Cu7 Off Comm4nd

Quantifying Pitcher Positioning on Outfield Throws

Team 93

https://cutoffcommand.shinyapps.io/myApp/

Abstract

This project delivers a data-driven look at how game context shapes the way pitchers position themselves when the ball is in play. Using heatmaps generated from anonymized minor league data, we identified where pitchers most frequently align on base hits into the outfield at the moment an outfielder releases a throw. Our analysis incorporates key variables — fielder position, fielder throw strength, exit velocity, and base runners — to quantify and recommend context-specific pitcher positioning. To support practical application, we developed an interactive interface that allows users to explore this data based on customizable scenarios.

Introduction

The stage is set. It's Game Four of the 2020 World Series. Brett Phillips steps to the plate against Kenley Jansen in the bottom of the ninth inning — two outs, runners on first and second, and a 1-2 count.

What happens next is forever etched into baseball history. Phillips laces Jansen's signature 92-mile-per-hour cutter into right-center field. Chris Taylor mishandles the ball. Max Muncy's relay throw veers towards the first-base side of home plate, and Will Smith fails to secure it. In the chaos, Randy Arozarena scrambles home, and the Rays even the series at two games apiece.

Postgame discussion focused on the obvious mistakes. Taylor was blamed for the error in center field. Muncy was blamed for the off-line throw. Smith was blamed for muffing the ball. But one aspect was missed altogether: who was meant to be backing up the catcher? Why was Jansen standing on the grass up the third-base line instead of behind him? And, how, in the biggest moment, on the biggest stage, was no one there to stop the ball from getting all the way to the backstop (Figure 1)?

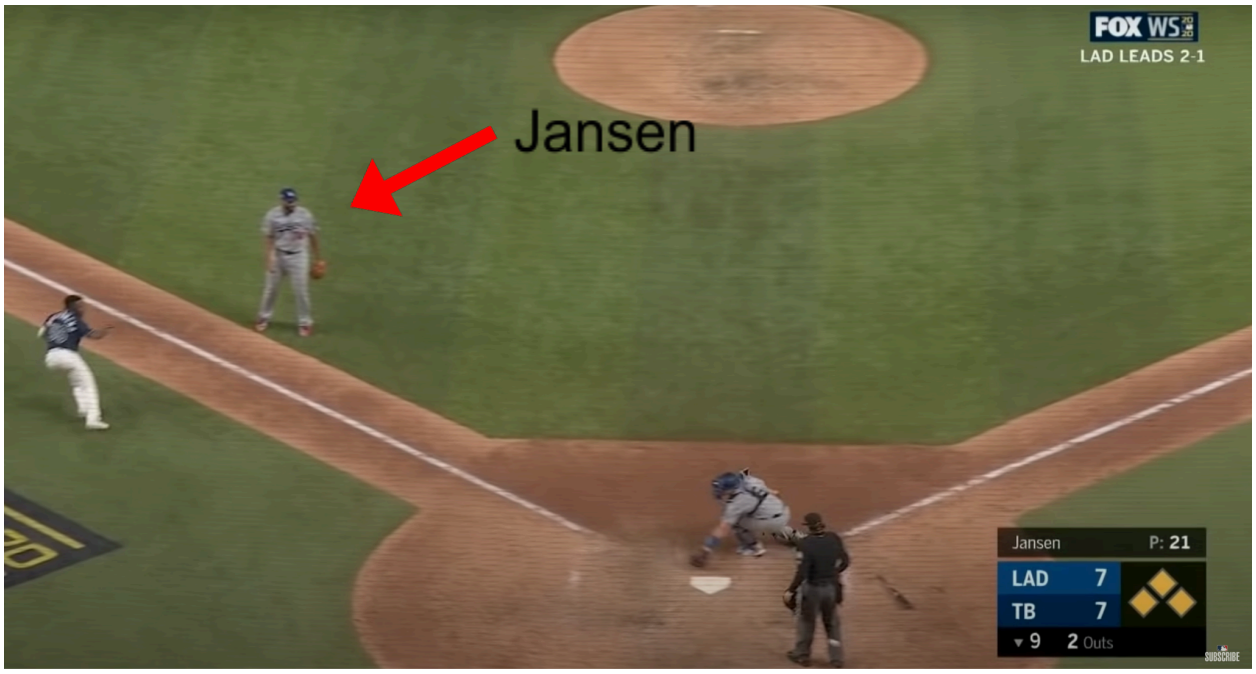

FIGURE 1: Kenley Jansen's positioning in the 2020 World Series

When you ask someone in baseball what pitchers are meant to do when a ball is hit to the outfield, the general answer you'll hear is "back up a base." But beyond those four words, there's hardly any guidance on exactly *where* a pitcher should be. One of our team members, a former high school pitcher, knows this firsthand. Growing up on the mound, he was never shown or told precisely where to stand on relays, cutoffs, or plays where he wasn't directly involved on defense. It was mainly intuition that led him to back up the catcher or move up the third base line to field a poor throw. Why? Because coaches lacked the data to offer specifics, defaulting to the same advice they received growing up: "Back up home!"

The issue is clear: there's no widely-adopted, data-driven framework for pitcher positioning. But with access to player tracking data, we can finally move beyond instinct. To this end, we built a tool that takes in key independent variables to uncover trends in professional baseball and

provide concrete, data-informed guidance on exactly where a pitcher should be in the field of play. By quantifying what's long been dismissed as a minor detail, we aim to bring clarity to a defensive strategy — one that, while often overlooked, can be the difference between a championship and heartbreak.

Data and Methodology

Our analysis is based on 247 games of anonymized minor league data that was provided by SportsMEDIA Technology (SMT). The data describes game events, ball and player positioning, and team rosters across two baseball seasons. It captures each game down to every 50 milliseconds.

We designed a function that extracts every play where the ball bounces into the outfield. Since most flyball outs result in routine tosses back into the infield, we chose to exclude these from the dataset and only observe base hits into the outfield. The function iterates through each game, identifying plays that involve these hits and filtering them based on conditions:

- Base state: 8 states of baserunners (e.g., first-and-third, bases loaded, etc.).
- Outfield position fielding the ball: Whether the ball is fielded by LF, CF, or RF.
- Exit velocity: Separated into "Hard," "Soft," or "All" conditions, with 86 MPH designated as the threshold through trial and error to maximize data points for each combination of inputs.
- Outfielder arm strength: These intervals were selected to separate outfielders' throw velocities into performance tiers.
    - <70 mph
    - 70-85 mph
    - 85+ mph
    - All Velocities

Finally, we filter for pitcher positioning at the time of the throw from the outfield and create our CSV files using their x and y coordinates.

