Evaluating Player Relationships in Stolen Base Defense

2023 SMT Data Challenge Submission

Github Link: kaifranke/SMT-Data-Challenge-2023 (github.com)

Abstract

Catcher defense has been a widely researched topic in the baseball analytics community over the past few years. However, stolen base relationships, and the ability of a catcher to throw out potential base stealers are difficult to study without player tracking data. This project delves into the relationship between the catcher, pitcher, and middle infielder in generating an efficient defense against base-stealers. A GLM was used to predict the probability of a stolen base at four different stages: pitch release, catcher retrieval, catcher release, and middle infielder retrieval. These probabilities can then be used to tell the story of how defensive players combine in their attempt to throw out baserunners. An app was also made for coaches and players to view these relationships in action. Understanding this information can lead to better evaluation of players themselves, but deeper understanding about the mechanics of stolen base defense, and how player chemistry helps in these situations.

Introduction

Catcher defense has been a widely researched topic in the baseball analytics community over the past few years. A catcher's defensive contributions can be broken into three main parts: their ability to "frame" a pitch as a strike, their ability to block a wild pitch, and their ability to stop runners from stealing extra bases. The last piece of catcher defense, stolen base prevention, is a hard topic to approach with current data limitations. Key information isn't publically available, such as the quality of a throw to second base or a pitcher's time to the plate. With the tracking data provided by SMT for this competition, we have the ability to derive all of these features.

Unlike these other pieces of a catcher's skillset, controlling baserunners involves several players whose contributions affect each other in different ways. The pitcher is responsible for holding the baserunner on, the catcher is responsible for a quick and accurate throw to second base, and the middle infielders are responsible for receiving the throw and applying the tag. When building a system for evaluating stolen base defense we need to have the ability to evaluate all of the individual contributions as interactions of these players.

In order to define this relationship and the interaction between defensive players, we split a stolen base into four stages: pitch release, catcher reception and transfer to their throw, catcher throw accuracy, and middle infielder receiving and tagging. At each stage we generate a probability of a successful stolen base, which allows us to see how probability changes at each step and the individual contributions to the defensive effort by each fielder.

Data Cleaning and Manipulation

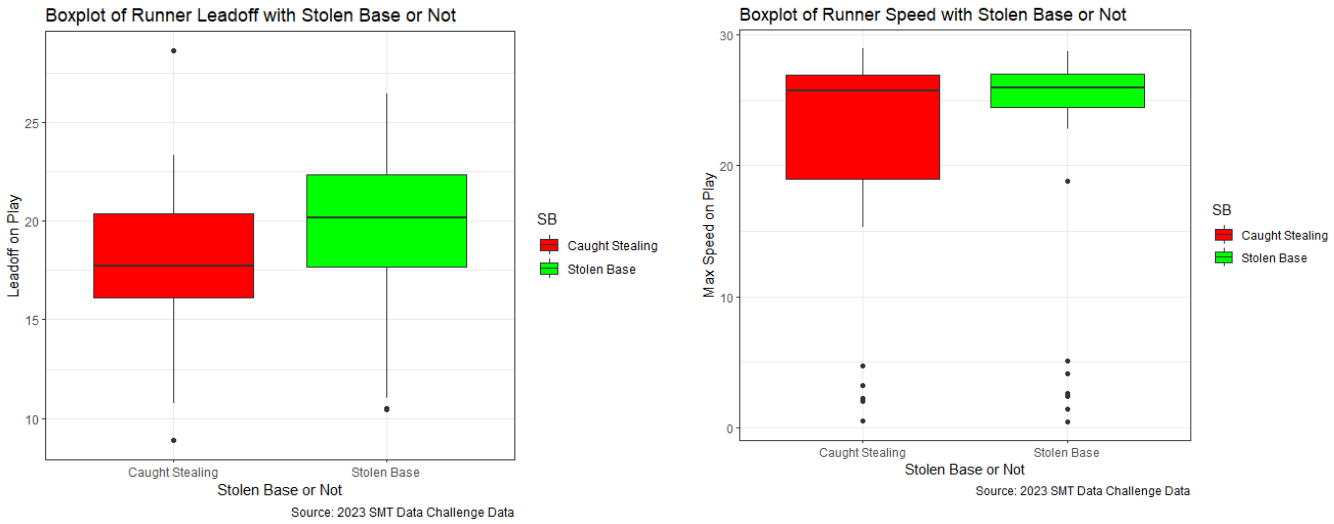
The first stage of a project with big data is being able to take raw data and turn it into a format that is easily interpretable and digestible from the model.

