Baseball Umpire Analysis

Project for MILESTONE 1, SIADS 591 & 592

May 27, 2020 - Submitted by Anthony Giove (agiove@umich.edu), Avinash Reddy (avimads@umich.edu), and Ryan Maley (rjmaley@umich.edu).

# MOTIVATION

New York Yankees leads the Atlanta Braves in the world series finals by a run. It’s the 9th innings and the count reads 3 balls and 2 strikes. Garrett Cole throws the pitch. Freddie Freeman does not swing, and the umpire calls it a strike and New York Yankees with the World Series. What if the umpire had called it a ball?

Our project is motivated by our interest in how umpires affect Major League Baseball (MLB) games. It is our hypothesis that game situations and batter characteristics affect umpire’s definition of a strike zone. We are not aware of other research specifically addressing individual umpire behavior to the degree we want to explore.

In this project, we explore how an umpire strike zone varies with different parameters. With publicly available data, we explored strike zones for individual umpires, compared this to other umpires, and analyzed umpire behavior relative to individual players and teams.

# DATA SOURCES



# DATA MANIPULATION

All the code was modularized and converted into a package. Below we have listed the process and data manipulation methods used for each source of data. As the data sets are large, we have created a **Sample.ipynb** which is included in the package as an aid to walk through a smaller portion of the data and doesn’t utilize AWS like the rest of the process.

## Pitches Data:

Pitch data was downloaded from [BaseballSavant.com](https://baseballsavant.mlb.com/statcast_search). The site holds information of every single pitch recorded in MLB history. The dataset includes a number of other information including pitch coordinates, pitcher ID, batter ID, pitch result, etc.

Challenges:

* The size of individual CSV downloads is limited to approximately 40,000 records.
* The web site performance is variable and a full data file cannot be reliably downloaded in a single instance.
* The actual baseball season start and end dates are variable.
* Given the amount of required data, manually downloading would be onerous.

A program was created to automate the download process (see **PitchData.ipynb**). The process was:

1. A data sample CSV was downloaded manually.
2. The data sample was used to create the Postgres table pitches with SQL code**.**
3. The URL from the manual download was analyzed and used as a “base URL” with specific download options including a range of dates.
4. A function was written to create a list of target date tuples which would (1) account for variability in the baseball season and (2) cover a small enough range of dates to not overload the source website.
5. The main routine was written to download the data in chunks and add to the Postgres database. This routine
6. Loops through the list of target dates
7. Creates a target URL
8. Uses pandas read\_CSV to download the data
9. Renames two columns to be compatible with Postgres (replaces “.” with “-”).
10. Uses psycopg2 library to connect to AWS Postgres database.
11. Uses pandas to\_CSV to write data to database.

The data was seemingly complete without significant NaN or other issues. Some manipulation and outlier detection was done during the analysis and is discussed below.

## Players Data

The players data was downloaded from [baseballreference.com](https://www.baseball-reference.com/players/b/bryankr01.shtml). The site hosts information about every baseball player ever played in the MLB. Each player is given a page which holds all the information regarding that player such as height, weight, batting side, throwing arm, teams, debut, career statistics. All the information provided was scraped from the web.

Challenges:

* Extracting all the URLs for different players listed in the website.
* Although there is some uniformity in how the information for each player is displayed, there is a lot of variation between players. For example, some players have their agent’s information and some have who have passed away have death information. We discovered players who have just played practice games and never played a regular season game.
* Baseball batters and pitchers have different career statistics measures.

A program was created to automate the whole process. (see **PlayerData.ipynb**). A lot of ‘Try’ and ‘Except’ statements were used to handle various scenarios. The process was:

* Player information is stored alphabetically by individual letter on individual pages. For each letter of the alphabet, a web crawler goes through the appropriate page and collects all the individual player URLs into a list.
* For each player URL, the script queries the URL and extracts relevant information. To solve the variation between different players, the script goes through the player information page and collects all the information into a dictionary. When the dictionary is converted to a dataframe, the missing information is automatically filled as ‘NA’.
* A list of such dictionaries containing data of an individual player is generated and a dataframe is generated.
* The dataframe was modified:
  + To extract birth year, month, height in inches, weight in lbs., country of birth etc.
  + Relevant columns were changed into suitable datatypes like floats for height, career statistics, datetime for columns with date etc.
  + ‘\n’ were replaced with “ ” to solve error caused when the SQL reader reads the information for the exported CSV.
* Created a function to create a SQL table in AWS using this dataframe and its column information.
* The dataframe is exported to CSV and is loaded into the AWS relational Database for further analysis.