User Interface

Simplicity was central to the design of our Shiny application: Cu7 Off Comm4nd — a nod to Kenley Jansen's jersey number 74. Our focus was on building a tool that would be intuitive and accessible to a wide range of users, from casual fans to veteran coaches to the little leaguers they coach. We prioritized easy use and speed — not only with quick input and loading times, but also in the sense that our user doesn't need to sift through dense tables or complex visualizations to find what matters. Instead, we envision a coach pulling out a phone mid-game, tapping through a few quick inputs, and instantly showing a pitcher where they should have been standing on a certain play. This efficiency is important because we understand that how to back up a throw after allowing a line drive double in the gap isn't something pitchers want to spend a ton of time thinking about. Limiting that time to just a quick few words from the coach allows the pitcher to spend more time resetting and focusing on the next pitch.

When a user opens Cu7 Off Comm4nd, they're greeted with a super simple layout: an infield with bases, three gloves (representing three outfielders), and two multiple-choice options where they can select hit quality and outfielder throw velocity.

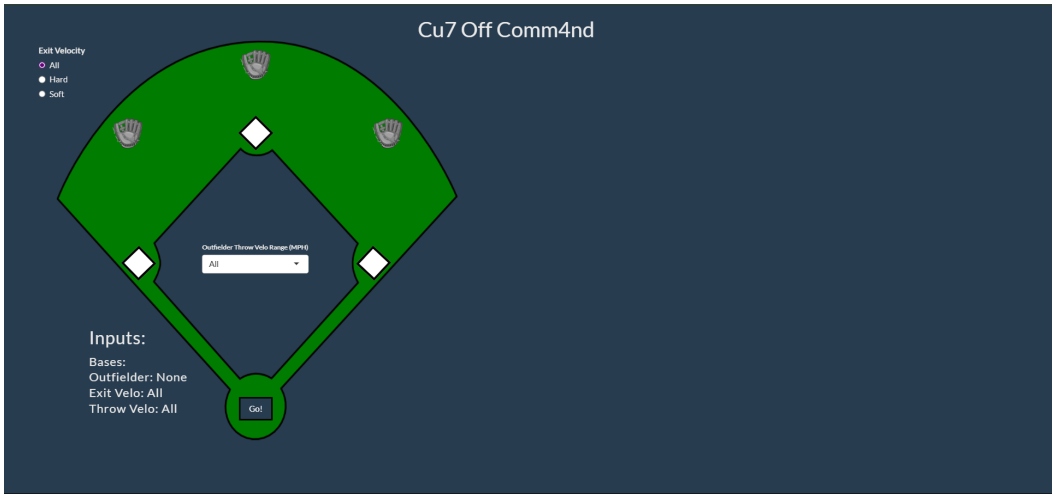

FIGURE 2: A look at the application when first opened

The user can quickly change inputs with just one click. Bases toggle on (yellow) and off (white) to indicate if a runner occupies it. One outfielder can be selected at a time to choose which position is making the throw. All inputs are updated live at the bottom left of the field, displaying to the user exactly what situation is currently selected. Once everything is chosen, the user can click the "Go!" button on home plate, and the corresponding heatmap is displayed to the right.

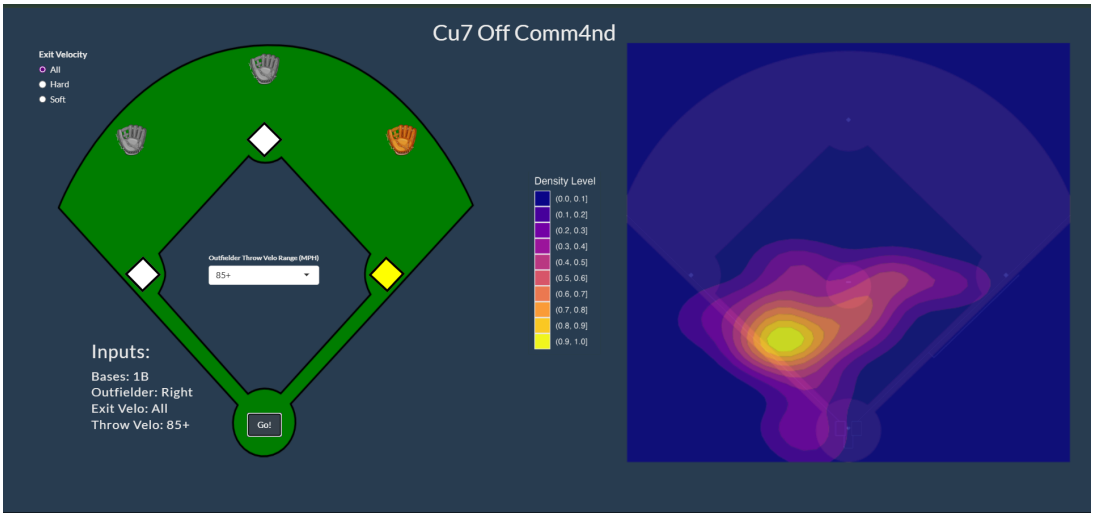

FIGURE 3: Heatmap being displayed for 1B runner, all exit velocities, 85+ mph throws, and RF fielding

Staying true to our vision of a quick and easy interface, we designed Cu7 Off Comm4nd to be fully optimized for touchscreens as well as computers. This makes it ideal for use on virtually any device — perfect for coaches or pitchers reviewing plays on an iPad in the dugout. Every function can be accessed with just a tap — no typing required.

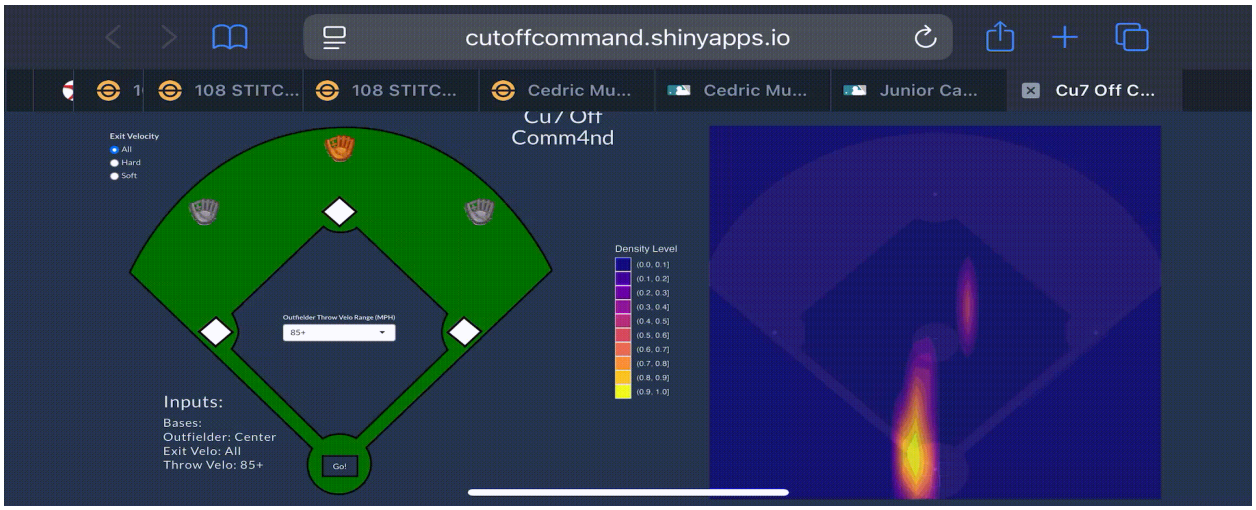

FIGURE 4 (GIF): Demonstration of Cu7 Off Comm4nd's quick and easy interface on iPhone

Discussion + Analysis

We begin our analysis with one of the most common pitcher backup scenarios: a runner starting on second base and a ball hit to the ield.