Our first step was to clean and turn the raw tracking data into useful information that is interpretable by coaches and player development. The dataset provided for this project contains over 20,000,000 observations including ball tracking, player tracking, and game information. There are several steps detailed in the appendix on how we made this data more interpretable and easier to work with. The most difficult step in this process was determining if a runner was successful in their stolen base attempt. The dataset did not include information on the outcome of the play, so we had to use information on the status of the next play to determine the result.This also gave us indirect information about things to look out for on the play, such as wild throws or errors where a runner took multiple bases on a stolen base attempt. In total, we found 66 stolen base attempts and 40 total successfully stolen bases.

Data Exploration

The next step in this analysis was to explore how these different features that we engineered relate to base stealing. This not only informs our later modeling decisions, but also allows us to watch out for errors in the data. Below we include some plots created in this initial stage of analysis. We examined many of the features of the dataset and they lined up with prior expectations, but for succinctness we will take a deeper look at two important ones as examples.

The plots below compare the average lead and sprint speed on failed and successful stolen base attempts.



As expected, we can see that if the runner is faster or has a bigger lead, they are more likely to steal successfully. The median leadoff for a successful stolen base is about two feet longer than when runners get caught. Similarly, we can see that there are many more players that fail at stealing a base when they are slower. These two differences make sense in the context of a stolen base and reinforce the quality of our data collection metrics

Model Building

We want to ensure that our model has interpretable results and inputs aimed at coaches and player development. This ensures our tools have value and are useful for both evaluation of results and finding ways to improve players skill sets. In this process, we tried an array of modeling techniques but leaned on those that are more explainable to a non-technical audience. Lastly, and most important to our project is a focus on probabilities themselves rather than outcomes. Many metrics used to evaluate the effectiveness of machine learning models rely on a probability threshold rather than looking at the actual predicted probability itself. This can treat probabilities like .99 and .51 the same, which takes away from the effectiveness of a model. To

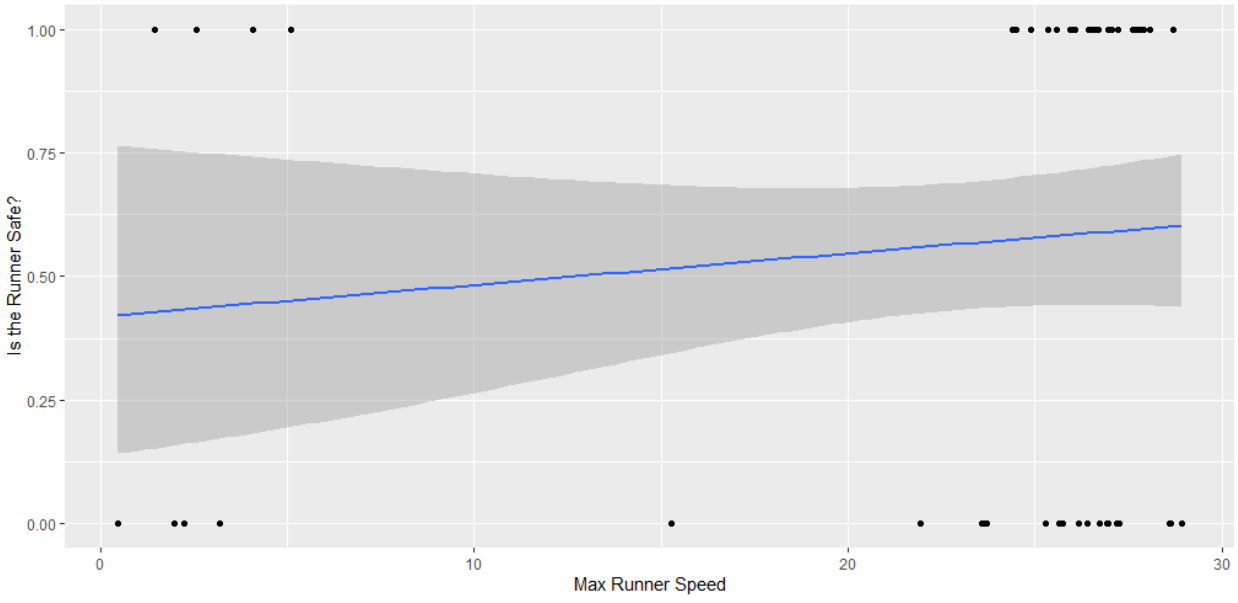
combat this we examined the probabilities in the context of each play, which allowed us to use our baseball intuition to ensure the probabilities are well calibrated.