## Game Data

The umpire data was downloaded from [retrosheet.com](https://www.retrosheet.org/game.htm). This website contains text files with information regarding all the games played in MLB. Information include home team, away team, attendance, weather, homeplate umpire, etc.

Challenges:

* The required information was spread across a lot of files (~300). Each file had information about all the games played by one team.

A program was created to automate the whole process. (see **GameData.ipynb**). The process was:

* Created a function to go through all the files in the directory and extract relevant file names into a list:
* The script goes through each line in the file and extracts relevant information into a dictionary. Each file contains information of all the games played by one team.
* The dictionaries are converted to a dataframe, and the dataframe is appended with information of all teams in a particular time period.
* The dataframe is modified to include correct datatypes. Using this dataframe and its column information, the SQL table creation function is reused to create a corresponding SQL table in AWS. The dataframe is exported to CSV and is loaded into the AWS relational Database for further analysis. A copy of this CSV is available in the sample data folder provided in the ZIP file.

## Umpire ID and Player ID Data

Umpire ID information was downloaded from [retrosheet.com](https://www.retrosheet.org/retroID.htm). Player ID was downloaded from [SmartFantasyBaseball.com](http://www.SmartFantasyBaseball.com). Umpire ID maps umpire ID information captured in game table to actual umpire names and their debut information. Player ID contains MLBID and baseball reference ID for any player, MLB ID links to pitches data and baseball reference ID links to player data. The player ID data acts as a link between pitches and player data.

A program was created to automate the whole process (see **GameData.ipynb**). The process was:

Umpire ID:

* Read the webpage, go through the text and extract relevant information into a dictionary and convert into dataframe.

Player ID

* Information is directly downloaded as CSV and converted to a dataframe
* Dataframes were modified to include correct datatypes.
* Using these dataframes and its column information, SQL table creation function is reused to create corresponding SQL tables in AWS.
* The dataframes are exported to CSV and is loaded into the AWS relational Database for further analysis. A copy of these CSVs are available in the sample data folder provided in the ZIP file.

## Joining Data Tables

The joining fields are detailed in the Data Sources section of this report.

## Extract, Transform, Load

Because of the size of the data, queries were performing poorly. WE implemented an Extract, Transform, Load (ETL) process to improve performance. The following steps were taken:

* Created all individual details as described above.
* Determined columns most used during the analysis.
* Indexed the columns used in SQL JOIN and WHERE statements.
* Created a new single table called **pitches\_expanded** and performed all queries against it.

Query times were improved by a factor of 10.

# ANALYSIS AND VISUALIZATION

The umpire strike zone has evolved over the years. The current official definition reads “The official strike zone is the area over home plate from the midpoint between a batter's shoulders and the top of the uniform pants -- when the batter is in his stance and prepared to swing at a pitched ball -- and a point just below the kneecap. In order to get a strike call, part of the ball must cross over part of home plate while in the area mentioned above.” [[1]](#footnote-1)

Given the variance inherent in the definition, strike zones are difficult to quantify. We explored various ways to define and analyze a strike zone.

## Defining Strikes

The pitches data contains a “type” column with values for strike, ball, and in-play. There are multiple reasons a pitch may be called a strike which are not related to an umpire’s call (e.g. a foul ball). A function was created to extract information from the description column and classify a pitch into 20 further categories.

## Visualization 1

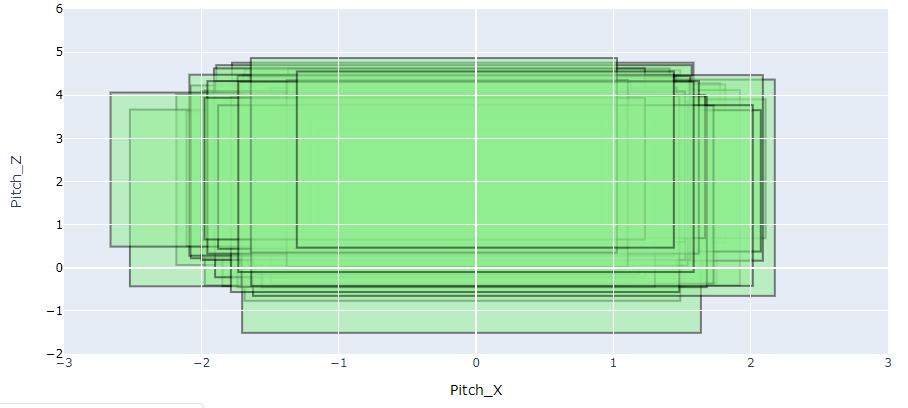
Figure 1: Initial visualization by new pitch types. (see NOTEBOOK SECTION ???).

