error	$\mathbb{P}(> z)$
(Intercept)	-4.088	0.041	$< 2 \times 10^{-16}$
Strikes	-0.148	0.008	$< 2 \times 10^{-16}$
Balls	-0.243	0.007	$< 2 \times 10^{-16}$
Outs	0.128	0.007	$< 2 \times 10^{-16}$
Batter_homeTRUE	-0.028	0.012	0.016
Inning2	-0.055	0.024	0.022
Inning3	-0.012	0.022	0.570
Inning4	-0.050	0.024	0.032
Inning5	-0.072	0.023	0.002
Inning6	-0.121	0.024	$< 2 \times 10^{-7}$
Inning7	-0.202	0.025	3×10^{-16}
Inning8	-0.286	0.026	$< 2 \times 10^{-16}$
Inning9	-0.612	0.031	$< 2 \times 10^{-16}$
Year2021	-0.115	0.051	0.023
Year2022	-0.049	0.049	0.316
Year2023	0.051	0.062	0.412
LeagueAAA	-0.163	0.039	3×10^{-5}
LeagueMLB	-0.407	0.037	$< 2 \times 10^{-16}$
AB_pickoffs1	0.656	0.014	$< 2 \times 10^{-16}$
AB_pickoffs2	0.885	0.023	$< 2 \times 10^{-16}$
AB_pickoffs3	0.941	0.044	$< 2 \times 10^{-16}$
AB_pickoffs4	1.068	0.083	$< 2 \times 10^{-16}$
AB_pickoffs5	0.747	0.198	0.0002
AB_pickoffs6	0.396	0.516	0.442
AB_pickoffs7	0.673	1.033	0.515
AB_pickoffs8	-3.702	31.906	0.908
AB_pickoffs9	-3.239	43.811	0.941
AB_pickoffs10	-4.112	43.387	0.924
ABSTRUE	-0.061	0.036	0.089
Pitch_timerv1	-0.022	0.052	0.672
Pitch_timerv2	0.200	0.061	0.001
Bigger_basesTRUE	0.063	0.055	0.251
Defensive_shift_limitsv2	-0.078	0.046	0.088

Table 12: Coefficients of Original SBA Model

like the p-value and can therefore be misleading.

The estimates of the coefficients can be interpreted relative to how they add to or take away from the log-odds of an SBA relative to their values- e.g, the log-odds of a SBA for a pitch that has two strikes, given everything else holds constant, will be decreased by .296 $(-.148 \times 2)$. The standard error is a reflection of the sureness of the calculated coefficient's magnitude. Bigger standard errors suggest that there is more uncertainty in the coefficient's estimate, while smaller standard errors suggest its corresponding coefficient is a proper description of the relationship between the feature and the response variable relative to the data it was trained on.

The z-scores were calculated directly from the estimate and standard error, by dividing the former by the latter. It represents how many standard errors the estimated coefficient is from zero. The higher the z-score the better, as low z-scores might include

zero within its 95% confidence interval, which would suggest no relationship between the respective feature and response variable. The output p-values were calculated using an asymptotic Wald hypothesis test. The asymptotic Wald test relies on the assumption that as the sample size increases, the distribution of the Wald statistic converges to a standard normal distribution under the null hypothesis (in this case, the Wald statistic is just the z-score of each coefficient). It also expects the log-likelihood surface to be close to quadratic and that the z-score can be approximated by a chi-square distribution. However, the former might not hold because of potential nonlinear relationships between my predictors and response. The latter might also be affected, as there is sparse data relative to the number of SBAs, the model is quite complex, and could be mismodeling the overall relationship between the predictors and response. Therefore, while the following analysis is done in respect to the coefficient's p-values, again, the majority of it will be used as a baseline for the subsequent models.

As shown in the table, there are many coefficients with associated p-values of less than .001. Specifically, the intercept, "Strikes", "Balls", "Outs", "Inning6", "Inning7", "Inning8", "Inning9", "LeagueAAA", "LeagueMLB", "AB_pickoffs1", "AB_pickoffs2", "AB_pickoffs3", "AB_pickoffs4", and "AB_pickoffs5".

First, the intercept. The intercept considers the situation in which none of the features are active. In the context of this project, this is when the count is 0-0 with no outs or attempted pickoffs, the batter is playing away, the game is still in its first inning and is being played in Double-A in the 2014 season with no active versions of ABS, the pitch timer, defensive shift limits, or bigger bases. Plugging (-4.088) into the equation $\frac{1}{1+e^{-x}}$, the baserunner is given a roughly 1.6% chance of attempting a steal.

In terms of both "Strikes" and "Balls", as the of either increase, the probability of a SBA decreases, as both respective coefficients will subtract from the intercept. This suggests that the data reflects that baserunners are more likely to record an SBA within the first few pitches of a count rather than later. "Outs" has the opposite relationship, its positive coefficient adding to the log-odds of an SBA as the number increases. This

makes intuitive sense, as the number of outs increase, the chances a team has to get on base decrease, and teams are more desperate to get players into scoring position (second or third)- and to do so from first base one can steal second.

