Pitch Prediction

Predicting Pitches Thrown by Major League Baseball (MLB) Starting Pitchers Using Machine Learning

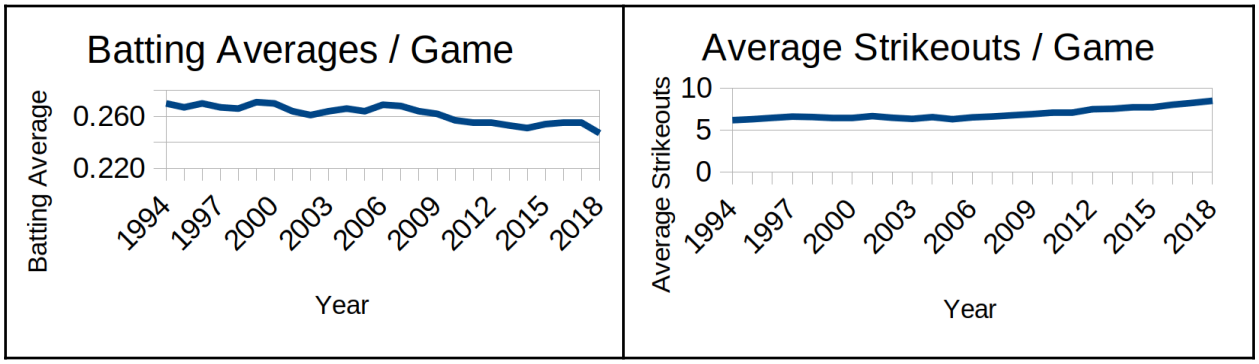
Capstone I Project: Milestone Report
Springboard Data Science Career Track

Mark Rojas
November 2018

Table of Contents:

I. Problem Statement

According to Ted Williams, one of the greatest hitters of all time, "Hitting is the single most difficult thing to do in a sport". Having less than 400 milliseconds to hit a fastball traveling about 95 mph, it should be humanly impossible to hit a baseball. Even the best hitters today average less than 4 hits per 10 at-bats and since 1994, **the average hits per game have decreased while the number of strikeouts per game has increased** [1]. **In addition, the overall attendance and TV ratings per season continue to drop, costing owners money and some employees jobs.**



It appears that improved training methods, technology advancements, and preparation seem to have benefited pitchers more so than the hitters. While hitters can rely on historical data and past experiences of facing a pitcher multiple times to guesstimate the next pitch, this does not take into account the current game-time variables that impact the pitchers decision making, and as the great Yogi Berra once stated, “You can’t think and hit at the same time”.

Because of this, to assist hitters, opposing teams have resorted to stealing signs from the catcher and observing the pitcher, like in poker, looking for telltale signs that could give away the next pitch. For example, in the 2017 Major League Baseball (MLB) World Series between the Houston Astros and Los Angeles Dodgers, the Dodgers starting pitcher for game 3 and the decisive game 7 was Yu Darvish. In both games, Darvish was quickly removed in the 2nd inning after giving up 4 and 5 runs, respectively. Following the series in which the Astros won the best of seven, it was later revealed in a Sports Illustrated article [2] that Darvish was “tipping” his pitches, meaning he was unknowingly telegraphing what pitch he was going to throw next.

The problem is that not every pitcher “tips” their pitches and stealing signs is extremely frowned upon and usually leads to retaliation through hit batters. Thus, the need for new approaches to improving hitters batting averages, providing them real-time updates during the game including the next expected pitch with > 70% accuracy.

1. https://www.baseball-reference.com/leagues/MLB/bat.shtml
2. https://www.si.com/mlb/2017/12/11/dodgers-yu-darvish-tipped-pitches-world-series-astros

II. Data Wrangling

A. Data Collection

In the 2006 postseason (playoffs), Major League Baseball (MLB) first began using a camera-based system to track the trajectory, speed, spin, break, and location of a pitched ball, called **PITCH*f/x***. This information makes it possible to determine the “pitch type” (e.g., Fastball, Cutter, Change-up) of each pitch thrown by the pitcher. Early models for classifying pitch types, however, were not very accurate for pitches similar in speed and break and the data from early years includes many misclassified pitch types. Since 2015, a more advanced system (**Statcast**) is being used which integrates doppler radar with high definition video to track all aspects of a game, including pitches, hits, and players. For the 2017 season, **Trackman**, a component of **Statcast**, officially replaced the previous **PITCH*f/x*** system.