An initial visualization was created using the new pitch type.

Pitch coordinates in feet relative to home plate are recorded in pitches data and were used to create a scatter plot with pitch type encoded in color.

While this visualization clearly shows the location of pitches, it does not provide enough information to define a strike zone. It also shows the complexity in defining a strike zone for the game of baseball.

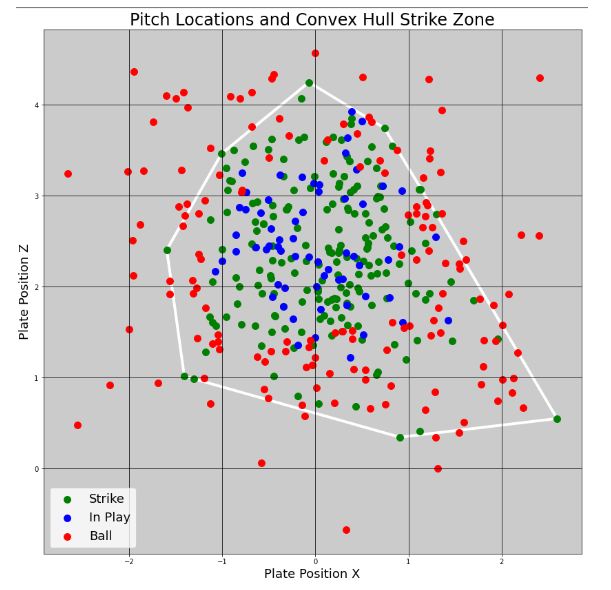
## Visualization 2

After reviewing various options, we decided to limit the analysis to “called strike” and ‘ball’ which are the pitch type over which the umpires have control.

The next visualization attempted to define a strike zone directly by taking the maximum and minimum x and y coordinates per umpire.

Strike zones become clearer, but it is obvious that some zones do not make sense since they are more than two feet from the center of home plate.

Figure 2: Rectangular strike zones defined by maximum and minimum x and y coordinates of strikes. (see NOTEBOOK SECTION ???).

A strike zone is not clear from the strike coordinates alone and, due to some outliers, the strike zone changes drastically.

## Visualization 3

We used the SciPy library to calculate a convex hull based on strike data. The resulting visualization of strike convex hulls, balls, and in-play pitches revealed some interesting patterns.

The convex hull certainly also encloses the all the in-play pitches which matches the intuition that the pitches in the strike zone can reasonably be hit by a batter. However, the area also encloses many balls. The convex hull shape itself does not indicate where an umpire is more likely to call “ball” vs “strike”.

Upon reflection, some variance may be accounted for by stance of the batters (left or right), lack of pitches to certain areas of the expected strike zone, or simply mistaken calls of the umpire.

Figure 3: Convex hull visualization of sample pitches (see NOTEBOOK SECTION ???).

### Visualization 4

Outlier data points were likely an issue and it was decided to remove outlying pitches. Any strike that was 2.2 or more standard deviations from the mean coordinates of the strikes were excluded from the analysis.

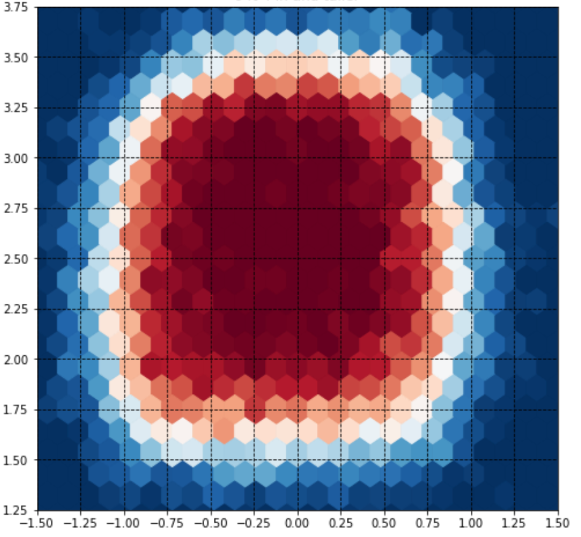
Another set of convex hulls was calculated and a strike zone was visualized for multiple umpires resulting in more realistic visualization.

