SMT Data Challenge 2023

https://github.com/ConorD28/SMT-Data-Challenge

Abstract

Within the game of baseball, defensive analytics have always differed from hitting and pitching analytics in that quantifying such information isn't so straightforward. There are a multitude of factors influencing defensive analytics that aren't always present when evaluating hitting and pitching; these factors include movement from one location to another, as well as defensive decision-making with regard to where to go with the baseball. Our work specifically focused on these types of variables, as we compared the average time it takes to get someone out from different locations in the outfield to different locations on the basepaths while examining throws that were and weren't cut off from the outfield. In addition, we examined the accuracy and success rates of such throws. Specifically, we looked at five regions in the outfield (left field, left-center field, center field, right-center field, and right field) while evaluating three regions to create an out on the basepaths (second base, third base, and home plate).

We began by forming an understanding of the data and learning what each specific variable meant (for example, 2 = catching the ball, while 3 = throwing the ball), before deciding how to proceed with our project. Ultimately, we selected an idea proposed by SMT to compare cutoff throws to non-cutoff throws. We began by evaluating non-cutoff throws, and once we finished gauging that category, we continued by evaluating cutoff throws. We evolved our work by adding variables such as distance, time, accuracy, success rate, and the number of throws to a specific location within a distance interval.

Introduction

Various attempts over the years to quantify and refine defensive performance in baseball have led to a rich history of statistical innovations used to evaluate each player's defensive performance on the whole. Bill James, a pioneering baseball statistician responsible for creating sabermetrics, introduced a number of catch-all defensive statistics such as Range Factor and Defensive Efficiency Rating. While these metrics lacked the depth and complexity of newer metrics, they ultimately set in motion the mission of quantifying defense, long considered the least measurable aspect of the game. Since the early days of sabermetrics, the statistics we rely upon to gauge defensive performance have improved, with Defensive Runs Saved (DRS) and Outs Above Average (OAA) most notably having filtered their way into the mainstream.

While these catch-all statistics are useful in quickly summarizing a player's defensive capabilities, they often do not tell the full story of the specifics behind a player's defensive performance. Perhaps nowhere on the diamond is this more apparent than in the outfield, as there are many different components to outfield defense that might not be immediately apparent when looking at a catch-all statistic. Some examples of the various components of outfield defense include outfielder jump, sprint speed, arm strength, and arm accuracy, along with the efficiency of the routes outfielders take toward the baseball. With various elements contributing to defensive performance in the outfield, it can be difficult to properly assess outfield defense, as a player might excel in one component of outfield defense while struggling with another.

One notable example of this can be found when evaluating outfielder Hunter Renfroe's defensive performance during the 2021 season. It is clear that Renfroe excelled in the areas of arm strength and arm accuracy while playing right field for the Red Sox, as he ranked in the 100th percentile for arm strength in 2021 (Baseball Savant) while also tying Adolis Garcia for the league lead with 16 outfield assists (Sports Reference). Renfroe was also middle-of-the-pack when it came to his outfielder jump metrics, which were in the 53rd percentile during 2021, and his sprint speed, which was in the 57th percentile that year (Baseball Savant). One might surmise that Renfroe's catch-all defensive statistics would peg him as an above-average defender for 2021 due to his prowess throwing from the outfield and his mobility in the outfield, which was at the very least considered to be mediocre by the metrics. This, however, would be a false assumption, as Renfroe only ranked in the 18th percentile for Outs Above Average (OAA) with -3 on the year (Baseball Savant). This largely could be pinned on Renfroe's inefficiency in the routes he took to track down balls in play, which ultimately undermined his adequate mobility and exceptional arm. Through this, we see that it is imperative to evaluate each individual component of outfield defense separately so that we can properly encapsulate the defensive strengths and weaknesses of each outfielder.

With this in mind, our focus was on one particular area of outfield defense: the average time it takes to get someone out from different locations in the outfield to different locations on the basepaths. To do this, we evaluated throws that were and weren't cut off from the outfield to determine their accuracy and success rates. As previously mentioned, we reviewed five regions in the outfield (left field, left-center field, center field, right-center field, and right field) while evaluating three regions to create an out on the basepaths (second base, third base, and home plate). The goal in performing this evaluation was to answer our problem question: Should the outfielder use the cutoff man or instead skip the cutoff man when attempting to throw a runner out at different locations?

Coming out of this evaluation, we anticipated seeing a number of outcomes and surmised that the average time to get someone out on the basepaths would likely be shorter without using the cutoff man in certain situations. Overall, we believe that such an examination allows for practical evaluation to determine whether or not to use a cutoff man in different scenarios. This could present itself in numerous ways; for example, a player could test if they possess the ability to throw a baseball over a specific distance in the average time or less to determine if they should skip the cutoff man based on what the data says, while this information could also be useful for team drills to improve synergy on plays with and without the cutoff man.

In the next section, we will review the data used, what the different tables and variables mean, and why they are important. After that, we will delve into the methodology behind our work while also touching upon assumptions that were made and limitations we encountered. Once these have been addressed, we will list the findings, discuss the significance of the results, and touch upon future forms of exploration that may derive from these findings.

Data and Methodology

When looking at our work, the tables we used were called “game events,” “game info,” and “player position.” For the “game events” table, four variables were used: “play_id,” “timestamp,” “player_position,” and “event_code.” The variable “play_id” is an ID starting at 1 and increasing by 1 for each play until the game is over. The variable “timestamp” tracks the time in milliseconds from the start of the game. The variable “player_position” uses numbers to represent what position the player is playing, ranging from 1-9. The numbers for each player's position are the same baseball position numbers used for officially scoring a baseball game. The variable “event_code” uses numbers to represent an event; an event refers to something that happened during the play such as a hit ball, a caught ball, a ball bouncing, etc. For example, - 2 represents catching the ball, 3 is throwing the ball, etc. The variable “play_id” was used to check if we were still on the current play during a loop and it was used to connect to the other tables to obtain information about the specific play from those tables. The variable “timestamp” was used to capture the time when the ball was caught by the outfielder, the cutoff man, and the fielder covering the base or plate. These times were then used for calculating average times. It was also used to obtain the outfielder’s position when he caught the ball. The variable “player_position” was used to check the position of the player who caught the ball, the position of the player who threw the ball, and to make sure the cutoff man threw it to the intended target to create an out; this was done because we only sought to compare direct throws from the outfield to the target area with throws that followed an identical path involving a cutoff man. The variable “event_code” was used to see what occurred during the play, as we checked to see if the ball was caught and if the ball was thrown. This variable, used with “player_position,” allowed for the selection of desired plays, which in this case meant plays involving catches from the outfielder that were then thrown to a base or to a player who may cut the ball off.

For the “game info” table, the variables used were “play_per_game,” “top_bottom_inning,” “second_baserunner,” and “third_baserunner.” The variable “play_per_game” is an ID starting at 1 and increasing by 1 for each play until the game is over, while the variable “top_bottom_inning” represents if it is the top or bottom of the inning. The variable “second_baserunner” is the ID of the player at second base; if there is no player at second base, then the value is 0. Similarly, the variable “third_baserunner” is the ID of the player at third base; if there is no player at third base, then the value is 0. The variable “play_per_game” was used to obtain the desired specific play from the “game info” table, allowing for the examination of important statistics from that table. The variable “top_bottom_inning” was used to see if the runner was out at home plate. If the half of the inning changed after a play at the plate, we were confident the runner was out at home. The baserunner variables were used to see if the runner was out or safe at the bases, which then allowed for the calculation of statistics regarding the success rates on throws to those bases.

For the "player position" table, the variables used were "player_position," "timestamp," "field_x," and "field_y." The variable "player_position" is the same as the "player_positon" variable from the game events table, while the variable" timestamp" represents the time in milliseconds of when the event happened. The variables "field_x" and "field_y" represent the x and y positions of the fielder. The variable "player_position" was used to check which fielder caught the ball, which allowed us to make sure the initial catch was made by an outfielder and that the cutoff man's position was either 1B, 2B, SS, or 3B. The variable "timestamp" was used to obtain the player's x and y positions on the field when the event happened. The variables "field_x" and "field_y" were used to procure the players' x and y positions on the field, which allowed for the calculation of throw distance and throw accuracy.

Utilizing the aforementioned tables and variables, we compared cutoff throws from the outfield to non-cutoff throws from the outfield. We looked at throws to different bases and from different 50-foot interval distances away from those bases, successfully obtaining the number of throws, throw distances, average throw times, average throw accuracy, average cutoff time (for cutoff throws), number of outs, number of times safe, and success rates for the various types of throws; this allowed us to draw conclusions from the results. We ensured that the cutoff man was within a 30-foot interval on cutoff throws to help guarantee he was cutting the ball off and not doing something else; this relied upon an assumption that the cutoff man caught the ball within a 30-foot distance interval. Throughout most of the code, information was printed out on the individual throws, while the overall information on the various groupings of throws was printed out at the end.

As for equations utilized within the code, they were used to sum up the number of throws, the number of times safe, total times, total accuracies, and the number of outs. The distance formula was to compute the throw distances given two pairs of xy coordinates. To obtain the times, we used simple subtraction followed by division by 1000 and converted them to seconds. While this process was effective, our code did contain some limitations. Due to the nature of the data, we could only tell if the runner was out at home plate if there were two outs. To do so, we had to check to see if the half of the inning had changed to see if the runner was out with 2 outs; with less than 2 outs, there was no out/safe variable to see if he was out. We also lacked the time to create additional graphs and we did not examine how specific players performed on throws, which would have elevated our findings.

Main Findings

For home plate non-cutoff throws from 250-300 feet, there were 10 throws where it was confirmed if the runner was out or safe. On throws where the runner was out, 3.7 seconds was the average time to throw the ball to the base, and 4.32 feet off the bag was the average accuracy. For second base non-cutoff throws from 150-199 feet, there were 26 throws where it was confirmed if the runner was out or safe. On throws where the runner was out, there was a 66.67% success rate. 3.35 seconds was the average time to throw to the base and 7.96 feet was the average accuracy.

```
 Average_Time_Second_Outs
1                      3.22
2                      3.66
3                      3.35
4                      3.33
```

Discussion

It seemed that there was not enough data for cutoff throws and/or the bounds for the cutoff man should have been extended. Looking at the average throw time to second, it seems an outfielder needs to be able to throw the ball to second in 3.35 seconds or less to get the runner out. Otherwise, he should use a cutoff man. Drills can be run before the game to see if the outfielder can get the ball to second in time, and if not, he can try to increase his arm strength or perhaps change something mechanically else to get the ball there quicker. If he still is unable to get the throw there in 3.35 seconds or less, then the cutoff man could be a better option. Now, the outfielder can make a decision based on data of whether to use the cutoff man, which could be a better-informed decision than going off his gut-feeling. The table that contains 62 non cutoff throws to second it reaffirms that a player needs to throw the ball to second in 3.2-3.7 seconds. In the future, we can analyze specific players' throws and what would be best suited for them, we could make graphs, and we could use more data.

Acknowledgments

We would like to acknowledge Dr. Meredith Wills of SportsMEDIA Technology Corp (SMT) for providing us with guidance throughout this process. While consulting with her, she offered actionable advice and made useful recommendations. Her advice on how to structure a paper played an integral role in the layout of these findings, and we are thankful for her guidance. We would also like to acknowledge SMT as a whole for facilitating this data challenge and for providing us with the data used to carry out our project.

Citations

Baseball Savant. (n.d.-a). Hunter Renfroe Statcast, Visuals & Advanced Metrics: MLB.com. baseballsavant.com.

https://baseballsavant.mlb.com/savant-player/hunter-renfroe-592669?stats=statcast-r-fielding-mlb

Baseball Savant. (n.d.-b). Statcast Outs Above Average Leaderboard. baseballsavant.com. https://baseballsavant.mlb.com/leaderboard/outs_above_average?type=Fielder&startYear=2021&endYear=2021&split=yes&team=BOS&range=year&min=q&pos=&roles=&viz=hide

Sports Reference. (n.d.). Year-by-Year Top-Tens Leaders & Records for Assists as OF. Baseball Reference. https://www.baseball-reference.com/leaders/A_of_top_ten.shtml

Appendix

###Non_Cutoff Throws Variables:

out_catcher_NC = 0

out_catcher_NC_150 = 0

out_catcher_NC_200 = 0

out_catcher_NC_250 = 0

out_catcher_NC_300 = 0

out_catcher_NC_350 = 0

out_catcher_NC_400 = 0

out_second_NC = 0

out_second_NC_150 = 0

out_second_NC_200 = 0

out_second_NC_250 = 0

out_second_NC_300 = 0

out_second_NC_350 = 0

out_second_NC_400 = 0

out_third_NC = 0

out_third_NC_150 = 0

out_third_NC_200 = 0

out_third_NC_250 = 0

out_third_NC_300 = 0

out_third_NC_350 = 0

out_third_NC_400 = 0

safe_second_NC = 0

safe_second_NC_150 = 0

safe_second_NC_200 = 0

safe_second_NC_250 = 0

safe_second_NC_300 = 0

safe_second_NC_350 = 0

safe_second_NC_400 = 0

safe_third_NC = 0

safe_third_NC_150 = 0

safe_third_NC_200 = 0

safe_third_NC_250 = 0

safe_third_NC_300 = 0

safe_third_NC_350 = 0

safe_third_NC_400 = 0

success_rate_2B_NC = 0

```
success_rate_3B_NC = 0

throws_home_NC = 0

throws_home_NC_150 = 0

throws_home_NC_200 = 0

throws_home_NC_250 = 0

throws_home_NC_300 = 0

throws_home_NC_350 = 0

throws_home_NC_400 = 0

throws_third_NC = 0

throws_third_NC_150 = 0

throws_third_NC_200 = 0

throws_third_NC_250 = 0

throws_third_NC_300 = 0

throws_third_NC_350 = 0

throws_third_NC_400 = 0

throws_second_NC = 0

throws_second_NC_150 = 0

throws_second_NC_200 = 0

throws_second_NC_250 = 0

throws_second_NC_300 = 0

throws_second_NC_350 = 0

throws_second_NC_400 = 0

total_time_home_NC = 0
```

```
total_time_home_NC_150 = 0

total_time_home_NC_200 = 0

total_time_home_NC_250 = 0

total_time_home_NC_300 = 0

total_time_home_NC_350 = 0

total_time_home_NC_400 = 0

total_time_second_NC = 0

total_time_second_NC_150 = 0

total_time_second_NC_200 = 0

total_time_second_NC_250 = 0

total_time_second_NC_300 = 0

total_time_second_NC_350 = 0

total_time_second_NC_400 = 0

total_time_third_NC = 0

total_time_third_NC_150 = 0

total_time_third_NC_200 = 0

total_time_third_NC_250 = 0

total_time_third_NC_300 = 0

total_time_third_NC_350 = 0

total_time_third_NC_400 = 0

total_accuracyC_NC = 0

total_accuracyC_NC_150 = 0

total_accuracyC_NC_200 = 0
```

```
total_accuracyC_NC_250 = 0
total_accuracyC_NC_300 = 0
total_accuracyC_NC_350 = 0
total_accuracyC_NC_400 = 0
total_accuracy3B_NC = 0
total_accuracy3B_NC_150 = 0
total_accuracy3B_NC_200 = 0
total_accuracy3B_NC_250 = 0
total_accuracy3B_NC_300 = 0
total_accuracy3B_NC_350 = 0
total_accuracy3B_NC_400 = 0
total_accuracy2B_NC = 0
total_accuracy2B_NC_150 = 0
total_accuracy2B_NC_200 = 0
total_accuracy2B_NC_250 = 0
total_accuracy2B_NC_300 = 0
total_accuracy2B_NC_350 = 0
total_accuracy2B_NC_400 = 0
average_timeC_NC = 0
average_timeC_NC_150 = 0
average_timeC_NC_200 = 0
average_timeC_NC_250 = 0
average_timeC_NC_300 = 0
```

```
average_timeC_NC_350 = 0
average_timeC_NC_400 = 0
average_time2B_NC = 0
average_time2B_NC_150 = 0
average_time2B_NC_200 = 0
average_time2B_NC_250 = 0
average_time2B_NC_300 = 0
average_time2B_NC_350 = 0
average_time2B_NC_400 = 0
average_time3B_NC = 0
average_time3B_NC_150 = 0
average_time3B_NC_200 = 0
average_time3B_NC_250 = 0
average_time3B_NC_300 = 0
average_time3B_NC_350 = 0
average_time3B_NC_400 = 0
Average_Time_NC = 0
Average_Time_NC_150 = 0
Average_Time_NC_200 = 0
Average_Time_NC_250 = 0
Average_Time_NC_300 = 0
Average_Time_NC_350 = 0
Average_Time_NC_400 = 0
```

```
average_accuracyC_NC = 0

average_accuracyC_NC_150 = 0

average_accuracyC_NC_200 = 0

average_accuracyC_NC_250 = 0

average_accuracyC_NC_300 = 0

average_accuracyC_NC_350 = 0

average_accuracyC_NC_400 = 0

average_accuracy2B_NC = 0

average_accuracy2B_NC_150 = 0

average_accuracy2B_NC_200 = 0

average_accuracy2B_NC_250 = 0

average_accuracy2B_NC_300 = 0

average_accuracy2B_NC_350 = 0

average_accuracy2B_NC_400 = 0

average_accuracy3B_NC = 0

average_accuracy3B_NC_150 = 0

average_accuracy3B_NC_200 = 0

average_accuracy3B_NC_250 = 0

average_accuracy3B_NC_300 = 0

average_accuracy3B_NC_350 = 0

average_accuracy3B_NC_400 = 0

diff_catcher = 0

diff_second = 0
```

```
diff_third = 0
checkOF_NC = 0
check_home_NC = 0
check_third_NC = 0
check_second_NC = 0
throw_distance_C_NC = 0
throw_distance_2B_NC = 0
throw_distance_3B_NC = 0
success_rate_2B_NC = 0
success_rate_2B_NC_150 = 0
success_rate_2B_NC_200 = 0
success_rate_2B_NC_250 = 0
success_rate_2B_NC_300 = 0
success_rate_2B_NC_350 = 0
success_rate_2B_NC_400 = 0
success_rate_3B_NC = 0
success_rate_3B_NC_150 = 0
success_rate_3B_NC_200 = 0
success_rate_3B_NC_250 = 0
success_rate_3B_NC_300 = 0
success_rate_3B_NC_350 = 0
success_rate_3B_NC_400 = 0
```

```
#Cutoff Throws Variables:

out_home_cu = 0

out_home_cu_150 = 0

out_home_cu_200 = 0

out_home_cu_250 = 0

out_home_cu_300 = 0

out_home_cu_350 = 0

out_home_cu_400 = 0

out_second_cu = 0

out_second_cu_150 = 0

out_second_cu_200 = 0

out_second_cu_250 = 0

out_second_cu_300 = 0

out_second_cu_350 = 0

out_second_cu_400 = 0

out_third_cu = 0

out_third_cu_150 = 0

out_third_cu_200 = 0

out_third_cu_250 = 0

out_third_cu_300 = 0

out_third_cu_350 = 0

out_third_cu_400 = 0

safe_home_cu = 0
```

```
safe_home_cu_150 = 0
safe_home_cu_200 = 0
safe_home_cu_250 = 0
safe_home_cu_300 = 0
safe_home_cu_350 = 0
safe_home_cu_400 = 0
safe_second_cu = 0
safe_second_cu_150 = 0
safe_second_cu_200 = 0
safe_second_cu_250 = 0
safe_second_cu_300 = 0
safe_second_cu_350 = 0
safe_second_cu_400 = 0
safe_third_cu = 0
safe_third_cu_150 = 0
safe_third_cu_200 = 0
safe_third_cu_250 = 0
safe_third_cu_300 = 0
safe_third_cu_350 = 0
safe_third_cu_400 = 0
throws_home_cu = 0
throws_home_cu_150 = 0
throws_home_cu_200 = 0
```

```
throws_home_cu_250 = 0

throws_home_cu_300 = 0

throws_home_cu_350 = 0

throws_home_cu_400 = 0

throws_third_cu = 0

throws_third_cu_150 = 0

throws_third_cu_200 = 0

throws_third_cu_250 = 0

throws_third_cu_300 = 0

throws_third_cu_350 = 0

throws_third_cu_400 = 0

throws_second_cu = 0

throws_second_cu_150 = 0

throws_second_cu_200 = 0

throws_second_cu_250 = 0

throws_second_cu_300 = 0

throws_second_cu_350 = 0

throws_second_cu_400 = 0

total_time_home_cu = 0

total_time_home_cu_150 = 0

total_time_home_cu_200 = 0

total_time_home_cu_250 = 0

total_time_home_cu_300 = 0
```

```
total_time_home_cu_350 = 0

total_time_home_cu_400 = 0

total_time_second_cu = 0

total_time_second_cu_150 = 0

total_time_second_cu_200 = 0

total_time_second_cu_250 = 0

total_time_second_cu_300 = 0

total_time_second_cu_350 = 0

total_time_second_cu_400 = 0

total_time_third_cu = 0

total_time_third_cu_150 = 0

total_time_third_cu_200 = 0

total_time_third_cu_250 = 0

total_time_third_cu_300 = 0

total_time_third_cu_350 = 0

total_time_third_cu_400 = 0

total_accuracyC_cu = 0

total_accuracyC_cu_150 = 0

total_accuracyC_cu_200 = 0

total_accuracyC_cu_250 = 0

total_accuracyC_cu_300 = 0

total_accuracyC_cu_350 = 0

total_accuracyC_cu_400 = 0
```

```
total_accuracy2B_cu = 0
total_accuracy2B_cu_150 = 0
total_accuracy2B_cu_200 = 0
total_accuracy2B_cu_250 = 0
total_accuracy2B_cu_300 = 0
total_accuracy2B_cu_350 = 0
total_accuracy2B_cu_400 = 0
total_accuracy3B_cu = 0
total_accuracy3B_cu_150 = 0
total_accuracy3B_cu_200 = 0
total_accuracy3B_cu_250 = 0
total_accuracy3B_cu_300 = 0
total_accuracy3B_cu_350 = 0
total_accuracy3B_cu_400 = 0
average_timeC = 0
average_timeC_150 = 0
average_timeC_200 = 0
average_timeC_250 = 0
average_timeC_300 = 0
average_timeC_350 = 0
average_timeC_400 = 0
average_time2B = 0
average_time2B_150 = 0
```

```
average_time2B_200 = 0

average_time2B_250 = 0

average_time2B_300 = 0

average_time2B_350 = 0

average_time2B_400 = 0

average_time3B = 0

average_time3B_150 = 0

average_time3B_200 = 0

average_time3B_250 = 0

average_time3B_300 = 0

average_time3B_350 = 0

average_time3B_400 = 0

Average_Time_cu = 0

Average_Time_cu_150 = 0

Average_Time_cu_200 = 0

Average_Time_cu_250 = 0

Average_Time_cu_300 = 0

Average_Time_cu_350 = 0

Average_Time_cu_400 = 0

average_cutoff_timeC = 0

average_cutoff_timeC_150 = 0

average_cutoff_timeC_200 = 0

average_cutoff_timeC_250 = 0
```

```
average_cutoff_timeC_300 = 0
average_cutoff_timeC_350 = 0
average_cutoff_timeC_400 = 0
average_cutoff_time2B = 0
average_cutoff_time2B_150 = 0
average_cutoff_time2B_200 = 0
average_cutoff_time2B_250 = 0
average_cutoff_time2B_300 = 0
average_cutoff_time2B_350 = 0
average_cutoff_time2B_400 = 0
average_cutoff_time3B = 0
average_cutoff_time3B_150 = 0
average_cutoff_time3B_200 = 0
average_cutoff_time3B_250 = 0
average_cutoff_time3B_300 = 0
average_cutoff_time3B_350 = 0
average_cutoff_time3B_400 = 0
average_accuracy2B = 0
average_accuracy2B_150 = 0
average_accuracy2B_200 = 0
average_accuracy2B_250 = 0
average_accuracy2B_300 = 0
average_accuracy2B_350 = 0
```

```
average_accuracy2B_400 = 0

average_cutoff_time2B_150 = 0

average_cutoff_time2B_200 = 0

average_cutoff_time2B_250 = 0

average_cutoff_time2B_300 = 0

average_cutoff_time2B_350 = 0

average_cutoff_time2B_400 = 0

average_accuracy3B = 0

average_accuracy3B_150 = 0

average_accuracy3B_200 = 0

average_accuracy3B_250 = 0

average_accuracy3B_300 = 0

average_accuracy3B_350 = 0

average_accuracy3B_400 = 0

average_accuracyC = 0

average_accuracyC_150 = 0

average_accuracyC_200 = 0

average_accuracyC_250 = 0

average_accuracyC_300 = 0

average_accuracyC_350 = 0

average_accuracyC_400 = 0

cutoff_timeC = 0

cutoff_time2B = 0
```

```
cutoff_time3B = 0
checkOF = 0
check_home = 0
check_second = 0
check_third = 0
success_rate_2B = 0
success_rate_2B_150 = 0
success_rate_2B_200 = 0
success_rate_2B_250 = 0
success_rate_2B_300 = 0
success_rate_2B_350 = 0
success_rate_2B_400 = 0
success_rate_3B = 0
success_rate_3B_150 = 0
success_rate_3B_200 = 0
success_rate_3B_250 = 0
success_rate_3B_300 = 0
success_rate_3B_350 = 0
success_rate_3B_400 = 0

throws_caught_NC <- data.frame(matrix(ncol = 6, nrow = 0)) #data frame for visualization - only looking at outs
```

```
colnames(throws_caught_NC) <- c('Catcher_X', 'Catcher_Y', 'Second_X', 'Second_Y',
'Third_X', 'Third_Y')

throws_data_NC <- data.frame(matrix(ncol = 19, nrow = 0))
colnames(throws_data_NC) <- c('Success_Rate_2B', 'Success_Rate_3B',
                              'Throws_Home', 'Throws_Second', 'Throws_Third',
                              'Out_Home', 'Average_Time_Home_Outs',
'Average_Accuracy_Home_Outs',
                              'Average_Time_Second_Outs', 'Average_Accuracy_Second_Outs',
                              'Average_Time_Third_Outs', 'Average_Accuracy_Third_Outs',
                              'Average_Time_Outs', 'Average_Time_Outs_150',
'Average_Time_Outs_200',
                              'Average_Time_Outs_250', 'Average_Time_Outs_300',
'Average_Time_Outs_350',
                              'Average_Time_Outs_400') #Major Results Table

throws_caught_cu <- data.frame(matrix(ncol = 6, nrow = 0)) #data frame for visualization -
only looking at outs
colnames(throws_caught_cu) <- c('Catcher_X', 'Catcher_Y', 'Second_X', 'Second_Y',
'Third_X', 'Third_Y')

throws_data_cu <- data.frame(matrix(ncol = 19, nrow = 0))
colnames(throws_data_cu) <- c('Success_Rate_2B', 'Success_Rate_3B',
                              'Throws_Home', 'Throws_Second', 'Throws_Third',
                              'Out_Home', 'Average_Time_Home_Outs',
'Average_Accuracy_Home_Outs',
                              'Average_Time_Second_Outs', 'Average_Accuracy_Second_Outs',
                              'Average_Time_Third_Outs', 'Average_Accuracy_Third_Outs',
                              'Average_Time_Outs', 'Average_Time_Outs_150',
'Average_Time_Outs_200',
                              'Average_Time_Outs_250', 'Average_Time_Outs_300',
'Average_Time_Outs_350',
                              'Average_Time_Outs_400') #Major Results Table

get_x_y_i <- function(final_timeF, pos2){
  i2 = 1
  x2 = -10000
  y2 = -10000

  for(n in 1:end2){
    if(pl_pos2$timestamp[n] == final_timeF && pl_pos2$player_position[n] == pos2){
      i2 = n
      x2 = pl_pos2$field_x[n] #gets x distance from plate when ball is caught
      y2 = pl_pos2$field_y[n] #gets y distance from plate when ball is caught
      break
    }
```

```
 }
 return(c(i2, x2, y2))
}

for(index in 1:10){
 if(index==1){
   player_pos = read.csv("player_pos-1903_10_TeamNJ_TeamB.csv", header = TRUE)
   player_pos2 = player_pos %>% select(timestamp, player_position, field_x, field_y)
   pl_pos2 = player_pos2 %>% filter(complete.cases(.))
   game_events = read.csv("game_events-1903_10_TeamNJ_TeamB.csv", header = TRUE)
   game_events2 = game_events %>% select(play_id, timestamp, player_position,
event_code)
   ge = game_events2 %>% filter(complete.cases(.))
   game_info = read.csv("game_info-1903_10_TeamNJ_TeamB.csv", header = TRUE)
   game_info2 = game_info %>% select(play_per_game, top_bottom_inning,
second_baserunner, third_baserunner)
   gi = game_info2 %>% filter(complete.cases(.))
 }
 if(index==2){
   player_pos = read.csv("player_pos-1903_07_TeamND_TeamA2.csv", header = TRUE)
   player_pos2 = player_pos %>% select(timestamp, player_position, field_x, field_y)
   pl_pos2 = player_pos2 %>% filter(complete.cases(.))
```

```
    game_events = read.csv("game_events-1903_07_TeamND_TeamA2.csv", header = TRUE)
    game_events2 = game_events %>% select(play_id, timestamp, player_position, event_code)
    ge = game_events2 %>% filter(complete.cases(.))
    game_info = read.csv("game_info-1903_07_TeamND_TeamA2.csv", header = TRUE)
    game_info2 = game_info %>% select(play_per_game, top_bottom_inning, second_baserunner, third_baserunner)
    gi = game_info2 %>% filter(complete.cases(.))
  }
  if(index==3){ #1 throw home - confirmed out
    player_pos = read.csv("player_pos-1903_02_TeamNE_TeamA2.csv", header = TRUE)
    player_pos2 = player_pos %>% select(timestamp, player_position, field_x, field_y)
    pl_pos2 = player_pos2 %>% filter(complete.cases(.))
    game_events = read.csv("game_events-1903_02_TeamNE_TeamA2.csv", header = TRUE)
    game_events2 = game_events %>% select(play_id, timestamp, player_position, event_code)
    ge = game_events2 %>% filter(complete.cases(.))
    game_info = read.csv("game_info-1903_02_TeamNE_TeamA2.csv", header = TRUE)
    game_info2 = game_info %>% select(play_per_game, top_bottom_inning, second_baserunner, third_baserunner)
    gi = game_info2 %>% filter(complete.cases(.))
```

```
 }
 if(index==4){
   player_pos = read.csv("player_pos-1903_04_TeamNC_TeamA1.csv", header = TRUE)
   player_pos2 = player_pos %>% select(timestamp, player_position, field_x, field_y)
   pl_pos2 = player_pos2 %>% filter(complete.cases(.))
   game_events = read.csv("game_events-1903_04_TeamNC_TeamA1.csv", header = TRUE)
   game_events2 = game_events %>% select(play_id, timestamp, player_position, event_code)
   ge = game_events2 %>% filter(complete.cases(.))
   game_info = read.csv("game_info-1903_04_TeamNC_TeamA1.csv", header = TRUE)
   game_info2 = game_info %>% select(play_per_game, top_bottom_inning, second_baserunner, third_baserunner)
   gi = game_info2 %>% filter(complete.cases(.))
 }
 if(index==5){
   player_pos = read.csv("player_pos-1903_03_TeamNE_TeamA2.csv", header = TRUE)
   player_pos2 = player_pos %>% select(timestamp, player_position, field_x, field_y)
   pl_pos2 = player_pos2 %>% filter(complete.cases(.))
   game_events = read.csv("game_events-1903_03_TeamNE_TeamA2.csv", header = TRUE)
   game_events2 = game_events %>% select(play_id, timestamp, player_position, event_code)
```

```
    ge = game_events2 %>% filter(complete.cases(.))
    game_info = read.csv("game_info-1903_03_TeamNE_TeamA2.csv", header = TRUE)
    game_info2 = game_info %>% select(play_per_game, top_bottom_inning, second_baserunner, third_baserunner)
    gi = game_info2 %>% filter(complete.cases(.))
  }
  if(index==6){
    player_pos = read.csv("player_pos-1903_06_TeamND_TeamA2.csv", header = TRUE)
    player_pos2 = player_pos %>% select(timestamp, player_position, field_x, field_y)
    pl_pos2 = player_pos2 %>% filter(complete.cases(.))
    game_events = read.csv("game_events-1903_06_TeamND_TeamA2.csv", header = TRUE)
    game_events2 = game_events %>% select(play_id, timestamp, player_position, event_code)
    ge = game_events2 %>% filter(complete.cases(.))
    game_info = read.csv("game_info-1903_06_TeamND_TeamA2.csv", header = TRUE)
    game_info2 = game_info %>% select(play_per_game, top_bottom_inning, second_baserunner, third_baserunner)
    gi = game_info2 %>% filter(complete.cases(.))
  }
  if(index==7){
    player_pos = read.csv("player_pos-1903_08_TeamNJ_TeamB.csv", header = TRUE)
    player_pos2 = player_pos %>% select(timestamp, player_position, field_x, field_y)
    pl_pos2 = player_pos2 %>% filter(complete.cases(.))
    game_events = read.csv("game_events-1903_08_TeamNJ_TeamB.csv", header = TRUE)
    game_events2 = game_events %>% select(play_id, timestamp, player_position,
event_code)
    ge = game_events2 %>% filter(complete.cases(.))
    game_info = read.csv("game_info-1903_08_TeamNJ_TeamB.csv", header = TRUE)
    game_info2 = game_info %>% select(play_per_game, top_bottom_inning,
second_baserunner, third_baserunner)
    gi = game_info2 %>% filter(complete.cases(.))
  }
  if(index==8){
    player_pos = read.csv("player_pos-1903_09_TeamNJ_TeamB.csv", header = TRUE)
    player_pos2 = player_pos %>% select(timestamp, player_position, field_x, field_y)
    pl_pos2 = player_pos2 %>% filter(complete.cases(.))
    game_events = read.csv("game_events-1903_09_TeamNJ_TeamB.csv", header = TRUE)
    game_events2 = game_events %>% select(play_id, timestamp, player_position,
event_code)
    ge = game_events2 %>% filter(complete.cases(.))
    game_info = read.csv("game_info-1903_09_TeamNJ_TeamB.csv", header = TRUE)
    game_info2 = game_info %>% select(play_per_game, top_bottom_inning,
second_baserunner, third_baserunner)
    gi = game_info2 %>% filter(complete.cases(.))
  }
  if(index==9){
    player_pos = read.csv("player_pos-1903_05_TeamND_TeamA2.csv", header = TRUE)
    player_pos2 = player_pos %>% select(timestamp, player_position, field_x, field_y)
    pl_pos2 = player_pos2 %>% filter(complete.cases(.))
    game_events = read.csv("game_events-1903_05_TeamND_TeamA2.csv", header =
TRUE)
    game_events2 = game_events %>% select(play_id, timestamp, player_position,
event_code)
    ge = game_events2 %>% filter(complete.cases(.))
    game_info = read.csv("game_info-1903_05_TeamND_TeamA2.csv", header = TRUE)
    game_info2 = game_info %>% select(play_per_game, top_bottom_inning,
second_baserunner, third_baserunner)
    gi = game_info2 %>% filter(complete.cases(.))
  }

  if(index==10){
    player_pos = read.csv("player_pos-1903_01_TeamNE_TeamA2.csv", header = TRUE)
    player_pos2 = player_pos %>% select(timestamp, player_position, field_x, field_y)
    pl_pos2 = player_pos2 %>% filter(complete.cases(.))
    game_events = read.csv("game_events-1903_01_TeamNE_TeamA2.csv", header =
TRUE)
```

```
   game_events2 = game_events %>% select(play_id, timestamp, player_position, event_code)
   ge = game_events2 %>% filter(complete.cases(.))
   game_info = read.csv("game_info-1903_01_TeamNE_TeamA2.csv", header = TRUE)
   game_info2 = game_info %>% select(play_per_game, top_bottom_inning, second_baserunner, third_baserunner)
   gi = game_info2 %>% filter(complete.cases(.))
  }

  end = nrow(ge)
  end2 = nrow(pl_pos2)
  end3 = nrow(gi)

  final_time_catcher = 0
  final_time_second = 0
  final_time_third = 0
  a = 1
  j = 1
  k = 1
  m = 1

  ###Non-Cutoff Throws:
```

```
  while(a<(end-2)){ #Start at index of 2 because play won't begin with a throw by an
outfielder.
    while(ge$event_code[a] != 5 && a <= (end-2)){ #runs until play ends
      if(ge$event_code[a]==2 && (ge$player_position[a]==7 || ge$player_position[a]==8 ||
                                  ge$player_position[a]==9) && ge$event_code[a+1]==3){ #checks if ball
was acquired by an outfielder then thrown
        checkOF_NC = 1
        j=a+2 #set to after throw is made by outfielder
        beginning_time = ge$timestamp[a] #gets time when ball is caught by outfielder
        for(b in 1:end2){
          if(pl_pos2$timestamp[b] == ge$timestamp[a+1] && (pl_pos2$player_position[b] ==
7 ||
                                                             pl_pos2$player_position[b] == 8 ||
                                                             pl_pos2$player_position[b] == 9)){
            of_x = pl_pos2$field_x[b] #gets x distance from plate when ball is thrown by
outfielder
            of_y = pl_pos2$field_y[b] #gets y distance from plate when ball is thrown by
outfielder
          }
        }

        #Throws home: (0, .7)
    while(ge$event_code[j-1] != 2 && ge$play_id[j] == ge$play_id[j-1]){ #runs until right
after ball is acquired or until play ends
     changed = 0
     if(ge$event_code[j] == 2 && ge$player_position[j] == 2){ #Makes sure throw goes to
catcher.
      final_time_catcher = ge$timestamp[j]
      catcher_x = get_x_y_i(final_time_catcher, 2)[2]
      catcher_y = get_x_y_i(final_time_catcher, 2)[3]
      if(is.na(catcher_x)){
       catcher_x = 0
       changed = 1
      }
      if(is.na(catcher_y)){
       catcher_y = 0.7
       changed = 1
      }
      if(catcher_x < 16 && catcher_x > -16 && catcher_y < 16.7 && catcher_y > -15.7
&& checkOF_NC == 1){ #Checks that catcher caught the ball near or on home plate so is
not backing up a throw
       check_home_NC = 1
       throw_distance_C_NC = sqrt((of_x - catcher_x)^2 + (of_y-catcher_y)^2)
       diff_catcher = (final_time_catcher - beginning_time)/1000
       for(n in 1:(end3-1)){
```

```
if(gi$play_per_game[n] == ge$play_id[j]){
 if(gi$top_bottom_inning[n] != gi$top_bottom_inning[n+1]){
  print("The runner was out at home.")
  out_catcher_NC = out_catcher_NC + 1
  throws_caught_NC = throws_caught_NC %>% add_row(Catcher_X = catcher_x, Catcher_Y = catcher_y)
  if(throw_distance_C_NC >= 150 && throw_distance_C_NC <= 199.99){
   out_catcher_NC_150 = out_catcher_NC_150 + 1
   throws_home_NC_150 = throws_home_NC_150 + 1
   total_time_home_NC_150 = total_time_home_NC_150 + diff_catcher
  }
  if(throw_distance_C_NC >= 200 && throw_distance_C_NC <= 249.99){
   out_catcher_NC_200 = out_catcher_NC_200 + 1
   throws_home_NC_200 = throws_home_NC_200 + 1
   total_time_home_NC_200 = total_time_home_NC_200 + diff_catcher
  }
  if(throw_distance_C_NC >= 250 && throw_distance_C_NC <= 299.99){
   out_catcher_NC_250 = out_catcher_NC_250 + 1
   throws_home_NC_250 = throws_home_NC_250 + 1
   total_time_home_NC_250 = total_time_home_NC_250 + diff_catcher
  }
  if(throw_distance_C_NC >= 300 && throw_distance_C_NC <= 349.99){
   out_catcher_NC_300 = out_catcher_NC_300 + 1
```

```
          throws_home_NC_300 = throws_home_NC_300 + 1
          total_time_home_NC_300 = total_time_home_NC_300 + diff_catcher
        }
        if(throw_distance_C_NC >= 350 && throw_distance_C_NC <= 399.99){
          out_catcher_NC_350 = out_catcher_NC_350 + 1
          throws_home_NC_350 = throws_home_NC_350 + 1
          total_time_home_NC_350 = total_time_home_NC_350 + diff_catcher
        }
        if(throw_distance_C_NC >= 400){
          out_catcher_NC_400 = out_catcher_NC_400 + 1
          throws_home_NC_400 = throws_home_NC_400 + 1
          total_time_home_NC_400 = total_time_home_NC_400 + diff_catcher
        }
        total_time_home_NC = total_time_home_NC + diff_catcher
        if(changed == 0){
          accuracyC = sqrt((catcher_x)^2 + (catcher_y - 0.7)^2) #Checks how far off
throw is from C
          total_accuracyC_NC = total_accuracyC_NC + accuracyC #Gets how far off
the throw to home is
          print(paste("Throw was", round(accuracyC, 2), "feet off from home."))

          if(throw_distance_C_NC >= 150 && throw_distance_C_NC <= 199.99){
            accuracyC_150 = accuracyC
```

```
   total_accuracyC_NC_150 = total_accuracyC_NC_150 + accuracyC_150
  }
  if(throw_distance_C_NC >= 200 && throw_distance_C_NC <= 249.99){
   accuracyC_200 = accuracyC
   total_accuracyC_NC_200 = total_accuracyC_NC_200 + accuracyC_200
  }
  if(throw_distance_C_NC >= 250 && throw_distance_C_NC <= 299.99){
   accuracyC_250 = accuracyC
   total_accuracyC_NC_250 = total_accuracyC_NC_250 + accuracyC_250
  }
  if(throw_distance_C_NC >= 300 && throw_distance_C_NC <= 349.99){
   accuracyC_300 = accuracyC
   total_accuracyC_NC_300 = total_accuracyC_NC_300 + accuracyC_300
  }
  if(throw_distance_C_NC >= 350 && throw_distance_C_NC <= 399.99){
   accuracyC_350 = accuracyC
   total_accuracyC_NC_350 = total_accuracyC_NC_350 + accuracyC_350
  }
  if(throw_distance_C_NC >= 400){
   accuracyC_400 = accuracyC
   total_accuracyC_NC_400 = total_accuracyC_NC_400 + accuracyC_400
  }
 }
```

```
  break
}
else{
  print("Unsure if out or safe at home.")
  if(throw_distance_C_NC >= 150 && throw_distance_C_NC <= 199.99){
    throws_home_NC_150 = throws_home_NC_150 + 1
  }
  if(throw_distance_C_NC >= 200 && throw_distance_C_NC <= 249.99){
    throws_home_NC_200 = throws_home_NC_200 + 1
  }
  if(throw_distance_C_NC >= 250 && throw_distance_C_NC <= 299.99){
    throws_home_NC_250 = throws_home_NC_250 + 1
  }
  if(throw_distance_C_NC >= 300 && throw_distance_C_NC <= 349.99){
    throws_home_NC_300 = throws_home_NC_300 + 1
  }