The **Statcast**/**Trackman** pitch data was collected using an application programming interface (**API**) provided by **Sportradar.com**. Access to the MLB feed through standard **HTTP Requests** using **Python 3.6** with a **version 6.5 API Key** includes data for leagues, conferences, teams, games, and players in **JSON** format. MLB utilized the **PITCH*f/x*** system from 2006 – 2014 and then switched to **Statcast** in 2015. Because of the change in systems used to track pitches, I decided I would not consider any data prior to 2015 and for this project, I decided to collect data from the 2016-2018 seasons. Because the **API** is fee-based, I limited the number of starting pitchers of interest to **twenty** (**20**) who were considered “top” pitchers during the **2016**, **2017**, or **2018** seasons. The list of pitchers of interest are:

```
pitchers = ['aaron_nola', 'carlos_carrasco', 'carlos_martinez',
            'chris_archer', 'chris_sale', 'clayton_kershaw',
            'corey_kluber', 'dallas_keuchel', 'david_price',
            'gerrit_cole', 'jacob_degrom', 'jake_arrieta',
            'jose_quintana', 'marcus_stroman', 'justin_verlander',
            'max_scherzer', 'michael_fulmer', 'stephen_strasburg',
            'yu_darvish',  'zack_greinke']
```

Approximately half of the pitchers in the list play for the **National League** while the other half play for the **American League** with some switching teams during the off-season or getting traded mid-season. I also obtained pitching statistics from **Baseball-Reference.com** and **MLB.com** using Web Scraper or simply downloading CSV files directly.

B. Data Cleaning

Cleaning the play-by-play data collected from Sportradar.com was the most challenging and time consuming process, yet it was also the most informative and educational experience gained from this project thus far. Many of the challenges encountered were, I believe, a result of attempting to store too much information in a single file as compared to utilizing a relational database to spread the data across multiple tables. The play-by-play JSON files consists of detailed real-time information on every pitch and game event. When converting the JSON file to a Python Pandas DataFrame to review shape, info, and description, it was plain to see that many of the columns were deeply nested with dictionaries and lists of dictionaries.

The goal for cleaning the play-by-play data was to eliminate redundancy and unnecessary information such as jersey numbers, lineups, warm-ups, and any information that is not related to the pitcher of interest. Using Python, I created functions to iterate through the list of pitchers of interest for each year (2016-2018) and normalize JSON files, convert to a DataFrame, de-nest nested values, drop and rename columns, handle null values and merge cleaned data into a single CSV file for each pitcher.

End result:

- **Sixty (60) cleaned CSV files:** One (1) CSV for every game started by each pitcher (x 20 pitchers) for each season (x3 seasons)

Some challenges encountered during the cleaning process include:

- **Inconsistencies in columns:** Not all JSON files contained the same columns. Some included 'Hitter' and 'Pitcher' columns while others did not. Also, some columns only existed if an event occurred, such as 'Runners' and 'Errors'. These columns may not exist at times if there were no runners on base or no errors committed during the game. To address occasional missing columns, I either drop the columns if not needed, or if needed, I add the columns with null values where not present in the DataFrame.

- **Double-headers:** It is possible for a team to play two (2) games on the same day, referred to as a double-header. To address this, I download and check both games for pitcher of interest. If pitcher of interest did not pitch a game, his ID will not be found and I simply ignore the game and move on.

- **Null Values for Pitch Speeds and Pitch Types:** I found that there are some events where the Pitch Speed and Pitch Type are null. To address Pitch Speeds, I replace the null values with mean() pitch speed for that game. To address Pitch Types, I first check if the pitch outcome was an intentional walk and correctly label the pitch type as 'IB'. For other cases, I label as 'UN' for unknown. For cases where majority or all of the Pitch Speeds or Types are null, I ignore the game to avoid any bias.

C. Data Formatting

Once the collection and cleaning of the data was complete, it was time to perform data wrangling. In other words, transform and map data from the original "raw" data form into another format with the intent of making it more appropriate and valuable for downstream analysis, aka Machine Learning. To achieve this, it was first important to identify variables/features that significantly impact a pitcher's pitch selection. These variables can be separated into two groups:

Past Events

Previous pitch type
Previous pitch velocity
Previous pitch location and movement (as pitch crosses home plate)
Previous pitch outcome (e.g., hit, strike, ball, strikeout, homerun)

Current Game Situation(s)

Current Inning (total of 9 innings in standard game, excluding extra innings)
Pitch count (number of pitches pitcher has thrown to batter, inning, and game)
Hitter count (number of balls and strikes during at-bat)
Score
Runners on base
Weather conditions

While it is important to consider all variables that may impact the pitch type selection, not all data is relevant and due to time constraints, I selected only a handful of variables to utilize in this project. Should time allow and if an additional feature proves statistically significant to analysis, I will expand the number of variables considered in the machine learning approach(es). As of now, the following **Past Events** variables will be considered:

1. Pitch Types (10 different pitch types: 7 are actual pitches, 3 are not)
2. Pitch Outcomes (82 different possible outcomes, but will only consider top 15)

Pitch velocity, *location* and *movement* will be excluded so as to not over-complicate the project by adding additional dimensions to the equation. For **Current Game Situations**, the following variables will be considered in analysis:

3. Inning
4. Pitch Count
5. Hitter Count
6. Score
7. Runners on base

Weather conditions, while can play an important role in pitch selection, will be excluded as well. This is because this information is only available for 2018 and would require more time than

currently available to trace past weather conditions during date and time game was played.