In Figure 4, please note the significantly overlapping strike zone borders, the common enclosed center of the visualized zone, and border edges that are within 1.5 feet of the center of home plate.

At this point, we believed that we had a reasonably defined zone that could be used for further analysis.

Figure 4: Strike zone convex hulls with outliers excluded for multiple umpires.

### Umpire Strike Zone

It was still difficult to define a proper boundary to differentiate a called strike from a ball. It was clear that there is no perfect boundary. In order to better understand and visualize individual umpire strike zones, the transition of the strike and ball zone needed to be visualized.

For this, the coordinate system was divided into zones/bins and a color was assigned based on how many strikes and pitches were called in that zone/bin.

This changed the perspective of the strike zone from a defined area to how likely a pitch would be called a strike or a ball in a given area.

Figure 5 shows the strike zone for all umpires with left-handed batters. Dark red indicates that pitches were predominantly called as strikes. Dark blue indicates zones where pitches were predominantly called as balls. The color white indicates a zone where pitches were called as balls and strikes in equal proportion. We explored plotting custom zone shapes and colors on Plotly, but found Matplotlib’s performance to be much faster because of integrations with pandas dataframes.

Figure 5: Strike zone hexbin heat map for all umpires with left-handed batters.

Figure 6: Strike zone hexbin heat map for all umpires with left-handed batte

### Umpire Strike Zone Comparison

Once the strike zone visualization was finalized, the modular nature of the code enabled endless comparisons. For instance, an umpire’s zones could be compared for four combinations of batter stand (left vs right) and pitching arm (left vs. right). We compared various conditions.

#### Left and Right-Handed Batter Comparison

Figure 6 shows the strike zone comparisons between left-hand and right-hand batters for all umpires.

A picture containing sitting, white, clock, large

Description automatically generated

Figure 6: All umpire right-hand vs. left hand batter strike zones.

The visualization shows the difference in calls closer to the body as expected for any umpire. The scale at the bottom shows the aggregated color in the axes, which also indicated the shift in strike zone away from the body.

A picture containing red, clock, white

Description automatically generatedA plot was created to highlight just the differences between conditions. Figure 7 shows the areas where the strike zone for a right-handed batter is different than a left-handed batter.

Figure 7: Left-Hand Strike zone and differential plot comparing left-hand to right-hand. See NOTEBOOK SECTION ???).

#### Left and Right-Handed Pitcher Comparison

A picture containing clock

Description automatically generatedFigure 8 shows the strike zone comparisons between left-hand and right-hand pitchers for all umpires. There is a similar pattern of difference to left and right-handed batters, but less pronounced.

Figure 8: Left and right-handed pitcher comparison among all umpires.

#### Player Height Comparison

Figure 9 shows strike zone comparison between players shorter than 5 ft 10 in and taller than 6 ft 2 in for Umpire Joe West. The figure shows the strike zone moves up for a taller person in comparison.