  if(throw_distance_C_NC >= 350 && throw_distance_C_NC <= 399.99){
    throws_home_NC_350 = throws_home_NC_350 + 1
  }
  if(throw_distance_C_NC >= 400){
    throws_home_NC_400 = throws_home_NC_400 + 1
  }
  break
```

```
         }
        }
       }
       print(paste("Took", round(diff_catcher, 2), "seconds to throw ball home. Throw distance was",
             round(throw_distance_C_NC, 2), "feet to home")) #prints how long it took to throw ball to home after catching it and throw distance
       cat("\n")
      }
     }
     j=j+1
    }

    #Throws to third: (-63.7, 63.7)
    k=a+2
    while(ge$event_code[k-1] != 2 && ge$play_id[k] == ge$play_id[k-1]){ #runs until right after ball is acquired or until play ends
     changed = 0
     if(ge$event_code[k] == 2 && ge$player_position[k] == 5){ #Checks that third baseman caught the ball
      final_time_third = ge$timestamp[j]
      third_x = get_x_y_i(final_time_third, 5)[2]
      third_y = get_x_y_i(final_time_third, 5)[3]
```

```
if(is.na(third_x)){
  third_x = -63.7
  changed = 1
}
if(is.na(third_y)){
  third_x = 63.7
  changed = 1
}
if(third_x < -57.7 && third_x > -79.7 && third_y < 79.7 && third_y > 57.7 && checkOF_NC == 1){ #Checks that third baseman caught the ball near or on third base so is not cutting it off
  check_third_NC = 1
  throw_distance_3B_NC = sqrt((of_x - third_x)^2 + (of_y - third_y)^2)
  diff_third = (final_time_third - beginning_time)/1000
  for(n in 1:(end3-1)){
    if(gi$play_per_game[n] == ge$play_id[k]){
      if(gi$third_baserunner[n+1] != 0){
        print("The runner was safe at third.")
        safe_third_NC = safe_third_NC + 1
        if(throw_distance_3B_NC >= 150 && throw_distance_3B_NC <= 199.99){
          throws_third_NC_150 = throws_third_NC_150 + 1
          safe_third_NC_150 = safe_third_NC_150 + 1
        }
        if(throw_distance_3B_NC >= 200 && throw_distance_3B_NC <= 249.99){
          throws_third_NC_200 = throws_third_NC_200 + 1
          safe_third_NC_200 = safe_third_NC_200 + 1
        }
        if(throw_distance_3B_NC >= 250 && throw_distance_3B_NC <= 299.99){
          throws_third_NC_250 = throws_third_NC_250 + 1
          safe_third_NC_250 = safe_third_NC_250 + 1
        }
        if(throw_distance_3B_NC >= 300 && throw_distance_3B_NC <= 349.99){
          throws_third_NC_300 = throws_third_NC_300 + 1
          safe_third_NC_300 = safe_third_NC_300 + 1
        }
        if(throw_distance_3B_NC >= 350 && throw_distance_3B_NC <= 399.99){
          throws_third_NC_350 = throws_third_NC_350 + 1
          safe_third_NC_350 = safe_third_NC_350 + 1
        }
        if(throw_distance_3B_NC >= 400){
          throws_third_NC_400 = throws_third_NC_400 + 1
          safe_third_NC_400 = safe_third_NC_400 + 1
        }
        break
      }
      else{
```

```
print("The runner was out at third")
out_third_NC = out_third_NC + 1
total_time_third_NC = total_time_third_NC + diff_third
throws_caught_NC = throws_caught_NC %>% add_row(Third_X = third_x, Third_Y = third_y)
if(throw_distance_3B_NC >= 150 && throw_distance_3B_NC <= 199.99){
  out_third_NC_150 = out_third_NC_150 + 1
  throws_third_NC_150 = throws_third_NC_150 + 1
  total_time_third_NC_150 = total_time_third_NC_150 + diff_third
}
if(throw_distance_3B_NC >= 200 && throw_distance_3B_NC <= 249.99){
  out_third_NC_200 = out_third_NC_200 + 1
  throws_third_NC_200 = throws_third_NC_200 + 1
  total_time_third_NC_200 = total_time_third_NC_200 + diff_third
}
if(throw_distance_3B_NC >= 250 && throw_distance_3B_NC <= 299.99){
  out_third_NC_250 = out_third_NC_250 + 1
  throws_third_NC_250 = throws_third_NC_250 + 1
  total_time_third_NC_250 = total_time_third_NC_250 + diff_third
}
if(throw_distance_3B_NC >= 300 && throw_distance_3B_NC <= 349.99){
  out_third_NC_300 = out_third_NC_300 + 1
  throws_third_NC_300 = throws_third_NC_300 + 1
  total_time_third_NC_300 = total_time_third_NC_300 + diff_third
}
if(throw_distance_3B_NC >= 350 && throw_distance_3B_NC <= 399.99){
  out_third_NC_350 = out_third_NC_350 + 1
  throws_third_NC_350 = throws_third_NC_350 + 1
  total_time_third_NC_350 = total_time_home_NC_350 + diff_third
}
if(throw_distance_3B_NC >= 400){
  out_third_NC_400 = out_third_NC_400 + 1
  throws_third_NC_400 = throws_third_NC_400 + 1
  total_time_third_NC_400 = total_time_third_NC_400 + diff_third
}
if(changed == 0){
  accuracy3B = sqrt((third_x + 63.7)^2 + (third_y - 63.7)^2) #Checks how far off throw is from third
  total_accuracy3B_NC = total_accuracy3B_NC + accuracy3B #Gets how far off the throw to third is
  print(paste("Throw was", round(accuracy3B, 2), "feet off from third."))

  if(throw_distance_3B_NC >= 150 && throw_distance_3B_NC <= 199.99){
    accuracy3B_150 = accuracy3B
    total_accuracy3B_NC_150 = total_accuracy3B_NC_150 + accuracy3B_150
  }
```

```
    if(throw_distance_3B_NC >= 200 && throw_distance_3B_NC <= 249.99){
      accuracy3B_200 = accuracy3B
      total_accuracy3B_NC_200 = total_accuracy3B_NC_200 + accuracy3B_200
    }
    if(throw_distance_3B_NC >= 250 && throw_distance_3B_NC <= 299.99){
      accuracy3B_250 = accuracy3B
      total_accuracy3B_NC_250 = total_accuracy3B_NC_250 + accuracy3B_250
    }
    if(throw_distance_3B_NC >= 300 && throw_distance_3B_NC <= 349.99){
      accuracy3B_300 = accuracy3B
      total_accuracy3B_NC_300 = total_accuracy3B_NC_300 + accuracy3B_300
    }
    if(throw_distance_3B_NC >= 350 && throw_distance_3B_NC <= 399.99){
      accuracy3B_350 = accuracy3B
      total_accuracy3B_NC_350 = total_accuracy3B_NC_350 + accuracy3B_350
    }
    if(throw_distance_3B_NC >= 400){
      accuracy3B_400 = accuracy3B
      total_accuracy3B_NC_400 = total_accuracy3B_NC_400 + accuracy3B_400
    }
  }
  break
}
```

```
        }
       }
       print(paste("Took", round(diff_third, 2), "seconds to throw ball to third. Throw
distance was",
            round(throw_distance_3B_NC, 2), "feet to third")) #prints how long it took
to throw ball to third after catching it and throw distance
       cat("\n")
      }
     }
     k=k+1
    }

    #Throws to second: (0, 127.3)
    m = a + 2
    while(ge$event_code[m-1] != 2 && ge$play_id[m] == ge$play_id[m-1]){ #runs until
right after ball is acquired or until play ends
     changed = 0
     if(ge$event_code[m] == 2 && ge$player_position[m] == 4){ #Checks that second
baseman caught the ball
      final_time_second = ge$timestamp[m] #Gets time ball is caught by 2B
      second_x = get_x_y_i(final_time_second, 4)[2]
      second_y = get_x_y_i(final_time_second, 4)[3]
      if(is.na(second_x))
       second_x = 0
      if(is.na(second_y))
       second_y = 127.3
      if(second_x < 16 &&
         second_x > -16 && second_y < 143.3 &&
         second_y > 111.3 && checkOF_NC == 1){ #Checks that second baseman caught
the ball near or on second base so is not cutting it off
       check_second_NC = 1
       throw_distance_2B_NC = sqrt((of_x - second_x)^2 + (of_y - second_y)^2)
       diff_second = (final_time_second - beginning_time)/1000
       for(n in 1:(end3-1)){
        if(gi$play_per_game[n] == ge$play_id[m]){
          if(gi$second_baserunner[n+1] != 0){
            print("The runner was safe at second.")
            safe_second_NC = safe_second_NC + 1
            if(throw_distance_2B_NC >= 150 && throw_distance_2B_NC <= 199.99){
              throws_second_NC_150 = throws_second_NC_150 + 1
              safe_second_NC_150 = safe_second_NC_150 + 1
            }
            if(throw_distance_2B_NC >= 200 && throw_distance_2B_NC <= 249.99){
              throws_second_NC_200 = throws_second_NC_200 + 1
              safe_second_NC_200 = safe_second_NC_200 + 1
            }
            if(throw_distance_2B_NC >= 250 && throw_distance_2B_NC <= 299.99){
              throws_second_NC_250 = throws_second_NC_250 + 1
              safe_second_NC_250 = safe_second_NC_250 + 1
            }
            if(throw_distance_2B_NC >= 300 && throw_distance_2B_NC <= 349.99){
              throws_second_NC_300 = throws_second_NC_300 + 1
              safe_second_NC_300 = safe_second_NC_300 + 1
            }
            if(throw_distance_2B_NC >= 350 && throw_distance_2B_NC <= 399.99){
              throws_second_NC_350 = throws_second_NC_350 + 1
              safe_second_NC_350 = safe_second_NC_350 + 1
            }
            if(throw_distance_2B_NC >= 400){
              throws_second_NC_400 = throws_second_NC_400 + 1
              safe_second_NC_400 = safe_second_NC_400 + 1
            }
          }
          else{
            print("The runner was out at second")
            out_second_NC = out_second_NC + 1
            total_time_second_NC = total_time_second_NC + diff_second
             throws_caught_NC = throws_caught_NC %>% add_row(Second_X = second_x, Second_Y = second_y)
```

```
if(throw_distance_2B_NC >= 150 && throw_distance_2B_NC <= 199.99){
  out_second_NC_150 = out_second_NC_150 + 1
  throws_second_NC_150 = throws_second_NC_150 + 1
  total_time_second_NC_150 = total_time_second_NC_150 + diff_second


}
if(throw_distance_2B_NC >= 200 && throw_distance_2B_NC <= 249.99){
  out_second_NC_200 = out_second_NC_200 + 1
  throws_second_NC_200 = throws_second_NC_200 + 1
  total_time_second_NC_200 = total_time_second_NC_200 + diff_second
}
if(throw_distance_2B_NC >= 250 && throw_distance_2B_NC <= 299.99){
  out_second_NC_250 = out_second_NC_250 + 1
  throws_second_NC_250 = throws_second_NC_250 + 1
  total_time_second_NC_150 = total_time_second_NC_150 + diff_second
}
if(throw_distance_2B_NC >= 300 && throw_distance_2B_NC <= 349.99){
  out_second_NC_300 = out_second_NC_300 + 1
  throws_second_NC_300 = throws_second_NC_300 + 1
  total_time_second_NC_300 = total_time_second_NC_300 + diff_second
}
if(throw_distance_2B_NC >= 350 && throw_distance_2B_NC <= 399.99){
  out_second_NC_350 = out_second_NC_350 + 1
  throws_second_NC_350 = throws_second_NC_350 + 1
  total_time_second_NC_350 = total_time_second_NC_350 + diff_second
}
if(throw_distance_2B_NC >= 400){
  out_second_NC_400 = out_second_NC_400 + 1
  throws_second_NC_400 = throws_second_NC_400 + 1
  total_time_second_NC_400 = total_time_second_NC_400 + diff_second
}
if(changed == 0){
  accuracy2B = sqrt((second_x)^2 + (second_y - 127.3)^2) #Checks how far off throw is
  total_accuracy2B_NC = total_accuracy2B_NC + accuracy2B #Gets how far off every throw to second is
  print(paste("Throw was", round(accuracy2B, 2), "feet off from second base"))

  if(throw_distance_2B_NC >= 150 && throw_distance_2B_NC <= 199.99){
    accuracy2B_150 = accuracy2B
    total_accuracy2B_NC_150 = total_accuracy2B_NC_150 + accuracy2B_150
  }
  if(throw_distance_2B_NC >= 200 && throw_distance_2B_NC <= 249.99){
    accuracy2B_200 = accuracy2B
    total_accuracy2B_NC_200 = total_accuracy2B_NC_200 + accuracy2B_200
    }
    if(throw_distance_2B_NC >= 250 && throw_distance_2B_NC <= 299.99){
     accuracy2B_250 = accuracy2B
     total_accuracy2B_NC_250 = total_accuracy2B_NC_250 + accuracy2B_250
    }
    if(throw_distance_2B_NC >= 300 && throw_distance_2B_NC <= 349.99){
     accuracy2B_300 = accuracy2B
     total_accuracy2B_NC_300 = total_accuracy2B_NC_300 + accuracy2B_300
    }
    if(throw_distance_2B_NC >= 350 && throw_distance_2B_NC <= 399.99){
     accuracy2B_350 = accuracy2B
     total_accuracy2B_NC_350 = total_accuracy2B_NC_350 + accuracy2B_350
    }
    if(throw_distance_2B_NC >= 400){
     accuracy2B_400 = accuracy2B
     total_accuracy2B_NC_400 = total_accuracy2B_NC_400 + accuracy2B_400
    }
   }
  }
 }
}
print(paste("Took", round(diff_second, 2), "seconds to throw ball to second.
Throw distance was",
             round(throw_distance_2B_NC, 2), "feet to second")) #prints how long it
       #took to throw ball to second after catching it and throw distance
       cat("\n")
      }
     }
     m=m+1
    }
   }
   if(checkOF_NC==1 && check_home_NC==1) #checks if a throw was made by an
outfielder to home
    throws_home_NC = throws_home_NC + 1
   if(checkOF_NC==1 && check_third_NC==1) #checks if a throw was made by an
outfielder to third with no cutoff.
    throws_third_NC = throws_third_NC + 1
   if(checkOF_NC==1 && check_second_NC==1) #checks if a throw was made by an
outfielder to second with no cutoff.
    throws_second_NC = throws_second_NC + 1

   checkOF_NC = 0
   check_home_NC = 0
   check_second_NC = 0
   check_third_NC = 0
   a=a+1
```

```
  }
  a = a+1
}

###Cutoff Throws:
#3B: (-63.7, 63.7)
#2B: (0, 127.3)


final_time_home = 0
final_time_second = 0
final_time_third = 0
a = 1
j = 1
k = 1
catcher_x = -1000
catcher_y = -1000
second_x = -1000
second_y = -1000
third_x = -1000
third_y = -1000
pos = 0
```

```
 cutoff_pos_home <- function(z, x1, x2, y1, y2, position, ofX, catcherX, ofY, catcherY,
finalTime, beginningTime, e, checkO){
  throw_distance_C = 0
  check_home = 0
  safe_home = 0
  diff = 0
  cutoff_diff = 0
  cutoff_throw_dist = 0 #Cutoff throw distance
  total_cutoff_time = 0
  changed = 0
  total_accuracyC = 0
  total_accuracyC_150 = 0
  total_accuracyC_200 = 0
  total_accuracyC_250 = 0
  total_accuracyC_300 = 0
  total_accuracyC_350 = 0
  total_accuracyC_400 = 0
  outs_home_cu = 0
  outs_home_cu_150 = 0
  outs_home_cu_200 = 0
  outs_home_cu_250 = 0
  outs_home_cu_300 = 0
  outs_home_cu_350 = 0
```

```
outs_home_cu_400 = 0
total_time_home = 0
total_time_home_150 = 0
total_time_home_200 = 0
total_time_home_250 = 0
total_time_home_300 = 0
total_time_home_350 = 0
total_time_home_400 = 0
total_accuracyC_150 = 0
total_accuracyC_200 = 0
total_accuracyC_250 = 0
total_accuracyC_300 = 0
total_accuracyC_350 = 0
total_accuracyC_400 = 0
throws_home_150 = 0
throws_home_200 = 0
throws_home_250 = 0
throws_home_300 = 0
throws_home_350 = 0
throws_home_400 = 0

cutoff_time = ge$timestamp[z] #time when cutoff man got the ball
cutoff_x = get_x_y_i(cutoff_time, position)[2] #gets exact x position of cutoff man
```

```
  cutoff_y = get_x_y_i(cutoff_time, position)[3] #gets exact y position of cutoff man


  if(cutoff_x < x1 &&
   cutoff_x > x2 && cutoff_y < y1 &&
   cutoff_y > y2  && checkO == 1){ #Checks that cutoff man is around the cutoff position
   check_home = 1
   if(catcherX == -1000){
    catcherX = 0  #sets catcher baseman's x value to x value of catcher base if didn't get exact x value of catcher baseman
    changed = 1
   }
   if(catcherY == -1000) {
    catcherY = 0.7 #sets catcher baseman's y value to y value of catcher base if didn't get exact y value of catcher baseman
    changed = 1
   }
   throw_distance_C = sqrt((ofX - catcherX)^2 + (ofY - catcherY)^2)
   diff = (finalTime - beginningTime)/1000 #time to throw ball to catcher from outfield with cutoff
   cutoff_diff = (finalTime - cutoff_time)/1000 #time to throw ball to catcher from cutoff man
   cutoff_throw_dist = sqrt((cutoff_x - catcherX)^2 + (cutoff_y - catcherY)^2)
```

```
for(n in 1:(e-1)){
  if(!is.na(gi$play_per_game[n]) && !is.na(ge$play_id[z])){
    if(gi$play_per_game[n] == ge$play_id[z]){
      if(gi$top_bottom_inning[n] != gi$top_bottom_inning[n+1]){
        print("The runner was out at home.")
        outs_home_cu = outs_home_cu + 1
        total_time_home = total_time_home + diff
        if(throw_distance_C >= 150 && throw_distance_C <= 199.99){
          outs_home_cu_150 = 1
          throws_home_150 = 1
          total_time_home_150 = total_time_home_150 + diff
        }
        if(throw_distance_C >= 200 && throw_distance_C <= 249.99){
          outs_home_cu_200 = 1
          throws_home_200 = 1
          total_time_home_200 = total_time_home_200 + diff
        }
        if(throw_distance_C >= 250 && throw_distance_C <= 299.99){
          outs_home_cu_250 = 1
          throws_home_250 = 1
          total_time_home_250 = total_time_home_250 + diff
        }
        if(throw_distance_C >= 300 && throw_distance_C <= 349.99){
```

```
  outs_home_cu_300 =  1
  throws_home_300 = 1
  total_time_home_300 = total_time_home_300 + diff
}
if(throw_distance_C >= 350 && throw_distance_C <= 399.99){
  outs_home_cu_350 =  1
  throws_home_350 = 1
  total_time_home_350 = total_time_home_350 + diff
}
if(throw_distance_C >= 400){
  outs_home_cu_400 =  1
  throws_home_400 = 1
  total_time_home_400 = total_time_home_400 + diff
}
if(changed == 0){
  accuracyC = sqrt((catcher_x)^2 + (catcher_y - 0.7)^2) #Checks how far off throw is from C
  total_accuracyC = total_accuracyC + accuracyC #Gets how far off every throw to catcher is
  print(paste("Throw was", round(accuracyC, 2), "feet off from home."))
  if(throw_distance_C >= 150 && throw_distance_C <= 199.99){
    accuracyC_150 = accuracyC #Checks how far off throw is from C
    total_accuracyC_150 = total_accuracyC_150 + accuracyC_150 #Gets how far off every throw to catcher is
  }
  if(throw_distance_C >= 200 && throw_distance_C <= 249.99){
    accuracyC_200 = accuracyC
    total_accuracyC_200 = total_accuracyC_200 + accuracyC_200
  }
  if(throw_distance_C >= 250 && throw_distance_C <= 299.99){
    accuracyC_250 = accuracyC
    total_accuracyC_250 = total_accuracyC_250 + accuracyC_250
  }
  if(throw_distance_C >= 300 && throw_distance_C <= 349.99){
    accuracyC_300 = accuracyC
    total_accuracyC_300 = total_accuracyC_300 + accuracyC_300
  }
  if(throw_distance_C >= 350 && throw_distance_C <= 399.99){
    accuracyC_350 = accuracyC
    total_accuracyC_350 = total_accuracyC_350 + accuracyC_350
  }
  if(throw_distance_C >= 400){
    accuracyC_400 = accuracyC
    total_accuracyC_400 = total_accuracyC_400 + accuracyC_400
  }
  }
  break
}
else{
  print("Unsure if out or safe at home.")
  if(throw_distance_C >= 150 && throw_distance_C <= 199.99){
    throws_home_150 = 1
  }
  if(throw_distance_C >= 200 && throw_distance_C <= 249.99){
    throws_home_200 = 1
  }
  if(throw_distance_C >= 250 && throw_distance_C <= 299.99){
    throws_home_250 = 1
  }
  if(throw_distance_C >= 300 && throw_distance_C <= 349.99){
    throws_home_300 = 1
  }
  if(throw_distance_C >= 350 && throw_distance_C <= 399.99){
    throws_home_350 = 1
  }
  if(throw_distance_C >= 400){
    throws_home_400 = 1
  }
```

```
       break
      }
     }
    }
   }

   if(check_home==1){
    if(finalTime != 0){ #Only print if have final time
     print(paste("Took", round(diff, 2), "seconds to get the ball from the outfielder to
catcher",
           "Took", round(cutoff_diff, 2), "seconds for cutoff man to throw ball to
catcher"))
    }
    print(paste("Throw distance from outfielder to catcher was",
round(throw_distance_C, 2), "feet to catcher",
          "Throw distance from cutoff man to catcher was", round(cutoff_throw_dist, 2),
"feet to catcher"))
    cat("\n")
   }
  }
  return(c(check_home, outs_home_cu, safe_home, total_time_home, total_cutoff_time,
total_accuracyC,
            outs_home_cu_150, outs_home_cu_200, outs_home_cu_250, outs_home_cu_300,
outs_home_cu_350,
            outs_home_cu_400, total_time_home_150, total_time_home_200,
total_time_home_250,
            total_time_home_300, total_time_home_350, total_time_home_400,
total_accuracyC_150,
            total_accuracyC_200, total_accuracyC_250, total_accuracyC_300,
total_accuracyC_350,
            total_accuracyC_400, throws_home_150, throws_home_200, throws_home_250,
throws_home_300,
            throws_home_350, throws_home_400))
  }

  cutoff_pos_second <- function(z, x1, x2, y1, y2, position, ofX, secondX, ofY, secondY,
finalTime, beginningTime, e, checkO){
    throw_distance_2B = 0
    check_second = 0
    safe_second = 0
    outs_second_cu = 0
    outs_second_cu_150 = 0
    outs_second_cu_200 = 0
    outs_second_cu_250 = 0
    outs_second_cu_300 = 0
```

```
outs_second_cu_350 = 0

outs_second_cu_400 = 0

total_time_second = 0

total_time_second_150 = 0

total_time_second_200 = 0

total_time_second_250 = 0

total_time_second_300 = 0

total_time_second_350 = 0

total_time_second_400 = 0

diff = 0

cutoff_diff = 0

cutoff_throw_dist = 0 #Cutoff throw distance

total_cutoff_time = 0

changed = 0

total_accuracy2B = 0

total_accuracy2B_150 = 0

total_accuracy2B_200 = 0

total_accuracy2B_250 = 0

total_accuracy2B_300 = 0

total_accuracy2B_350 = 0

total_accuracy2B_400 = 0

outs_second_cu = 0

outs_second_cu_150 = 0
```

```
outs_second_cu_200 = 0

outs_second_cu_250 = 0

outs_second_cu_300 = 0

outs_second_cu_350 = 0

outs_second_cu_400 = 0

total_time_second = 0

total_time_second_150 = 0

total_time_second_200 = 0

total_time_second_250 = 0

total_time_second_300 = 0

total_time_second_350 = 0

total_time_second_400 = 0

total_accuracy2B_150 = 0

total_accuracy2B_200 = 0

total_accuracy2B_250 = 0

total_accuracy2B_300 = 0

total_accuracy2B_350 = 0

total_accuracy2B_400 = 0

throws_second_150 = 0

throws_second_200 = 0

throws_second_250 = 0

throws_second_300 = 0

throws_second_350 = 0
```

```
    throws_second_400 = 0


    cutoff_time = ge$timestamp[z] #time when cutoff man got the ball
    cutoff_x = get_x_y_i(cutoff_time, position)[2] #gets exact x position of cutoff man
    cutoff_y = get_x_y_i(cutoff_time, position)[3] #gets exact y position of cutoff man


    if(cutoff_x < x1 &&
       cutoff_x > x2 && cutoff_y < y1 &&
       cutoff_y > y2 && checkO == 1){ #Checks that cutoff man is around the cutoff position
      check_second = 1
      if(secondX == -1000){
        secondX = 0  #sets second baseman's x value to x value of second base if didn't get
exact x value of second baseman
        changed = 1
      }
      if(secondY == -1000) {
        secondY = 127.3 #sets second baseman's y value to y value of second base if didn't get
exact y value of second baseman
        changed = 1
      }
      throw_distance_2B = sqrt((ofX - secondX)^2 + (ofY - secondY)^2)
      diff = (finalTime - beginningTime)/1000 #time to throw ball to second from outfield
with cutoff
```

```
  cutoff_diff = (finalTime - cutoff_time)/1000 #time to throw ball to second from cutoff man
  cutoff_throw_dist = sqrt((cutoff_x - secondX)^2 + (cutoff_y - secondY)^2)


  for(n in 1:(e-1)){
   if(!is.na(gi$play_per_game[n]) && !is.na(ge$play_id[z])){
    if(gi$play_per_game[n] == ge$play_id[z]){
     if(gi$second_baserunner[n+1] != 0){#Only print if there is a base runner on second
      print("The runner was safe at second.")
      safe_second = 1
      if(throw_distance_2B >= 150 && throw_distance_2B <= 199.99){
          safe_second_cu_150 = 1
          throws_second_150 = 1
         }
         if(throw_distance_2B >= 200 && throw_distance_2B <= 249.99){
          safe_second_cu_200 = 1
          throws_second_200 = 1
         }
         if(throw_distance_2B >= 250 && throw_distance_2B <= 299.99){
          safe_second_cu_250 = 1
          throws_second_250 = 1
         }
         if(throw_distance_2B >= 300 && throw_distance_2B <= 349.99){
          safe_second_cu_300 = 1
          throws_second_300 = 1
        }
        if(throw_distance_2B >= 350 && throw_distance_2B <= 399.99){
          safe_second_cu_350 = 1
          throws_second_350 = 1
        }
        if(throw_distance_2B >= 400){
          safe_second_cu_400 = 1
          throws_second_400 = 1
        }
  break
}
else{
  print("The runner was out at second")
  outs_second_cu = 1
  total_time_second = diff
  total_cutoff_time = cutoff_diff
  if(throw_distance_2B >= 150 && throw_distance_2B <= 199.99){
    outs_second_cu_150 = 1
    total_time_second_150 = total_time_second_150 + diff
    throws_second_150 = 1
  }
```

```
if(throw_distance_2B >= 200 && throw_distance_2B <= 249.99){
  outs_second_cu_200 = 1
  total_time_second_200 = total_time_second_200 + diff
  throws_second_200 = 1
}
if(throw_distance_2B >= 250 && throw_distance_2B <= 299.99){
  outs_second_cu_250 = 1
  total_time_second_250 = total_time_second_250 + diff
  throws_second_250 = 1
}
if(throw_distance_2B >= 300 && throw_distance_2B <= 349.99){
  outs_second_cu_300 = 1
  total_time_second_300 = total_time_second_300 + diff
  throws_second_300 = 1
}
if(throw_distance_2B >= 350 && throw_distance_2B <= 399.99){
  outs_second_cu_350 =  1
  total_time_second_350 = total_time_second_350 + diff
  throws_second_350 = 1
}
if(throw_distance_2B >= 400){
  outs_second_cu_400 = 1
  total_time_second_400 = total_time_second_400 + diff
```

```
      throws_second_400 = 1
    }
    if(changed == 0){ #Measures accuracy if I didn't put the x or y position to an
estimated spot of where it may be
      accuracy2B = sqrt((secondX)^2 + (secondY - 0.7)^2) #Checks how far off throw is
from second
      total_accuracy2B = total_accuracy2B + accuracy2B #Gets how far off every
throw to second is
      print(paste("Throw was", round(accuracy2B, 2), "feet off from second."))
      if(throw_distance_2B >= 150 && throw_distance_2B <= 199.99){
        accuracy2B_150 = accuracy2B #Checks how far off throw is from 2B
        total_accuracy2B_150 = total_accuracy2B_150 + accuracy2B_150 #Gets how far
off every throw to second is
      }
      if(throw_distance_2B >= 200 && throw_distance_2B <= 249.99){
        accuracy2B_200 = accuracy2B
        total_accuracy2B_200 = total_accuracy2B_200 + accuracy2B_200
      }
      if(throw_distance_2B >= 250 && throw_distance_2B <= 299.99){