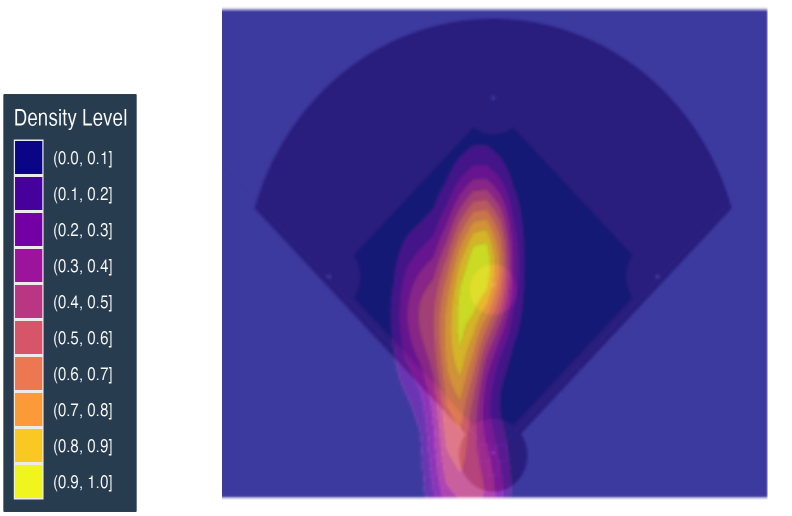

FIGURE 4: Pitcher positioning on outfield throws from right field with a runner on second and a throw velocity less than 70 mph.

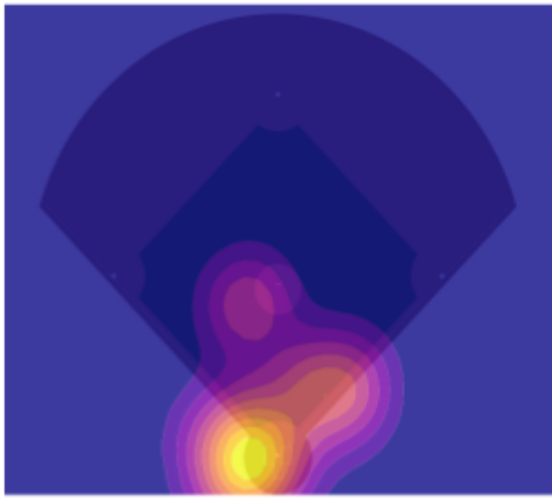

FIGURE 5: Pitcher positioning on outfield throws from right field with a runner on second and a throw velocity of 85 mph or above.

On plays where the right fielder throws below 70 miles per hour, we observe that pitchers remain near the mound (Figure 4). This makes sense — throws at that velocity are rarely made with intent, so we can assume they are not meant to challenge a runner at the plate. Instead, the outfielder is likely hitting a cut-off man or routinely tossing the ball back into the infield. Here, the pitcher acts as an observer, positioning themselves to respond to an overthrow or miscue but not anticipating a high-leverage defensive play. However, in the arm bucket of 85+ mph, the results are drastically different. Pitchers are now positioned densely behind and around home plate because this velocity bucket is likely to contain an outfielder's hardest throws — when they are trying to nail a runner at the plate. In these situations, pitchers are backing up home plate for any misplays or overthrows — like what happened in the Jansen example. Throw velocity significantly influences pitcher positioning on plays with a runner on second.

The next significant factor in pitcher positioning is base state.

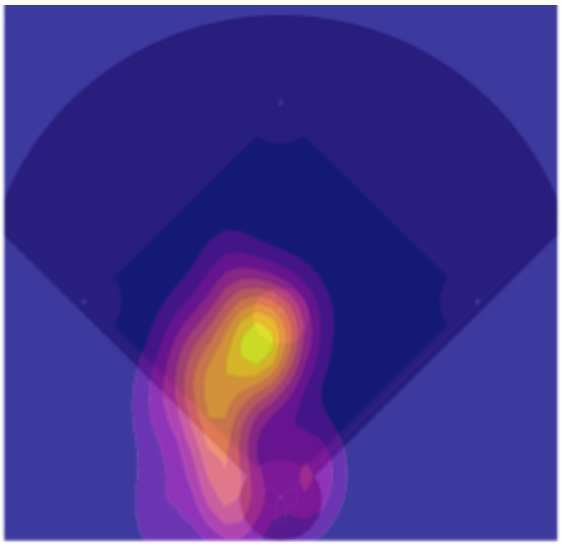

FIGURE 6: Pitcher positioning on outfield throws from center field with runners on first and third.

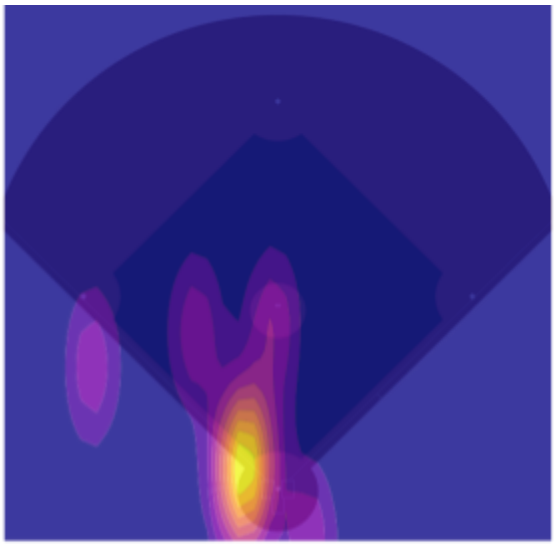

FIGURE 7: Pitcher positioning on outfield throws from center field with runners on first, second, and third.

For center field throws, pitchers have the greatest density near the mound for plays where there is a runner on first and third, while in bases-loaded situations, they move closer to home plate. In the first scenario, the defense most likely does not have a chance to throw out the lead runner on a hit and instead prioritizes halting the advancement of the runner on first and the batter. Just like in the below 70 mph arm bucket, the pitcher's job is to stay closer to the mound and back up any mishaps. However, with bases loaded, the pitcher's priority is to back up the plate because two runners are already in scoring position, significantly increasing the probability of a play at the plate. We also observe that the heatmap becomes denser farther up the third base line with the bases loaded, compared to when there's only a runner on second. This pattern likely reflects pitchers "hedging" their positioning — situating themselves between third base and home plate — when they're unsure where the play will occur. With multiple runners on, pitchers can anticipate a potential throw and position themselves accordingly so they can react quickly and

have a positive impact on the play. Base state drives pitcher positioning through pre-play recognition and probabilities of cutoffs versus plays at the plate.