The final four innings of a standard 9 inning game (6th, 7th, 8th, 9th) indicate to decrease the chances of an SBA. As they escalate, the absolute value of their coefficients continue to increase, signaling that the later it is in the game the probability of an SBA decreases even more. The league coefficients follow a similar pattern, relative to talent level. As talent level increases, SBAs become less likely, the baseline situation DoubleA, Triple-A with a negative coefficient, and MLB one with an even bigger absolute magnitude.

“AB_pickoffs1” through “AB_pickoffs5” have the biggest positive coefficients relative to the rest of the pool. They also increase in absolute value as the X in “AB_pickoffsX” increases, or as the number of pickoff attempts prior to the pitch increases. This makes intuitive sense, as the more pickoffs a baserunner has incentivized the pitcher to try, the more he is showing real potential of stealing. I suspected the general factor type feature “AB_pickoffs” to be the most descriptive of when SBAs were actually tried, which as my hypothesis was proven, makes the investigation of the model without including the feature even more relevant. Also something to note is that in 2022 for the Minors, and in 2023 for all, pitchers were limited to two pickoff attempts. As shown above, the year coefficients were seemingly irrelevant. Therefore, I suspect the model that ignores “AB_pickoffs” to perhaps have more significant year coefficients.

The only coefficient that describes levels of rule implementation that is somewhat informative is “Pitch_timerv2”, which has a positive coefficient. This was the version of the rule that, in addition to the more time limit between pitches, enforced the limit of two pickoff attempts per attempt. The positive coefficient is consistent with the intentions of the rule, as after two pickoff attempts pitchers were ineligible to throw another one, and baserunners stole. Again, I suspect change within this coefficient with the removal of the “AB_pickoffs”.

As revealing as the coefficients with notable p-values are, those that fall above the significance threshold provide information as well, in the capacity that perhaps their associated features truly have no effect on SBAs. The coefficients with the highest p-values include the first couple innings with indicators (2nd, 3rd, and 4th innings), the years, the later “AB_pickoff” values, “Pitch_timerv1”, and “Bigger_basesTRUE”. The irrelevance of the early innings are interesting, specifically in contrast to how the later innings are significant- perhaps this can be explained by how game-state in the later innings are more descriptive of the result of the game, as in the early innings teams have more time to rebound. However, if this were the true case, I would have expected the coefficients of later innings to be more positive, as teams become more desperate to score. The indifference of the years, as discussed, I suspect to be because of the “AB_pickoffs”. The later “AB_pickoffs” most likely do not have enough occurrences to be truly indicative of if SBAs will occur or not. The first version of the pitch timer was less aggressive and did not have the pickoff attempt limits, thus its indifference in respect to SBAs makes sense, and proves relative success of the second version. Finally, it is surprising that even with a full season of isolated impact (Triple-A 2021), the bigger base indicator was not considered to bring much to SBAs. Though further investigation will be done relative to the sub-models, this hints that the increase in base size didn’t have as much as its intended impact relative to SBAs.

Additionally, the “Defensive_shift_limitsv1” coefficient was dropped from the model completely. Though I did indeed fit the model to consider this coefficient (and checked again to make sure once I realized it was missing), the model automatically dropped it because it was too collinear with other coefficients. It is my understanding that the issue was between “Defensive_shift_limitsv1” and “Defensive_shift_limitsControl” (which was not fit as an official indicator, rather considered in the model as a base case). This drop of coefficient in order to preserve rank efficiency further indicates that there was not enough of a difference within the data marked with the “Defensive_shift_limitsv1” and “Defensive_shift_limitsControl”, suggesting that the first version of the rule did not

have significant or perhaps any impact on SBAs. In hindsight, rather than a simple factor variable type, perhaps I should have considered an ordinal factor type, as the versions of the defensive shift limits can be seen as an escalation/

6.3.1.3 Random Effect Analysis

The random effects controlled for in the SBA model, “Batter_id”, “Pitcher_id”, and “firstID” were fit by the model to have following variances:

Groups	Variance	Std. Dev.
Batter_id	0.0007643	0.02765
Pitcher_id	0.1623628	0.40294
firstID	0.8170969	0.90393

Table 13: Random Effect Descriptive Statistics for Original SBA Model

Note that the random effects were not fit to have estimated means, but rather were each respectively fit to be normally distributed with a mean of 0. A mean of zero is used to center their effects, as because there is no prior information known about their impact, there is no other value that might make sense as center. Perhaps another iteration of this project would first run analysis relative to different player groups in order to see if there are means that fit the distributions better. But alas that is not the point of this project, and such deviation would take away from the intention of gaining a holistic understanding of the impact of the new rules rather than in respect to players.

Of the three random effects, the one to have the greatest variance among players relative to the response variable is “firstID”. Its variance (and therefore standard deviation) relative to the other two random effects is the greatest by a far margin. This intuitively makes sense, as “firstID” is the unique identification of the baserunner on first, and in the capacity of SBAs, is quite more involved than both the pitcher and batter. Certain players also tend to steal rather than others- some players will go their entire careers without even recording a single SBA. I am surprised the “Pitcher_id” is not a little higher, as I would have expected the role of the pitcher to be a little more involved. I suppose because batters (and thus potential baserunners) are more likely

to play the entire game, rather than a pitcher who will most likely get subbed out, pitchers have a lesser role in SBAs.