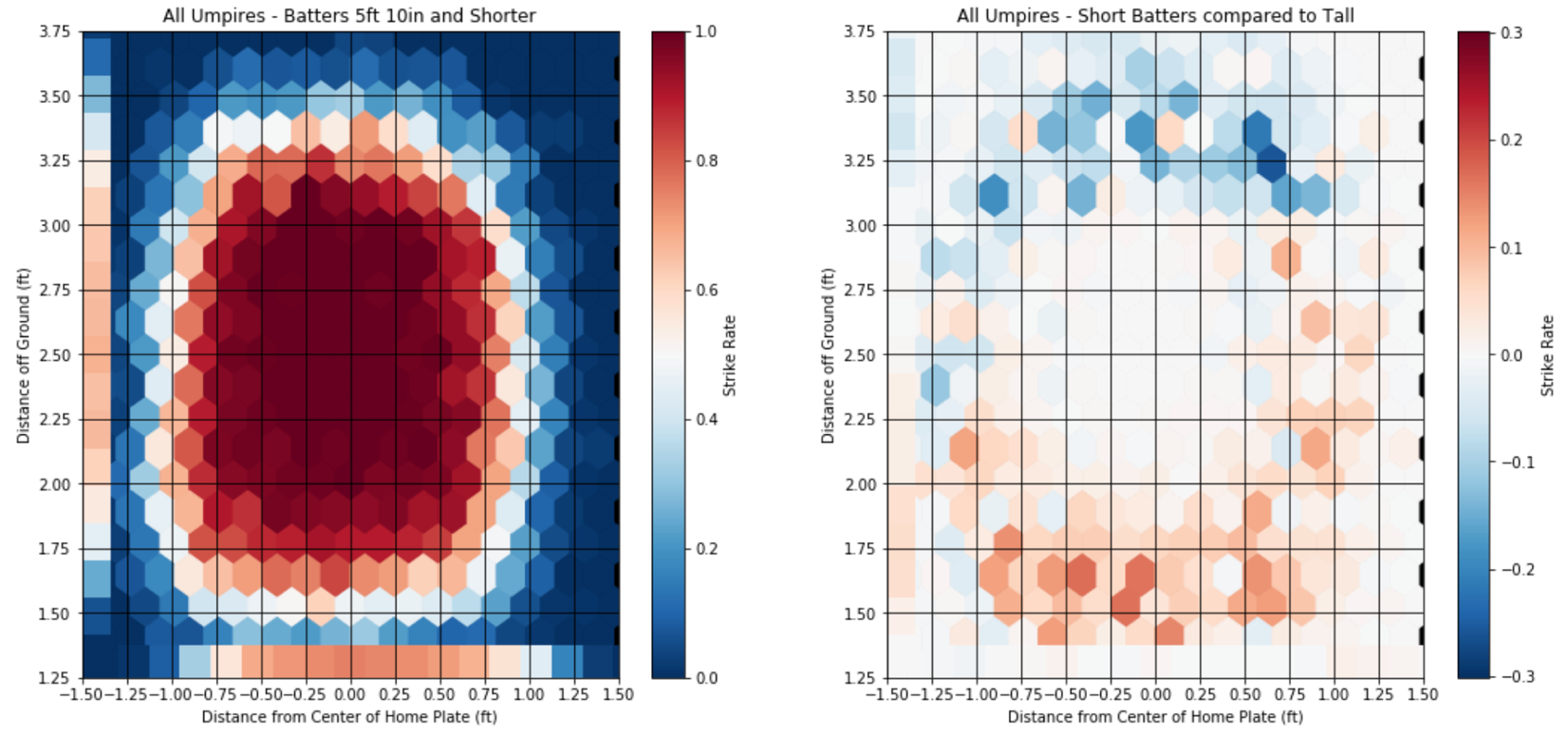


Figure 9: Umpire Joe West strike zones for 5’10” players and 6’2” players.

#### Day / Night Comparison

Figure 10 shows a comparison between call for day and night games. We see a small skewing of balls called at the lower range which may be a result of reduced visibility or shadows nearer the ground. Please note the scale. Even though we can see difference, they are not significant.

A picture containing clock

Description automatically generated

Figure 10: Day and night game pitch call comparison for all umpires.