        accuracy2B_250 = accuracy2B
        total_accuracy2B_250 = total_accuracy2B_250 + accuracy2B_250
      }
      if(throw_distance_2B >= 300 && throw_distance_2B <= 349.99){
         accuracy2B_300 = accuracy2B

         total_accuracy2B_300 = total_accuracy2B_300 + accuracy2B_300

        }

        if(throw_distance_2B >= 350 && throw_distance_2B <= 399.99){

         accuracy2B_350 = accuracy2B

         total_accuracy2B_350 = total_accuracy2B_350 + accuracy2B_350

        }

        if(throw_distance_2B >= 400){

         accuracy2B_400 = accuracy2B

         total_accuracy2B_400 = total_accuracy2B_400 + accuracy2B_400

        }

        break

       }

      }

     }

    }

   }



   if(check_second==1){

    if(finalTime != 0){ #Only print if have final time

     print(paste("Took", round(diff, 2), "seconds to get the ball from the outfielder to

second.",
          "Took", round(cutoff_diff, 2), "seconds for cutoff man to throw ball to
second."))
    }
    print(paste("Throw distance from outfielder to second was",
round(throw_distance_2B, 2), "feet to second.",
          "Throw distance from cutoff man to second was", round(cutoff_throw_dist, 2),
"feet to second"))
    cat("\n")
   }
  }
  return(c(check_second, outs_second_cu, safe_second, total_time_second,
total_cutoff_time, total_accuracy2B,
       outs_second_cu_150, outs_second_cu_200, outs_second_cu_250,
outs_second_cu_300, outs_second_cu_350,
       outs_second_cu_400, total_time_second_150, total_time_second_200,
total_time_second_250,
       total_time_second_300, total_time_second_350, total_time_second_400,
total_accuracy2B_150,
       total_accuracy2B_200, total_accuracy2B_250, total_accuracy2B_300,
total_accuracy2B_350,
       total_accuracy2B_400, throws_second_150, throws_second_200,
throws_second_250, throws_second_300,
       throws_second_350, throws_second_400))
```

```
 }

 cutoff_pos_third <- function(z, x1, x2, y1, y2, position, ofX, thirdX, ofY, thirdY, finalTime,
beginningTime, e, checkO){
   throw_distance_3B = 0
   check_third = 0
   safe_third = 0
   outs_third = 0
   total_time_third = 0
   diff = 0
   cutoff_diff = 0
   cutoff_throw_dist = 0 #Cutoff throw distance
   total_cutoff_time = 0
   changed = 0
   total_accuracy3B = 0
   outs_third_cu = 0
   outs_third_cu_150 = 0
   outs_third_cu_200 = 0
   outs_third_cu_250 = 0
   outs_third_cu_300 = 0
   outs_third_cu_350 = 0
   outs_third_cu_400 = 0
   total_time_third = 0
```

```
total_time_third_150 = 0
total_time_third_200 = 0
total_time_third_250 = 0
total_time_third_300 = 0
total_time_third_350 = 0
total_time_third_400 = 0
total_accuracy3B_150 = 0
total_accuracy3B_200 = 0
total_accuracy3B_250 = 0
total_accuracy3B_300 = 0
total_accuracy3B_350 = 0
total_accuracy3B_400 = 0
throws_third_150 = 0
throws_third_200 = 0
throws_third_250 = 0
throws_third_300 = 0
throws_third_350 = 0
throws_third_400 = 0

cutoff_time = ge$timestamp[z] #time when cutoff man got the ball
cutoff_x = get_x_y_i(cutoff_time, position)[2] #gets exact x position of cutoff man
cutoff_y = get_x_y_i(cutoff_time, position)[3] #gets exact y position of cutoff man
```

```
    if(cutoff_x < x1 &&
       cutoff_x > x2 && cutoff_y < y1 &&
       cutoff_y > y2 && checkO == 1){ #Checks that cutoff man is around the cutoff position
      check_third = 1
      if(thirdX == -1000){
        thirdX = -63.7  #sets third baseman's x value to x value of third base if didn't get exact x value of third baseman
        changed = 1
      }
      if(thirdY == -1000){
        thirdY = 63.7 #sets third baseman's y value to y value of third base if didn't get exact y value of third baseman
        changed = 1
      }
      throw_distance_3B = sqrt((ofX - thirdX)^2 + (ofY - thirdY)^2)
      diff = (finalTime - beginningTime)/1000 #time to throw ball to second from outfield with cutoff
      cutoff_diff = (finalTime - cutoff_time)/1000
      cutoff_throw_dist = sqrt((cutoff_x - thirdX)^2 + (cutoff_y - thirdY)^2)
      for(n in 1:(e-1)){
        if(!is.na(gi$play_per_game[n]) && !is.na(ge$play_id[z])){
          if(gi$play_per_game[n] == ge$play_id[z]){
            if(gi$third_baserunner[n+1] != 0){ #Only print if there is a base runner on third
```

```
print("The runner was safe at third")

safe_third = 1

if(throw_distance_3B >= 150 && throw_distance_3B <= 199.99){

     throws_third_150 = 1

     safe_third_150 = 1

    }

    if(throw_distance_3B >= 200 && throw_distance_3B <= 249.99){

     throws_third_200 = 1

     safe_third_200 = 1

    }

    if(throw_distance_3B >= 250 && throw_distance_3B <= 299.99){

     throws_third_250 = 1

     safe_third_250 = 1

    }

    if(throw_distance_3B >= 300 && throw_distance_3B <= 349.99){

     throws_third_300 = 1

     safe_third_300 = 1

    }

    if(throw_distance_3B >= 350 && throw_distance_3B <= 399.99){

     throws_third_350 = 1

     safe_third_350 = 1

    }

    if(throw_distance_3B >= 400){
          throws_third_400 = 1
          safe_third_400 = 1
        }
  break
}
else{
  print("The runner was out at third")
  outs_third = 1
  total_time_third = diff
  total_cutoff_time = cutoff_diff
  if(throw_distance_3B >= 150 && throw_distance_3B <= 199.99){
    outs_third_cu_150 = 1
    throws_third_150 = 1
    total_time_third_150 = total_time_third_150 + diff
  }
  if(throw_distance_3B >= 200 && throw_distance_3B <= 249.99){
    outs_third_cu_200 = 1
    throws_third_200 = 1
    total_time_third_200 = total_time_third_200 + diff
  }
  if(throw_distance_3B >= 250 && throw_distance_3B <= 299.99){
    outs_third_cu_250 = 1
    throws_third_250 = 1
```

```
            total_time_third_250 = total_time_third_250 + diff
          }
          if(throw_distance_3B >= 300 && throw_distance_3B <= 349.99){
            outs_third_cu_300 = 1
            throws_third_300 = 1
            total_time_third_300 = total_time_third_300 + diff
          }
          if(throw_distance_3B >= 350 && throw_distance_3B <= 399.99){
            outs_third_cu_350 = 1
            throws_third_350 = 1
            total_time_third_350 = total_time_third_350 + diff
          }
          if(throw_distance_3B >= 400){
            outs_third_cu_400 = 1
            throws_third_400 = 1
            total_time_third_400 = total_time_third_400 + diff
          }
          if(changed == 0){
            accuracy3B = sqrt((third_x + 63.7)^2 + (third_y - 63.7)^2) #Checks how far off
throw is from third
            total_accuracy3B = total_accuracy3B + accuracy3B #Gets how far off every
throw to third is
            print(paste("Throw was", round(accuracy3B, 2), "feet off from third."))
```

```
if(throw_distance_3B >= 150 && throw_distance_3B <= 199.99){
  accuracy3B_150 = accuracy3B #Checks how far off throw is from 3B
  total_accuracy3B_150 = total_accuracy3B_150 + accuracy3B_150 #Gets how far off every throw to third is
}
if(throw_distance_3B >= 200 && throw_distance_3B <= 249.99){
  accuracy3B_200 = accuracy3B
  total_accuracy3B_200 = total_accuracy3B_200 + accuracy3B_200
}
if(throw_distance_3B >= 250 && throw_distance_3B <= 299.99){
  accuracy3B_250 = accuracy3B
  total_accuracy3B_250 = total_accuracy3B_250 + accuracy3B_250
}
if(throw_distance_3B >= 300 && throw_distance_3B <= 349.99){
  accuracy3B_300 = accuracy3B
  total_accuracy3B_300 = total_accuracy3B_300 + accuracy3B_300
}
if(throw_distance_3B >= 350 && throw_distance_3B <= 399.99){
  accuracy3B_350 = accuracy3B
  total_accuracy3B_350 = total_accuracy3B_350 + accuracy3B_350
}
if(throw_distance_3B >= 400){
  accuracy3B_400 = accuracy3B
```

```
         total_accuracy3B_400 = total_accuracy3B_400 + accuracy3B_400
        }
        break
       }
      }
     }
    }
   }

   if(check_third==1){
    if(finalTime != 0){ #Only print if have final time
     print(paste("Took", round(diff, 2), "seconds to get the ball from the outfielder to
third.",
           "Took", round(cutoff_diff, 2), "seconds for cutoff man to throw ball to
third."))
    }
    print(paste("Throw distance from outfielder to third was", round(throw_distance_3B,
2), "feet to third.",
          "Throw distance from cutoff man to third was", round(cutoff_throw_dist, 2),
"feet to third."))
    cat("\n")
   }
  }
```

```
    return(c(check_third, outs_third, safe_third, total_time_third, total_cutoff_time,
total_accuracy3B,
             outs_third_cu_150, outs_third_cu_200, outs_third_cu_250, outs_third_cu_300,
             outs_third_cu_350, outs_third_cu_400, total_time_third_150, total_time_third_200,
             total_time_third_250, total_time_third_300, total_time_third_350,
total_time_third_400,
             total_accuracy3B_150, total_accuracy3B_200, total_accuracy3B_250,
total_accuracy3B_300,
             total_accuracy3B_350, total_accuracy3B_400, throws_third_150,
throws_third_200, throws_third_250, throws_third_300,
             throws_third_350, throws_third_400))
  }

  while(a<(end-2)){ #Start at index of 2 because play won't begin with a throw by an
outfielder.
    while(ge$event_code[a] != 5 && a <= (end-2)){ #runs until play ends
      if(ge$event_code[a]==2 && (ge$player_position[a]==7 || ge$player_position[a]==8 ||
                           ge$player_position[a]==9) && ge$event_code[a+1]==3){ #checks if ball
was acquired by an outfielder then thrown
        checkOF = 1
        j=a+2 #set to after throw is made by outfielder
        beginning_time = ge$timestamp[a] #gets time when ball is caught by outfielder
    for(b in 1:end2){
     if(pl_pos2$timestamp[b] == ge$timestamp[a+1] && (pl_pos2$player_position[b] ==
7 ||
                              pl_pos2$player_position[b] == 8 ||
                              pl_pos2$player_position[b] == 9)){
      of_x = pl_pos2$field_x[b] #gets x distance from plate when ball is thrown by
outfielder
      of_y = pl_pos2$field_y[b] #gets y distance from plate when ball is thrown by
outfielder
     }
    }

    while(ge$play_id[j] == ge$play_id[j-1] && ge$event_code[j-1] != 2){ #runs until play
ends or right after ball is caught

     #First Baseman Cutting Off:
     if(ge$event_code[j] == 2 && ge$player_position[j] == 4 && ge$event_code[j+1] == 3
&& ge$player_position[j+1]==3){ #Checks that first baseman caught ball then threw it
      k = j+2
      while(ge$play_id[k] == ge$play_id[k-1] && ge$event_code[k-1] != 2){ #runs until
play ends or right after ball is caught
       if(ge$event_code[k] == 2 && ge$player_position[k] == 2){ #Checks that catcher
caught the ball
```

```
        final_time_home = ge$timestamp[k] #Gets time ball is caught by catcher baseman
        catcher_x = get_x_y_i(final_time_home, 2)[2]
        catcher_y = get_x_y_i(final_time_home, 2)[3]
        break
      }

      if(ge$event_code[k] == 2 && ge$player_position[k] == 5){ #Checks that third
baseman caught the ball
        final_time_third = ge$timestamp[k] #Gets time ball is caught by third baseman
        third_x = get_x_y_i(final_time_third, 5)[2]
        third_y = get_x_y_i(final_time_third, 5)[3]
        break
      }
      k = k+1
    }


    #1B cutoff of throw to third from hit to right field: (26.9, 145.6)
    if(ge$event_code[k] == 2 && ge$player_position[k] == 5){
      list_1B_cu = cutoff_pos_third(j, 41.9, 11.9, 160.6, 130.6, 3, of_x, third_x, of_y,
                               third_y, final_time_third, beginning_time, end3, checkOF)
      if(list_1B_cu[1]==1){
        check_third = 1
```

```
out_third_cu = out_third_cu + list_1B_cu[2]

safe_third_cu = safe_third_cu + list_1B_cu[3]

total_time_third_cu = total_time_third_cu + list_1B_cu[4]

cutoff_time3B = cutoff_time3B + list_1B_cu[5]

total_accuracy3B_cu = total_accuracy3B_cu + list_1B_cu[6]

out_third_cu_150 = out_third_cu_150 + list_1B_cu[7]

out_third_cu_200 = out_third_cu_200 + list_1B_cu[8]

out_third_cu_250 = out_third_cu_250 + list_1B_cu[9]

out_third_cu_300 = out_third_cu_300 + list_1B_cu[10]

out_third_cu_350 = out_third_cu_350 + list_1B_cu[11]

out_third_cu_400 = out_third_cu_400 + list_1B_cu[12]

total_time_third_cu_150 = total_time_third_cu_150 + list_1B_cu[13]

total_time_third_cu_200 = total_time_third_cu_200 + list_1B_cu[14]

total_time_third_cu_250 = total_time_third_cu_250 + list_1B_cu[15]

total_time_third_cu_300 = total_time_third_cu_300 + list_1B_cu[16]

total_time_third_cu_350 = total_time_third_cu_350 + list_1B_cu[17]

total_time_third_cu_400 = total_time_third_cu_400 + list_1B_cu[18]

total_accuracy3B_cu_150 = total_accuracy3B_cu_150 + list_1B_cu[19]

total_accuracy3B_cu_200 = total_accuracy3B_cu_200 + list_1B_cu[20]

total_accuracy3B_cu_250 = total_accuracy3B_cu_250 + list_1B_cu[21]

total_accuracy3B_cu_300 = total_accuracy3B_cu_300 + list_1B_cu[22]

total_accuracy3B_cu_350 = total_accuracy3B_cu_350 + list_1B_cu[23]

total_accuracy3B_cu_400 = total_accuracy3B_cu_400 + list_1B_cu[24]
```

```
        throws_third_cu_150 = throws_third_cu_150 + list_1B_cu[25]
        throws_third_cu_200 = throws_third_cu_200 + list_1B_cu[26]
        throws_third_cu_250 = throws_third_cu_250 + list_1B_cu[27]
        throws_third_cu_300 = throws_third_cu_300 + list_1B_cu[28]
        throws_third_cu_350 = throws_third_cu_350 + list_1B_cu[29]
        throws_third_cu_400 = throws_third_cu_400 + list_1B_cu[30]
        throws_caught_cu = throws_caught_cu %>% add_row(Third_X = third_x,
Third_Y = third_y)
        break
       }
      }

      #1B Cutoff of throw to home from single to right field: (26.7, 53.7)
      if(ge$event_code[k] == 2 && ge$player_position[k] == 2){
       list_1B_cu = cutoff_pos_home(j, 11.7, 41.7, 68.7, 38.7, 3, of_x, catcher_x, of_y,
                     catcher_y, final_time_home, beginning_time, end3, checkOF)
       if(list_1B_cu[1]==1){
        check_home = 1
        out_home_cu = out_home_cu + list_1B_cu[2]
        safe_home_cu = safe_home_cu + list_1B_cu[3]
        total_time_home_cu = total_time_home_cu + list_1B_cu[4]
        cutoff_timeC = cutoff_timeC + list_1B_cu[5]
        total_accuracyC_cu = total_accuracyC_cu + list_1B_cu[6]
```

```
out_home_cu_150 = out_home_cu_150 + list_1B_cu[7]

out_home_cu_200 = out_home_cu_200 + list_1B_cu[8]

out_home_cu_250 = out_home_cu_250 + list_1B_cu[9]

out_home_cu_300 = out_home_cu_300 + list_1B_cu[10]

out_home_cu_350 = out_home_cu_350 + list_1B_cu[11]

out_home_cu_400 = out_home_cu_400 + list_1B_cu[12]

total_time_home_cu_150 = total_time_home_cu_150 + list_1B_cu[13]

total_time_home_cu_200 = total_time_home_cu_200 + list_1B_cu[14]

total_time_home_cu_250 = total_time_home_cu_250 + list_1B_cu[15]

total_time_home_cu_300 = total_time_home_cu_300 + list_1B_cu[16]

total_time_home_cu_350 = total_time_home_cu_350 + list_1B_cu[17]

total_time_home_cu_400 = total_time_home_cu_400 + list_1B_cu[18]

total_accuracyC_cu_150 = total_accuracyC_cu_150 + list_1B_cu[19]

total_accuracyC_cu_200 = total_accuracyC_cu_200 + list_1B_cu[20]

total_accuracyC_cu_250 = total_accuracyC_cu_250 + list_1B_cu[21]

total_accuracyC_cu_300 = total_accuracyC_cu_300 + list_1B_cu[22]

total_accuracyC_cu_350 = total_accuracyC_cu_350 + list_1B_cu[23]

total_accuracyC_cu_400 = total_accuracyC_cu_400 + list_1B_cu[24]

throws_home_cu_150 = throws_home_cu_150 + list_1B_cu[25]

throws_home_cu_200 = throws_home_cu_200 + list_1B_cu[26]

throws_home_cu_250 = throws_home_cu_250 + list_1B_cu[27]

throws_home_cu_300 = throws_home_cu_300 + list_1B_cu[28]

throws_home_cu_350 = throws_home_cu_350 + list_1B_cu[29]
```

```
          throws_home_cu_400 = throws_home_cu_400 + list_1B_cu[30]
          throws_caught_cu = throws_caught_cu %>% add_row(Catcher_X = catcher_x,
Catcher_Y = catcher_y)
          break
        }
        #1B Cutoff of throw to home from single to center field: (0, 75.6)
        list_1B_cu = cutoff_pos_home(j, -15, 15, 60.6, 90.6, 3, of_x, catcher_x, of_y,
                                     catcher_y, final_time_home, beginning_time, end3, checkOF)
        if(list_1B_cu[1]==1){
          check_home = 1
          out_home_cu = out_home_cu + list_1B_cu[2]
          safe_home_cu = safe_home_cu + list_1B_cu[3]
          total_time_home_cu = total_time_home_cu + list_1B_cu[4]
          cutoff_timeC = cutoff_timeC + list_1B_cu[5]
          total_accuracyC_cu = total_accuracyC_cu + list_1B_cu[6]
          out_home_cu_150 = out_home_cu_150 + list_1B_cu[7]
          out_home_cu_200 = out_home_cu_200 + list_1B_cu[8]
          out_home_cu_250 = out_home_cu_250 + list_1B_cu[9]
          out_home_cu_300 = out_home_cu_300 + list_1B_cu[10]
          out_home_cu_350 = out_home_cu_350 + list_1B_cu[11]
          out_home_cu_400 = out_home_cu_400 + list_1B_cu[12]
          total_time_home_cu_150 = total_time_home_cu_150 + list_1B_cu[13]
          total_time_home_cu_200 = total_time_home_cu_200 + list_1B_cu[14]
          total_time_home_cu_250 = total_time_home_cu_250 + list_1B_cu[15]
          total_time_home_cu_300 = total_time_home_cu_300 + list_1B_cu[16]
          total_time_home_cu_350 = total_time_home_cu_350 + list_1B_cu[17]
          total_time_home_cu_400 = total_time_home_cu_400 + list_1B_cu[18]
          total_accuracyC_cu_150 = total_accuracyC_cu_150 + list_1B_cu[19]
          total_accuracyC_cu_200 = total_accuracyC_cu_200 + list_1B_cu[20]
          total_accuracyC_cu_250 = total_accuracyC_cu_250 + list_1B_cu[21]
          total_accuracyC_cu_300 = total_accuracyC_cu_300 + list_1B_cu[22]
          total_accuracyC_cu_350 = total_accuracyC_cu_350 + list_1B_cu[23]
          total_accuracyC_cu_400 = total_accuracyC_cu_400 + list_1B_cu[24]
          throws_home_cu_150 = throws_home_cu_150 + list_1B_cu[25]
          throws_home_cu_200 = throws_home_cu_200 + list_1B_cu[26]
          throws_home_cu_250 = throws_home_cu_250 + list_1B_cu[27]
          throws_home_cu_300 = throws_home_cu_300 + list_1B_cu[28]
          throws_home_cu_350 = throws_home_cu_350 + list_1B_cu[29]
          throws_home_cu_400 = throws_home_cu_400 + list_1B_cu[30]
          throws_caught_cu = throws_caught_cu %>% add_row(Catcher_X = catcher_x,
Catcher_Y = catcher_y)
          break
        }
       }
      }
```

```
    #Second Baseman Cutting Off:
    if(ge$event_code[j] == 2 && ge$player_position[j] == 4 && ge$event_code[j+1] == 3
&& ge$player_position[j+1]==4){ #Checks that second baseman caught ball then threw it
     k = j+2
     while(ge$play_id[k] == ge$play_id[k-1] && ge$event_code[k-1] != 2){ #runs until
play ends or right after ball is caught
      if(ge$event_code[k] == 2 && ge$player_position[k] == 2){ #Checks that catcher
caught the ball
       final_time_home = ge$timestamp[k] #Gets time ball is caught by catcher baseman
       catcher_x = get_x_y_i(final_time_home, 2)[2]
       catcher_y = get_x_y_i(final_time_home, 2)[3]
       break
      }

      if(ge$event_code[k] == 2 && ge$player_position[k] == 6){ #Checks that shortstop
caught the ball since he's covering second on these plays
       final_time_second = ge$timestamp[k] #Gets time ball is caught by shortstop
       second_x = get_x_y_i(final_time_second, 6)[2]
       second_y = get_x_y_i(final_time_second, 6)[3]
       break
      }
      if(ge$event_code[k] == 2 && ge$player_position[k] == 5){ #Checks that third
baseman caught the ball
        final_time_third = ge$timestamp[k] #Gets time ball is caught by third baseman
        third_x = get_x_y_i(final_time_third, 5)[2]
        third_y = get_x_y_i(final_time_third, 5)[3]
        break
      }
      k = k+1
    }

    #2B cutoff of throw to home from double to LF: (-107.4, 122.3)
    if(ge$event_code[k] == 2 && ge$player_position[k] == 2){
      list_2B_cu = cutoff_pos_home(j, -97.4, -127.4, 137.3, 107.3, 4, of_x, catcher_x, of_y,
                             catcher_y, final_time_home, beginning_time, end3, checkOF)
      if(list_2B_cu[1]==1){
        check_home = 1
        out_home_cu = out_home_cu + list_2B_cu[2]
        safe_home_cu = safe_home_cu + list_2B_cu[3]
        total_time_home_cu = total_time_home_cu + list_2B_cu[4]
        cutoff_timeC = cutoff_timeC + list_2B_cu[5]
        total_accuracyC_cu = total_accuracyC_cu + list_2B_cu[6]
        out_home_cu_150 = out_home_cu_150 + list_2B_cu[7]
        out_home_cu_200 = out_home_cu_200 + list_2B_cu[8]
```

```
out_home_cu_250 = out_home_cu_250 + list_2B_cu[9]

out_home_cu_300 = out_home_cu_300 + list_2B_cu[10]

out_home_cu_350 = out_home_cu_350 + list_2B_cu[11]

out_home_cu_400 = out_home_cu_400 + list_2B_cu[12]

total_time_home_cu_150 = total_time_home_cu_150 + list_2B_cu[13]

total_time_home_cu_200 = total_time_home_cu_200 + list_2B_cu[14]

total_time_home_cu_250 = total_time_home_cu_250 + list_2B_cu[15]

total_time_home_cu_300 = total_time_home_cu_300 + list_2B_cu[16]

total_time_home_cu_350 = total_time_home_cu_350 + list_2B_cu[17]

total_time_home_cu_400 = total_time_home_cu_400 + list_2B_cu[18]

total_accuracyC_cu_150 = total_accuracyC_cu_150 + list_2B_cu[19]

total_accuracyC_cu_200 = total_accuracyC_cu_200 + list_2B_cu[20]

total_accuracyC_cu_250 = total_accuracyC_cu_250 + list_2B_cu[21]

total_accuracyC_cu_300 = total_accuracyC_cu_300 + list_2B_cu[22]

total_accuracyC_cu_350 = total_accuracyC_cu_350 + list_2B_cu[23]

total_accuracyC_cu_400 = total_accuracyC_cu_400 + list_2B_cu[24]

throws_home_cu_150 = throws_home_cu_150 + list_2B_cu[25]

throws_home_cu_200 = throws_home_cu_200 + list_2B_cu[26]

throws_home_cu_250 = throws_home_cu_250 + list_2B_cu[27]

throws_home_cu_300 = throws_home_cu_300 + list_2B_cu[28]

throws_home_cu_350 = throws_home_cu_350 + list_2B_cu[29]

throws_home_cu_400 = throws_home_cu_400 + list_2B_cu[30]
```

```
      throws_caught_cu = throws_caught_cu %>% add_row(Catcher_X = catcher_x,
Catcher_Y = catcher_y)
      break
    }

    #2B cutoff of throw to home from double to left center: (-57.3, 170.6)
    list_2B_cu = cutoff_pos_home(j, -42.3, -72.3, 185.6, 155.6, 4, of_x, catcher_x, of_y,
                                 catcher_y, final_time_home, beginning_time, end3, checkOF)
    if(list_2B_cu[1]==1){
      check_home = 1
      out_home_cu = out_home_cu + list_2B_cu[2]
      safe_home_cu = safe_home_cu + list_2B_cu[3]
      total_time_home_cu = total_time_home_cu + list_2B_cu[4]
      cutoff_timeC = cutoff_timeC + list_2B_cu[5]
      total_accuracyC_cu = total_accuracyC_cu + list_2B_cu[6]
      out_home_cu_150 = out_home_cu_150 + list_2B_cu[7]
      out_home_cu_200 = out_home_cu_200 + list_2B_cu[8]
      out_home_cu_250 = out_home_cu_250 + list_2B_cu[9]
      out_home_cu_300 = out_home_cu_300 + list_2B_cu[10]
      out_home_cu_350 = out_home_cu_350 + list_2B_cu[11]
      out_home_cu_400 = out_home_cu_400 + list_2B_cu[12]
      total_time_home_cu_150 = total_time_home_cu_150 + list_2B_cu[13]
      total_time_home_cu_200 = total_time_home_cu_200 + list_2B_cu[14]
```

```
        total_time_home_cu_250 = total_time_home_cu_250 + list_2B_cu[15]
        total_time_home_cu_300 = total_time_home_cu_300 + list_2B_cu[16]
        total_time_home_cu_350 = total_time_home_cu_350 + list_2B_cu[17]
        total_time_home_cu_400 = total_time_home_cu_400 + list_2B_cu[18]
        total_accuracyC_cu_150 = total_accuracyC_cu_150 + list_2B_cu[19]
        total_accuracyC_cu_200 = total_accuracyC_cu_200 + list_2B_cu[20]
        total_accuracyC_cu_250 = total_accuracyC_cu_250 + list_2B_cu[21]
        total_accuracyC_cu_300 = total_accuracyC_cu_300 + list_2B_cu[22]
        total_accuracyC_cu_350 = total_accuracyC_cu_350 + list_2B_cu[23]
        total_accuracyC_cu_400 = total_accuracyC_cu_400 + list_2B_cu[24]
        throws_home_cu_150 = throws_home_cu_150 + list_2B_cu[25]
        throws_home_cu_200 = throws_home_cu_200 + list_2B_cu[26]
        throws_home_cu_250 = throws_home_cu_250 + list_2B_cu[27]
        throws_home_cu_300 = throws_home_cu_300 + list_2B_cu[28]
        throws_home_cu_350 = throws_home_cu_350 + list_2B_cu[29]
        throws_home_cu_400 = throws_home_cu_400 + list_2B_cu[30]
        throws_caught_cu = throws_caught_cu %>% add_row(Catcher_X = catcher_x, Catcher_Y = catcher_y)
        break
    }


    #2B cutoff of throw to home from double to CF: (0, 180.6)
```

```
list_2B_cu = cutoff_pos_home(j, 20, -20, 200.6, 160.6, 4, of_x, catcher_x, of_y,
                              catcher_y, final_time_home, beginning_time, end3, checkOF)
if(list_2B_cu[1]==1){
  check_home = 1
  out_home_cu = out_home_cu + list_2B_cu[2]
  safe_home_cu = safe_home_cu + list_2B_cu[3]
  total_time_home_cu = total_time_home_cu + list_2B_cu[4]
  cutoff_timeC = cutoff_timeC + list_2B_cu[5]
  total_accuracyC_cu = total_accuracyC_cu + list_2B_cu[6]
  out_home_cu_150 = out_home_cu_150 + list_2B_cu[7]
  out_home_cu_200 = out_home_cu_200 + list_2B_cu[8]
  out_home_cu_250 = out_home_cu_250 + list_2B_cu[9]
  out_home_cu_300 = out_home_cu_300 + list_2B_cu[10]
  out_home_cu_350 = out_home_cu_350 + list_2B_cu[11]
  out_home_cu_400 = out_home_cu_400 + list_2B_cu[12]
  total_time_home_cu_150 = total_time_home_cu_150 + list_2B_cu[13]
  total_time_home_cu_200 = total_time_home_cu_200 + list_2B_cu[14]
  total_time_home_cu_250 = total_time_home_cu_250 + list_2B_cu[15]
  total_time_home_cu_300 = total_time_home_cu_300 + list_2B_cu[16]