6.3.1.4 Comparing the "No AB_pickoffs" Model

Statistic	Value
AIC	290671.7
BIC	290990.1
Log-Likelihood	-145309.8
Deviance	290619.7
df.resid	1577794

Table 14: Evaluation Statistics of Overarching SBA Model disregarding the "AB_pickoffs" feature

This model performed even worse than the first model. Its AIC, BIC, and deviance values are higher, while its log-likelihood value is more negative. This suggests that the "AB_pickoffs" coefficient was valuable in respect to improving the model's performance.

Variable	Estimate	Std. Error	$\mathbb{P}(> z)$
(Intercept)	-4.031976	0.042843	< 2e-16
Strikes	-0.068946	0.007633	< 2e-16
Balls	-0.197851	0.006825	< 2e-16
Outs	0.125013	0.007303	< 2e-16
Batter_homeTRUE	-0.035582	0.011700	0.00236
Inning2	-0.024172	0.024194	0.31775
Inning3	0.027455	0.021928	0.21055
Inning4	-0.017583	0.023512	0.45457
Inning5	-0.052765	0.022939	0.02144
Inning6	-0.124163	0.023807	1.83e-07
Inning7	-0.232185	0.024823	< 2e-16
Inning8	-0.348320	0.026053	< 2e-16
Inning9	-0.711178	0.031483	< 2e-16
Year2021	-0.073269	0.053247	0.16882
Year2022	-0.021665	0.051325	0.67295
Year2023	0.073013	0.064752	0.25950
LeagueAAA	-0.164321	0.039823	3.69e-05
LeagueMLB	-0.412517	0.037834	< 2e-16
ABSTRUE	-0.050111	0.036476	0.16950
Pitch_timerv1	-0.040958	0.053088	0.44040
Pitch_timerv2	0.117773	0.061984	0.05743
Bigger_basesTRUE	-0.008884	0.055699	0.87328
Defensive_shift_limitsv2	-0.080632	0.046760	0.08464

Table 15: Coefficients of SBA Model without "AB_pickoffs" feature

Many of the coefficients' p-values increased, suggesting that without the addition of "AB_pickoffs", less information is known about the respective coefficients, and we are therefore less confident about their ability to help predict our response. The intercept is slightly less negative than the previous, boasting a baseline probability of 1.7% of

Groups	Variance	Std. Dev.
Batter_id	0.007145	0.08453
Pitcher_id	0.172086	0.41483
firstID	1.041482	1.02053

Table 16: Random Effect Descriptive Statistics for SBA Model without “AB_pickoffs” feature

a player stealing rather than 1.6%. The only remaining positive coefficient with a p-value that is less than .001 is “Outs”, which is only slightly different than the feature’s previous. The lack of additional positive coefficients with significant p-values indicate that almost all of the observations were automatically blanketed with no SBA.

In this version of the model, the only coefficients that became more statistically significant are “Batter_homeTRUE” and “Defensive_shift_limitsv2”. Neither estimate has an associated p-value that is less than .001, however. Also, while the p-value of “Defensive_shift_limitsv2” became more significant, the p-value of “Pitch_timerv2” increased to substantially exceed the .001 threshold of which it previously was under. While perhaps the last version of the model’s results suggested impact of the second version of the pitch timer, there is not enough evidence to indicate the same from this one. I hypothesized that the coefficients associated with the years would encapsulate more of the variety in the model, though that did not happen, and they remained relatively irrelevant.

Both “firstID” and “Pitcher_id”’s respective variances increased. This suggests that in this variant of the model, the baserunner on first base proved a more significant role relative to the SBA outcome, as did who the pitcher was.

6.3.2 Sub-Model Comparisons

As previously discussed, the count of a pitch is perhaps the most important context of by-pitch gamestate. Before looking at each of the sub-models, we must consider the number of SBAs that occurred at each, in order to prevent unnecessary model fitting.

Each of the counts (Batter’s, Empty, Full, Pitcher’s) have the following SBA count

in the entirety of the data:

Table 17: Count of SBAs by Sub-Model

Sub-Model	Total Pitches	SBAs	Proportion
Batter's	20314	149	.0073
Empty	415066	9283	.0223
Full	75937	266	.0035
Pitcher's	101496	1712	.01687

Both Batter's and Full counts have less than even a 1% SBA rate, therefore it doesn't make sense to move forward with the model nor analysis. Possible explanations for such minimal rates could include waiting for their teammate to be granted the base (Batter's count), or not wanting to take the risk in such a high-tension situation (Full count).