#### Beginning of Game vs End of Game Comparison

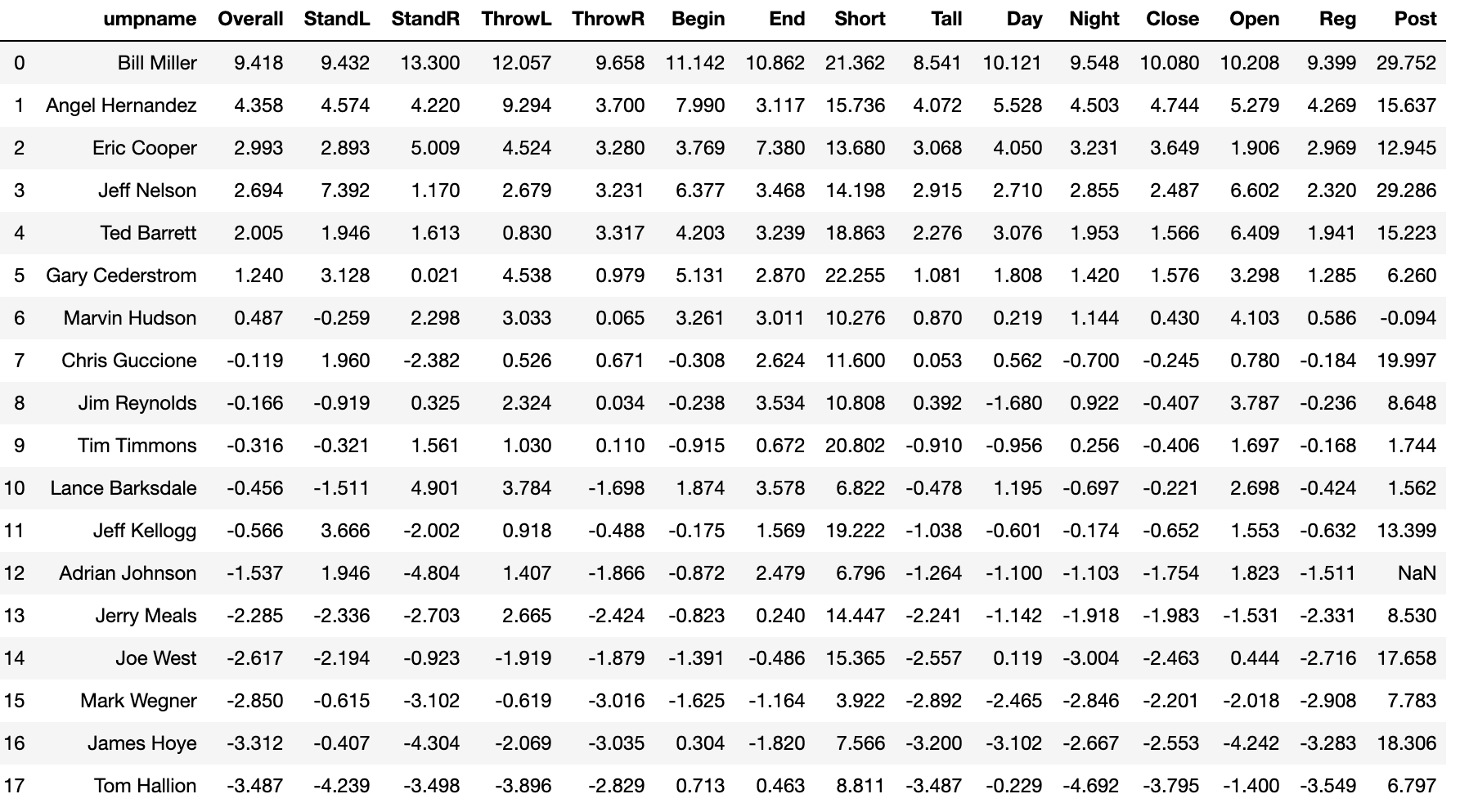
Figure 11 shows a comparison between beginning and ending of games. Similar to the visualization above, while there are detectable differences, they are not large.

A picture containing food, clock

Description automatically generated

Figure 11: Beginning of the game strike zone and difference to end of game zone.

## Umpire Comparison

This table shows all the umpires with over 9,500 called strikes in all investigated scenarios and the percentage difference in called strikes. Positive values indicate higher likelihood of calling strikes (aka aggressive), negative values indicate higher likelihood of call balls (aka conservative). Please see **umpireEDA.ipynb** for details of calculations.