  total_time_home_cu_350 = total_time_home_cu_350 + list_2B_cu[17]
  total_time_home_cu_400 = total_time_home_cu_400 + list_2B_cu[18]
  total_accuracyC_cu_150 = total_accuracyC_cu_150 + list_2B_cu[19]
  total_accuracyC_cu_200 = total_accuracyC_cu_200 + list_2B_cu[20]
```

```
    total_accuracyC_cu_250 = total_accuracyC_cu_250 + list_2B_cu[21]
    total_accuracyC_cu_300 = total_accuracyC_cu_300 + list_2B_cu[22]
    total_accuracyC_cu_350 = total_accuracyC_cu_350 + list_2B_cu[23]
    total_accuracyC_cu_400 = total_accuracyC_cu_400 + list_2B_cu[24]
    throws_home_cu_150 = throws_home_cu_150 + list_2B_cu[25]
    throws_home_cu_200 = throws_home_cu_200 + list_2B_cu[26]
    throws_home_cu_250 = throws_home_cu_250 + list_2B_cu[27]
    throws_home_cu_300 = throws_home_cu_300 + list_2B_cu[28]
    throws_home_cu_350 = throws_home_cu_350 + list_2B_cu[29]
    throws_home_cu_400 = throws_home_cu_400 + list_2B_cu[30]
    throws_caught_cu = throws_caught_cu %>% add_row(Catcher_X = catcher_x, Catcher_Y = catcher_y)
    break
  }

  #2B cutoff of throw to home from double to right center: (57.3, 170.6)
  list_2B_cu = cutoff_pos_home(j, 72.3, 42.3, 185.6, 155.6, 4, of_x, catcher_x, of_y,
                               catcher_y, final_time_home, beginning_time, end3, checkOF)
  if(list_2B_cu[1]==1){
    check_home = 1
    out_home_cu = out_home_cu + list_2B_cu[2]
    safe_home_cu = safe_home_cu + list_2B_cu[3]
    total_time_home_cu = total_time_home_cu + list_2B_cu[4]
```

```
cutoff_timeC = cutoff_timeC + list_2B_cu[5]
total_accuracyC_cu = total_accuracyC_cu + list_2B_cu[6]
out_home_cu_150 = out_home_cu_150 + list_2B_cu[7]
out_home_cu_200 = out_home_cu_200 + list_2B_cu[8]
out_home_cu_250 = out_home_cu_250 + list_2B_cu[9]
out_home_cu_300 = out_home_cu_300 + list_2B_cu[10]
out_home_cu_350 = out_home_cu_350 + list_2B_cu[11]
out_home_cu_400 = out_home_cu_400 + list_2B_cu[12]
total_time_home_cu_150 = total_time_home_cu_150 + list_2B_cu[13]
total_time_home_cu_200 = total_time_home_cu_200 + list_2B_cu[14]
total_time_home_cu_250 = total_time_home_cu_250 + list_2B_cu[15]
total_time_home_cu_300 = total_time_home_cu_300 + list_2B_cu[16]
total_time_home_cu_350 = total_time_home_cu_350 + list_2B_cu[17]
total_time_home_cu_400 = total_time_home_cu_400 + list_2B_cu[18]
total_accuracyC_cu_150 = total_accuracyC_cu_150 + list_2B_cu[19]
total_accuracyC_cu_200 = total_accuracyC_cu_200 + list_2B_cu[20]
total_accuracyC_cu_250 = total_accuracyC_cu_250 + list_2B_cu[21]
total_accuracyC_cu_300 = total_accuracyC_cu_300 + list_2B_cu[22]
total_accuracyC_cu_350 = total_accuracyC_cu_350 + list_2B_cu[23]
total_accuracyC_cu_400 = total_accuracyC_cu_400 + list_2B_cu[24]
throws_home_cu_150 = throws_home_cu_150 + list_2B_cu[25]
throws_home_cu_200 = throws_home_cu_200 + list_2B_cu[26]
throws_home_cu_250 = throws_home_cu_250 + list_2B_cu[27]
```

```
      throws_home_cu_300 = throws_home_cu_300 + list_2B_cu[28]
      throws_home_cu_350 = throws_home_cu_350 + list_2B_cu[29]
      throws_home_cu_400 = throws_home_cu_400 + list_2B_cu[30]
      throws_second_cu_150 = throws_second_cu_150 + list_2B_cu[25]
      throws_second_cu_200 = throws_second_cu_200 + list_2B_cu[26]
      throws_second_cu_250 = throws_second_cu_250 + list_2B_cu[27]
      throws_second_cu_300 = throws_second_cu_300 + list_2B_cu[28]
      throws_second_cu_350 = throws_second_cu_350 + list_2B_cu[29]
      throws_second_cu_400 = throws_second_cu_400 + list_2B_cu[30]
      throws_caught_cu = throws_caught_cu %>% add_row(Catcher_X = catcher_x,
Catcher_Y = catcher_y)
      break
     }
    }



    #2B Cutoff of throw to second from hit to right field: (36.9, 155.6) - x is 1/2 way
between 2nd and 3rd plus some feet
    if(ge$event_code[k] == 2 && ge$player_position[k] == 6){
     list_2B_cu = cutoff_pos_second(j, 51.9, 21.9, 170.6, 140.6, 4, of_x, second_x, of_y,
                    second_y, final_time_second, beginning_time, end3, checkOF)
     if(list_2B_cu[1]==1){
      check_second = 1
```

```
out_second_cu = out_second_cu + list_2B_cu[2]
safe_second_cu = safe_second_cu + list_2B_cu[3]
total_time_second_cu = total_time_second_cu + list_2B_cu[4]
cutoff_time2B = cutoff_time2B + list_2B_cu[5]
total_accuracy2B_cu = total_accuracy2B_cu + list_2B_cu[6]
out_second_cu_150 = out_second_cu_150 + list_2B_cu[7]
out_second_cu_200 = out_second_cu_200 + list_2B_cu[8]
out_second_cu_250 = out_second_cu_250 + list_2B_cu[9]
out_second_cu_300 = out_second_cu_300 + list_2B_cu[10]
out_second_cu_350 = out_second_cu_350 + list_2B_cu[11]
out_second_cu_400 = out_second_cu_400 + list_2B_cu[12]
total_time_second_cu_150 = total_time_second_cu_150 + list_2B_cu[13]
total_time_second_cu_200 = total_time_second_cu_200 + list_2B_cu[14]
total_time_second_cu_250 = total_time_second_cu_250 + list_2B_cu[15]
total_time_second_cu_300 = total_time_second_cu_300 + list_2B_cu[16]
total_time_second_cu_350 = total_time_second_cu_350 + list_2B_cu[17]
total_time_second_cu_400 = total_time_second_cu_400 + list_2B_cu[18]
total_accuracy2B_cu_150 = total_accuracy2B_cu_150 + list_2B_cu[19]
total_accuracy2B_cu_200 = total_accuracy2B_cu_200 + list_2B_cu[20]
total_accuracy2B_cu_250 = total_accuracy2B_cu_250 + list_2B_cu[21]
total_accuracy2B_cu_300 = total_accuracy2B_cu_300 + list_2B_cu[22]
total_accuracy2B_cu_350 = total_accuracy2B_cu_350 + list_2B_cu[23]
total_accuracy2B_cu_400 = total_accuracy2B_cu_400 + list_2B_cu[24]
```

```
      throws_second_cu_150 = throws_second_cu_150 + list_2B_cu[25]

      throws_second_cu_200 = throws_second_cu_200 + list_2B_cu[26]

      throws_second_cu_250 = throws_second_cu_250 + list_2B_cu[27]

      throws_second_cu_300 = throws_second_cu_300 + list_2B_cu[28]

      throws_second_cu_350 = throws_second_cu_350 + list_2B_cu[29]

      throws_second_cu_400 = throws_second_cu_400 + list_2B_cu[30]

      throws_caught_cu = throws_caught_cu %>% add_row(Second_X = second_x,
Second_Y = second_y)

      break

    }


    #2B cutoff of throw to second from hit to right center: (20, 175.6)

    list_2B_cu = cutoff_pos_second(j, 35, 5, 190.6, 150.6, 4, of_x, second_x, of_y,

                             second_y, final_time_second, beginning_time, end3, checkOF)

    if(list_2B_cu[1]==1){

      check_second = 1

      out_second_cu = out_second_cu + list_2B_cu[2]

      safe_second_cu = safe_second_cu + list_2B_cu[3]

      total_time_second_cu = total_time_second_cu + list_2B_cu[4]

      cutoff_time2B = cutoff_time2B + list_2B_cu[5]

      total_accuracy2B_cu = total_accuracy2B_cu + list_2B_cu[6]

      out_second_cu_150 = out_second_cu_150 + list_2B_cu[7]

      out_second_cu_200 = out_second_cu_200 + list_2B_cu[8]
```

```
out_second_cu_250 = out_second_cu_250 + list_2B_cu[9]

out_second_cu_300 = out_second_cu_300 + list_2B_cu[10]

out_second_cu_350 = out_second_cu_350 + list_2B_cu[11]

out_second_cu_400 = out_second_cu_400 + list_2B_cu[12]

total_time_second_cu_150 = total_time_second_cu_150 + list_2B_cu[13]

total_time_second_cu_200 = total_time_second_cu_200 + list_2B_cu[14]

total_time_second_cu_250 = total_time_second_cu_250 + list_2B_cu[15]

total_time_second_cu_300 = total_time_second_cu_300 + list_2B_cu[16]

total_time_second_cu_350 = total_time_second_cu_350 + list_2B_cu[17]

total_time_second_cu_400 = total_time_second_cu_400 + list_2B_cu[18]

total_accuracy2B_cu_150 = total_accuracy2B_cu_150 + list_2B_cu[19]

total_accuracy2B_cu_200 = total_accuracy2B_cu_200 + list_2B_cu[20]

total_accuracy2B_cu_250 = total_accuracy2B_cu_250 + list_2B_cu[21]

total_accuracy2B_cu_300 = total_accuracy2B_cu_300 + list_2B_cu[22]

total_accuracy2B_cu_350 = total_accuracy2B_cu_350 + list_2B_cu[23]

total_accuracy2B_cu_400 = total_accuracy2B_cu_400 + list_2B_cu[24]

throws_second_cu_150 = throws_second_cu_150 + list_2B_cu[25]

throws_second_cu_200 = throws_second_cu_200 + list_2B_cu[26]

throws_second_cu_250 = throws_second_cu_250 + list_2B_cu[27]

throws_second_cu_300 = throws_second_cu_300 + list_2B_cu[28]

throws_second_cu_350 = throws_second_cu_350 + list_2B_cu[29]

throws_second_cu_400 = throws_second_cu_400 + list_2B_cu[30]
```

```
  throws_caught_cu = throws_caught_cu %>% add_row(Second_X = second_x, Second_Y = second_y)
  break
}

#2B Cutoff of throw to second from hit to center field: (0, 175.6)
list_2B_cu = cutoff_pos_second(j, 15, -15, 190.6, 150.6, 4, of_x, second_x, of_y,
                         second_y, final_time_second, beginning_time, end3, checkOF)
if(list_2B_cu[1]==1){
  check_second = 1
  out_second_cu = out_second_cu + list_2B_cu[2]
  safe_second_cu = safe_second_cu + list_2B_cu[3]
  total_time_second_cu = total_time_second_cu + list_2B_cu[4]
  cutoff_time2B = cutoff_time2B + list_2B_cu[5]
  total_accuracy2B_cu = total_accuracy2B_cu + list_2B_cu[6]
  out_second_cu_150 = out_second_cu_150 + list_2B_cu[7]
  out_second_cu_200 = out_second_cu_200 + list_2B_cu[8]
  out_second_cu_250 = out_second_cu_250 + list_2B_cu[9]
  out_second_cu_300 = out_second_cu_300 + list_2B_cu[10]
  out_second_cu_350 = out_second_cu_350 + list_2B_cu[11]
  out_second_cu_400 = out_second_cu_400 + list_2B_cu[12]
  total_time_second_cu_150 = total_time_second_cu_150 + list_2B_cu[13]
  total_time_second_cu_200 = total_time_second_cu_200 + list_2B_cu[14]
```

```
      total_time_second_cu_250 = total_time_second_cu_250 + list_2B_cu[15]

      total_time_second_cu_300 = total_time_second_cu_300 + list_2B_cu[16]

      total_time_second_cu_350 = total_time_second_cu_350 + list_2B_cu[17]

      total_time_second_cu_400 = total_time_second_cu_400 + list_2B_cu[18]

      total_accuracy2B_cu_150 = total_accuracy2B_cu_150 + list_2B_cu[19]

      total_accuracy2B_cu_200 = total_accuracy2B_cu_200 + list_2B_cu[20]

      total_accuracy2B_cu_250 = total_accuracy2B_cu_250 + list_2B_cu[21]

      total_accuracy2B_cu_300 = total_accuracy2B_cu_300 + list_2B_cu[22]

      total_accuracy2B_cu_350 = total_accuracy2B_cu_350 + list_2B_cu[23]

      total_accuracy2B_cu_400 = total_accuracy2B_cu_400 + list_2B_cu[24]

      throws_second_cu_150 = throws_second_cu_150 + list_2B_cu[25]

      throws_second_cu_200 = throws_second_cu_200 + list_2B_cu[26]

      throws_second_cu_250 = throws_second_cu_250 + list_2B_cu[27]

      throws_second_cu_300 = throws_second_cu_300 + list_2B_cu[28]

      throws_second_cu_350 = throws_second_cu_350 + list_2B_cu[29]

      throws_second_cu_400 = throws_second_cu_400 + list_2B_cu[30]

      throws_caught_cu = throws_caught_cu %>% add_row(Second_X = second_x,

Second_Y = second_y)

      break

     }

    }


    #2B Cutoff of throw to third from hit to right field: (83.7, 158.6)
```

```
if(ge$event_code[k] == 2 && ge$player_position[k] == 5){
  list_2B_cu = cutoff_pos_third(j, 98.7, 68.7, 173.6, 143.6, 4, of_x, third_x, of_y,
                               third_y, final_time_third, beginning_time, end3, checkOF)
  if(list_2B_cu[1]==1){
    check_third = 1
    out_third_cu = out_third_cu + list_2B_cu[2]
    safe_third_cu = safe_third_cu + list_2B_cu[3]
    total_time_third_cu = total_time_third_cu + list_2B_cu[4]
    cutoff_time3B = cutoff_time3B + list_2B_cu[5]
    total_accuracy3B_cu = total_accuracy3B_cu + list_2B_cu[6]
    out_third_cu_150 = out_third_cu_150 + list_2B_cu[7]
    out_third_cu_200 = out_third_cu_200 + list_2B_cu[8]
    out_third_cu_250 = out_third_cu_250 + list_2B_cu[9]
    out_third_cu_300 = out_third_cu_300 + list_2B_cu[10]
    out_third_cu_350 = out_third_cu_350 + list_2B_cu[11]
    out_third_cu_400 = out_third_cu_400 + list_2B_cu[12]
    total_time_third_cu_150 = total_time_third_cu_150 + list_2B_cu[13]
    total_time_third_cu_200 = total_time_third_cu_200 + list_2B_cu[14]
    total_time_third_cu_250 = total_time_third_cu_250 + list_2B_cu[15]
    total_time_third_cu_300 = total_time_third_cu_300 + list_2B_cu[16]
    total_time_third_cu_350 = total_time_third_cu_350 + list_2B_cu[17]
    total_time_third_cu_400 = total_time_third_cu_400 + list_2B_cu[18]
    total_accuracy3B_cu_150 = total_accuracy3B_cu_150 + list_2B_cu[19]
```

```
      total_accuracy3B_cu_200 = total_accuracy3B_cu_200 + list_2B_cu[20]
      total_accuracy3B_cu_250 = total_accuracy3B_cu_250 + list_2B_cu[21]
      total_accuracy3B_cu_300 = total_accuracy3B_cu_300 + list_2B_cu[22]
      total_accuracy3B_cu_350 = total_accuracy3B_cu_350 + list_2B_cu[23]
      total_accuracy3B_cu_400 = total_accuracy3B_cu_400 + list_2B_cu[24]
      throws_third_cu_150 = throws_third_cu_150 + list_2B_cu[25]
      throws_third_cu_200 = throws_third_cu_200 + list_2B_cu[26]
      throws_third_cu_250 = throws_third_cu_250 + list_2B_cu[27]
      throws_third_cu_300 = throws_third_cu_300 + list_2B_cu[28]
      throws_third_cu_350 = throws_third_cu_350 + list_2B_cu[29]
      throws_third_cu_400 = throws_third_cu_400 + list_2B_cu[30]
      throws_caught_cu = throws_caught_cu %>% add_row(Third_X = third_x,
Third_Y = third_y)
      break
    }

    #2B cutoff of throw to third from hit to right center: (-5, 170.6)
    list_2B_cu = cutoff_pos_third(j, 10, -20, 185.6, 155.6, 4, of_x, third_x, of_y,
                          third_y, final_time_third, beginning_time, end3, checkOF)
    if(list_2B_cu[1]==1){
      check_third = 1
      out_third_cu = out_third_cu + list_2B_cu[2]
      safe_third_cu = safe_third_cu + list_2B_cu[3]
```

```
total_time_third_cu = total_time_third_cu + list_2B_cu[4]
cutoff_time3B = cutoff_time3B + list_2B_cu[5]
total_accuracy3B_cu = total_accuracy3B_cu + list_2B_cu[6]
out_third_cu_150 = out_third_cu_150 + list_2B_cu[7]
out_third_cu_200 = out_third_cu_200 + list_2B_cu[8]
out_third_cu_250 = out_third_cu_250 + list_2B_cu[9]
out_third_cu_300 = out_third_cu_300 + list_2B_cu[10]
out_third_cu_350 = out_third_cu_350 + list_2B_cu[11]
out_third_cu_400 = out_third_cu_400 + list_2B_cu[12]
total_time_third_cu_150 = total_time_third_cu_150 + list_2B_cu[13]
total_time_third_cu_200 = total_time_third_cu_200 + list_2B_cu[14]
total_time_third_cu_250 = total_time_third_cu_250 + list_2B_cu[15]
total_time_third_cu_300 = total_time_third_cu_300 + list_2B_cu[16]
total_time_third_cu_350 = total_time_third_cu_350 + list_2B_cu[17]
total_time_third_cu_400 = total_time_third_cu_400 + list_2B_cu[18]
total_accuracy3B_cu_150 = total_accuracy3B_cu_150 + list_2B_cu[19]
total_accuracy3B_cu_200 = total_accuracy3B_cu_200 + list_2B_cu[20]
total_accuracy3B_cu_250 = total_accuracy3B_cu_250 + list_2B_cu[21]
total_accuracy3B_cu_300 = total_accuracy3B_cu_300 + list_2B_cu[22]
total_accuracy3B_cu_350 = total_accuracy3B_cu_350 + list_2B_cu[23]
total_accuracy3B_cu_400 = total_accuracy3B_cu_400 + list_2B_cu[24]
throws_third_cu_150 = throws_third_cu_150 + list_2B_cu[25]
throws_third_cu_200 = throws_third_cu_200 + list_2B_cu[26]
```

```
      throws_third_cu_250 = throws_third_cu_250 + list_2B_cu[27]
      throws_third_cu_300 = throws_third_cu_300 + list_2B_cu[28]
      throws_third_cu_350 = throws_third_cu_350 + list_2B_cu[29]
      throws_third_cu_400 = throws_third_cu_400 + list_2B_cu[30]
      throws_caught_cu = throws_caught_cu %>% add_row(Third_X = third_x,
Third_Y = third_y)
      break
    }

    #2B Cutoff of throw to third from hit to center field: (0, 180.6)
    list_2B_cu = cutoff_pos_third(j, 15, -15, 195.6, 165.6, 4, of_x, third_x, of_y,
                                  third_y, final_time_third, beginning_time, end3, checkOF)
    if(list_2B_cu[1]==1){
      check_third = 1
      out_third_cu = out_third_cu + list_2B_cu[2]
      safe_third_cu = safe_third_cu + list_2B_cu[3]
      total_time_third_cu = total_time_third_cu + list_2B_cu[4]
      cutoff_time3B = cutoff_time3B + list_2B_cu[5]
      total_accuracy3B_cu = total_accuracy3B_cu + list_2B_cu[6]
      out_third_cu_150 = out_third_cu_150 + list_2B_cu[7]
      out_third_cu_200 = out_third_cu_200 + list_2B_cu[8]
      out_third_cu_250 = out_third_cu_250 + list_2B_cu[9]
      out_third_cu_300 = out_third_cu_300 + list_2B_cu[10]
```

```
        out_third_cu_350 = out_third_cu_350 + list_2B_cu[11]
        out_third_cu_400 = out_third_cu_400 + list_2B_cu[12]
        total_time_third_cu_150 = total_time_third_cu_150 + list_2B_cu[13]
        total_time_third_cu_200 = total_time_third_cu_200 + list_2B_cu[14]
        total_time_third_cu_250 = total_time_third_cu_250 + list_2B_cu[15]
        total_time_third_cu_300 = total_time_third_cu_300 + list_2B_cu[16]
        total_time_third_cu_350 = total_time_third_cu_350 + list_2B_cu[17]
        total_time_third_cu_400 = total_time_third_cu_400 + list_2B_cu[18]
        total_accuracy3B_cu_150 = total_accuracy3B_cu_150 + list_2B_cu[19]
        total_accuracy3B_cu_200 = total_accuracy3B_cu_200 + list_2B_cu[20]
        total_accuracy3B_cu_250 = total_accuracy3B_cu_250 + list_2B_cu[21]
        total_accuracy3B_cu_300 = total_accuracy3B_cu_300 + list_2B_cu[22]
        total_accuracy3B_cu_350 = total_accuracy3B_cu_350 + list_2B_cu[23]
        total_accuracy3B_cu_400 = total_accuracy3B_cu_400 + list_2B_cu[24]
        throws_third_cu_150 = throws_third_cu_150 + list_2B_cu[25]
        throws_third_cu_200 = throws_third_cu_200 + list_2B_cu[26]
        throws_third_cu_250 = throws_third_cu_250 + list_2B_cu[27]
        throws_third_cu_300 = throws_third_cu_300 + list_2B_cu[28]
        throws_third_cu_350 = throws_third_cu_350 + list_2B_cu[29]
        throws_third_cu_400 = throws_third_cu_400 + list_2B_cu[30]
        throws_caught_cu = throws_caught_cu %>% add_row(Third_X = third_x,
Third_Y = third_y)
        break
```

```
}


#2B Cutoff of throw to third from hit to left center: (-66.7, 165.6)
list_2B_cu = cutoff_pos_third(j, -51.7, -81.7, 165.6, 135.6, 4, of_x, third_x, of_y,
                              third_y, final_time_third, beginning_time, end3, checkOF)
if(list_2B_cu[1]==1){
  check_third = 1
  out_third_cu = out_third_cu + list_2B_cu[2]
  safe_third_cu = safe_third_cu + list_2B_cu[3]
  total_time_third_cu = total_time_third_cu + list_2B_cu[4]
  cutoff_time3B = cutoff_time3B + list_2B_cu[5]
  total_accuracy3B_cu = total_accuracy3B_cu + list_2B_cu[6]
  out_third_cu_150 = out_third_cu_150 + list_2B_cu[7]
  out_third_cu_200 = out_third_cu_200 + list_2B_cu[8]
  out_third_cu_250 = out_third_cu_250 + list_2B_cu[9]
  out_third_cu_300 = out_third_cu_300 + list_2B_cu[10]
  out_third_cu_350 = out_third_cu_350 + list_2B_cu[11]
  out_third_cu_400 = out_third_cu_400 + list_2B_cu[12]
  total_time_third_cu_150 = total_time_third_cu_150 + list_2B_cu[13]
  total_time_third_cu_200 = total_time_third_cu_200 + list_2B_cu[14]
  total_time_third_cu_250 = total_time_third_cu_250 + list_2B_cu[15]
  total_time_third_cu_300 = total_time_third_cu_300 + list_2B_cu[16]
  total_time_third_cu_350 = total_time_third_cu_350 + list_2B_cu[17]
```

```
    total_time_third_cu_400 = total_time_third_cu_400 + list_2B_cu[18]
    total_accuracy3B_cu_150 = total_accuracy3B_cu_150 + list_2B_cu[19]
    total_accuracy3B_cu_200 = total_accuracy3B_cu_200 + list_2B_cu[20]
    total_accuracy3B_cu_250 = total_accuracy3B_cu_250 + list_2B_cu[21]
    total_accuracy3B_cu_300 = total_accuracy3B_cu_300 + list_2B_cu[22]
    total_accuracy3B_cu_350 = total_accuracy3B_cu_350 + list_2B_cu[23]
    total_accuracy3B_cu_400 = total_accuracy3B_cu_400 + list_2B_cu[24]
    throws_third_cu_150 = throws_third_cu_150 + list_2B_cu[25]
    throws_third_cu_200 = throws_third_cu_200 + list_2B_cu[26]
    throws_third_cu_250 = throws_third_cu_250 + list_2B_cu[27]
    throws_third_cu_300 = throws_third_cu_300 + list_2B_cu[28]
    throws_third_cu_350 = throws_third_cu_350 + list_2B_cu[29]
    throws_third_cu_400 = throws_third_cu_400 + list_2B_cu[30]
    throws_caught_cu = throws_caught_cu %>% add_row(Third_X = third_x,
Third_Y = third_y)
    break
  }

  #2B Cutoff of throw to third from hit to left field: (-105.6, 122.3)
  list_2B_cu = cutoff_pos_third(j, -90.6, -120.6, 137.3, 107.3, 4, of_x, third_x, of_y,
                        third_y, final_time_third, beginning_time, end3, checkOF)
  if(list_2B_cu[1]==1){
    check_third = 1
```

```
out_third_cu = out_third_cu + list_2B_cu[2]
safe_third_cu = safe_third_cu + list_2B_cu[3]
total_time_third_cu = total_time_third_cu + list_2B_cu[4]
cutoff_time3B = cutoff_time3B + list_2B_cu[5]
total_accuracy3B_cu = total_accuracy3B_cu + list_2B_cu[6]
out_third_cu_150 = out_third_cu_150 + list_2B_cu[7]
out_third_cu_200 = out_third_cu_200 + list_2B_cu[8]
out_third_cu_250 = out_third_cu_250 + list_2B_cu[9]
out_third_cu_300 = out_third_cu_300 + list_2B_cu[10]
out_third_cu_350 = out_third_cu_350 + list_2B_cu[11]
out_third_cu_400 = out_third_cu_400 + list_2B_cu[12]
total_time_third_cu_150 = total_time_third_cu_150 + list_2B_cu[13]
total_time_third_cu_200 = total_time_third_cu_200 + list_2B_cu[14]
total_time_third_cu_250 = total_time_third_cu_250 + list_2B_cu[15]
total_time_third_cu_300 = total_time_third_cu_300 + list_2B_cu[16]
total_time_third_cu_350 = total_time_third_cu_350 + list_2B_cu[17]
total_time_third_cu_400 = total_time_third_cu_400 + list_2B_cu[18]
total_accuracy3B_cu_150 = total_accuracy3B_cu_150 + list_2B_cu[19]
total_accuracy3B_cu_200 = total_accuracy3B_cu_200 + list_2B_cu[20]
total_accuracy3B_cu_250 = total_accuracy3B_cu_250 + list_2B_cu[21]
total_accuracy3B_cu_300 = total_accuracy3B_cu_300 + list_2B_cu[22]
total_accuracy3B_cu_350 = total_accuracy3B_cu_350 + list_2B_cu[23]
total_accuracy3B_cu_400 = total_accuracy3B_cu_400 + list_2B_cu[24]
```

```
              throws_third_cu_150 = throws_third_cu_150 + list_2B_cu[25]

              throws_third_cu_200 = throws_third_cu_200 + list_2B_cu[26]

              throws_third_cu_250 = throws_third_cu_250 + list_2B_cu[27]

              throws_third_cu_300 = throws_third_cu_300 + list_2B_cu[28]

              throws_third_cu_350 = throws_third_cu_350 + list_2B_cu[29]

              throws_third_cu_400 = throws_third_cu_400 + list_2B_cu[30]

              throws_caught_cu = throws_caught_cu %>% add_row(Third_X = third_x,
Third_Y = third_y)

              break

            }

          }

        }


        #Shortstop Cutting Off:

        if(ge$event_code[j] == 2 && ge$player_position[j] == 6 && ge$event_code[j+1] == 3
&& ge$player_position[j+1]==6){#Checks that SS caught ball then threw it

          k = j+2

          while(ge$play_id[k] == ge$play_id[k-1] && ge$event_code[k-1] != 2){ #runs until
play ends or right after ball is caught

            if(ge$event_code[k] == 2 && ge$player_position[k] == 2){ #Checks that catcher
caught the ball

              final_time_home = ge$timestamp[k] #Gets time ball is caught by catcher baseman

              catcher_x = get_x_y_i(final_time_home, 2)[2]
```

```
    catcher_y = get_x_y_i(final_time_home, 2)[3]
    break
   }
   if(ge$event_code[k] == 2 && ge$player_position[k] == 4){ #Checks that second baseman caught the ball
    final_time_second = ge$timestamp[k] #Gets time ball is caught by second baseman
    second_x = get_x_y_i(final_time_second, 4)[2]
    second_y = get_x_y_i(final_time_second, 4)[3]
    break
   }
   if(ge$event_code[k] == 2 && ge$player_position[k] == 5){ #Checks that third baseman caught the ball
    final_time_third = ge$timestamp[k] #Gets time ball is caught by third baseman
    third_x = get_x_y_i(final_time_third, 5)[2]
    third_y = get_x_y_i(final_time_third, 5)[3]
    break
   }
   k = k+1
  }

  #SS cutoff of throw to home from double to LF: (-62.3, 195.6)
  if(ge$event_code[k] == 2 && ge$player_position[k] == 2){
```

```
list_SS_cu = cutoff_pos_home(j, -47.3, -77.3, 210.6, 180.6, 6, of_x, catcher_x, of_y,
                              catcher_y, final_time_home, beginning_time, end3, checkOF)
if(list_SS_cu[1]==1){
  check_home = 1
  out_home_cu = out_home_cu + list_SS_cu[2]
  safe_home_cu = safe_home_cu + list_SS_cu[3]
  total_time_home_cu = total_time_home_cu + list_SS_cu[4]
  cutoff_timeC = cutoff_timeC + list_SS_cu[5]
  total_accuracyC_cu = total_accuracyC_cu + list_SS_cu[6]
  out_home_cu_150 = out_home_cu_150 + list_SS_cu[7]
  out_home_cu_200 = out_home_cu_200 + list_SS_cu[8]
  out_home_cu_250 = out_home_cu_250 + list_SS_cu[9]
  out_home_cu_300 = out_home_cu_300 + list_SS_cu[10]
  out_home_cu_350 = out_home_cu_350 + list_SS_cu[11]
  out_home_cu_400 = out_home_cu_400 + list_SS_cu[12]
  total_time_home_cu_150 = total_time_home_cu_150 + list_SS_cu[13]
  total_time_home_cu_200 = total_time_home_cu_200 + list_SS_cu[14]
  total_time_home_cu_250 = total_time_home_cu_250 + list_SS_cu[15]
  total_time_home_cu_300 = total_time_home_cu_300 + list_SS_cu[16]
  total_time_home_cu_350 = total_time_home_cu_350 + list_SS_cu[17]
  total_time_home_cu_400 = total_time_home_cu_400 + list_SS_cu[18]
  total_accuracyC_cu_150 = total_accuracyC_cu_150 + list_SS_cu[19]
  total_accuracyC_cu_200 = total_accuracyC_cu_200 + list_SS_cu[20]
```

```
      total_accuracyC_cu_250 = total_accuracyC_cu_250 + list_SS_cu[21]
      total_accuracyC_cu_300 = total_accuracyC_cu_300 + list_SS_cu[22]
      total_accuracyC_cu_350 = total_accuracyC_cu_350 + list_SS_cu[23]
      total_accuracyC_cu_400 = total_accuracyC_cu_400 + list_SS_cu[24]
      throws_home_cu_150 = throws_home_cu_150 + list_SS_cu[25]
      throws_home_cu_200 = throws_home_cu_200 + list_SS_cu[26]