A limiting factor in the dataset is that not every outfielder registered a throw with their top velocity. In 2025, the average throw from center field is 89.6 mph, and only one player has registered an average velocity below 70 mph. So instead of discussing how max velocity from outfielders affects positioning, we can quantify how pitcher positioning changes based on the intentions of the throw from the outfield. Our heatmaps display how positioning is much more of a fluid alignment, rather than a set designation.

Applying our analysis to Kenley Jansen in Game Four of the World Series, we can conclude that he *was not* in the standard defensive positioning. Although we do not have access to how fast Chris Taylor's throw to Max Muncy was, we can reasonably place it in the 85+ mph arm bucket with the understanding that there was urgency to beat a runner. Based on our heatmaps, Jansen should have crashed behind home plate to back up Muncy's errant throw, and maybe that postgame discussion would have been highlighted with the reminder that defense – or in this case, smart positioning – *can* win championships.

Takeaways

The main takeaway we hope to leave you with isn't tied to any single positioning scenario or even to how variables — like arm strength or base state — affect positioning (though these are important!). We're looking to highlight something broader.

When first discussing project ideas, this idea of focusing on pitcher positioning jumped out at us. It stemmed from a real gap in knowledge — something one of our group members experienced firsthand as a pitcher himself. So we ran with it, and what you see today is the result. But more than just the app itself, what we really want to showcase is this blueprint we've built. In the research world, what we've done is what is often called a "proof of concept."

While our work was shaped by realities of data availability, scope, and our still-budding coding knowledge, it provides a strong foundation for future development. Our app is a fully functional user interface that brings meaningful insights into pitcher positioning — an area often overlooked in defensive analysis. Still, many input combinations lack sufficient data to generate meaningful visualizations, and many potentially impactful variables remain unexplored. But what excites us so much about this tool is its potential. The same approach we took with pitchers can be expanded to any of the other eight positions on the field. Additionally, with more comprehensive data, possibilities arise to implement deeper customization, such as runner speed, real outfielder metrics, or more precise ball tracking to separate groundball singles from doubles in the gap, just to name a few.

Ultimately, we accomplished exactly what we set out to do: build a simple, intuitive, and effective tool that allows anyone to explore underappreciated pitcher fielding dynamics.

Acknowledgements

We would like to thank Dr. Meredith Wills and Billy Fryer for their coordination of the SMT Data Challenge and their tremendous help throughout the summer.

We would also like to thank the judges for their time and consideration in examining our submission.

Citations and References

Baseball Savant. “Arm Strength Leaderboard.” *MLB Advanced Media*, n.d., https://baseballsavant.mlb.com/leaderboard/arm-strength.

RStudio. “Shinythemes.” *RStudio*, n.d., https://rstudio.github.io/shinythemes/.

Posit. “Images in Shiny.” *Posit*, n.d., https://shiny.posit.co/r/articles/build/images/.

Wickham, Hadley. “Action Graphics.” *Mastering Shiny*, n.d., https://mastering-shiny.org/action-graphics.html.

RStudio. “Issue #555: Shiny Action Buttons.” *GitHub*, 2014, https://github.com/rstudio/shiny/issues/555.

“Shiny: How Not to Select Any Option in RadioButtons.” *Stack Overflow*, 2017, https://stackoverflow.com/questions/45300600/shiny-how-not-to-select-any-option-in-radiobuttons.

Attali, Dean. “Add Pretty Popups to Your Shiny Apps with Shinyalert.” *Dean Attali Blog*, 2018, https://deanattali.com/blog/shinyalert-package/.

Attali, Dean. “Shinyalert Demo.” *Dean Attali*, 2018, https://daattali.com/shiny/shinyalert-demo/.

"RenderImage Issue on Shiny R." *Stack Overflow*, 2020, https://stackoverflow.com/questions/62939450/renderimage-issue-on-shiny-r.

Clker. "Baseball Field Clipart." *Clker.com*, n.d., https://www.clker.com/clipart-baseball-field-4.html.

"Baseball Glove Illustrations and Clipart." *Dreamstime*, n.d., https://www.dreamstime.com/illustration/baseball-glove.html.

Wickham, Hadley, et al. *ggplot2: Elegant Graphics for Data Analysis*. R package version 3.4.0, 2023, https://ggplot2.tidyverse.org/.

Wickham, Hadley, et al. *dplyr: A Grammar of Data Manipulation*. R package version 1.1.2, 2023, https://dplyr.tidyverse.org/.

Lane, Bill, et al. *sportyR: Visualize Play-by-Play Data for Sports*. R package version 2.0.2, 2022, https://sportyr.rinterface.com/.

Appendix

Data Manipulation:
We used the tidyverse package as the foundation of our data analysis, assisting with data manipulation, cleaner code and iteration, and the creation of CSV files.

The **process_game** function was developed to extract the pitcher positioning for each relevant play and associate it with the contextual variables of base state, exit velocity, and arm strength. The base states were created through the "game_info" dataset and a conditional statement that applied a "1" to any occupied base, and a "0" to an empty base. These values were combined

into a three-digit string representing the base paths (ex, “101”). Exit velocity and arm strength were calculated the same way — distance divided by time. Distance was solved using the Pythagorean theorem, where the squared changes in the horizontal and vertical ball position were summed and square-rooted. Tracking data was recorded at 50-millisecond intervals, allowing time to be calculated from the frame differences. The resulting speed was converted from meters per second to miles per hour by dividing by 1.467 and then multiplying by 1,000. In total, the function assigned context for 2,825 data points across all three outfield positions.

For the creation of our heatmaps, we utilized ggplot2 for plotting and sportyR to render baseball field visualizations. The **plot_density** function processes every datapoint from a dataframe and maps pitcher positioning onto a heatmap. The **generate_plots_for_field** function iterates through all unique combinations of outfielder position, exit velocity classification, base state, and arm strength bucket to produce context-specific heatmaps. Figure 8 illustrates a grid of center field heatmap outputs, classified by base state, arm strength, and exit velocity categories generated by our **plot_density** function. Scenarios with fewer than five data points were excluded to ensure meaningful visualization, resulting in some scenarios not passing the threshold.

Across all situations, our function generated 204 graphs out of the possible 288.