SBA Models fit to only Empty Count and Pitcher's Count have the following descriptive metrics:

	AIC	BIC	LogLikelihood	Deviance	DF.Residuals
Empty	80404.2	80732.3	-40172.1	80344.2	415036
Empty No_AB	81001.1	81263.5	-40476.5	80953.1	415042
Pitchers	16757.3	17043.2	-8348.7	16697.3	101466
Pitchers No_AB	16973.5	17202.1	-8462.7	16925.5	101472

Table 18: Empty Count and Pitcher's Count Model Metrics

Variance	Empty	Empty No_AB	Pitcher's	Pitcher's No_AB
Batter_id	0.001431	0.01379	0.000932	0.09292
Pitcher_id	0.298422	0.29811	0.039746	0.10976
firstID	1.406975	1.51097	0.006257	0.41175

Table 19: Empty Count and Pitcher's Count Random Effects

Consistent with the models fit on the entirety of the data, in both the empty count and pitcher's count case, the model fit including the "AB_pickoffs" feature has lower AIC, BIC, and deviance values, and a more negative log-likelihood. Given the results of the overarching model, this is not that surprising, "AB_pickoffs" further holds as a feature whose inclusion makes a tangible difference in model quality.

In both empty count models, the variance of the random effects "Pitcher_id " and "firstID " are greater than in the overarching model, suggesting that both the pitcher and baserunner on first have more of a relative impact when there is no count. The irrelevance of "Batter_id" remains. Conversely, in the Pitcher's count models, "Pitcher_id"

and “firstID” make much less of an impact. Intuitively, perhaps SBAs tried on Pitcher’s counts are done in a desperate attempt to get on base- as their variances are lower, it indicates that there aren’t typical baserunners who attempt, therefore the strategy might be used on a more situational basis.

Variable	Empty Count Model		Empty Count No AB		Pitcher’s Count		Pitcher’s Count No AB	
	Estimate	$\mathbb{P}(> z)$	Estimate	$\mathbb{P}(> z)$	Estimate	$\mathbb{P}(> z)$	Estimate	$\mathbb{P}(> z)$
(Intercept)	-4.68286	$< 2 \times 10^{-16}$	-4.5198842	$< 2 \times 10^{-16}$	-4.3468	$< 2 \times 10^{-16}$	-3.9915	$< 2 \times 10^{-16}$
Outs	0.19919	$< 2 \times 10^{-16}$	0.1965634	$< 2 \times 10^{-16}$	0.38813	$< 2 \times 10^{-16}$	0.3803	$< 2 \times 10^{-16}$
Batter_homeTRUE	-0.01479	0.50825	-0.0178443	0.422202	-0.01677	0.730362	-0.02303	0.650801
Inning2	-0.06608	0.15739	-0.0303755	0.513399	-0.2966	0.001795	-0.27581	0.005838
Inning3	0.03555	0.39467	0.0765854	0.064923	-0.15299	0.096979	-0.12534	0.19467
Inning4	-0.01419	0.75176	0.0259813	0.559906	-0.31103	0.001134	-0.27766	0.005593
Inning5	-0.02856	0.51143	-0.0032885	0.939399	-0.12415	0.177012	-0.12337	0.201016
Inning6	-0.08067	0.07452	-0.0666668	0.138106	-0.36541	0.000188	-0.39511	0.000118
Inning7	-0.14746	0.00157	-0.1473531	0.0015	-0.22138	1.94E-02	-0.2773	0.005352
Inning8	-0.27151	3.50E-08	-0.3143949	1.75E-10	-0.44446	1.22E-05	-0.53073	6.36E-07
Inning9	-0.58898	$< 2 \times 10^{-16}$	-0.6757585	$< 2 \times 10^{-16}$	-0.71984	5.20E-09	-0.88666	6.50E-12
Year2021	-0.18033	0.03689	-0.1792506	0.039647	-0.3044	0.014183	-0.2588	0.050402
Year2022	-0.08144	0.3172	-0.0594194	0.468497	-0.20358	0.081113	-0.23189	0.06371
Year2023	-0.01842	0.86441	-0.0006631	0.995122	-0.26029	0.109948	-0.31476	0.078056
LeagueAAA	-0.21679	0.00119	-0.2567259	0.000136	-0.29588	0.00269	-0.36208	0.000801
LeagueMLB	-0.39576	3.41E-10	-0.4242469	2.64E-11	-0.31215	0.000149	-0.35787	0.000126
AB_pickoffs1	0.74905	$< 2 \times 10^{-16}$	N/A	N/A	0.75656	$< 2 \times 10^{-16}$	N/A	N/A
AB_pickoffs2	0.68349	$< 2 \times 10^{-16}$	N/A	N/A	1.34255	$< 2 \times 10^{-16}$	N/A	N/A
AB_pickoffs3	0.61135	0.00259	N/A	N/A	1.58099	$< 2 \times 10^{-16}$	N/A	N/A
AB_pickoffs4	0.92286	0.01855	N/A	N/A	1.56501	4.38E-06	N/A	N/A
AB_pickoffs5	-4.11089	0.63367	N/A	N/A	1.22178	0.219024	N/A	N/A
AB_pickoffs6	-4.53567	0.88182	N/A	N/A	-2.69432	0.809931	N/A	N/A
ABSTRUE	0.1076	0.09405	0.1160415	0.071691	-0.07562	0.496845	-0.01482	0.904596
Pitch_timerv1	-0.08387	0.38628	-0.0729195	0.452043	0.21134	0.177041	0.08541	0.609999
Pitch_timerv2	-0.04047	0.71848	-0.0906962	0.419459	0.43161	0.028407	0.25926	0.228798
Bigger_basesTRUE	0.30734	0.00165	0.2614927	0.007645	0.11664	0.467361	-0.0178	0.919682
Defensive_shift_limitsv2	-0.10519	0.2015	-0.1194344	0.148336	-0.14091	0.308766	-0.09704	0.53056

Table 20: Empty Count and Pitcher’s Count SBA Coefficient Analysis

Note that because of the set values of “Balls” and “Strikes”, they were not fit as coefficients in either situation.