      throws_home_cu_250 = throws_home_cu_250 + list_SS_cu[27]
      throws_home_cu_300 = throws_home_cu_300 + list_SS_cu[28]
      throws_home_cu_350 = throws_home_cu_350 + list_SS_cu[29]
      throws_home_cu_400 = throws_home_cu_400 + list_SS_cu[30]
      throws_caught_cu = throws_caught_cu %>% add_row(Catcher_X = catcher_x,
Catcher_Y = catcher_y)
      break
    }

    #SS cutoff of throw to home from double to left center: (-127.4, 137.3)
    list_SS_cu = cutoff_pos_home(j, -112.4, -142.4, 152.3, 122.3, 6, of_x, catcher_x,
of_y,
                              catcher_y, final_time_home, beginning_time, end3, checkOF)
    if(list_SS_cu[1]==1){
      check_home = 1
      out_home_cu = out_home_cu + list_SS_cu[2]
      safe_home_cu = safe_home_cu + list_SS_cu[3]
```

```
total_time_home_cu = total_time_home_cu + list_SS_cu[4]

cutoff_timeC = cutoff_timeC + list_SS_cu[5]

total_accuracyC_cu = total_accuracyC_cu + list_SS_cu[6]

out_home_cu_150 = out_home_cu_150 + list_SS_cu[7]

out_home_cu_200 = out_home_cu_200 + list_SS_cu[8]

out_home_cu_250 = out_home_cu_250 + list_SS_cu[9]

out_home_cu_300 = out_home_cu_300 + list_SS_cu[10]

out_home_cu_350 = out_home_cu_350 + list_SS_cu[11]

out_home_cu_400 = out_home_cu_400 + list_SS_cu[12]

total_time_home_cu_150 = total_time_home_cu_150 + list_SS_cu[13]

total_time_home_cu_200 = total_time_home_cu_200 + list_SS_cu[14]

total_time_home_cu_250 = total_time_home_cu_250 + list_SS_cu[15]

total_time_home_cu_300 = total_time_home_cu_300 + list_SS_cu[16]

total_time_home_cu_350 = total_time_home_cu_350 + list_SS_cu[17]

total_time_home_cu_400 = total_time_home_cu_400 + list_SS_cu[18]

total_accuracyC_cu_150 = total_accuracyC_cu_150 + list_SS_cu[19]

total_accuracyC_cu_200 = total_accuracyC_cu_200 + list_SS_cu[20]

total_accuracyC_cu_250 = total_accuracyC_cu_250 + list_SS_cu[21]

total_accuracyC_cu_300 = total_accuracyC_cu_300 + list_SS_cu[22]

total_accuracyC_cu_350 = total_accuracyC_cu_350 + list_SS_cu[23]

total_accuracyC_cu_400 = total_accuracyC_cu_400 + list_SS_cu[24]

throws_home_cu_150 = throws_home_cu_150 + list_SS_cu[25]

throws_home_cu_200 = throws_home_cu_200 + list_SS_cu[26]
```

```
      throws_home_cu_250 = throws_home_cu_250 + list_SS_cu[27]
      throws_home_cu_300 = throws_home_cu_300 + list_SS_cu[28]
      throws_home_cu_350 = throws_home_cu_350 + list_SS_cu[29]
      throws_home_cu_400 = throws_home_cu_400 + list_SS_cu[30]
      throws_caught_cu = throws_caught_cu %>% add_row(Catcher_X = catcher_x,
Catcher_Y = catcher_y)
      break
    }

    #SS cutoff of throw to home from double to CF: (0, 200.6)
    list_SS_cu = cutoff_pos_home(j, 20, -20, 220.6, 180.6, 6, of_x, catcher_x, of_y,
                         catcher_y, final_time_home, beginning_time, end3, checkOF)
    if(list_SS_cu[1]==1){
      check_home = 1
      out_home_cu = out_home_cu + list_SS_cu[2]
      safe_home_cu = safe_home_cu + list_SS_cu[3]
      total_time_home_cu = total_time_home_cu + list_SS_cu[4]
      cutoff_timeC = cutoff_timeC + list_SS_cu[5]
      total_accuracyC_cu = total_accuracyC_cu + list_SS_cu[6]
      out_home_cu_150 = out_home_cu_150 + list_SS_cu[7]
      out_home_cu_200 = out_home_cu_200 + list_SS_cu[8]
      out_home_cu_250 = out_home_cu_250 + list_SS_cu[9]
      out_home_cu_300 = out_home_cu_300 + list_SS_cu[10]
```

```
out_home_cu_350 = out_home_cu_350 + list_SS_cu[11]
out_home_cu_400 = out_home_cu_400 + list_SS_cu[12]
total_time_home_cu_150 = total_time_home_cu_150 + list_SS_cu[13]
total_time_home_cu_200 = total_time_home_cu_200 + list_SS_cu[14]
total_time_home_cu_250 = total_time_home_cu_250 + list_SS_cu[15]
total_time_home_cu_300 = total_time_home_cu_300 + list_SS_cu[16]
total_time_home_cu_350 = total_time_home_cu_350 + list_SS_cu[17]
total_time_home_cu_400 = total_time_home_cu_400 + list_SS_cu[18]
total_accuracyC_cu_150 = total_accuracyC_cu_150 + list_SS_cu[19]
total_accuracyC_cu_200 = total_accuracyC_cu_200 + list_SS_cu[20]
total_accuracyC_cu_250 = total_accuracyC_cu_250 + list_SS_cu[21]
total_accuracyC_cu_300 = total_accuracyC_cu_300 + list_SS_cu[22]
total_accuracyC_cu_350 = total_accuracyC_cu_350 + list_SS_cu[23]
total_accuracyC_cu_400 = total_accuracyC_cu_400 + list_SS_cu[24]
throws_home_cu_150 = throws_home_cu_150 + list_SS_cu[25]
throws_home_cu_200 = throws_home_cu_200 + list_SS_cu[26]
throws_home_cu_250 = throws_home_cu_250 + list_SS_cu[27]
throws_home_cu_300 = throws_home_cu_300 + list_SS_cu[28]
throws_home_cu_350 = throws_home_cu_350 + list_SS_cu[29]
throws_home_cu_400 = throws_home_cu_400 + list_SS_cu[30]
throws_caught_cu = throws_caught_cu %>% add_row(Catcher_X = catcher_x, Catcher_Y = catcher_y)
break
```

```
}


#SS cutoff of throw to home from double to right center: (127.4, 137.3)
list_SS_cu = cutoff_pos_home(j, 142.4, 112.4, 152.3, 122.3, 6, of_x, catcher_x, of_y,
                              catcher_y, final_time_home, beginning_time, end3, checkOF)
if(list_SS_cu[1]==1){
  check_home = 1


  out_home_cu = out_home_cu + list_SS_cu[2]
  safe_home_cu = safe_home_cu + list_SS_cu[3]
  total_time_home_cu = total_time_home_cu + list_SS_cu[4]
  cutoff_timeC = cutoff_timeC + list_SS_cu[5]
  total_accuracyC_cu = total_accuracyC_cu + list_SS_cu[6]
  out_home_cu_150 = out_home_cu_150 + list_SS_cu[7]
  out_home_cu_200 = out_home_cu_200 + list_SS_cu[8]
  out_home_cu_250 = out_home_cu_250 + list_SS_cu[9]
  out_home_cu_300 = out_home_cu_300 + list_SS_cu[10]
  out_home_cu_350 = out_home_cu_350 + list_SS_cu[11]
  out_home_cu_400 = out_home_cu_400 + list_SS_cu[12]
  total_time_home_cu_150 = total_time_home_cu_150 + list_SS_cu[13]
  total_time_home_cu_200 = total_time_home_cu_200 + list_SS_cu[14]
  total_time_home_cu_250 = total_time_home_cu_250 + list_SS_cu[15]
  total_time_home_cu_300 = total_time_home_cu_300 + list_SS_cu[16]
```

```
          total_time_home_cu_350 = total_time_home_cu_350 + list_SS_cu[17]

          total_time_home_cu_400 = total_time_home_cu_400 + list_SS_cu[18]

          total_accuracyC_cu_150 = total_accuracyC_cu_150 + list_SS_cu[19]

          total_accuracyC_cu_200 = total_accuracyC_cu_200 + list_SS_cu[20]

          total_accuracyC_cu_250 = total_accuracyC_cu_250 + list_SS_cu[21]

          total_accuracyC_cu_300 = total_accuracyC_cu_300 + list_SS_cu[22]

          total_accuracyC_cu_350 = total_accuracyC_cu_350 + list_SS_cu[23]

          total_accuracyC_cu_400 = total_accuracyC_cu_400 + list_SS_cu[24]

          throws_home_cu_150 = throws_home_cu_150 + list_SS_cu[25]

          throws_home_cu_200 = throws_home_cu_200 + list_SS_cu[26]

          throws_home_cu_250 = throws_home_cu_250 + list_SS_cu[27]

          throws_home_cu_300 = throws_home_cu_300 + list_SS_cu[28]

          throws_home_cu_350 = throws_home_cu_350 + list_SS_cu[29]

          throws_home_cu_400 = throws_home_cu_400 + list_SS_cu[30]

          throws_caught_cu = throws_caught_cu %>% add_row(Catcher_X = catcher_x,

Catcher_Y = catcher_y)

          break

        }



        #SS cutoff of throw to home from double to RF: (62.3, 195.6)

        list_SS_cu = cutoff_pos_home(j, 77.3, 47.3, 210.6, 180.6, 6, of_x, catcher_x, of_y,

                        catcher_y, final_time_home, beginning_time, end3, checkOF)

        if(list_SS_cu[1]==1){
```

```
check_home = 1
out_home_cu = out_home_cu + list_SS_cu[2]
safe_home_cu = safe_home_cu + list_SS_cu[3]
total_time_home_cu = total_time_home_cu + list_SS_cu[4]
cutoff_timeC = cutoff_timeC + list_SS_cu[5]
total_accuracyC_cu = total_accuracyC_cu + list_SS_cu[6]
out_home_cu_150 = out_home_cu_150 + list_SS_cu[7]
out_home_cu_200 = out_home_cu_200 + list_SS_cu[8]
out_home_cu_250 = out_home_cu_250 + list_SS_cu[9]
out_home_cu_300 = out_home_cu_300 + list_SS_cu[10]
out_home_cu_350 = out_home_cu_350 + list_SS_cu[11]
out_home_cu_400 = out_home_cu_400 + list_SS_cu[12]
total_time_home_cu_150 = total_time_home_cu_150 + list_SS_cu[13]
total_time_home_cu_200 = total_time_home_cu_200 + list_SS_cu[14]
total_time_home_cu_250 = total_time_home_cu_250 + list_SS_cu[15]
total_time_home_cu_300 = total_time_home_cu_300 + list_SS_cu[16]
total_time_home_cu_350 = total_time_home_cu_350 + list_SS_cu[17]
total_time_home_cu_400 = total_time_home_cu_400 + list_SS_cu[18]
total_accuracyC_cu_150 = total_accuracyC_cu_150 + list_SS_cu[19]
total_accuracyC_cu_200 = total_accuracyC_cu_200 + list_SS_cu[20]
total_accuracyC_cu_250 = total_accuracyC_cu_250 + list_SS_cu[21]
total_accuracyC_cu_300 = total_accuracyC_cu_300 + list_SS_cu[22]
total_accuracyC_cu_350 = total_accuracyC_cu_350 + list_SS_cu[23]
```

```
        total_accuracyC_cu_400 = total_accuracyC_cu_400 + list_SS_cu[24]
        throws_home_cu_150 = throws_home_cu_150 + list_SS_cu[25]
        throws_home_cu_200 = throws_home_cu_200 + list_SS_cu[26]
        throws_home_cu_250 = throws_home_cu_250 + list_SS_cu[27]
        throws_home_cu_300 = throws_home_cu_300 + list_SS_cu[28]
        throws_home_cu_350 = throws_home_cu_350 + list_SS_cu[29]
        throws_home_cu_400 = throws_home_cu_400 + list_SS_cu[30]
        throws_caught_cu = throws_caught_cu %>% add_row(Catcher_X = catcher_x,
Catcher_Y = catcher_y)
        break
      }
    }

    #SS cutoff of throw to second from hit to LF - (1/2 way between 2nd and 1st plus a
few feet, 155.6): (-36.9, 155.6) - x is 1/2 way between 2nd and 3rd plus some feet
    if(ge$event_code[k] == 2 && ge$player_position[k] == 4){
      list_SS_cu = cutoff_pos_second(j, -21.9, -51.9, 170.6, 140.6, 6, of_x, second_x, of_y,
                          second_y, final_time_second, beginning_time, end3, checkOF)
      if(list_SS_cu[1]==1){
        check_second = 1

        out_second_cu = out_second_cu + list_SS_cu[2]
        safe_second_cu = safe_second_cu + list_SS_cu[3]
```

```
total_time_second_cu = total_time_second_cu + list_SS_cu[4]
cutoff_time2B = cutoff_time2B + list_SS_cu[5]
total_accuracy2B_cu = total_accuracy2B_cu + list_SS_cu[6]
out_second_cu_150 = out_second_cu_150 + list_SS_cu[7]
out_second_cu_200 = out_second_cu_200 + list_SS_cu[8]
out_second_cu_250 = out_second_cu_250 + list_SS_cu[9]
out_second_cu_300 = out_second_cu_300 + list_SS_cu[10]
out_second_cu_350 = out_second_cu_350 + list_SS_cu[11]
out_second_cu_400 = out_second_cu_400 + list_SS_cu[12]
total_time_second_cu_150 = total_time_second_cu_150 + list_SS_cu[13]
total_time_second_cu_200 = total_time_second_cu_200 + list_SS_cu[14]
total_time_second_cu_250 = total_time_second_cu_250 + list_SS_cu[15]
total_time_second_cu_300 = total_time_second_cu_300 + list_SS_cu[16]
total_time_second_cu_350 = total_time_second_cu_350 + list_SS_cu[17]
total_time_second_cu_400 = total_time_second_cu_400 + list_SS_cu[18]
total_accuracy2B_cu_150 = total_accuracy2B_cu_150 + list_SS_cu[19]
total_accuracy2B_cu_200 = total_accuracy2B_cu_200 + list_SS_cu[20]
total_accuracy2B_cu_250 = total_accuracy2B_cu_250 + list_SS_cu[21]
total_accuracy2B_cu_300 = total_accuracy2B_cu_300 + list_SS_cu[22]
total_accuracy2B_cu_350 = total_accuracy2B_cu_350 + list_SS_cu[23]
total_accuracy2B_cu_400 = total_accuracy2B_cu_400 + list_SS_cu[24]
throws_second_cu_150 = throws_second_cu_150 + list_SS_cu[25]
throws_second_cu_200 = throws_second_cu_200 + list_SS_cu[26]
```

```
      throws_second_cu_250 = throws_second_cu_250 + list_SS_cu[27]
      throws_second_cu_300 = throws_second_cu_300 + list_SS_cu[28]
      throws_second_cu_350 = throws_second_cu_350 + list_SS_cu[29]
      throws_second_cu_400 = throws_second_cu_400 + list_SS_cu[30]
      throws_caught_cu = throws_caught_cu %>% add_row(Second_X = second_x,
Second_Y = second_y)
      break
    }

    #SS cutoff of throw to second from hit to left center: (-68.7, 185.6)
    list_SS_cu = cutoff_pos_second(j, -53.7, -83.7, 200.6, 170.6, 6, of_x, second_x, of_y,
                                   second_y, final_time_second, beginning_time, end3, checkOF)
    if(list_SS_cu[1]==1){
      check_second = 1

      out_second_cu = out_second_cu + list_SS_cu[2]
      safe_second_cu = safe_second_cu + list_SS_cu[3]
      total_time_second_cu = total_time_second_cu + list_SS_cu[4]
      cutoff_time2B = cutoff_time2B + list_SS_cu[5]
      total_accuracy2B_cu = total_accuracy2B_cu + list_SS_cu[6]
      out_second_cu_150 = out_second_cu_150 + list_SS_cu[7]
      out_second_cu_200 = out_second_cu_200 + list_SS_cu[8]
      out_second_cu_250 = out_second_cu_250 + list_SS_cu[9]
```

```
        out_second_cu_300 = out_second_cu_300 + list_SS_cu[10]
        out_second_cu_350 = out_second_cu_350 + list_SS_cu[11]
        out_second_cu_400 = out_second_cu_400 + list_SS_cu[12]
        total_time_second_cu_150 = total_time_second_cu_150 + list_SS_cu[13]
        total_time_second_cu_200 = total_time_second_cu_200 + list_SS_cu[14]
        total_time_second_cu_250 = total_time_second_cu_250 + list_SS_cu[15]
        total_time_second_cu_300 = total_time_second_cu_300 + list_SS_cu[16]
        total_time_second_cu_350 = total_time_second_cu_350 + list_SS_cu[17]
        total_time_second_cu_400 = total_time_second_cu_400 + list_SS_cu[18]
        total_accuracy2B_cu_150 = total_accuracy2B_cu_150 + list_SS_cu[19]
        total_accuracy2B_cu_200 = total_accuracy2B_cu_200 + list_SS_cu[20]
        total_accuracy2B_cu_250 = total_accuracy2B_cu_250 + list_SS_cu[21]
        total_accuracy2B_cu_300 = total_accuracy2B_cu_300 + list_SS_cu[22]
        total_accuracy2B_cu_350 = total_accuracy2B_cu_350 + list_SS_cu[23]
        total_accuracy2B_cu_400 = total_accuracy2B_cu_400 + list_SS_cu[24]
        throws_second_cu_150 = throws_second_cu_150 + list_SS_cu[25]
        throws_second_cu_200 = throws_second_cu_200 + list_SS_cu[26]
        throws_second_cu_250 = throws_second_cu_250 + list_SS_cu[27]
        throws_second_cu_300 = throws_second_cu_300 + list_SS_cu[28]
        throws_second_cu_350 = throws_second_cu_350 + list_SS_cu[29]
        throws_second_cu_400 = throws_second_cu_400 + list_SS_cu[30]
        throws_caught_cu = throws_caught_cu %>% add_row(Second_X = second_x,
Second_Y = second_y)
```

```
  break
}


#SS Cutoff of throw to second from hit to center field: (0, 175.6)
list_SS_cu = cutoff_pos_second(j, 15, -15, 190.6, 150.6, 6, of_x, second_x, of_y,
                              second_y, final_time_second, beginning_time, end3, checkOF)
if(list_SS_cu[1]==1){
  check_second = 1
  out_second_cu = out_second_cu + list_SS_cu[2]
  safe_second_cu = safe_second_cu + list_SS_cu[3]
  total_time_second_cu = total_time_second_cu + list_SS_cu[4]
  cutoff_time2B = cutoff_time2B + list_SS_cu[5]
  total_accuracy2B_cu = total_accuracy2B_cu + list_SS_cu[6]
  out_second_cu_150 = out_second_cu_150 + list_SS_cu[7]
  out_second_cu_200 = out_second_cu_200 + list_SS_cu[8]
  out_second_cu_250 = out_second_cu_250 + list_SS_cu[9]
  out_second_cu_300 = out_second_cu_300 + list_SS_cu[10]
  out_second_cu_350 = out_second_cu_350 + list_SS_cu[11]
  out_second_cu_400 = out_second_cu_400 + list_SS_cu[12]
  total_time_second_cu_150 = total_time_second_cu_150 + list_SS_cu[13]
  total_time_second_cu_200 = total_time_second_cu_200 + list_SS_cu[14]
  total_time_second_cu_250 = total_time_second_cu_250 + list_SS_cu[15]
  total_time_second_cu_300 = total_time_second_cu_300 + list_SS_cu[16]
```

```
        total_time_second_cu_350 = total_time_second_cu_350 + list_SS_cu[17]

        total_time_second_cu_400 = total_time_second_cu_400 + list_SS_cu[18]

        total_accuracy2B_cu_150 = total_accuracy2B_cu_150 + list_SS_cu[19]

        total_accuracy2B_cu_200 = total_accuracy2B_cu_200 + list_SS_cu[20]

        total_accuracy2B_cu_250 = total_accuracy2B_cu_250 + list_SS_cu[21]

        total_accuracy2B_cu_300 = total_accuracy2B_cu_300 + list_SS_cu[22]

        total_accuracy2B_cu_350 = total_accuracy2B_cu_350 + list_SS_cu[23]

        total_accuracy2B_cu_400 = total_accuracy2B_cu_400 + list_SS_cu[24]

        throws_second_cu_150 = throws_second_cu_150 + list_SS_cu[25]

        throws_second_cu_200 = throws_second_cu_200 + list_SS_cu[26]

        throws_second_cu_250 = throws_second_cu_250 + list_SS_cu[27]

        throws_second_cu_300 = throws_second_cu_300 + list_SS_cu[28]

        throws_second_cu_350 = throws_second_cu_350 + list_SS_cu[29]

        throws_second_cu_400 = throws_second_cu_400 + list_SS_cu[30]

        throws_caught_cu = throws_caught_cu %>% add_row(Second_X = second_x,

Second_Y = second_y)

        break

       }

      }



      #SS cutoff of throw to third from hit to left field: (-127.4, 145.6)

      if(ge$event_code[k] == 2 && ge$player_position[k] == 5){

       list_SS_cu = cutoff_pos_third(j, -112.4, -142.4, 160.6, 130.6, 6, of_x, third_x, of_y,
                    third_y, final_time_third, beginning_time, end3, checkOF)
if(list_SS_cu[1]==1){
  check_third = 1
  out_third_cu = out_third_cu + list_SS_cu[2]
  safe_third_cu = safe_third_cu + list_SS_cu[3]
  total_time_third_cu = total_time_third_cu + list_SS_cu[4]
  cutoff_time3B = cutoff_time3B + list_SS_cu[5]
  total_accuracy3B_cu = total_accuracy3B_cu + list_SS_cu[6]
  out_third_cu_150 = out_third_cu_150 + list_SS_cu[7]
  out_third_cu_200 = out_third_cu_200 + list_SS_cu[8]
  out_third_cu_250 = out_third_cu_250 + list_SS_cu[9]
  out_third_cu_300 = out_third_cu_300 + list_SS_cu[10]
  out_third_cu_350 = out_third_cu_350 + list_SS_cu[11]
  out_third_cu_400 = out_third_cu_400 + list_SS_cu[12]
  total_time_third_cu_150 = total_time_third_cu_150 + list_SS_cu[13]
  total_time_third_cu_200 = total_time_third_cu_200 + list_SS_cu[14]
  total_time_third_cu_250 = total_time_third_cu_250 + list_SS_cu[15]
  total_time_third_cu_300 = total_time_third_cu_300 + list_SS_cu[16]
  total_time_third_cu_350 = total_time_third_cu_350 + list_SS_cu[17]
  total_time_third_cu_400 = total_time_third_cu_400 + list_SS_cu[18]
  total_accuracy3B_cu_150 = total_accuracy3B_cu_150 + list_SS_cu[19]
  total_accuracy3B_cu_200 = total_accuracy3B_cu_200 + list_SS_cu[20]
  total_accuracy3B_cu_250 = total_accuracy3B_cu_250 + list_SS_cu[21]
```

```
          total_accuracy3B_cu_300 = total_accuracy3B_cu_300 + list_SS_cu[22]

          total_accuracy3B_cu_350 = total_accuracy3B_cu_350 + list_SS_cu[23]

          total_accuracy3B_cu_400 = total_accuracy3B_cu_400 + list_SS_cu[24]

          throws_third_cu_150 = throws_third_cu_150 + list_SS_cu[25]

          throws_third_cu_200 = throws_third_cu_200 + list_SS_cu[26]

          throws_third_cu_250 = throws_third_cu_250 + list_SS_cu[27]

          throws_third_cu_300 = throws_third_cu_300 + list_SS_cu[28]

          throws_third_cu_350 = throws_third_cu_350 + list_SS_cu[29]

          throws_third_cu_400 = throws_third_cu_400 + list_SS_cu[30]

          throws_caught_cu = throws_caught_cu %>% add_row(Third_X = third_x,

Third_Y = third_y)

          break

        }


        #SS cutoff of throw to third from hit to right center: (5, 190.6)

        list_SS_cu = cutoff_pos_third(j, 20, -10, 205.6, 175.6, 6, of_x, third_x, of_y,

                                 third_y, final_time_third, beginning_time, end3, checkOF)

        if(list_SS_cu[1]==1){

          check_third = 1

          out_third_cu = out_third_cu + list_SS_cu[2]

          safe_third_cu = safe_third_cu + list_SS_cu[3]

          total_time_third_cu = total_time_third_cu + list_SS_cu[4]

          cutoff_time3B = cutoff_time3B + list_SS_cu[5]
```

```
total_accuracy3B_cu = total_accuracy3B_cu + list_SS_cu[6]
out_third_cu_150 = out_third_cu_150 + list_SS_cu[7]
out_third_cu_200 = out_third_cu_200 + list_SS_cu[8]
out_third_cu_250 = out_third_cu_250 + list_SS_cu[9]
out_third_cu_300 = out_third_cu_300 + list_SS_cu[10]
out_third_cu_350 = out_third_cu_350 + list_SS_cu[11]
out_third_cu_400 = out_third_cu_400 + list_SS_cu[12]
total_time_third_cu_150 = total_time_third_cu_150 + list_SS_cu[13]
total_time_third_cu_200 = total_time_third_cu_200 + list_SS_cu[14]
total_time_third_cu_250 = total_time_third_cu_250 + list_SS_cu[15]
total_time_third_cu_300 = total_time_third_cu_300 + list_SS_cu[16]
total_time_third_cu_350 = total_time_third_cu_350 + list_SS_cu[17]
total_time_third_cu_400 = total_time_third_cu_400 + list_SS_cu[18]
total_accuracy3B_cu_150 = total_accuracy3B_cu_150 + list_SS_cu[19]
total_accuracy3B_cu_200 = total_accuracy3B_cu_200 + list_SS_cu[20]
total_accuracy3B_cu_250 = total_accuracy3B_cu_250 + list_SS_cu[21]
total_accuracy3B_cu_300 = total_accuracy3B_cu_300 + list_SS_cu[22]
total_accuracy3B_cu_350 = total_accuracy3B_cu_350 + list_SS_cu[23]
total_accuracy3B_cu_400 = total_accuracy3B_cu_400 + list_SS_cu[24]
throws_third_cu_150 = throws_third_cu_150 + list_SS_cu[25]
throws_third_cu_200 = throws_third_cu_200 + list_SS_cu[26]
throws_third_cu_250 = throws_third_cu_250 + list_SS_cu[27]
throws_third_cu_300 = throws_third_cu_300 + list_SS_cu[28]
```

```
          throws_third_cu_350 = throws_third_cu_350 + list_SS_cu[29]
          throws_third_cu_400 = throws_third_cu_400 + list_SS_cu[30]
          throws_caught_cu = throws_caught_cu %>% add_row(Third_X = third_x,
Third_Y = third_y)
          break
        }

        #SS Cutoff of throw to third from hit to left center: (-68.7, 190.6)
        list_SS_cu = cutoff_pos_third(j, -53.7, -83.7, 205.6, 175.6, 6, of_x, third_x, of_y,
                               third_y, final_time_third, beginning_time, end3, checkOF)
        if(list_SS_cu[1]==1){
          check_third = 1
          out_third_cu = out_third_cu + list_SS_cu[2]
          safe_third_cu = safe_third_cu + list_SS_cu[3]
          total_time_third_cu = total_time_third_cu + list_SS_cu[4]
          cutoff_time3B = cutoff_time3B + list_SS_cu[5]
          total_accuracy3B_cu = total_accuracy3B_cu + list_SS_cu[6]
          out_third_cu_150 = out_third_cu_150 + list_SS_cu[7]
          out_third_cu_200 = out_third_cu_200 + list_SS_cu[8]
          out_third_cu_250 = out_third_cu_250 + list_SS_cu[9]
          out_third_cu_300 = out_third_cu_300 + list_SS_cu[10]
          out_third_cu_350 = out_third_cu_350 + list_SS_cu[11]
          out_third_cu_400 = out_third_cu_400 + list_SS_cu[12]
          total_time_third_cu_150 = total_time_third_cu_150 + list_SS_cu[13]
          total_time_third_cu_200 = total_time_third_cu_200 + list_SS_cu[14]
          total_time_third_cu_250 = total_time_third_cu_250 + list_SS_cu[15]
          total_time_third_cu_300 = total_time_third_cu_300 + list_SS_cu[16]
          total_time_third_cu_350 = total_time_third_cu_350 + list_SS_cu[17]
          total_time_third_cu_400 = total_time_third_cu_400 + list_SS_cu[18]
          total_accuracy3B_cu_150 = total_accuracy3B_cu_150 + list_SS_cu[19]
          total_accuracy3B_cu_200 = total_accuracy3B_cu_200 + list_SS_cu[20]
          total_accuracy3B_cu_250 = total_accuracy3B_cu_250 + list_SS_cu[21]
          total_accuracy3B_cu_300 = total_accuracy3B_cu_300 + list_SS_cu[22]
          total_accuracy3B_cu_350 = total_accuracy3B_cu_350 + list_SS_cu[23]
          total_accuracy3B_cu_400 = total_accuracy3B_cu_400 + list_SS_cu[24]
          throws_third_cu_150 = throws_third_cu_150 + list_SS_cu[25]
          throws_third_cu_200 = throws_third_cu_200 + list_SS_cu[26]
          throws_third_cu_250 = throws_third_cu_250 + list_SS_cu[27]
          throws_third_cu_300 = throws_third_cu_300 + list_SS_cu[28]
          throws_third_cu_350 = throws_third_cu_350 + list_SS_cu[29]
          throws_third_cu_400 = throws_third_cu_400 + list_SS_cu[30]
          throws_caught_cu = throws_caught_cu %>% add_row(Third_X = third_x,
Third_Y = third_y)
          break
        }
       }
```

```
     }


     #3B Cutting Off:
     if(ge$event_code[j] == 2 && ge$player_position[j] == 5 && ge$event_code[j+1] == 3
&& ge$player_position[j+1]==5){#Checks that 3B caught ball then threw it
      k = j+2
      while(ge$play_id[k] == ge$play_id[k-1] && ge$event_code[k-1] != 2){ #runs until
play ends or right after ball is caught
       if(ge$event_code[k] == 2 && ge$player_position[k] == 2){ #Checks that catcher
caught the ball
        final_time_home = ge$timestamp[k] #Gets time ball is caught by catcher
        catcher_x = get_x_y_i(final_time_home, 2)[2]
        catcher_y = get_x_y_i(final_time_home, 2)[3]
        break
       }
       k = k+1
      }
      if(ge$event_code[k] == 2 && ge$player_position[k] == 2){
       #3B Cutoff of throw to home from single to left field: (-26.7, 53.7)
       list_3B_cu = cutoff_pos_home(j, -11.7, -41.7, 68.7, 38.7, 5, of_x, catcher_x, of_y,
                     catcher_y, final_time_home, beginning_time, end3, checkOF)
       if(list_3B_cu[1]==1){
        check_home = 1
```

```
out_home_cu = out_home_cu + list_3B_cu[2]

safe_home_cu = safe_home_cu + list_3B_cu[3]

total_time_home_cu = total_time_home_cu + list_3B_cu[4]

cutoff_timeC = cutoff_timeC + list_3B_cu[5]

total_accuracyC_cu = total_accuracyC_cu + list_3B_cu[6]

out_home_cu_150 = out_home_cu_150 + list_3B_cu[7]

out_home_cu_200 = out_home_cu_200 + list_3B_cu[8]

out_home_cu_250 = out_home_cu_250 + list_3B_cu[9]

out_home_cu_300 = out_home_cu_300 + list_3B_cu[10]

out_home_cu_350 = out_home_cu_350 + list_3B_cu[11]

out_home_cu_400 = out_home_cu_400 + list_3B_cu[12]

total_time_home_cu_150 = total_time_home_cu_150 + list_3B_cu[13]