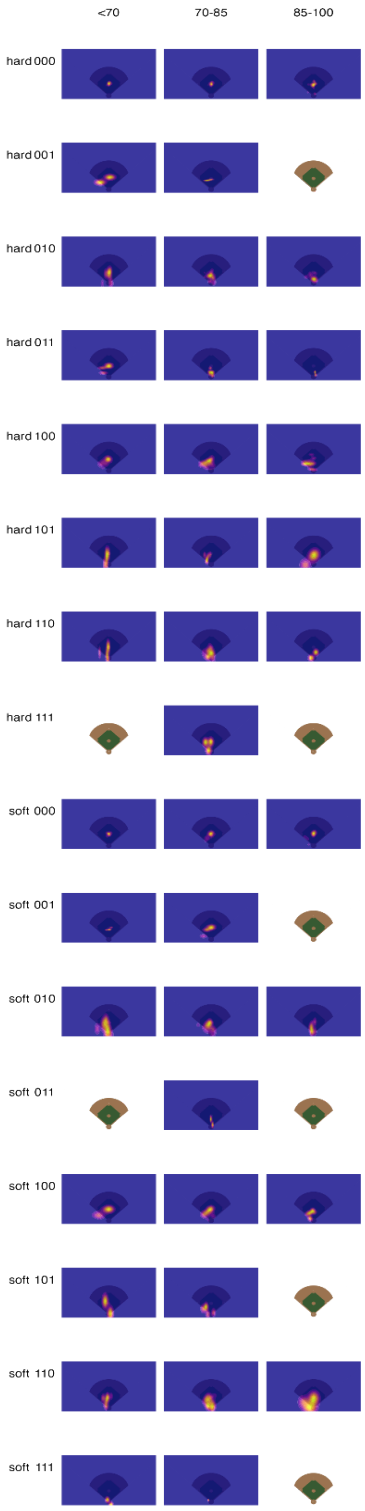

FIGURE 8: Heatmaps for center field - separated by exit velocity, base state, and arm strength

Cu7 Off Comm4nd User Guide:

The website opens up to an infield with a handful of different icons, buttons, and text.

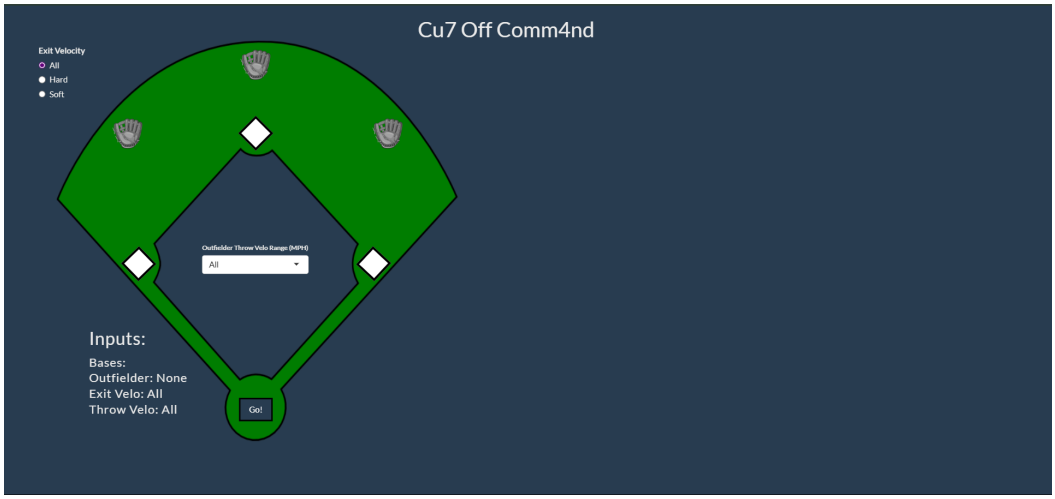

FIGURE 9

The **"Inputs"** section at the bottom left of your screen displays the status of each of the four inputs based on your selections. It starts with default values (bases empty, no fielder selected, all exit velocities, and all throw velocities).

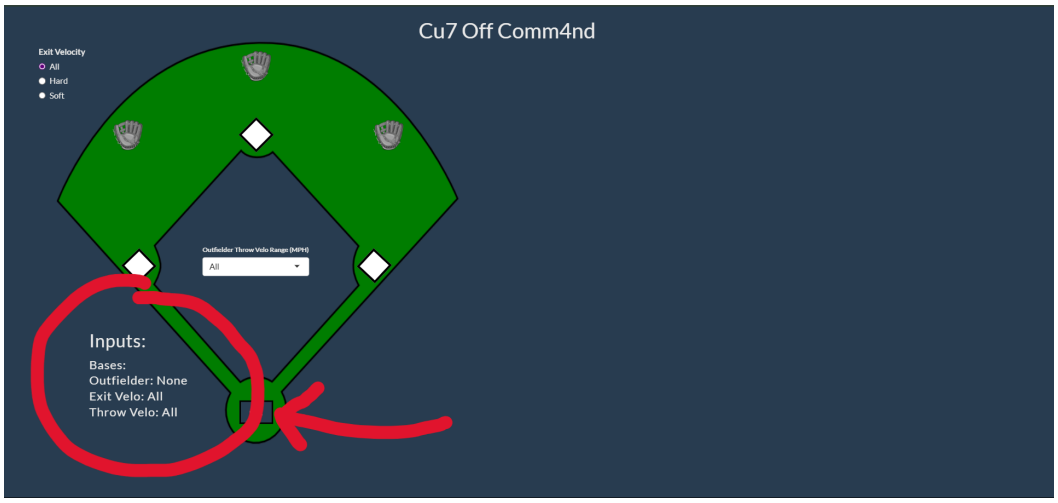

FIGURE 10

Your first input option is the base state. You can alter this by toggling on and off any of the three bases. Bases appear yellow if they are toggled on (representing a runner occupying that base) and

white if they are toggled off (representing an empty base). Notice how the “Inputs” section changes as you toggle different bases.

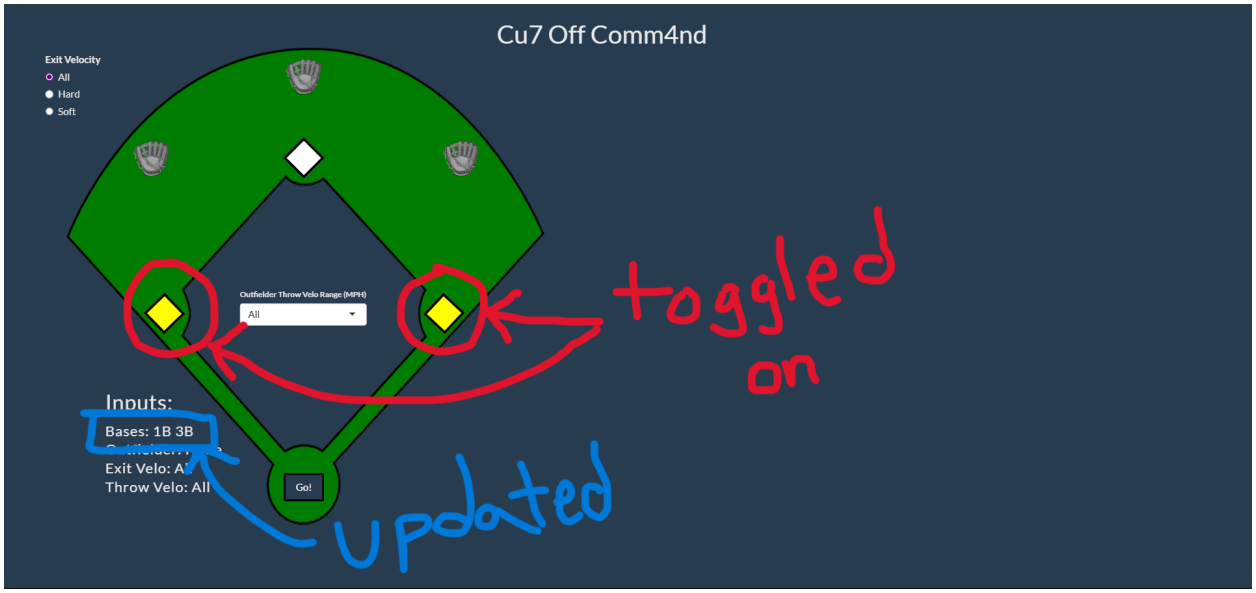

FIGURE 11

The next input is the fielder. This refers to whichever outfielder fielded the ball and is throwing it on all of the plays being used for the current heatmap. No outfielders are selected at first, but one must be selected to see a heatmap. If you press “Go!” and no outfielders are selected, the following error message pops up:

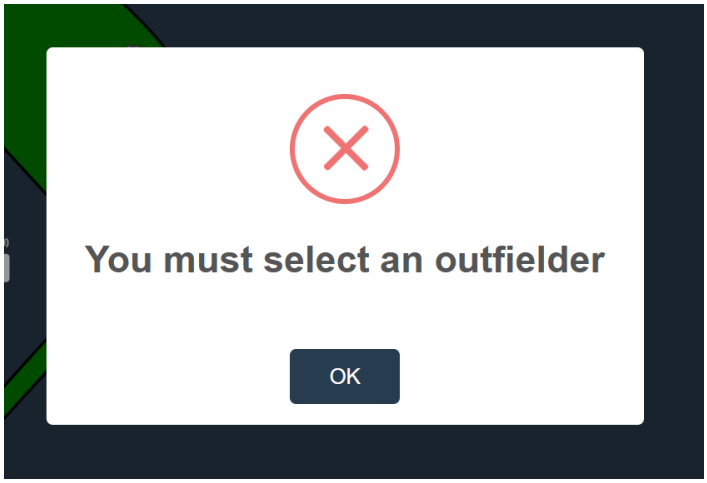

FIGURE 12

To exit that alert, you can click “OK,” press the ESC key, or click outside of the box.

To select an outfielder, press on the corresponding glove icon at the edge of the field graphic. Once selected, the glove icon will become colored in. Only one outfielder may be selected at a time. If you select a new outfielder, the previous outfielder will automatically deselect itself, turning grey again. These changes will once again be reflected in the “Inputs” section at the bottom.

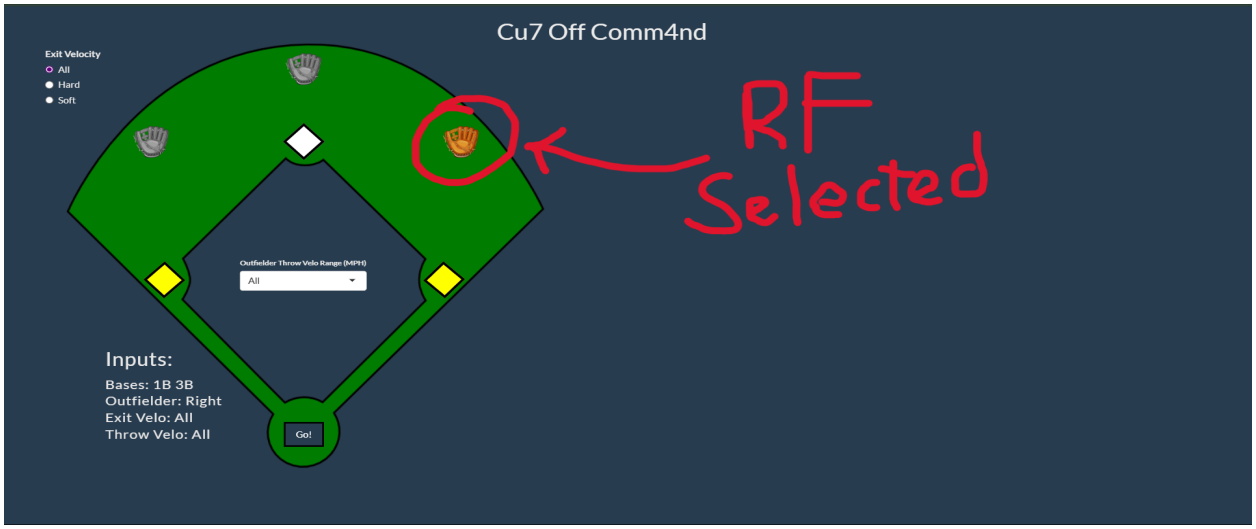

FIGURE 13

The third customizable input is the **“Exit Velocity”** input. You are given three options: “All,” “Hard,” and “Soft.” Selecting “All” will give you any exit velocity, while “Hard” will give you plays on balls hit at an exit velocity above the 86-mile-per-hour threshold, and “Soft” will give you plays on balls hit below that 86-mph mark. You can select one of these options by clicking on the word or the circle next to it. Your selected choice will appear with a purple circle next to it, indicating that it is selected while the other two options remain white.