The stealing process was broken down into four key stages:

1. The pre-pitch stage, which captures the base probability of a runner successfully stealing without any inputs from the interactions between the pitcher, catcher, and fielder.
2. The post-pitch stage, which captures how the time it takes the pitcher to pitch and the pitch location impact the probability of a successful steal.
3. The catcher throw stage, which captures how the catcher's ability to quickly transition to a throw to the fielder impacts the outcome.
4. The fielder catch stage, which captures how the catcher's throw location and fielder's tag of the baserunner impact the outcome of the play in its totality.

To build our probabilities we used a generalized linear model. A GLM allows for us to extract the probabilities of how an individual variable impacts the likelihood of success in conjunction with others and provides easily interpretable coefficients. At each stage of each play we can see exactly how the inputs produce the probabilistic prediction.

For the modeling at right before the point of pitch, the only independent variable used was the maximum speed of the runner on that play. Predictions at this stage can be thought of as the likelihood this runner will successfully steal without any impact of defensive teammate interaction. This stage will allow us to examine how the pitcher then impacts the probability of a successful steal in the next stage by finding the difference and creates a more informed value than just using the league average stolen base percentage. The below plot shows this relationship. As expected, faster runners are harder to throw out.



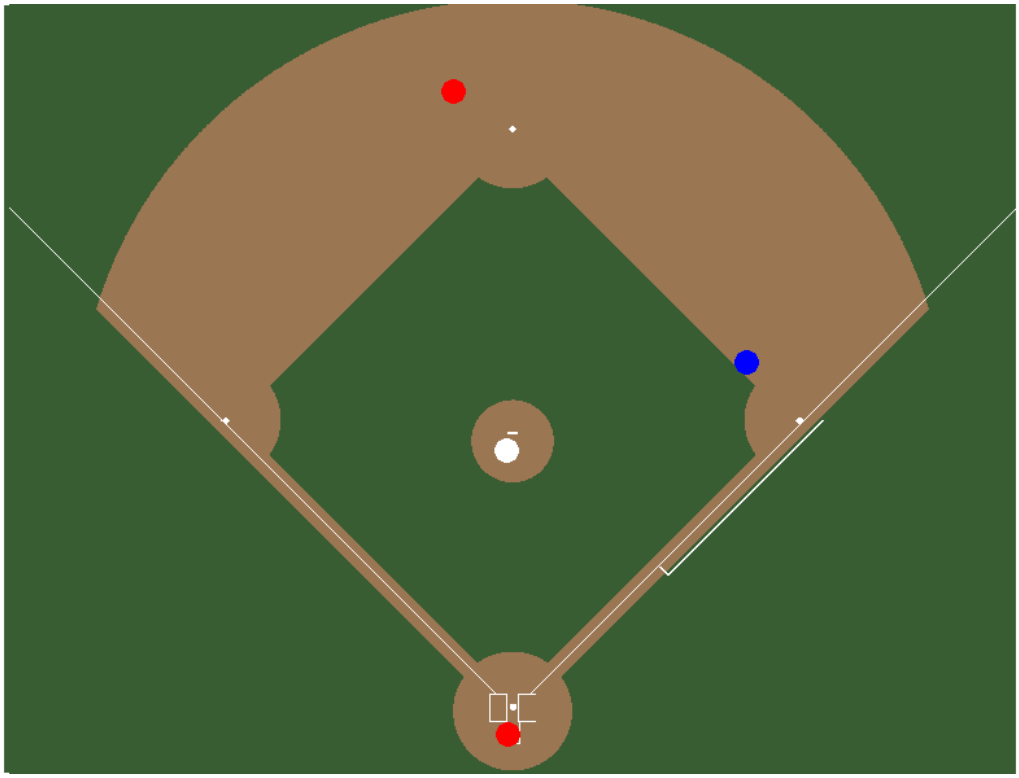
The second model utilizes the max speed variable from the first stage as well as newly included jump, which is the distance between the runner and second base as the ball hits the glove, and ball location. This allows for this model to include both the speed of the pitcher to the plate and the throw quality. These can be taken as the two largest impacts from the pitcher's contributions to the steal. This model had a logloss of .69 using a leave one out cross-validation approach that maximizes our small dataset.