## Aggressive and Conservative Umpires Visualization

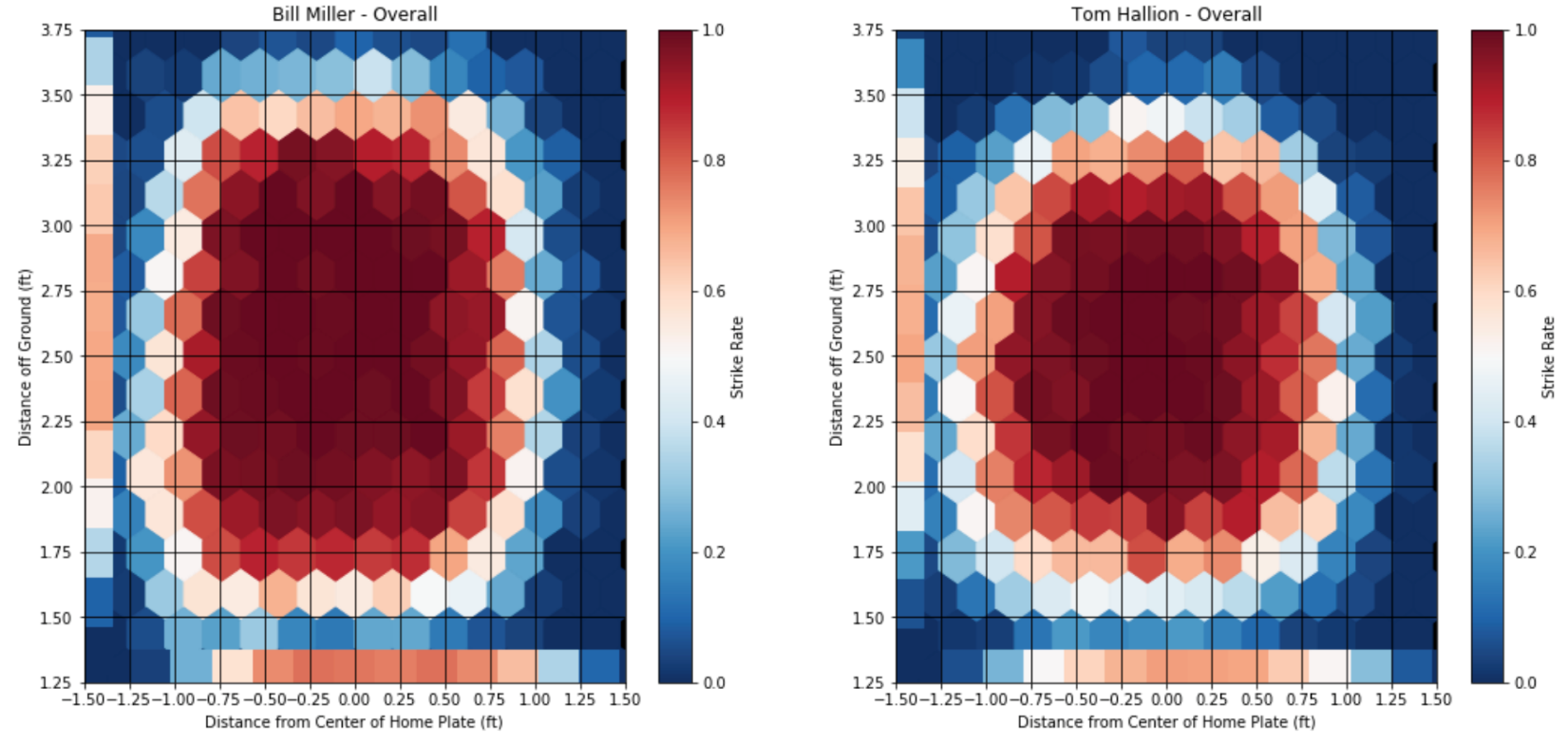
Figure 12 shows the most aggressive and most conservative umpires in baseball. Bill Miller shows the largest strike zone of all umpires in the sample with at least 9,500 strikes.

Figure 12: Bill Miller and Tom Hallion strike zone comparison.

## Aggressive and Conservative Umpires Differences

Figure 13 shows the strike zone for Bill Miller and visualizes differences compared to Tom Hallion. The image on the right illustrates the higher likelihood of Miller calling a strike as compared to Hallion. This illustrates the data shown in the table above by taking the most aggressive and most conservative umpires in the league.