The intercepts of the empty count models are the most negative of the six discussed. This implies that their base cases (2014, playing away, 0 outs, Double-A, Control rules all around), have the lowest likelihood of recording an SBA. This can be extended to imply that in order for any situation variation of an empty count to record an SBA, many of the positive coefficient features must be active. Of the non “AB_pickoffs” game-state features, the only significant positive coefficient is for outs, as it was for the overarching model. In the empty count case, however, the coefficient has a greater absolute value. This again suggests that as there are more outs for the start of a PA, the baserunner is more likely to steal, as holding everything else constant, it adds positive value to the negative intercept, which when plugged into the logit function produces a higher probability than the more negative intercept. As is seen in both overarching

models and the subsequently discussed pitcher’s count models, the “AB_pickoffs” provide higher probabilities of the baserunner recording an SBA as they increase- to an extent though, as for instances with five or more attempts the coefficient is negative- perhaps the baserunner got tired from diving back to the base, but more realistically, there are just fewer instances in the data. Most notably is the positive, significant coefficient associated with the “Bigger_basesTRUE” coefficient. This is the only rule-associated coefficient to prove significant. Combined with the higher variance relative to the random effect of “firstID”, it can be somewhat inferred that after the increase in base size, there was a group of baserunners that became more comfortable attempting to steal on an empty count.

By their intercepts, the pitcher’s count models claim that an average in the aforementioned base situation (2014, playing away, 0 outs, Double-A, Control rules all around) batter with a 0-2 count has a 1.3% and 1.8% percent chance of stealing, with and without “AB_pickoffs” considered, respectively. Besides variations of “AB_pickoffs”, the only feature with both a significant p-value and a positive coefficient is “Outs”, that has similar magnitude in both, and is much bigger than the models fit on all of the data, as well as empty count. There is a great difference in the “firstID” random effect’s variance between the two pitcher’s count models, as once the “AB_pickoffs” feature is removed it increases. Perhaps baserunners that typically attempted SBAs would also record more “AB_pickoffs”, thus with the removal of the feature their effect was encapsulated by the random effect instead. In both, the four later innings indicate to be somewhat significant, and contrasting the overarching model, rather than their coefficients continuing to increase in absolute value, both model’s 7th inning has less effect than their 6th and 8th. However, the most important takeaway from the pitcher’s count models’ coefficient is that the rules play no significant role in either model. This is different from the overarching model, which had indications suggesting the relevance of the second version of the pitch timer. The insignificance of the rules, combined with the lower variability of the random effects, can be interpreted

as the rules having no effect on incentivizing baserunners to steal on a pitcher’s count—because of the lower random effects, it might suggest that situational factors might have more influence. Though I dive deeper at the end of this chapter, I’d be interested to know if there was any relationship between SBAs and the score of the game. Or, if perhaps more steals during pitcher’s counts were attempted during close games or perhaps ones that were influential to playoff results.

6.4 Final SBA Model Thoughts

Overarchingly, my developed SBA models were somewhat informative relative to the goal at hand, to attempt to understand the effects of the pitch timer, defensive shift limits, and bigger bases on the in-game situation of stolen base attempts.

The first version of the pitch timer, as well as the use of ABS, and the second version of the defensive shift limits indicated no significant impact to the situation. In the model trained on the entirety of the data, the second version of the pitch clock had impact, which as discussed earlier, is likely due to the limitation of pitchers to two pickoff attempts per PA. When fitting the model on subset counts, only the instances of empty count and pitcher’s count had enough relative successes to justify going through with the analysis. In the empty count models, there was indication of an increase in SBAs relative to the increase in base size, which in addition to the increase in the “firstID” random effects variance could suggest a successful increase in incentive of certain players to steal. The pitcher’s count models, however, suggested no significance relative to the rule coefficients. Such indifference prompts the consideration that there are other game-state or situational variables that might have an impact that I did not include. Also, the lack of the second pitch timer proving significant in the empty count (0-0) and pitcher’s count (0-2) models suggests that the rule was of impact to SBAs relative to situations in the middle of PAs, with counts that were in between the initial or close to terminal.

In all three model levels (overarching, empty count, pitcher’s count), I fit two ver-

sions of the models- one that included the “AB_pickoffs” factor feature, that summarized the number of pickoff attempts in the PA prior to the current pitch, and one without. The model that included the variable outperformed the other in terms of AIC, BIC, deviance, and log-likelihood values every time. It suggests that it was an informative feature. My reasoning for fitting models without it, however, was because I had concerns about the availability of the feature, as well as its reproductivity, as recall in the Data chapter how I described the many steps taken to create it in the first place.

If I were to re-do this model fit, or continue with this part of the project, I would take what I have learned and apply it to make the model more efficient and perhaps more successful.