The third model utilizes the second stage's inputs and the time to exchange the ball from catch to throw for the catcher. This stage is meant to emulate how well the catcher can transition from catching the pitch to releasing the ball. Adding exchange improved the model to a logloss of .67 with similar feature importances.

The fourth model utilizes the previous inputs and newly includes the speed of the catcher's throw. This final stage measures the interaction between the catcher's throw and fielder's catch. Since our data did not include play outcomes, we had to evaluate much of our information at the point of catch by the fielder. The tag and receiving ability of the fielder is then inferred as the gap between the predicted probability and the binary 0 or 1 outcome. This final model was tuned with a very low penalty term to push results to the extremes. It had a logloss of .55.

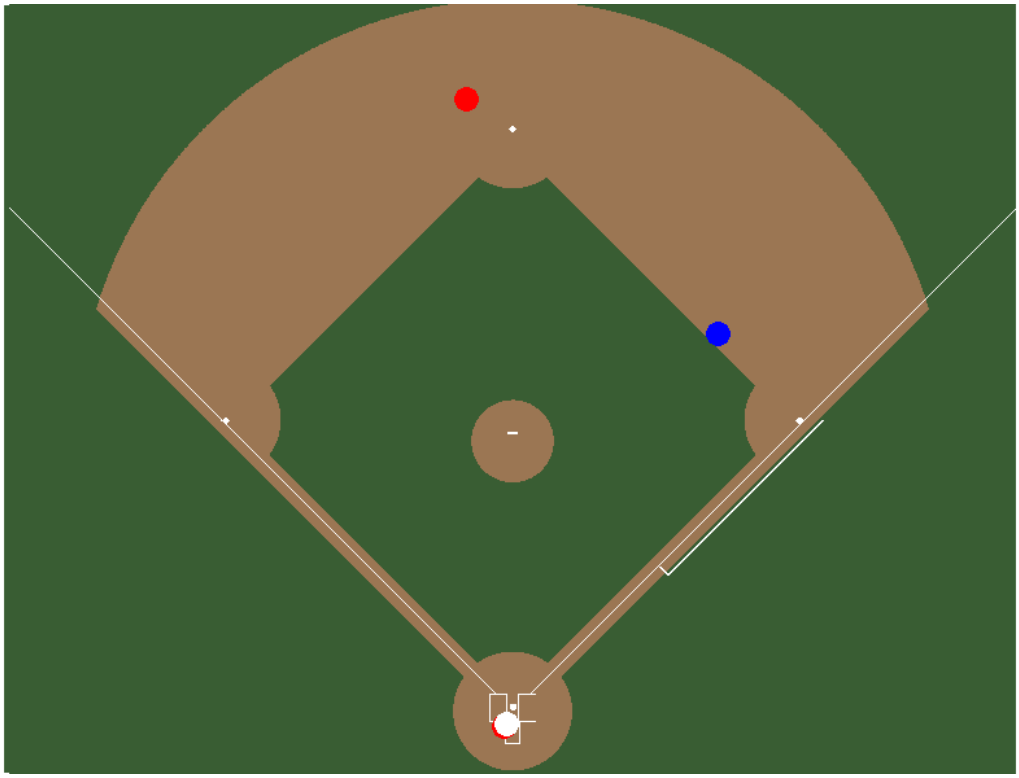
Analysis

In order to analyze our model and our findings, the simplest way is to break down a single play as our model changes and examine the individual impacts. This play will demonstrate the interaction between the pitcher, catcher, and shortstop from the red team as they try to stop a blue team runner from stealing second.



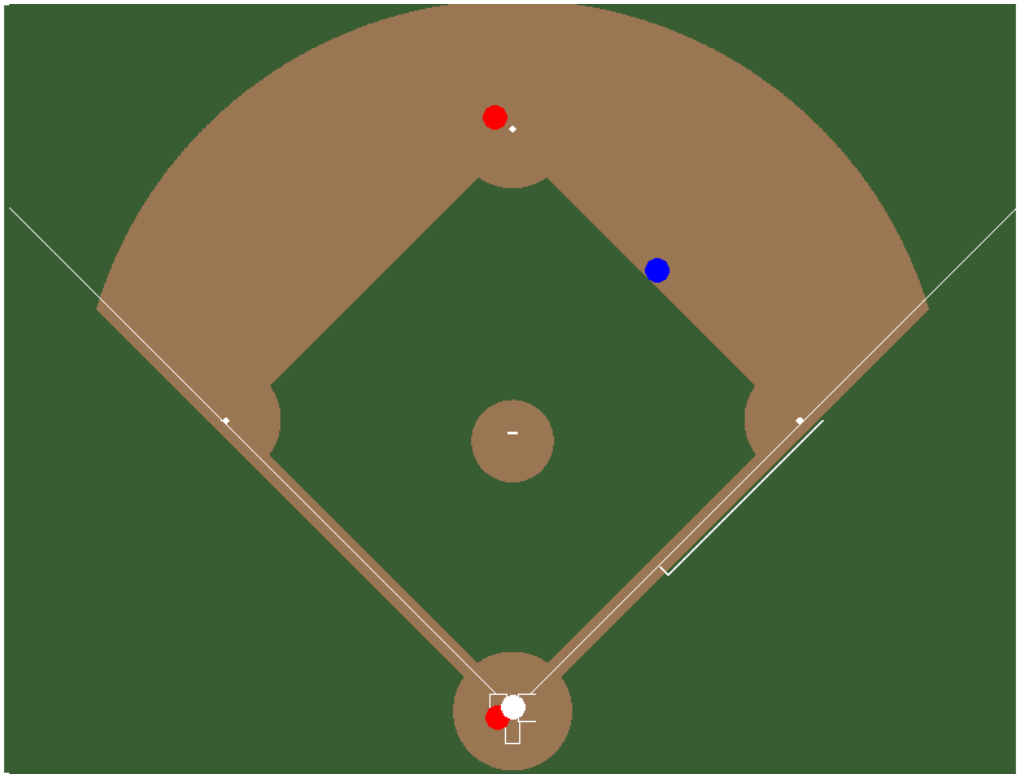
Stage 1: Pre-pitch

This stage is based on the speed of the runner alone. The runner has a max speed of 28.6 feet/second, which gives the runner a 0.6 chance of successfully stealing based on our model for the pre-pitch stage.