total_time_home_cu_200 = total_time_home_cu_200 + list_3B_cu[14]

total_time_home_cu_250 = total_time_home_cu_250 + list_3B_cu[15]

total_time_home_cu_300 = total_time_home_cu_300 + list_3B_cu[16]

total_time_home_cu_350 = total_time_home_cu_350 + list_3B_cu[17]

total_time_home_cu_400 = total_time_home_cu_400 + list_3B_cu[18]

total_accuracyC_cu_150 = total_accuracyC_cu_150 + list_3B_cu[19]

total_accuracyC_cu_200 = total_accuracyC_cu_200 + list_3B_cu[20]

total_accuracyC_cu_250 = total_accuracyC_cu_250 + list_3B_cu[21]

total_accuracyC_cu_300 = total_accuracyC_cu_300 + list_3B_cu[22]

total_accuracyC_cu_350 = total_accuracyC_cu_350 + list_3B_cu[23]

total_accuracyC_cu_400 = total_accuracyC_cu_400 + list_3B_cu[24]
```

```
        throws_home_cu_150 = throws_home_cu_150 + list_3B_cu[25]
        throws_home_cu_200 = throws_home_cu_200 + list_3B_cu[26]
        throws_home_cu_250 = throws_home_cu_250 + list_3B_cu[27]
        throws_home_cu_300 = throws_home_cu_300 + list_3B_cu[28]
        throws_home_cu_350 = throws_home_cu_350 + list_3B_cu[29]
        throws_home_cu_400 = throws_home_cu_400 + list_3B_cu[30]
        throws_caught_cu = throws_caught_cu %>% add_row(Catcher_X = catcher_x, Catcher_Y = catcher_y)
        break
       }
      }
     }
     j = j+1
    }
   }
   if(checkOF==1 && check_second==1) #checks if a throw was made by an outfielder to second with a cutoff.
    throws_second_cu = throws_second_cu + 1

   if(checkOF==1 && check_third==1) #checks if a throw was made by an outfielder to third with a cutoff.
    throws_third_cu = throws_third_cu + 1
    if(checkOF==1 && check_home==1) #checks if a throw was made by an outfielder to
home with a cutoff.
      throws_home_cu = throws_home_cu + 1


    checkOF = 0
    check_second = 0
    check_third = 0
    check_home = 0
    a=a+1
   }
   a = a+1
  }
  index = index + 1
}


if(out_second_NC > 0 || safe_second_NC > 0) #to avoid dividing by 0
  success_rate_2B_NC = (out_second_NC/(safe_second_NC + out_second_NC)) * 100


if(out_second_NC_150 > 0 || safe_second_NC_150 > 0)
  success_rate_2B_NC_150 = (out_second_NC_150/(safe_second_NC_150 +
out_second_NC_150)) * 100


if(out_second_NC_200 > 0 || safe_second_NC_200 > 0)
```

```
  success_rate_2B_NC_200 = (out_second_NC_200/(safe_second_NC_200 +
out_second_NC_200)) * 100

if(out_second_NC_250 > 0 || safe_second_NC_250 > 0)
  success_rate_2B_NC_250 = (out_second_NC_250/(safe_second_NC_250 +
out_second_NC_250)) * 100

if(out_second_NC_300 > 0 || safe_second_NC_300 > 0)
  success_rate_2B_NC_300 = (out_second_NC_300/(safe_second_NC_300 +
out_second_NC_300)) * 100

if(out_second_NC_350 > 0 || safe_second_NC_350 > 0)
  success_rate_2B_NC_350 = (out_second_NC_350/(safe_second_NC_350 +
out_second_NC_350)) * 100

if(out_second_NC_400 > 0 || safe_second_NC_400 > 0)
  success_rate_2B_NC_400 = (out_second_NC_400/(safe_second_NC_400 +
out_second_NC_400)) * 100

if(out_third_NC > 0 || safe_third_NC > 0)
  success_rate_3B_NC = (out_third_NC/(out_third_NC + safe_third_NC)) * 100

if(out_third_NC_150 > 0 || safe_third_NC_150 > 0)
```

```
  success_rate_3B_NC_150 = (out_third_NC_150/(safe_third_NC_150 +
out_third_NC_150)) * 100

if(out_third_NC_200 > 0 || safe_third_NC_200 > 0)
  success_rate_3B_NC_200 = (out_third_NC_200/(safe_third_NC_200 +
out_third_NC_200)) * 100

if(out_third_NC_250 > 0 || safe_third_NC_250 > 0)
  success_rate_3B_NC_250 = (out_third_NC_250/(safe_third_NC_250 +
out_third_NC_250)) * 100

if(out_third_NC_300 > 0 || safe_third_NC_300 > 0)
  success_rate_3B_NC_300 = (out_third_NC_300/(safe_third_NC_300 +
out_third_NC_300)) * 100

if(out_third_NC_350 > 0 || safe_third_NC_350 > 0)
  success_rate_3B_NC_350 = (out_third_NC_350/(safe_third_NC_350 +
out_third_NC_350)) * 100

if(out_third_NC_400 > 0 || safe_third_NC_400 > 0)
  success_rate_3B_NC_400 = (out_third_NC_400/(safe_third_NC_400 +
out_third_NC_400)) * 100
```

```
if(out_catcher_NC > 0){
  average_timeC_NC = round((total_time_home_NC/out_catcher_NC), 2) #Measures
average time on throws to home that resulted in outs
  average_accuracyC_NC = round((total_accuracyC_NC/out_catcher_NC), 2) #Measures
average accuracy per throw to catcher that resulted in an out
}
if(out_catcher_NC_150 > 0){
  average_timeC_NC_150 = round((total_time_home_NC_150/out_catcher_NC_150), 2)
  average_accuracyC_NC_150 = round((total_accuracyC_NC_150/out_catcher_NC_150),
2)
}
if(out_catcher_NC_200 > 0){
  average_timeC_NC_200 = round((total_time_home_NC_200/out_catcher_NC_200), 2)
  average_accuracyC_NC_200 = round((total_accuracyC_NC_200/out_catcher_NC_200),
2)
}
if(out_catcher_NC_250 > 0){
  average_timeC_NC_250 = round((total_time_home_NC_250/out_catcher_NC_250), 2)
  average_accuracyC_NC_250 = round((total_accuracyC_NC_250/out_catcher_NC_250),
2)
}
if(out_catcher_NC_300 > 0){
  average_timeC_NC_300 = round((total_time_home_NC_300/out_catcher_NC_300), 2)
  average_accuracyC_NC_300 = round((total_accuracyC_NC_300/out_catcher_NC_300),
2)
}
if(out_catcher_NC_350 > 0){
  average_timeC_NC_350 = round((total_time_home_NC_350/out_catcher_NC_350), 2)
  average_accuracyC_NC_350 = round((total_accuracyC_NC_350/out_catcher_NC_350),
2)
}
if(out_catcher_NC_400 > 0){
  average_timeC_NC_400 = round((total_time_home_NC_400/out_catcher_NC_400), 2)
  average_accuracyC_NC_400 = round((total_accuracyC_NC_400/out_catcher_NC_400),
2)
}
if(out_second_NC > 0){
  average_time2B_NC = round((total_time_second_NC/out_second_NC), 2) #Measures
average time on throws to second that resulted in outs
  average_accuracy2B_NC = round((total_accuracy2B_NC/out_second_NC), 2) #Measures
average accuracy per throw to 2B that resulted in an out
}
if(out_second_NC_150 > 0){
  average_time2B_NC_150 = round((total_time_second_NC_150/out_second_NC_150), 2)
  average_accuracy2B_NC_150 = round((total_accuracy2B_NC_150/out_second_NC_150),
2)
```

```
}
if(out_second_NC_200 > 0){
  average_time2B_NC_200 = round((total_time_second_NC_200/out_second_NC_200), 2)
  average_accuracy2B_NC_200 = round((total_accuracy2B_NC_200/out_second_NC_200),
2)
}
if(out_second_NC_250 > 0){
  average_time2B_NC_250 = round((total_time_second_NC_250/out_second_NC_250), 2)
  average_accuracy2B_NC_250 = round((total_accuracy2B_NC_250/out_second_NC_250),
2)
}
if(out_second_NC_300 > 0){
  average_time2B_NC_300 = round((total_time_second_NC_300/out_second_NC_300), 2)
  average_accuracy2B_NC_300 = round((total_accuracy2B_NC_300/out_second_NC_300),
2)
}
if(out_second_NC_350 > 0){
  average_time2B_NC_350 = round((total_time_second_NC_350/out_second_NC_350), 2)
  average_accuracy2B_NC_350 = round((total_accuracy2B_NC_350/out_second_NC_350),
2)
}
if(out_second_NC_400 > 0){
  average_time2B_NC_400 = round((total_time_second_NC_400/out_second_NC_400), 2)
  average_accuracy2B_NC_400 = round((total_accuracy2B_NC_400/out_second_NC_400),
2)
}


if(out_third_NC > 0){
  average_time3B_NC = round((total_time_third_NC/out_third_NC), 2) #Measures average
time on throws to third that resulted in outs
  average_accuracy3B_NC = round((total_accuracy3B_NC/out_third_NC), 2) #Measures
average accuracy per throw to 3B that resulted in an out
}
if(out_third_NC_150 > 0){
  average_time3B_NC_200 = round((total_time_third_NC_200/out_third_NC_200), 2)
  average_accuracy3B_NC_200 = round((total_accuracy3B_NC_200/out_third_NC_200), 2)
}
if(out_third_NC_200 > 0){
  average_time3B_NC_200 = round((total_time_third_NC_200/out_third_NC_200), 2)
  average_accuracy3B_NC_200 = round((total_accuracy3B_NC_200/out_third_NC_200), 2)
}
if(out_third_NC_250 > 0){
  average_time3B_NC_250 = round((total_time_third_NC_250/out_third_NC_250), 2)
  average_accuracy3B_NC_250 = round((total_accuracy3B_NC_250/out_third_NC_250), 2)
}
if(out_third_NC_300 > 0){
  average_time3B_NC_300 = round((total_time_third_NC_300/out_third_NC_300), 2)
  average_accuracy3B_NC_300 = round((total_accuracy3B_NC_300/out_third_NC_300), 2)
}
if(out_third_NC_350 > 0){
  average_time3B_NC_350 = round((total_time_third_NC_350/out_third_NC_350), 2)
  average_accuracy3B_NC_350 = round((total_accuracy3B_NC_350/out_third_NC_350), 2)
}
if(out_third_NC_400 > 0){
  average_time3B_NC_400 = round((total_time_third_NC_400/out_third_NC_400), 2)
  average_accuracy3B_NC_400 = round((total_accuracy3B_NC_400/out_third_NC_400), 2)
}

Total_Time_NC = total_time_home_NC + total_time_second_NC + total_time_third_NC
if(out_third_NC > 0 || out_second_NC > 0 || out_catcher_NC > 0) #to avoid dividing by 0
  Average_Time_NC = round((Total_Time_NC/(out_third_NC + out_second_NC +
out_catcher_NC)), 2)

Total_Time_NC_150 = total_time_home_NC_150 + total_time_second_NC_150 +
total_time_third_NC_150
if(out_third_NC_150 > 0 || out_second_NC_150 > 0 || out_catcher_NC_150 > 0)
  Average_Time_NC_150 = round((Total_Time_NC_150/(out_third_NC_150 +
out_second_NC_150 + out_catcher_NC_150)), 2)
```

```
Total_Time_NC_200 = total_time_home_NC_200 + total_time_second_NC_200 +
total_time_third_NC_200
if(out_third_NC_200 > 0 || out_second_NC_200 > 0 || out_catcher_NC_200 > 0)
  Average_Time_NC_200 = round((Total_Time_NC_200/(out_third_NC_200 +
out_second_NC_200 + out_catcher_NC_200)), 2)

Total_Time_NC_250 = total_time_home_NC_250 + total_time_second_NC_250 +
total_time_third_NC_250
if(out_third_NC_250 > 0 || out_second_NC_250 > 0 || out_catcher_NC_250 > 0)
  Average_Time_NC_250 = round((Total_Time_NC_250/(out_third_NC_250 +
out_second_NC_250 + out_catcher_NC_250)), 2)

Total_Time_NC_300 = total_time_home_NC_300 + total_time_second_NC_300 +
total_time_third_NC_300
if(out_third_NC_300 > 0 || out_second_NC_300 > 0 || out_catcher_NC_300 > 0)
  Average_Time_NC_300 = round((Total_Time_NC_300/(out_third_NC_300 +
out_second_NC_300 + out_catcher_NC_300)), 2)

Total_Time_NC_350 = total_time_home_NC_350 + total_time_second_NC_350 +
total_time_third_NC_350
if(out_third_NC_350 > 0 || out_second_NC_350 > 0 || out_catcher_NC_350 > 0)
  Average_Time_NC_350 = round((Total_Time_NC_350/(out_third_NC_350 +
out_second_NC_350 + out_catcher_NC_350)), 2)
```

```
Total_Time_NC_400 = total_time_home_NC_400 + total_time_second_NC_400 +
total_time_third_NC_400
if(out_third_NC_400 > 0 || out_second_NC_400 > 0 || out_catcher_NC_400 > 0)
  Average_Time_NC_400 = round((Total_Time_NC_400/(out_third_NC_400 +
out_second_NC_400 + out_catcher_NC_400)), 2)

throws_data_NC = throws_data_NC %>% add_row(Throws_Home = throws_home_NC)
throws_data_NC = throws_data_NC %>% add_row(Throws_Second =
throws_second_NC)
throws_data_NC = throws_data_NC %>% add_row(Throws_Third = throws_third_NC)
throws_data_NC = throws_data_NC %>% add_row(Success_Rate_3B =
success_rate_3B_NC)
throws_data_NC = throws_data_NC %>% add_row(Success_Rate_2B =
success_rate_2B_NC)
throws_data_NC = throws_data_NC %>% add_row(Out_Home = out_catcher_NC)
throws_data_NC = throws_data_NC %>% add_row(Average_Time_Home_Outs =
average_timeC_NC)
throws_data_NC = throws_data_NC %>% add_row(Average_Time_Second_Outs =
average_time2B_NC)
throws_data_NC = throws_data_NC %>% add_row(Average_Time_Third_Outs =
average_time3B_NC)
```

```
throws_data_NC = throws_data_NC %>% add_row(Average_Accuracy_Home_Outs =
average_accuracyC_NC)
throws_data_NC = throws_data_NC %>% add_row(Average_Accuracy_Second_Outs =
average_accuracy2B_NC)
throws_data_NC = throws_data_NC %>% add_row(Average_Accuracy_Third_Outs =
average_accuracy3B_NC)
throws_data_NC = throws_data_NC %>% add_row(Average_Time_Outs_150 =
Average_Time_NC_150)
throws_data_NC = throws_data_NC %>% add_row(Average_Time_Outs_200 =
Average_Time_NC_200)
throws_data_NC = throws_data_NC %>% add_row(Average_Time_Outs_250 =
Average_Time_NC_250)
throws_data_NC = throws_data_NC %>% add_row(Average_Time_Outs_300 =
Average_Time_NC_300)
throws_data_NC = throws_data_NC %>% add_row(Average_Time_Outs_350 =
Average_Time_NC_350)
throws_data_NC = throws_data_NC %>% add_row(Average_Time_Outs_400 =
Average_Time_NC_400)



if(out_second_cu > 0 || safe_second_cu > 0) #to avoid dividing by 0
  success_rate_2B = (out_second_cu/(safe_second_cu + out_second_cu)) * 100
```

```
if(out_second_cu_150 > 0 || safe_second_cu_150 > 0)

  success_rate_2B_150 = (out_second_cu_150/(safe_second_cu_150 + out_second_cu_150))

* 100


if(out_second_cu_200 > 0 || safe_second_cu_200 > 0)

  success_rate_2B_200 = (out_second_cu_200/(safe_second_cu_200 + out_second_cu_200))

* 100


if(out_second_cu_250 > 0 || safe_second_cu_250 > 0)

  success_rate_2B_250 = (out_second_cu_250/(safe_second_cu_250 + out_second_cu_250))

* 100


if(out_second_cu_300 > 0 || safe_second_cu_300 > 0)

  success_rate_2B_300 = (out_second_cu_300/(safe_second_cu_300 + out_second_cu_300))

* 100


if(out_second_cu_350 > 0 || safe_second_cu_350 > 0)

  success_rate_2B_350 = (out_second_cu_350/(safe_second_cu_350 + out_second_cu_350))

* 100


if(out_second_cu_400 > 0 || safe_second_cu_400 > 0)
```

```
  success_rate_2B_400 = (out_second_cu_400/(safe_second_cu_400 + out_second_cu_400))
* 100


if(out_second_cu > 0){
  average_time2B = round((total_time_second_cu/out_second_cu), 2) #Measures average
time on throws to second that resulted in outs
  average_cutoff_time2B = round((cutoff_time2B/out_second_cu), 2) #Measures average
time for cutoff man to throw ball to second, but only on plays that resulted in outs
  average_accuracy2B = round((total_accuracy2B_cu/out_second_cu), 2) #Measures
average accuracy per throw to 2B that resulted in an out
}
if(out_second_cu_150 > 0){
  average_time2B_150 = round((total_time_second_cu_150/out_second_cu_150), 2)
  average_cutoff_time2B_150 = round((cutoff_time2B/out_second_cu), 2)
  average_accuracy2B_150 = round((total_accuracy2B_cu_150/out_second_cu_150), 2)
}
if(out_second_cu_200 > 0){
  average_time2B_200 = round((total_time_second_cu_200/out_second_cu_200), 2)
  average_cutoff_time2B_200 = round((cutoff_time2B/out_second_cu), 2)
  average_accuracy2B_200 = round((total_accuracy2B_cu_200/out_second_cu_200), 2)
}
if(out_second_cu_250 > 0){
  average_time2B_250 = round((total_time_second_cu_250/out_second_cu_250), 2)
  average_cutoff_time2B_250 = round((cutoff_time2B/out_second_cu), 2)
  average_accuracy2B_250 = round((total_accuracy2B_cu_250/out_second_cu_250), 2)
}
if(out_second_cu_300 > 0){
  average_time2B_300 = round((total_time_second_cu_300/out_second_cu_300), 2)
  average_cutoff_time2B_300 = round((cutoff_time2B/out_second_cu), 2)
  average_accuracy2B_300 = round((total_accuracy2B_cu_300/out_second_cu_300), 2)
}
if(out_second_cu_350 > 0){
  average_time2B_350 = round((total_time_second_cu_350/out_second_cu_350), 2)
  average_cutoff_time2B_350 = round((cutoff_time2B/out_second_cu), 2)
  average_accuracy2B_350 = round((total_accuracy2B_cu_350/out_second_cu_350), 2)
}
if(out_second_cu_400 > 0){
  average_time2B_400 = round((total_time_second_cu_400/out_second_cu_400), 2)
  average_cutoff_time2B_400 = round((cutoff_time2B/out_second_cu), 2)
  average_accuracy2B_400 = round((total_accuracy2B_cu_400/out_second_cu_400), 2)
}
if(out_third_cu > 0 || safe_third_cu > 0) #to avoid dividing by 0
  success_rate_3B = (out_third_cu/(safe_third_cu + out_third_cu)) * 100
if(out_third_cu_150 > 0 || safe_third_cu_150 > 0)
  success_rate_3B_150 = (out_third_cu_150/(safe_third_cu_150 + out_third_cu_150)) * 100
```

```
if(out_third_cu_200 > 0 || safe_third_cu_200 > 0)
  success_rate_3B_200 = (out_third_cu_200/(safe_third_cu_200 + out_third_cu_200)) * 100


if(out_third_cu_250 > 0 || safe_third_cu_250 > 0)
  success_rate_3B_250 = (out_third_cu_250/(safe_third_cu_250 + out_third_cu_250)) * 100


if(out_third_cu_300 > 0 || safe_third_cu_300 > 0)
  success_rate_3B_300 = (out_third_cu_300/(safe_third_cu_300 + out_third_cu_300)) * 100


if(out_third_cu_350 > 0 || safe_third_cu_350 > 0)
  success_rate_3B_350 = (out_third_cu_350/(safe_third_cu_350 + out_third_cu_350)) * 100


if(out_third_cu_400 > 0 || safe_third_cu_400 > 0)
  success_rate_3B_400 = (out_third_cu_400/(safe_third_cu_400 + out_third_cu_400)) * 100


if(out_third_cu > 0){
  average_time3B = round((total_time_third_cu/out_third_cu), 2) #Measures average time
on throws to third that resulted in outs
  average_cutoff_time3B = round((cutoff_time3B/out_third_cu), 2) #Measures average time
for cutoff man to throw ball to third, but only on plays that resulted in outs
  average_accuracy3B = round((total_accuracy3B_cu/out_third_cu), 2) #Measures average
accuracy per throw to 3B that resulted in an out
}
```

```
if(out_third_cu_150 > 0){
  average_time3B_150 = round((total_time_third_cu_150/out_third_cu_150), 2)
  average_cutoff_time3B_150 = round((cutoff_time3B/out_third_cu), 2)
  average_accuracy3B_150 = round((total_accuracy3B_cu_150/out_third_cu_150), 2)
}
if(out_third_cu_200 > 0){
  average_time3B_200 = round((total_time_third_cu_200/out_third_cu_200), 2)
  average_cutoff_time3B_200 = round((cutoff_time3B/out_second_cu), 2)
  average_accuracy3B_200 = round((total_accuracy3B_cu_200/out_third_cu_200), 2)
}
if(out_third_cu_250 > 0){
  average_time3B_250 = round((total_time_third_cu_250/out_third_cu_250), 2)
  average_cutoff_time3B_250 = round((cutoff_time3B/out_second_cu), 2)
  average_accuracy3B_250 = round((total_accuracy3B_cu_250/out_third_cu_250), 2)
}
if(out_third_cu_300 > 0){
  average_time3B_300 = round((total_time_third_cu_300/out_third_cu_300), 2)
  average_cutoff_time3B_300 = round((cutoff_time3B/out_second_cu), 2)
  average_accuracy3B_300 = round((total_accuracy3B_cu_300/out_third_cu_300), 2)
}
if(out_third_cu_350 > 0){
  average_time3B_350 = round((total_time_third_cu_350/out_third_cu_350), 2)
  average_cutoff_time3B_350 = round((cutoff_time3B/out_second_cu), 2)
  average_accuracy3B_350 = round((total_accuracy3B_cu_350/out_third_cu_350), 2)
}
if(out_third_cu_400 > 0){
  average_time3B_400 = round((total_time_third_cu_400/out_third_cu_400), 2)
  average_cutoff_time3B_400 = round((cutoff_time3B/out_second_cu), 2)
  average_accuracy3B_400 = round((total_accuracy3B_cu_400/out_third_cu_400), 2)
}

if(out_home_cu > 0){
  average_timeC = round((total_time_home_cu/out_home_cu), 2) #Measures average time on throws to home that resulted in outs
  average_cutoff_timeC = round((cutoff_timeC/out_home_cu), 2) #Measures average time for cutoff man to throw ball to home, but only on plays that resulted in outs
  average_accuracyC = round((total_accuracyC_cu/out_home_cu), 2) #Measures average accuracy per throw to home that resulted in an out
}
if(out_home_cu_150 > 0){
  average_timeC_150 = round((total_time_home_cu_150/out_home_cu_150), 2)
  average_cutoff_timeC_150 = round((cutoff_timeC/out_home_cu), 2)
  average_accuracyC_150 = round((total_accuracyC_cu_150/out_home_cu_150), 2)
}
if(out_home_cu_200 > 0){
  average_timeC_200 = round((total_time_home_cu_200/out_home_cu_200), 2)
  average_cutoff_timeC_200 = round((cutoff_timeC/out_home_cu), 2)
  average_accuracyC_200 = round((total_accuracyC_cu_200/out_home_cu_200), 2)
}
if(out_home_cu_250 > 0){
  average_timeC_250 = round((total_time_home_cu_250/out_home_cu_250), 2)
  average_cutoff_timeC_250 = round((cutoff_timeC/out_home_cu), 2)
  average_accuracyC_250 = round((total_accuracyC_cu_250/out_home_cu_250), 2)
}
if(out_home_cu_300 > 0){
  average_timeC_300 = round((total_time_home_cu_300/out_home_cu_300), 2)
  average_cutoff_timeC_300 = round((cutoff_timeC/out_home_cu), 2)
  average_accuracyC_300 = round((total_accuracyC_cu_300/out_home_cu_300), 2)
}
if(out_home_cu_350 > 0){
  average_timeC_350 = round((total_time_home_cu_350/out_home_cu_350), 2)
  average_cutoff_timeC_350 = round((cutoff_timeC/out_home_cu), 2)
  average_accuracyC_350 = round((total_accuracyC_cu_350/out_home_cu_350), 2)
}
if(out_home_cu_400 > 0){
  average_timeC_400 = round((total_time_home_cu_400/out_home_cu_400), 2)
  average_cutoff_timeC_400 = round((cutoff_timeC/out_home_cu), 2)
  average_accuracyC_400 = round((total_accuracyC_cu_400/out_home_cu_400), 2)
}
```

```
Total_Time_cu = total_time_home_cu + total_time_second_cu + total_time_third_cu
if(out_third_cu > 0 || out_second_cu > 0 || out_home_cu > 0) #to avoid dividing by 0
  Average_Time_cu = round((Total_Time_cu/(out_third_cu + out_second_cu +
out_home_cu)), 2)

Total_Time_cu_150 = total_time_home_cu_150 + total_time_second_cu_150 +
total_time_third_cu_150
if(out_third_cu_150 > 0 || out_second_cu_150 > 0 || out_home_cu_150 > 0)
  Average_Time_cu_150 = round((Total_Time_cu_150/(out_third_cu_150 +
out_second_cu_150 + out_home_cu_150)), 2)

Total_Time_cu_200 = total_time_home_cu_200 + total_time_second_cu_200 +
total_time_third_cu_200
if(out_third_cu_200 > 0 || out_second_cu_200 > 0 || out_home_cu_200 > 0)
  Average_Time_cu_200 = round((Total_Time_cu_200/(out_third_cu_200 +
out_second_cu_200 + out_home_cu_200)), 2)

Total_Time_cu_250 = total_time_home_cu_250 + total_time_second_cu_250 +
total_time_third_cu_250
if(out_third_cu_250 > 0 || out_second_cu_250 > 0 || out_home_cu_250 > 0)
  Average_Time_cu_250 = round((Total_Time_cu_250/(out_third_cu_250 +
out_second_cu_250 + out_home_cu_250)), 2)
```

```
Total_Time_cu_300 = total_time_home_cu_300 + total_time_second_cu_300 +
total_time_third_cu_300
if(out_third_cu_300 > 0 || out_second_cu_300 > 0 || out_home_cu_300 > 0)
  Average_Time_cu_300 = round((Total_Time_cu_300/(out_third_cu_300 +
out_second_cu_300 + out_home_cu_300)), 2)

Total_Time_cu_350 = total_time_home_cu_350 + total_time_second_cu_350 +
total_time_third_cu_350
if(out_third_cu_350 > 0 || out_second_cu_350 > 0 || out_home_cu_350 > 0)
  Average_Time_cu_350 = round((Total_Time_cu_350/(out_third_cu_350 +
out_second_cu_350 + out_home_cu_350)), 2)

Total_Time_cu_400 = total_time_home_cu_400 + total_time_second_cu_400 +
total_time_third_cu_400
if(out_third_cu_400 > 0 || out_second_cu_400 > 0 || out_home_cu_400 > 0)
  Average_Time_cu_400 = round((Total_Time_cu_400/(out_third_cu_400 +
out_second_cu_400 + out_home_cu_400)), 2)

throws_data_cu = throws_data_cu %>% add_row(Throws_Home = throws_home_cu)
throws_data_cu = throws_data_cu %>% add_row(Throws_Second = throws_second_cu)
throws_data_cu = throws_data_cu %>% add_row(Throws_Third = throws_third_cu)
throws_data_cu = throws_data_cu %>% add_row(Success_Rate_3B = success_rate_3B)
throws_data_cu = throws_data_cu %>% add_row(Success_Rate_2B = success_rate_2B)
throws_data_cu = throws_data_cu %>% add_row(Out_Home = out_home_cu)
throws_data_cu = throws_data_cu %>% add_row(Average_Time_Home_Outs =
average_timeC)
throws_data_cu = throws_data_cu %>% add_row(Average_Time_Second_Outs =
average_time2B)
throws_data_cu = throws_data_cu %>% add_row(Average_Time_Third_Outs =
average_time3B)
throws_data_cu = throws_data_cu %>% add_row(Average_Accuracy_Home_Outs =
average_accuracyC)
throws_data_cu = throws_data_cu %>% add_row(Average_Accuracy_Second_Outs =
average_accuracy2B)
throws_data_cu = throws_data_cu %>% add_row(Average_Accuracy_Third_Outs =
average_accuracy3B)
throws_data_cu = throws_data_cu %>% add_row(Average_Time_Outs_150 =
Average_Time_cu_150)
throws_data_cu = throws_data_cu %>% add_row(Average_Time_Outs_200 =
Average_Time_cu_200)
throws_data_cu = throws_data_cu %>% add_row(Average_Time_Outs_250 =
Average_Time_cu_250)
throws_data_cu = throws_data_cu %>% add_row(Average_Time_Outs_300 =
Average_Time_cu_300)
```

```
throws_data_cu = throws_data_cu %>% add_row(Average_Time_Outs_350 =
Average_Time_cu_350)
throws_data_cu = throws_data_cu %>% add_row(Average_Time_Outs_400 =
Average_Time_cu_400)

print("Non-Cutoff Throws:")
print(paste("The number of throws home: ", throws_home_NC))
print(paste("Number of confirmed outs at home: ", out_catcher_NC))
print(paste("The average throw time to home where the throw got the runner out: ",
average_timeC_NC, "seconds"))
print(paste("The average distance off from home where the throw got the runner out: ",
average_accuracyC_NC, "feet"))
cat("\n")
print(paste("The number of throws home on throws between 150 feet and 199.99 feet
where the runner was confirmed to be out or safe: ", throws_home_NC_150))
print(paste("Number of confirmed outs at home on throws between 150 feet and 199.99
feet: ", out_catcher_NC_150))
print(paste("The average throw time to home where the throw got the runner out and the
throw was between 150 feet and 199.99 feet: ", average_timeC_NC_150, "seconds"))
print(paste("The average distance off from home where the throw got the runner out and
the throw was between 150 feet and 199.99 feet: ", average_accuracyC_NC_150, "feet"))
cat("\n")
```

```
print(paste("The number of throws home on throws between 200 feet and 249.99 feet
where the runner was confirmed to be out or safe: ", throws_home_NC_200))
print(paste("Number of confirmed outs at home on throws between 200 feet and 249.99
feet: ", out_catcher_NC_200))
print(paste("The average throw time to home where the throw got the runner out and the
throw was between 200 feet and 249.99 feet: ", average_timeC_NC_200, "seconds"))
print(paste("The average distance off from home where the throw got the runner out and
the throw was between 200 feet and 249.99 feet: ", average_accuracyC_NC_200, "feet"))
cat("\n")
print(paste("The number of throws home on throws between 250 feet and 299.99 feet