I would experiment with the adding of additional features in order to incorporate the score of the game, either through a score differential feature, indicator of which team was leading, or an indicator if the game was within a few runs, as perhaps more steals would be incentivized in closer games. I would also most likely further filter out games, in order to avoid games at the end of the season that might be closer and have an impact on playoff standings, as again, perhaps teams are more willing to take the risk of a SBA if it could provide more benefit.

Beyond feature considerations, I would experiment with reworkings of the model. In retrospect, the biggest error I made in fitting the SBA model was fitting the models with imbalanced classes of response variables. With the original overarching model being fit on data that only had 2% of pitches record an SBA, I more or less set the model up for failure. The fact that the model considered the baserunner a random effect only added to this, as it took away from any big influence baserunner identity might have on SBAs.

Further, I would incorporate an element of regularization in order to add penalty terms to the coefficients. However, there was no easy way to add regularization when using the “glmer” function, and though I considered shifting and fitting my model using functions from R’s “afex” package, research suggested elongated computation time was

associated with it, and thus I traded the potential of regularization for the efficiency of “glmer”. In hindsight, I could have worked around “afex”’s computation issues by randomly sampling out subsets of the data and attempting to fit multiple sub-models. I’m sure there could have been additional features of the “glmer” function that could have helped the algorithm fit my data better, however I did come to the conclusion of the set model parameters after attempting thorough research.

While I house further internal critiques of the methods I used to fit my SBA models, some of which will be discussed in the following conclusion section of the entire project, I believe my analysis proved somewhat conclusive in respect to which rules and game-state features are consistently significant or insignificant relative to SBAs.

7 Conclusion

This is the final chapter of my project, in which I wrap everything up, briefly discussing what I would do differently and what further changes MLB could make to baseball.

7.1 MLB’s Intentions

As previously discussed to great extents, the pitch timer, defensive shift limits, and bigger base sizes attempted to make great changes relative to the game of professional baseball when they were implemented by MLB in 2023.

Specifically, the pitch timer was aimed to speed up the game and result in more action, the defensive shift limits to allow more balls to be put in play, and the bigger bases to create greater confidence in player safety to therefore incentivize more stolen bases.

My overarching PA model model produced results that somewhat supported the intentions of the defensive shift limits, as the second version of the rule was given a positive coefficient with a significant p-value. This suggested that they increased the odds of subevents associated with higher value, i.e., more hits, and more balls into play. In the same model, however, the bigger base sizes were proven to be somewhat irrelevant, and the two versions of the pitch timer rather had negative coefficients with significant p-values, suggesting that when the pitch timer versions are active, the odds of higher number events actually decrease.

In my overarching SBA model, however, the second version of the pitch timer was the only rule to have a significant p-value. Its associated coefficient was positive, thus indicated that when it was active, there were more SBAs observed. While an increase in SBAs wasn’t a direct goal of the pitch timer, its sub-rule which limits the amount of pickoffs has a great deal of responsibility for this, as discussed within my SBA models’ evaluation of the “AB_pickoffs” features. Additionally, within the empty count sub-model, bigger bases was indicated to have a positive coefficient with a significant

p-value, indicating that in some situations, the bigger bases may have indeed followed through in its intention to incentivize stealing.

7.2 Analysis Technique Adjustments

Though each of my models proved to be somewhat successful in their respective intentions, I am forced to reflect on choices I made that might have hindered them from producing further results.

First and foremost, I regret not considering interaction terms across the treatment indicators. By making the assumption that each of them were independent of each other, though the arguments hold some ground, I prevented myself the understanding of how the rules might have changed through time. My specific concern comes from the fact that in neither the PA outcome model nor the SBA model did the Years features have significant coefficients. In retrospect, I believe that adding a term, perhaps simply among just the years and rules, would have been able to account for this better.

Though I previously discussed my hesitancy in introducing interaction terms, because of the complexity of my model, I would have taken steps to reduce such prior to adding the new features. I would have played around with many different ways to reduce both the complexity of the PA Outcomes model and the SBA model. I am still set on incorporating the random effects, as I believe to not is to not properly model, however I would consider changing the scope of my data. I believe that my choice in using pitch-by-pitch data limited me, and if I were to re-approach the project, would have reconsidered its use.

By reconsidering the scope of the data I would most likely no longer have to re-balance the classes of the data, as I indicated I would in both the conclusions of the respective models if I were to try again. I would have to readjust and reconsider my game-state parameters as well, and perhaps it might lead me to more encapsulating features. I would add at least one additional feature that considered the score of the game, or perhaps an indicator of which party was leading. I would further control for

how the results of the game might affect standings. I would additionally experiment with re-defining some of my factor features as ordinal factors, to see if that might reflect their relationships better.

All in all, I think that while I made a good effort to work with the data and the problems at hand, I ended up trying to do and control for too much, which ended up backfiring. While I have been taught to work from the bottom up in regards with modeling, I jumped into considering a lot at once, and while I did get some marginal results, I think if I had built up my models, evaluating the effects of each feature piece by piece, it might've worked better to my benefit. Regardless, this project was an immense learning experience that I'm very grateful to have completed.