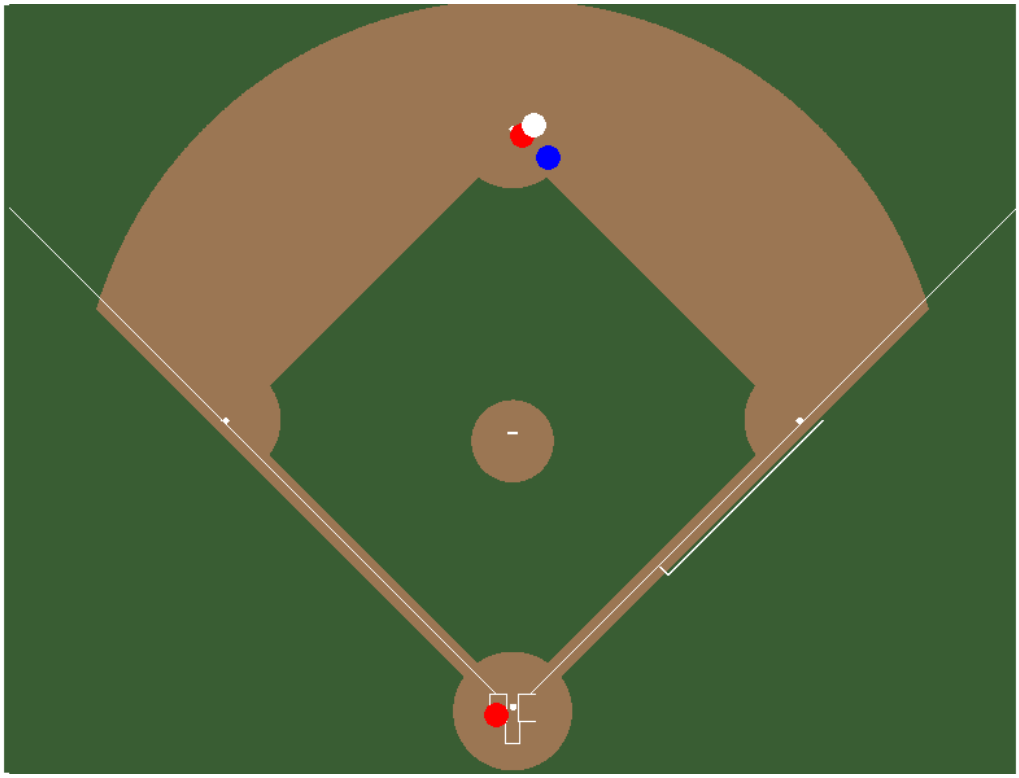
Stage 2: Post-pitch

After the pitch, the race is off. The runner is now only 63 feet from second base, but the pitcher has also done their part in defending the runner: their time to plate was 0.5 seconds. However, their pitch was also low. This bumped the probability of success down to only 0.567, giving the pitcher 0.033 outs of credit.



Stage 3: Catcher Exchange

Next, the catcher had an average exchange, taking 0.8 seconds. The runner is now 44 feet to second, bringing the probability of a successful stolen base slightly up to 0.578, dinging the catcher 0.011 outs of credit.



Stage 4: Catcher Throw and Fielder Tag

Finally, the red catcher had both an accurate and fast throw, getting the ball to the low runner side of the shortstop. This throw, taking only 1.2 seconds, dropped the probability of success all the way down to 0.114 for the runner, giving the catcher's throw 0.464 outs of credit and the catcher about a total of 0.45 outs of credit on the entire play. The tag by the fielder was also a good one, dropping the outcome to a zero chance of success and giving them 0.114 outs of credit.

Through the course of this play, we see the pitcher, catcher, and shortstop work together to bring a 0.6 probability of the runner successfully stealing down to catching the runner. The good throw by the pitcher dropped the probability of success down by 0.033, the catcher's exchange and throw dropped it down by 0.45, and the shortstop's final tag gave them the final 0.114 outs of credit.