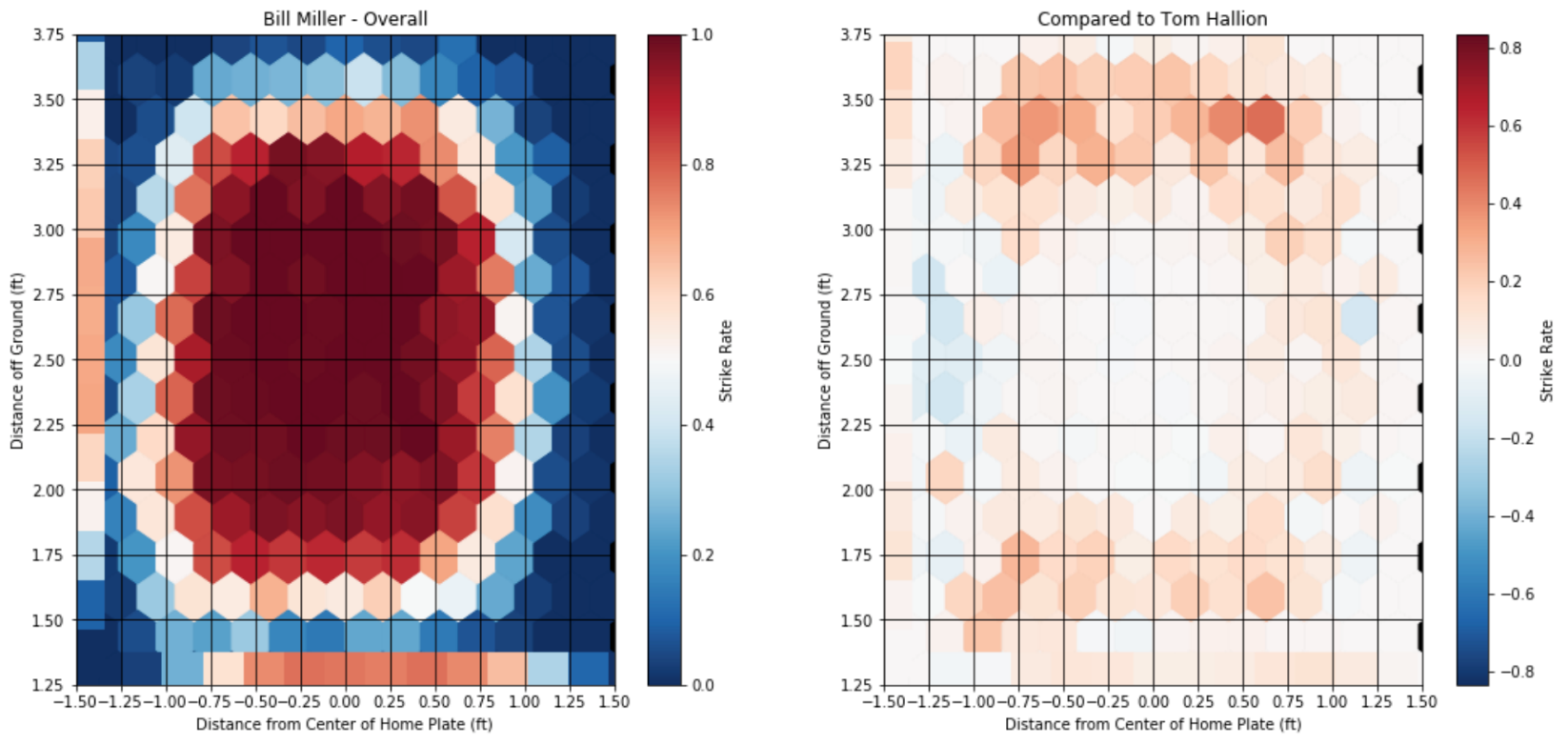


Figure 13: Illustration of differences between most aggressive and most conservative umpires.

# Conclusion

The strike zones do vary based on umpires and different game situations. A pitch called as ball by one umpire could have been called a strike by a different umpire on a different day with a different game situation. The complexity and the variation add to the beauty and excitement to the game.

The function based approach in our analysis gives us opportunity to visually explore how the strike zone varies across a number of situations for a given umpire like, first innings vs final innings, start of the game vs end of the game, left vs right-hand batter, left-hand pitcher vs right-hand batter, etc. The analysis also provides the option to see the difference between any two situations to know which zones specifically are different in the situations and to what degree the difference is. This analysis could be very useful for batters to practice based on the umpire’s history of calling balls or strikes.

As the final piece of the project, we added an option to generate a report for an umpire or umpires. This report provides these visualizations for a number of important parameters. The code allows to remove and add new visualizations based on the requirement. This report answers all the questions we wanted the answers for. Sample reports for the most aggressive and most conservative strike callers in baseball are included here:



# Next Steps

After completing this project, we have discovered some trends that will launch additional exploration. Quantifying the differences certain pitchers and/or hitters see while they are at the plate is one thing that can be investigated further. Creating classification models of umpire types and player types (pitcher and hitter) could result in determining optimal matchups for a given team.

# Statement of Work

Anthony Giove is project lead and a baseball subject matter expert. Avinash Reddy and Anthony will be developing the web scraping application for collecting raw Player and Umpire data. They will also perform data cleaning activity. Ryan Maley is setting up the Postgres database and collecting the Baseball Savant data.

Each team member performed solo initial explorations and discussed the best definitions of strike zones. Anthony created the final visualization routines.

Avinash created the first draft of the report, Ryan performed editing and formatting, and Anthony created the final project deliverable including all code. We were all equally invested in debugging and testing the package.

1. “What Is a Strike Zone?” Major League Baseball. Accessed May 18, 2020. http://m.mlb.com/glossary/rules/strike-zone. [↑](#footnote-ref-1)