7.3 MLB: What's Next

I want to end on a quote that I believe sums up the public's perception on the new rules. Said in a November 2023 interview following the first completed season of their implementation, special consultant to the Commissioner of Baseball Theo Epstein claimed, "I think the best part," Epstein said, "was how the rule changes themselves faded into the background so quickly — and what came to the fore was the best part of the game itself, the action and the players showing their athleticism. That all came to the fore, and what disappeared was some dead time," (Jayson Stark, [2023](#))

At this point, Major League Baseball continues to test different rules in order to both keep up with their fans and understand the game impacts of the rules they add. Right now, they continue to test within their own Minor Leagues, as well as by partnering with the independent Atlantic League, expanding the Robo Umps, limiting pickoff attempts further, having designated pinch runners, and even playing with a rule that would make a team lose its designated hitter and force the pitcher to hit if broken (Anthony Castrovine, [2023c](#)).

With so many different rule ideas out there, as well as other eccentric ways to change the game (such as the entire concept of the Savannah Bananas), the possibilities truly

are endless.

References

- Adam Berry. (n.d.). Different theories surround green light on 3-0 count [Accessed on April 1st, 2024].
- Anthony Castrovine. (2021, May 3). Rule changes to be tested in minors this year [Accessed on April 1st, 2024].
- Anthony Castrovine. (2022, April 5). Mlb informs clubs pitchcom is approved for '22 season [Accessed on April 1st, 2024].
- Anthony Castrovine. (2023a, February 1). Pitch timer, shift restrictions among announced rule changes for '23 [Accessed on April 1st, 2024].
- Anthony Castrovine. (2023b, March 29). New rules: 5 takeaways from spring training [Accessed on April 1st, 2024].
- Anthony Castrovine. (2023c, April 18). Mlb, atlantic league team up for more experimental rules [Accessed on April 1st, 2024].
- Associated Press. (2015, January 15). Pitch clock for double-a, triple-a use [Accessed on April 1st, 2024].
- Bates, D., Mächler, M., Bolker, B., & Walker, S. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, 67(1), 1–48. <https://doi.org/10.18637/jss.v067.i01>
- Benjamin Tzeel, J. W., Ryan Lowery. (2014). The relationship between various components of a plate appearance and the probability of success. *Pamukkale Journal of Sport Sciences*, 5(3), 23–24.
- Brill, R. S., Deshpande, S. K., & Wyner, A. J. (2022). A bayesian analysis of the time through the order penalty in baseball. *Journal of Quantitative Analysis in Sports*, 19, 245–262. <https://api.semanticscholar.org/CorpusID:252873373>
- Buster Olney. (2023, January 23). Sources: All aaa parks to use electronic strike zone in '23 [Accessed on April 1st, 2024].

- Cade Stanley. (2023). Modeling the probability of a successful stolen base attempt in major league baseball [Accessed on April 1st, 2024].
- Chris Gilligan. (2023, October 20). What happened to all those stolen bases? [Accessed on April 1st, 2024].
- Christensen, R. H. B. (2023a). *Ordinal—regression models for ordinal data* [R package version 2023.12-4]. <https://CRAN.R-project.org/package=ordinal>
- Christensen, R. H. B. (2023b). *A tutorial on fitting cumulative link mixed models with clmm2 from the ordinal package*. https://cran.r-project.org/web/packages/ordinal/vignettes/clmm2_tutorial.pdf
- Committee, J. C. (2023). *Official baseball rules* (2023 Edition). Major League Baseball.
- D. A. Rosenbaum, S. W. D. (2014). Injury risk due to collisions in major league baseball. *International Journal of Sports Medicine*, 35(8), 704–707.
- Dave Adler. (2020, June 16). The longest games in mlb history [Accessed on April 1st, 2024].
- David Adler. (2023, October 2). Attendance up, game time down in record-setting '23 season [Accessed on April 1st, 2024].
- Davis Weir. (2020, August 17). Reworking and improving the outcome machine [Accessed on April 1st, 2024].
- FanGraphs. (n.d.). Fangraphs: Baseball statistics and analysis [Accessed on April 1st, 2024].
- Glickman, M. (2023). Statistics 149 – ordinal response regression.
- Green, G., D'Angelo, J., Coyles, J., Penny, I., Golfinos, J. G., & Valadka, A. (2019). Association between a rule change to reduce home plate collisions and mild traumatic brain injury and other injuries in professional baseball players. *American Journal of Sports Medicine*, 47(11), 2704–2708.
- Jayson Stark. (2022, February 21). What would happen if baseball killed the shift? [Accessed on April 1st, 2024].