Shiny App Shoutout

As a part of this project, we also built an R Shiny app that allows a user to select different model inputs at the four different stages and see how the probabilities change. There are also animations for each of the stolen base plays from the dataset, along with stolen base probabilities at each of the 4 steps in the process we detailed above. A more detailed explanation is provided in the appendix.

Shiny App Link: Stolen Base Defense

Conclusion and Future Work

Through this project, we were able to separate and analyze the individual defensive contributions of teammates in trying to stop a stealing baserunner. By splitting the process into its individual steps rather than treating the stealing process as a single event where all data was considered in one go, we were able to help extract the story of how each player impacted the probability of a baserunner successfully stealing.

As stated throughout, the data limited our ability to study this topic. The biggest issue with this project is the limited dataset we worked with. There were only 66 clean stolen base plays, which severely limits the modeling techniques available to us. While this might mean our immediate results are worse, we believe that our process will scale well to larger datasets.

We believe that while this project was initially designed to revolve around the catcher's contributions to the effort of stopping a steal, it opened the door to evaluating fielder defensive positioning in this context and beginning to evaluate pitcher defensive fielding. With more data, perhaps from including other years or a larger selection of games, along with forthcoming advances in player tracking data collection, we feel as though this could be a major step towards fully mapping defensive contributions of players. By focusing on the interaction between teammates, we've set a stage to measure defensive contributions less on an individual level but on a teammate level. This may allow coaching staffs and front offices to optimize their defenses by utilizing better team chemistry and interactions.

Appendix

- Data Manipulation
    - Position names added instead of position numbers
        - Ex: “SS” used instead of “6”
    - Event codes updated to have actual play descriptions instead of numbers
        - Ex: “pitch” used instead of “1”
    - Data was all joined together by timestamp, game ID, and play ID
    - Figured out if there was runner on first base or second base on the next play
    - Filtered to only have plays where there was no runner on second base with a runner on first
        - This is a situation where a runner could hypothetically steal
    - ‘SB’ column
        - 1 if player stole base
            - If the runner on the next play was a second base or third base, then 1
        - 0 if player was caught or did not steal
            - If runner on the next play was no longer on the base paths or went back to first
    - Added columns showing locations of each of the players during the play
        - Catcher, pitcher, second baseman, shortstop, baserunner
    - Added columns showing x, y, and z locations of the ball during the play
    - Timediff column to show how much time has elapsed since the play started
    - Calculated the distance between the base runner and second and third base
        - Second for a feature in the model
    - Third for a sanity check to see if the data was filtered correctly
- R Shiny App
    - Here is the link to the App once again: Stolen Base Defense
    - Pictures of the Shiny App are on the github if access to the website is limited
    - The 4 models are represented in the app, with user inputs allowed for every feature in each of the 4 models
        - All user inputs are determined via sliding bars with preassigned minima and maxima, along with default starting values for each feature
    - Key Notes
        - 2nd Model
            - The pitch location graphic provides a strike zone as reference, however, the red dot represents where, horizontally and vertically, the catcher catches the ball
        - 3rd Model

            - Likewise to the 2nd model, the throw location is where, horizontally and vertically, the catcher releases the ball (a strike zone is provided as reference)
            - ‘Exchange Time’ is the time from ball retrieval to release point
        - 4th Model
            - ‘Fielder Catch Location’ is a graphic of second based with a view from home plate showing where, horizontally and vertically, the middle infielder catches the ball in relation to second base
            - ‘Catch Depth’ illustrates second base with a view from directly right (first base side) of the base. In order words, a ball to the left of the base is closer to home plate, while a ball to the right of the base is closer to the outfield
    - The second tab animates all of the stolen base plays within the database
        - The white dot is the baseball, the red dots are the defenders, and the blue dot is the runner
        - Users can select the specific game and play they want to see, while also filtering for only stolen base or non-stolen base plays
        - Probabilities of a successful stolen base are provided on the side of the animation using our 4 different models at the 4 distinct times we described
        - Disclaimer: Some plays may take very long to load
        - Disclaimer: Some plays have missing data points, causing ‘jumps’ in the animation