where the runner was confirmed to be out or safe: ", throws_home_NC_250))
print(paste("Number of confirmed outs at home on throws between 250 feet and 299.99
feet: ", out_catcher_NC_250))
print(paste("The average throw time to home where the throw got the runner out and the
throw was between 250 feet and 299.99 feet: ", average_timeC_NC_250, "seconds"))
print(paste("The average distance off from home where the throw got the runner out and
the throw was between 250 feet and 299.99 feet: ", average_accuracyC_NC_250, "feet"))
cat("\n")
print(paste("The number of throws home on throws between 300 feet and 349.99 feet
where the runner was confirmed to be out or safe: ", throws_home_NC_300))
print(paste("Number of confirmed outs at home on throws between 300 feet and 349.99
feet: ", out_catcher_NC_300))
```

```
print(paste("The average throw time to home where the throw got the runner out and the
throw was between 300 feet and 349.99 feet: ", average_timeC_NC_300, "seconds"))
print(paste("The average distance off from home where the throw got the runner out and
the throw was between 300 feet and 349.99 feet: ", average_accuracyC_NC_300, "feet"))
cat("\n")
print(paste("The number of throws home on throws between 350 feet and 399.99 feet
where the runner was confirmed to be out or safe: ", throws_home_NC_350))
print(paste("Number of confirmed outs at home on throws between 350 feet and 399.99
feet: ", out_catcher_NC_350))
print(paste("The average throw time to home where the throw got the runner out and the
throw was between 350 feet and 399.99 feet: ", average_timeC_NC_350, "seconds"))
print(paste("The average distance off from home where the throw got the runner out and
the throw was between 350 feet and 399.99 feet: ", average_accuracyC_NC_350, "feet"))
cat("\n")
print(paste("The number of throws home on throws at least 400 feet where the runner was
confirmed to be out or safe: ", throws_home_NC_400))
print(paste("Number of confirmed outs at home on throws at least 400 feet: ",
out_catcher_NC_400))
print(paste("The average throw time to home where the throw got the runner out and the
throw was at least 400 feet: ", average_timeC_NC_400, "seconds"))
print(paste("The average distance off from home where the throw got the runner out and
the throw was at least 400 feet: ", average_accuracyC_NC_400, "feet"))
cat("\n")
```

```
print(paste("The number of throws to third: ", throws_third_NC))
print(paste("The percentage of outs on throws to third: ", round(success_rate_3B_NC,
2),"%"))
print(paste("The average throw time to third base where the throw got the runner out: ",
average_time3B_NC, "seconds"))
print(paste("The average distance off from third base where the throw got the runner out:
", average_accuracy3B_NC, "feet"))
cat("\n")
print(paste("The number of throws to third between 150 and 199.99 feet where the runner
was confirmed to be out or safe: ", throws_third_NC_150))
print(paste("The percentage of outs on throws to third between 150 and 199.99 feet: ",
round(success_rate_3B_NC_150, 2),"%"))
print(paste("The average throw time to third base where the throw got the runner out and
the throw was between 150 and 199.99 feet: ", average_time3B_NC_150, "seconds"))
print(paste("The average distance from third base where the throw got the runner out and
the throw was between 150 and 199.99 feet: ", average_accuracy3B_NC_150, "feet"))
cat("\n")
print(paste("The number of throws to third between 200 and 249.99 feet where the runner
was confirmed to be out or safe: ", throws_third_NC_200))
print(paste("The percentage of outs on throws to third between 200 and 249.99 feet: ",
round(success_rate_3B_NC_200, 2),"%"))
```

```
print(paste("The average throw time to third base where the throw got the runner out and
the throw was between 200 and 249.99 feet: ", average_time3B_NC_200, "seconds"))
print(paste("The average distance from third base where the throw got the runner out and
the throw was between 200 and 249.99 feet: ", average_accuracy3B_NC_200, "feet"))
cat("\n")
print(paste("The number of throws to third between 250 and 299.99 feet where the runner
was confirmed to be out or safe: ", throws_third_NC_250))
print(paste("The percentage of outs on throws to third between 250 and 299.99 feet: ",
round(success_rate_3B_NC_250, 2),"%"))
print(paste("The average throw time to third base where the throw got the runner out and
the throw was between 250 and 299.99 feet: ", average_time3B_NC_250, "seconds"))
print(paste("The average distance from third base where the throw got the runner out and
the throw was between 250 and 299.99 feet: ", average_accuracy3B_NC_250, "feet"))
cat("\n")
print(paste("The number of throws to third between 300 and 349.99 feet where the runner
was confirmed to be out or safe: ", throws_third_NC_300))
print(paste("The percentage of outs on throws to third between 300 and 349.99 feet: ",
round(success_rate_3B_NC_300, 2),"%"))
print(paste("The average throw time to third base where the throw got the runner out and
the throw was between 300 and 349.99 feet: ", average_time3B_NC_300, "seconds"))
print(paste("The average distance from third base where the throw got the runner out and
the throw was between 300 and 349.99 feet: ", average_accuracy3B_NC_300, "feet"))
cat("\n")
```

```
print(paste("The number of throws to third between 350 and 399.99 feet where the runner
was confirmed to be out or safe: ", throws_third_NC_350))
print(paste("The percentage of outs on throws to third between 350 and 399.99 feet: ",
round(success_rate_3B_NC_350, 2),"%"))
print(paste("The average throw time to third base where the throw got the runner out and
the throw was between 350 and 399.99 feet: ", average_time3B_NC_350, "seconds"))
print(paste("The average distance from third base where the throw got the runner out and
the throw was between 350 and 399.99 feet: ", average_accuracy3B_NC_350, "feet"))
cat("\n")
print(paste("The number of throws to third at least 400 feet where the runner was
confirmed to be out or safe: ", throws_third_NC_400))
print(paste("The percentage of outs on throws at least 400 feet: ",
round(success_rate_3B_NC_400, 2),"%"))
print(paste("The average throw time to third base where the throw got the runner out and
the throw was at least 400 feet: ", average_time3B_NC_400, "seconds"))
print(paste("The average distance from third base where the throw got the runner out and
the throw was at least 400 feet: ", average_accuracy3B_NC_400, "feet"))
cat("\n")

print(paste("The number of throws to second: ", throws_second_NC))
print(paste("The percentage of outs on second to second: ", round(success_rate_2B_NC,
2),"%"))
```

```
print(paste("The average throw time to second base where the throw got the runner out: ",
average_time2B_NC, "seconds"))
print(paste("The average distance off from second base where the throw got the runner
out: ", average_accuracy2B_NC, "feet"))
cat("\n")
print(paste("The number of throws to second between 150 and 199.99 feet where the
runner was confirmed to be out or safe: ", throws_second_NC_150))
print(paste("The percentage of outs on throws to second between 150 and 199.99 feet: ",
round(success_rate_2B_NC_150, 2),"%"))
print(paste("The average throw time to second base where the throw got the runner out
and the throw was between 150 and 199.99 feet: ", average_time2B_NC_150, "seconds"))
print(paste("The average distance from second base where the throw got the runner out
and the throw was between 150 and 199.99 feet: ", average_accuracy2B_NC_150, "feet"))
cat("\n")
print(paste("The number of throws to second between 200 and 249.99 feet where the
runner was confirmed to be out or safe: ", throws_second_NC_200))
print(paste("The percentage of outs on throws to second between 200 and 249.99 feet: ",
round(success_rate_2B_NC_200, 2),"%"))
print(paste("The average throw time to second base where the throw got the runner out
and the throw was between 200 and 249.99 feet: ", average_time2B_NC_200, "seconds"))
print(paste("The average distance from second base where the throw got the runner out
and the throw was between 200 and 249.99 feet: ", average_accuracy2B_NC_200, "feet"))
cat("\n")
```

```
print(paste("The number of throws to second between 250 and 299.99 feet where the
runner was confirmed to be out or safe: ", throws_second_NC_250))
print(paste("The percentage of outs on throws to second between 250 and 299.99 feet: ",
round(success_rate_2B_NC_250, 2),"%"))
print(paste("The average throw time to second base where the throw got the runner out
and the throw was between 250 and 299.99 feet: ", average_time2B_NC_250, "seconds"))
print(paste("The average distance from second base where the throw got the runner out
and the throw was between 250 and 299.99 feet: ", average_accuracy2B_NC_250, "feet"))
cat("\n")
print(paste("The number of throws to second between 300 and 349.99 feet where the
runner was confirmed to be out or safe: ", throws_second_NC_300))
print(paste("The percentage of outs on throws to second between 300 and 349.99 feet: ",
round(success_rate_2B_NC_300, 2),"%"))
print(paste("The average throw time to second base where the throw got the runner out
and the throw was between 300 and 349.99 feet: ", average_time2B_NC_300, "seconds"))
print(paste("The average distance from second base where the throw got the runner out
and the throw was between 300 and 349.99 feet: ", average_accuracy2B_NC_300, "feet"))
cat("\n")
print(paste("The number of throws to second between 350 and 399.99 feet where the
runner was confirmed to be out or safe: ", throws_second_NC_350))
print(paste("The percentage of outs on throws to second between 350 and 399.99 feet: ",
round(success_rate_2B_NC_350, 2),"%"))
```

```
print(paste("The average throw time to second base where the throw got the runner out
and the throw was between 350 and 399.99 feet: ", average_time2B_NC_350, "seconds"))
print(paste("The average distance from second base where the throw got the runner out
and the throw was between 350 and 399.99 feet: ", average_accuracy2B_NC_350, "feet"))
cat("\n")
print(paste("The number of throws to second at least 400 feet where the runner was
confirmed to be out or safe: ", throws_second_NC_400))
print(paste("The percentage of outs on throws at least 400 feet: ",
round(success_rate_2B_NC_400, 2),"%"))
print(paste("The average throw time to second base where the throw got the runner out
and the throw was at least 400 feet: ", average_time2B_NC_400, "seconds"))
print(paste("The average distance from second base where the throw got the runner out
and the throw was at least 400 feet: ", average_accuracy2B_NC_400, "feet"))
cat("\n")
print(paste("The average throw time where the throw got the runner out: ",
round(Average_Time_NC, 2), "seconds"))
print(paste("The average throw time where the throw got the runner out on throws
between 150 and 199.99 feet: ", round(Average_Time_NC_150, 2), "seconds"))
print(paste("The average throw time where the throw got the runner out on throws
between 200 and 249.99 feet: ", round(Average_Time_NC_200, 2), "seconds"))
print(paste("The average throw time where the throw got the runner out on throws
between 250 and 299.99 feet: ", round(Average_Time_NC_250, 2), "seconds"))
```

```
print(paste("The average throw time where the throw got the runner out on throws between 300 and 349.99 feet: ", round(Average_Time_NC_300, 2), "seconds"))
print(paste("The average throw time where the throw got the runner out on throws between 350 and 399.99 feet: ", round(Average_Time_NC_350, 2), "seconds"))
print(paste("The average throw time where the throw got the runner out on throws at least 400 feet: ", round(Average_Time_NC_400, 2), "seconds"))
cat("\n")
cat("\n")

print("Cutoff Throws:")
print(paste("The number of throws to home is", throws_home_cu))
print(paste("The average total throws time to home where the throws got the runner out was", average_timeC, "seconds"))
print(paste("The average cutoff throw time to home where the cutoff throw got the runner out was", average_cutoff_timeC, "seconds"))
print(paste("The average distance off from home where the throw got the runner out: ", average_accuracyC, "feet"))
cat("\n")
print(paste("The number of throws home on throws between 150 feet and 199.99 feet where the runner was confirmed to be out or safe: ", throws_home_cu_150))
print(paste("Number of confirmed outs at home on throws between 150 feet and 199.99 feet: ", out_home_cu_150))
print(paste("The average throw time to home where the throw got the runner out and the
throw was between 150 feet and 199.99 feet: ", average_timeC_150, "seconds"))
print(paste("The average distance off from home where the throw got the runner out and
the throw was between 150 feet and 199.99 feet: ", average_accuracyC_150, "feet"))
cat("\n")
print(paste("The number of throws home on throws between 200 feet and 249.99 feet
where the runner was confirmed to be out or safe: ", throws_home_cu_200))
print(paste("Number of confirmed outs at home on throws between 200 feet and 249.99
feet: ", out_home_cu_200))
print(paste("The average throw time to home where the throw got the runner out and the
throw was between 200 feet and 249.99 feet: ", average_timeC_200, "seconds"))
print(paste("The average distance off from home where the throw got the runner out and
the throw was between 200 feet and 249.99 feet: ", average_accuracyC_200, "feet"))
cat("\n")
print(paste("The number of throws home on throws between 250 feet and 299.99 feet
where the runner was confirmed to be out or safe: ", throws_home_cu_250))
print(paste("Number of confirmed outs at home on throws between 250 feet and 299.99
feet: ", out_home_cu_250))
print(paste("The average throw time to home where the throw got the runner out and the
throw was between 250 feet and 299.99 feet: ", average_timeC_250, "seconds"))
print(paste("The average distance off from home where the throw got the runner out and
the throw was between 250 feet and 299.99 feet: ", average_accuracyC_250, "feet"))
cat("\n")
```

```
print(paste("The number of throws home on throws between 300 feet and 349.99 fee where
the runner was confirmed to be out or safe: ", throws_home_cu_300))
print(paste("Number of confirmed outs at home on throws between 300 feet and 349.99
feet: ", out_home_cu_300))
print(paste("The average throw time to home where the throw got the runner out and the
throw was between 300 feet and 349.99 feet: ", average_timeC_300, "seconds"))
print(paste("The average distance off from home where the throw got the runner out and
the throw was between 300 feet and 349.99 feet: ", average_accuracyC_300, "feet"))
cat("\n")
print(paste("The number of throws home on throws between 350 feet and 399.99 feet
where the runner was confirmed to be out or safe: ", throws_home_cu_350))
print(paste("Number of confirmed outs at home on throws between 350 feet and 399.99
feet: ", out_home_cu_350))
print(paste("The average throw time to home where the throw got the runner out and the
throw was between 350 feet and 399.99 feet: ", average_timeC_350, "seconds"))
print(paste("The average distance off from home where the throw got the runner out and
the throw was between 350 feet and 399.99 feet: ", average_accuracyC_350, "feet"))
cat("\n")
print(paste("The number of throws home on throws at least 400 feet where the runner was
confirmed to be out or safe: ", throws_home_cu_400))
print(paste("Number of confirmed outs at home on throws at least 400 feet: ",
out_home_cu_400))
```

```
print(paste("The average throw time to home where the throw got the runner out and the
throw was at least 400 feet: ", average_timeC_400, "seconds"))
print(paste("The average distance off from home where the throw got the runner out and
the throw was at least 400 feet: ", average_accuracyC_400, "feet"))
cat("\n")

print(paste("The number of throws to third is", throws_third_cu))
print(paste("The percentage of outs on throws to third was", round(success_rate_3B,
2),"%"))
print(paste("The average throws time to third base where the throws got the runner out
was", average_time3B, "seconds"))
print(paste("The average cutoff throw time to third base where the cutoff throw got the
runner out was", average_cutoff_time3B, "seconds"))
print(paste("The average distance off from third base where the throw got the runner out:
", average_accuracy3B, "feet"))
cat("\n")
print(paste("The number of throws to third between 150 and 199.99 feet where the runner
was confirmed to be out or safe: ", throws_third_cu_150))
print(paste("The percentage of outs on throws to third between 150 and 199.99 feet: ",
round(success_rate_3B_150, 2),"%"))
print(paste("The average throw time to third base where the throw got the runner out and
the throw was between 150 and 199.99 feet: ", average_time3B_150, "seconds"))
```

```
print(paste("The average distance from third base where the throw got the runner out and the throw was between 150 and 199.99 feet: ", average_accuracy3B_150, "feet"))
cat("\n")
print(paste("The number of throws to third between 200 and 249.99 feet where the runner was confirmed to be out or safe: ", throws_third_cu_200))
print(paste("The percentage of outs on throws to third between 200 and 249.99 feet: ", round(success_rate_3B_200, 2),"%"))
print(paste("The average throw time to third base where the throw got the runner out and the throw was between 200 and 249.99 feet: ", average_time3B_200, "seconds"))
print(paste("The average distance from third base where the throw got the runner out and the throw was between 200 and 249.99 feet: ", average_accuracy3B_200, "feet"))
cat("\n")
print(paste("The number of throws to third between 250 and 299.99 feet where the runner was confirmed to be out or safe: ", throws_third_cu_250))
print(paste("The percentage of outs on throws to third between 250 and 299.99 feet: ", round(success_rate_3B_250, 2),"%"))
print(paste("The average throw time to third base where the throw got the runner out and the throw was between 250 and 299.99 feet: ", average_time3B_250, "seconds"))
print(paste("The average distance from third base where the throw got the runner out and the throw was between 250 and 299.99 feet: ", average_accuracy3B_250, "feet"))
cat("\n")
print(paste("The number of throws to third between 300 and 349.99 feet where the runner was confirmed to be out or safe: ", throws_third_cu_300))
print(paste("The percentage of outs on throws to third between 300 and 349.99 feet: ",
round(success_rate_3B_300, 2),"%"))
print(paste("The average throw time to third base where the throw got the runner out and
the throw was between 300 and 349.99 feet: ", average_time3B_300, "seconds"))
print(paste("The average distance from third base where the throw got the runner out and
the throw was between 300 and 349.99 feet: ", average_accuracy3B_300, "feet"))
cat("\n")
print(paste("The number of throws to third between 350 and 399.99 feet where the runner
was confirmed to be out or safe: ", throws_third_cu_350))
print(paste("The percentage of outs on throws to third between 350 and 399.99 feet: ",
round(success_rate_3B_350, 2),"%"))
print(paste("The average throw time to third base where the throw got the runner out and
the throw was between 350 and 399.99 feet: ", average_time3B_350, "seconds"))
print(paste("The average distance from third base where the throw got the runner out and
the throw was between 350 and 399.99 feet: ", average_accuracy3B_350, "feet"))
cat("\n")
print(paste("The number of throws to third at least 400 feet where the runner was
confirmed to be out or safe: ", throws_third_cu_400))
print(paste("The percentage of outs on throws at least 400 feet: ",
round(success_rate_3B_400, 2),"%"))
print(paste("The average throw time to third base where the throw got the runner out and
the throw was at least 400 feet: ", average_time3B_400, "seconds"))
```

```
print(paste("The average distance from third base where the throw got the runner out and
the throw was at least 400 feet: ", average_accuracy3B_400, "feet"))
cat("\n")

print(paste("The number of throws to second is", throws_second_cu))
print(paste("The percentage of outs on throws to second was", round(success_rate_2B,
2),"%"))
print(paste("The average total throws time to second base where the throws got the runner
out was", average_time2B, "seconds"))
print(paste("The average cutoff throw time to second base where the cutoff throw got the
runner out was", average_cutoff_time2B, "seconds"))
print(paste("The average distance off from second base where the throw got the runner
out: ", average_accuracy2B, "feet"))
cat("\n")
print(paste("The number of throws to second between 150 and 199.99 feet: ",
throws_second_cu_150))
print(paste("The percentage of outs on throws to second between 150 and 199.99 feet: ",
round(success_rate_2B_150, 2),"%"))
print(paste("The average throw time to second base where the throw got the runner out
and the throw was between 150 and 199.99 feet: ", average_time2B_150, "seconds"))
print(paste("The average distance from second base where the throw got the runner out
and the throw was between 150 and 199.99 feet: ", average_accuracy2B_150, "feet"))
cat("\n")
```

```
print(paste("The number of throws to second between 200 and 249.99 feet: ",
throws_second_cu_200))
print(paste("The percentage of outs on throws to second between 200 and 249.99 feet: ",
round(success_rate_2B_200, 2),"%"))
print(paste("The average throw time to second base where the throw got the runner out
and the throw was between 200 and 249.99 feet: ", average_time2B_200, "seconds"))
print(paste("The average distance from second base where the throw got the runner out
and the throw was between 200 and 249.99 feet: ", average_accuracy2B_200, "feet"))
cat("\n")
print(paste("The number of throws to second between 250 and 299.99 feet: ",
throws_second_cu_250))
print(paste("The percentage of outs on throws to second between 250 and 299.99 feet: ",
round(success_rate_2B_250, 2),"%"))
print(paste("The average throw time to second base where the throw got the runner out
and the throw was between 250 and 299.99 feet: ", average_time2B_250, "seconds"))
print(paste("The average distance from second base where the throw got the runner out
and the throw was between 250 and 299.99 feet: ", average_accuracy2B_250, "feet"))
cat("\n")
print(paste("The number of throws to second between 300 and 349.99 feet: ",
throws_second_cu_300))
print(paste("The percentage of outs on throws to second between 300 and 349.99 feet: ",
round(success_rate_2B_300, 2),"%"))
```

```
print(paste("The average throw time to second base where the throw got the runner out
and the throw was between 300 and 349.99 feet: ", average_time2B_300, "seconds"))
print(paste("The average distance from second base where the throw got the runner out
and the throw was between 300 and 349.99 feet: ", average_accuracy2B_300, "feet"))
cat("\n")
print(paste("The number of throws to second between 350 and 399.99 feet: ",
throws_second_cu_350))
print(paste("The percentage of outs on throws to second between 350 and 399.99 feet: ",
round(success_rate_2B_350, 2),"%"))
print(paste("The average throw time to second base where the throw got the runner out
and the throw was between 350 and 399.99 feet: ", average_time2B_350, "seconds"))
print(paste("The average distance from second base where the throw got the runner out
and the throw was between 350 and 399.99 feet: ", average_accuracy2B_350, "feet"))
cat("\n")
print(paste("The number of throws to second at least 400 feet: ", throws_second_cu_400))
print(paste("The percentage of outs on throws at least 400 feet: ",
round(success_rate_2B_400, 2),"%"))
print(paste("The average throw time to second base where the throw got the runner out
and the throw was at least 400 feet: ", average_time2B_400, "seconds"))
print(paste("The average distance from second base where the throw got the runner out
and the throw was at least 400 feet: ", average_accuracy2B_400, "feet"))
cat("\n")
```

~~~r
print(paste("The average throw time where the throw got the runner out: ",
round(Average_Time_cu, 2), "seconds"))
cat("\n")
print(paste("The average throw time where the throw got the runner out on throws
between 150 and 199.99 feet: ", round(Average_Time_cu_150, 2), "seconds"))
print(paste("The average throw time where the throw got the runner out on throws
between 200 and 249.99 feet: ", round(Average_Time_cu_200, 2), "seconds"))
print(paste("The average throw time where the throw got the runner out on throws
between 250 and 299.99 feet: ", round(Average_Time_cu_250, 2), "seconds"))
print(paste("The average throw time where the throw got the runner out on throws
between 300 and 349.99 feet: ", round(Average_Time_cu_300, 2), "seconds"))
print(paste("The average throw time where the throw got the runner out on throws
between 350 and 399.99 feet: ", round(Average_Time_cu_350, 2), "seconds"))
print(paste("The average throw time where the throw got the runner out on throws at least
400 feet: ", round(Average_Time_cu_400, 2), "seconds"))
```
~~~