- Jayson Stark. (2023, November 23). Whole new ballgame: Mlb’s new rules changed everything [Accessed on April 1st, 2024].
- Jeff Seidel. (2017, April 21). Machado slide leads to tense sox-o’s ending [Accessed on April 1st, 2024].
- Jesse Rogers. (2022, June 7). ’damn, that should be a hit’: Mlb players sound off on the infield shift – before it could disappear forever [Accessed on April 1st, 2024].
- Joe Trezza. (2022, March 14). Experimental rules to be tested during 2022 minor league season [Accessed on April 1st, 2024].
- Jonah Pemstein. (2014, December 8). Updating and improving the outcome machine [Accessed on April 1st, 2024].
- Josh Norris. (2023, March 14). Mlb outlines 2023 minor league rules changes [Accessed on April 1st, 2024].
- Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B., Bussonnier, M., Frederic, J., Kelley, K., Hamrick, J., Grout, J., Corlay, S., Ivanov, P., Avila, D., Abdalla, S., & Willing, C. Jupyter notebooks – a publishing format for reproducible computational workflows (F. Loizides & B. Schmidt, Eds.). In: *Positioning and power in academic publishing: Players, agents and agendas* (F. Loizides & B. Schmidt, Eds.). Ed. by Loizides, F., & Schmidt, B. IOS Press. 2016, 87–90.
- Lme4 performance tips*. (n.d.). <https://cran.r-project.org/web/packages/lme4/vignettes/lmerperf.html>
- Major League Baseball. (n.d.). Rule changes for 2023 [Accessed on April 1st, 2024].
- Major League Baseball. (2022, September 9). Mlb announces rule changes for 2023 season [Accessed on April 1st, 2024]. <https://www.mlb.com/press-release/press-release-mlb-announces-rule-changes-for-2023-season>
- Major League Baseball. (2023, January 31). Mlb announces game times for 2023 regular season schedule [Accessed on April 1st, 2024]. <https://www.mlb.com/press-release/press-release-mlb-announces-game-times-for-2023-regular-season-schedule>

Major League Baseball and Major League Baseball Players Union. (2022). Mlb and mlbpa collective bargaining agreement [Accessed on April 1st, 2024].

Manny Randhawa. (2023, March 24). Pitchers now able to wear pitchcom transmitters in '23 [Accessed on April 1st, 2024].

Matt Kelly. (2019, May 27). What is a defensive shift? [Accessed on April 1st, 2024].

Mike Petriello. (2019, September 27). 9 reasons why october baseball is different [Accessed on April 1st, 2024].

MiLB. (2018, March 14). Minor league baseball announces pace of play regulations for 2018 [Accessed on April 1st, 2024].

MiLB. (2024). The minor leagues: About [Accessed on April 1st, 2024].

Minda Haas Kuhlman. (2022, April 20). Minor league hitters on automatic pitch clock strikes: “unnecessary chaos” [Accessed on April 1st, 2024].

MLB. (n.d.-a). At-bat (ab) [Accessed on April 1st, 2024].

MLB. (n.d.-b). Caught stealing (cs) [Accessed on April 1st, 2024].

MLB. (n.d.-c). Plate appearance (pa) [Accessed on April 1st, 2024].

MLB. (n.d.-d). Steamer [Accessed on April 1st, 2024].

MLB. (n.d.-e). Stolen base (sb) [Accessed on April 1st, 2024].

MLB. (2014). Collisions at home plate [Accessed on April 1st, 2024].

MLB. (2021, May 6). A guide to the 2021 mlb season [Accessed on April 1st, 2024].

MLB. (2023a). Base sizes (2023 rule change) [Accessed on April 1st, 2024].

MLB. (2023b). Defensive shift limits (2023 rule change) [Accessed on April 1st, 2024].

MLB. (2023c). Field dimensions [Accessed on April 1st, 2024].

MLB. (2023d). Pitch timer (2023 rule change) [Accessed on April 1st, 2024].

MLB Baseball Savant. (n.d.). Team positioning leaderboard [Accessed on April 1st, 2024].

Nathaniel Casey. (2016). A markov approach to modeling baseball at-bats and evaluating pitcher decision-making and performance [Accessed on April 1st, 2024].

- New York Times. (2005, May 28). Cleveland stumbles into triple steal [Accessed on April 1st, 2024].
- Petti, B., & Gilani, S. (2024). *Baseballr: Acquiring and analyzing baseball data* [R package version 1.6.0]. <https://CRAN.R-project.org/package=baseballr>
- Powers, S., Hastie, T. J., & Tibshirani, R. (2017). Nuclear penalized multinomial regression with an application to predicting at bat outcomes in baseball. *Statistical Modelling*, 18, 388–410. <https://api.semanticscholar.org/CorpusID:13367346>
- R Core Team. (2024). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. Vienna, Austria. <https://www.R-project.org/>
- Sal Baxamusa. (2006, December 27). The memory remains [Accessed on April 1st, 2024].
- Tanenbaum, J. E., Knapik, D. M., Karns, M. R., Salata, M. J., Gillespie, R. J., & Voos, J. E. (2022). Evaluating the chase utley rule: The association between a rule change and collision-related injuries in major league baseball. *Orthopaedic Journal of Sports Medicine*, 10(4).
- University of California Los Angeles Advanced Research Computing. (2021a). Mixed effects logistic regression — r data analysis examples [Accessed on April 1st, 2024].
- University of California Los Angeles Advanced Research Computing. (2021b). Ordinal logistic regression — r data analysis examples [Accessed on April 1st, 2024].
- Zachary Houghtaling. (2022). Analysis of minor league rule changes effect on stolen bases [Accessed on April 1st, 2024].