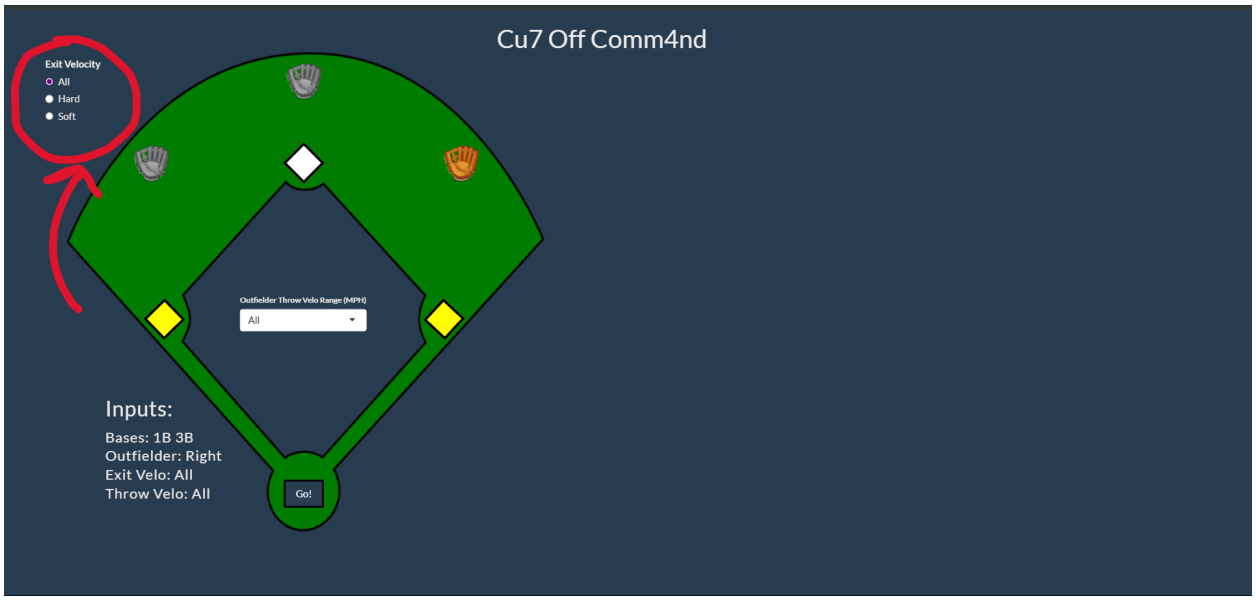

FIGURE 14

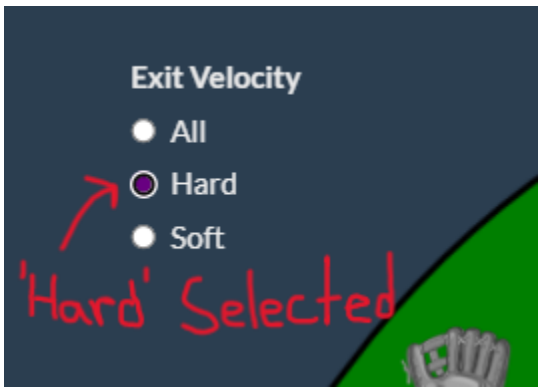

FIGURE 15

FIGURE 16

Your final input is the velocity of the throw from the outfielder. This can be altered with the dropdown menu in the middle of the field, where the pitcher's mound usually is. It is automatically set to "All" but can be changed to three other buckets: "<70" (throws less than 70 mph), "70-85" (throws between 70 and 85 mph), and "85+" (throws above 85 mph). To select a new velocity bucket, click on the white bar and select from the options given in the dropdown.

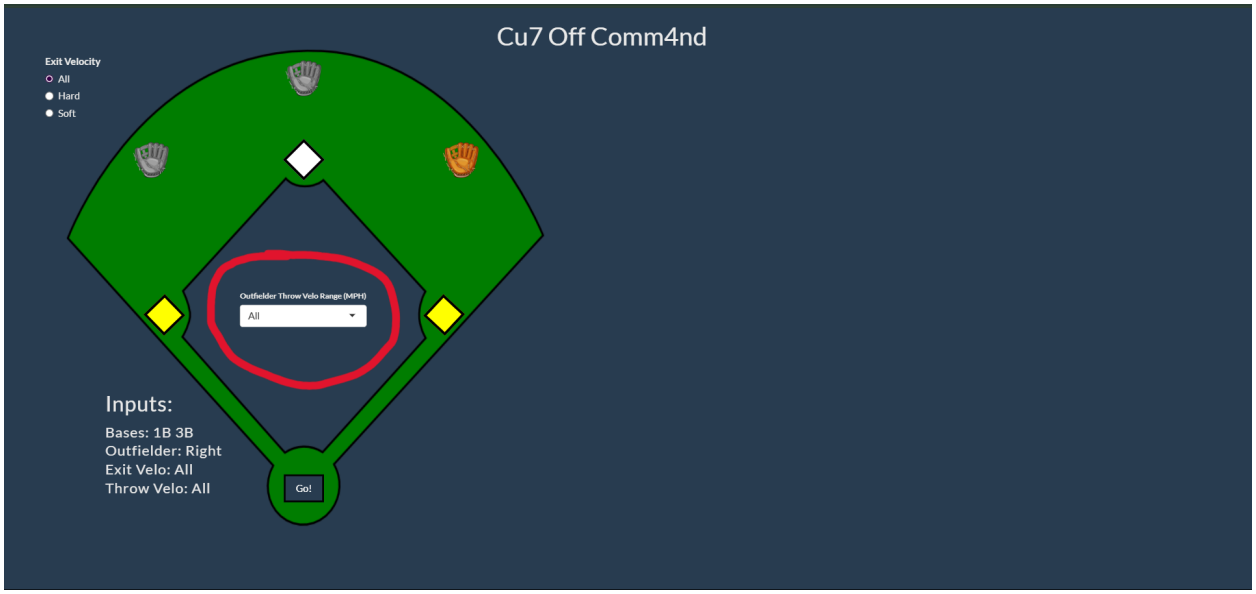

FIGURE 17

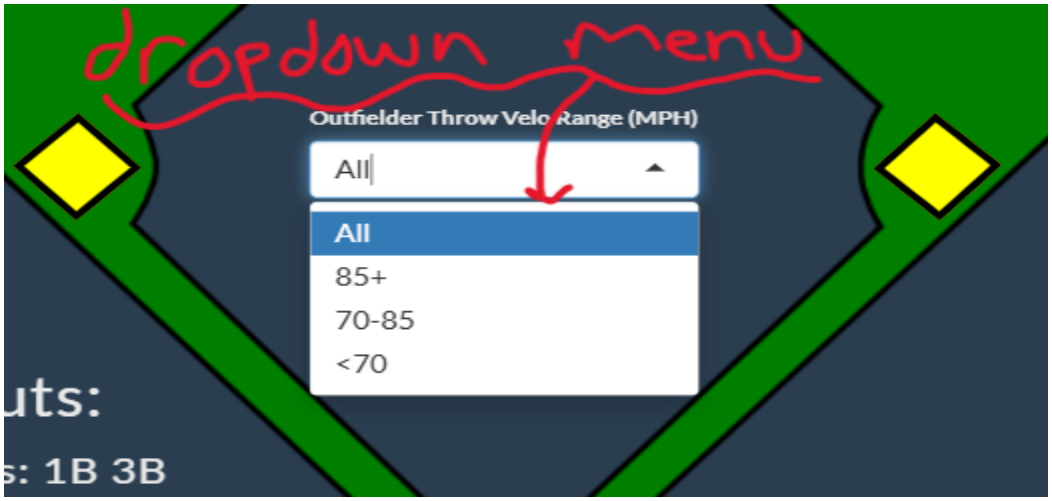

FIGURE 18

Once you have chosen all your inputs, double-check the **“Inputs”** section one more time, and hit the grey **“Go!”** button at the bottom of the field where home plate is normally located.

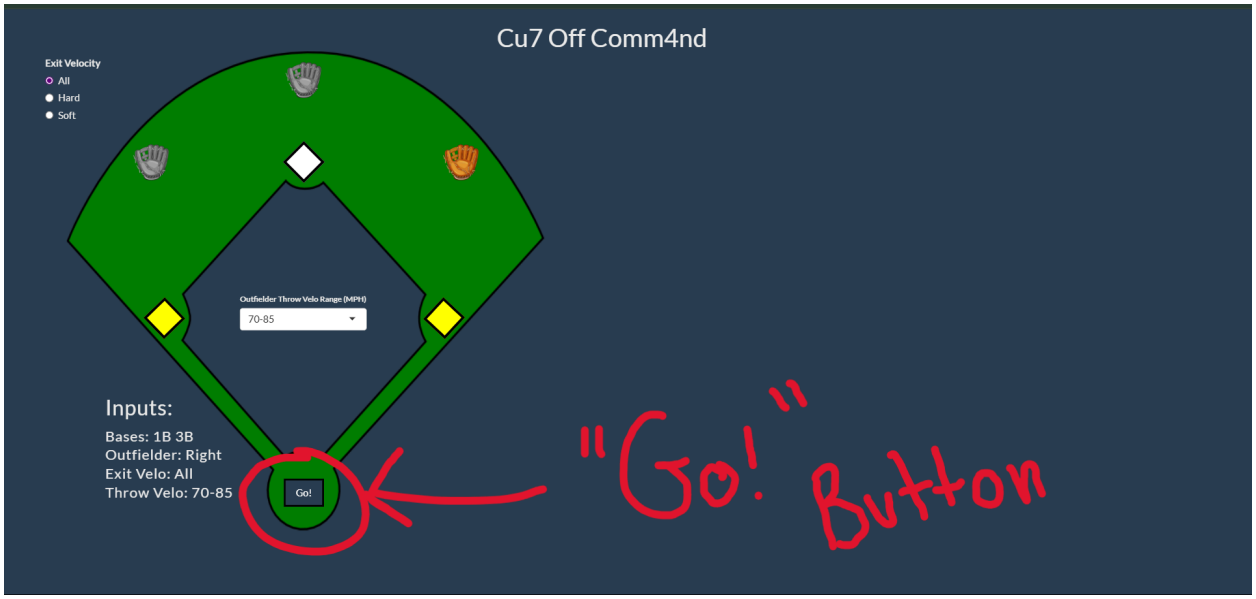

FIGURE 19

Voilà! Your heatmap should’ve appeared on the right side of your screen. If not, you’re probably seeing the following message:

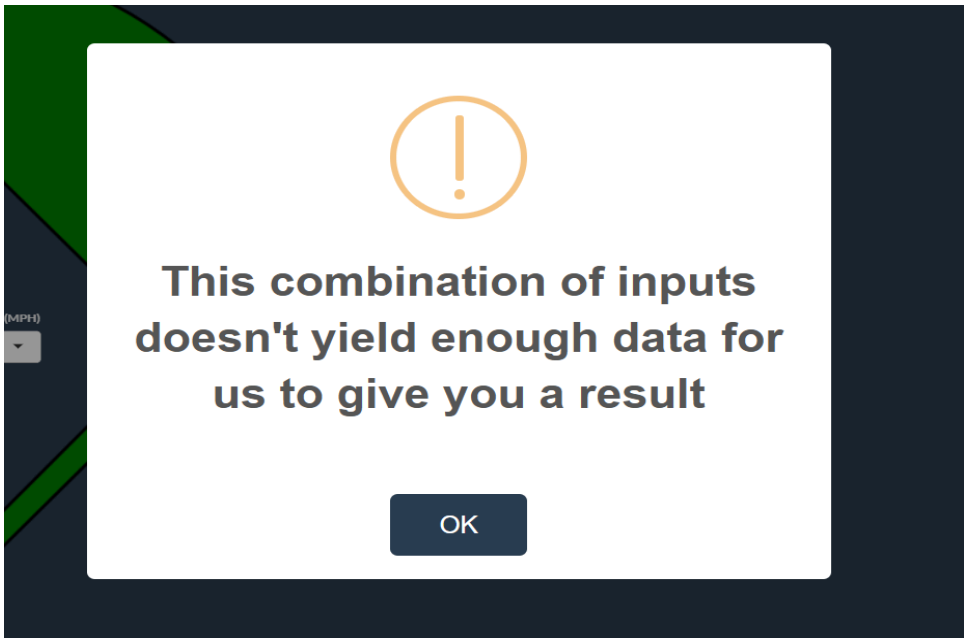

FIGURE 20

Essentially, the combination of inputs you selected didn't return enough data to display a heatmap. Unfortunately, just under 30% of the possible combinations don't have enough data in the set provided to us to create a heatmap. In this case, again, either click "OK," press ESC, or click out of the popup to hide the message. You'll now want to try a different combination of inputs. Once you press "Go!" one more time, hopefully, your screen will look more like this:

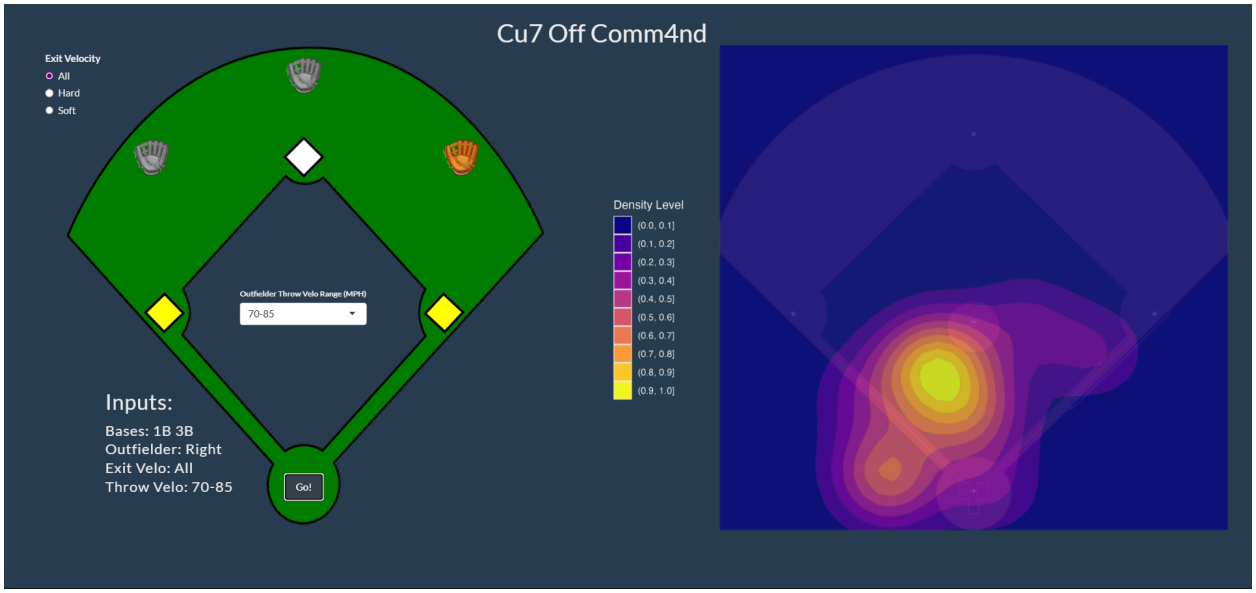

FIGURE 21

Here you see the full heatmap for the exact scenario that you selected. The colors of the key to the left of the image indicate what proportion of pitchers stand in each area.

To look at a different situation's heatmap, just change the inputs with the buttons on the left and press the grey "Go!" button again to generate an updated map. Enjoy!