For pitch types, types which are significantly similar in speed and movement were grouped together. For example, Sinkers are grouped with Fastballs, Screwballs are grouped with Curveballs, and Forkballs are grouped with Splitters. Overall, we end up with the following list of pitch types:

FB = Fastball	CB = Curveball	CH = Changeup	SP = Splitter	IB = Intentional Ball
CT = Cutter	SL = Slider	UN = Unknown	PI = Pitchout	KN = Knuckleball

By aggregating the data for each pitcher per year, I was able to compute the number of games started (GS), innings pitched (IP), pitches thrown (Pitches), and their ratios as seen in the plot below:

Pitching Stats / Year

	Year	Name	GS	IP	Pitches	IP/GS	Pitches/GS	Pitches/IP
0	2016	Aaron Nola	20	114	1800	5.700000	90.000000	15.789474
1	2017	Aaron Nola	27	171	2666	6.333333	98.740741	15.590643
2	2018	Aaron Nola	33	217	3217	6.575758	97.484848	14.824885
3	2016	Carlos Carrasco	25	154	2250	6.160000	90.000000	14.610390
4	2017	Carlos Carrasco	32	209	3063	6.531250	95.718750	14.655502
5	2018	Carlos Carrasco	32	212	2975	6.625000	92.968750	14.033019
6	2016	Carlos Martinez	31	199	3031	6.419355	97.774194	15.231156
7	2017	Carlos Martinez	32	212	3138	6.625000	98.062500	14.801887
8	2018	Carlos Martinez	33	230	1978	6.969697	59.939394	8.600000
9	2016	Chris Archer	33	209	3412	6.333333	103.393939	16.325359
10	2017	Chris Archer	34	211	3406	6.205882	100.176471	16.142180
11	2018	Chris Archer	27	154	2509	5.703704	92.925926	16.292208

A majority of the pitch data is ‘non-ordinal’ / ‘categorical’, such as the Hit Type and Outcome ID’s, so it was important that I utilize **one-hot encoding** to format the data so that it could be used in downstream analysis. Below is an example of format conversion:

BEFORE ENCODING:

	pitcher.pitch_type	hit_type	outcome_id
0	FB	NH	kKL
1	FB	NH	kF
2	CB	GB	oGO
3	FB	NH	bB
4	FB	NH	aHR
5	FB	NH	kKL
6	FB	NH	bB
7	CB	NH	kF
8	CB	NH	kF
9	FB	NH	bB
10	FB	GB	oGO
11	FB	NH	bB

AFTER ENCODING:

	pitcher.pitch_type	hitType_label	FB	GB	LD	NH	PU	outcome_label	aBK	aCI	...	oFO	oGO	oKST1	oKST2	oLO	oPO	oROET2	oSB	oSF	oST2
0	FB	3	0.0	0.0	0.0	1.0	0.0	29	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	FB	3	0.0	0.0	0.0	1.0	0.0	27	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	CB	1	0.0	1.0	0.0	0.0	0.0	35	0.0	0.0	...	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	FB	3	0.0	0.0	0.0	1.0	0.0	23	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	FB	3	0.0	0.0	0.0	1.0	0.0	7	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5	FB	3	0.0	0.0	0.0	1.0	0.0	29	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
6	FB	3	0.0	0.0	0.0	1.0	0.0	23	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7	CB	3	0.0	0.0	0.0	1.0	0.0	27	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
8	CB	3	0.0	0.0	0.0	1.0	0.0	27	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
9	FB	3	0.0	0.0	0.0	1.0	0.0	23	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
10	FB	1	0.0	1.0	0.0	0.0	0.0	35	0.0	0.0	...	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
11	FB	3	0.0	0.0	0.0	1.0	0.0	23	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

One concern is that by using **one-hot encoding** to convert Outcome ID's, we essentially added an additional 81 columns (**features**) to consider for each row (**observation**). Because only 1 outcome can occur per pitch, each observation will only have have one outcome with a **1:True value** in the **feature vector**. It may be necessary to exclude outcome's which are not significant to predicting pitch types or are rare occurrences.

III. Exploratory Data Analysis

Using the aggregated data, we observe from the "**Pitching Stats / Year**" table above in Data Formatting section that when healthy, starting pitchers average about 30 starts per year and pitch in approximately 200 innings per year. Not often, a starting pitcher completes a game (pitches 9 innings) and most make it only to the 6th or 7th inning before getting pulled for a relief pitcher. We can also expect starting pitchers to pitch around 100 pitches a game with an average of 15 pitches per inning.

To assure that the 20 selected pitcher's of interest (POI) are a good representation of all pitchers in the MLB, I opted to collect pre-aggregated statistics from https://www.baseball-reference.com/ for all starting pitchers for the past 9 regular seasons (2010-2018), and compare them to the selected 20 POI's. The objective was to show that my 20 selected POI's and other MLB Starters average approximately the same number of Innings pitched per game, Pitches thrown per game, and Pitches thrown per inning. Results shown in Figure 1, indicate that the number of Innings Pitched and Pitches Thrown by my 20 Selected Pitchers of Interest are comparable to those of all starting MLB pitchers for 2016, 2017, and 2018. Also, the line plot indicates that the 20 POI's tend to throw more pitches per game, however, this could also be due to them also pitching in slightly more innings per game on average.

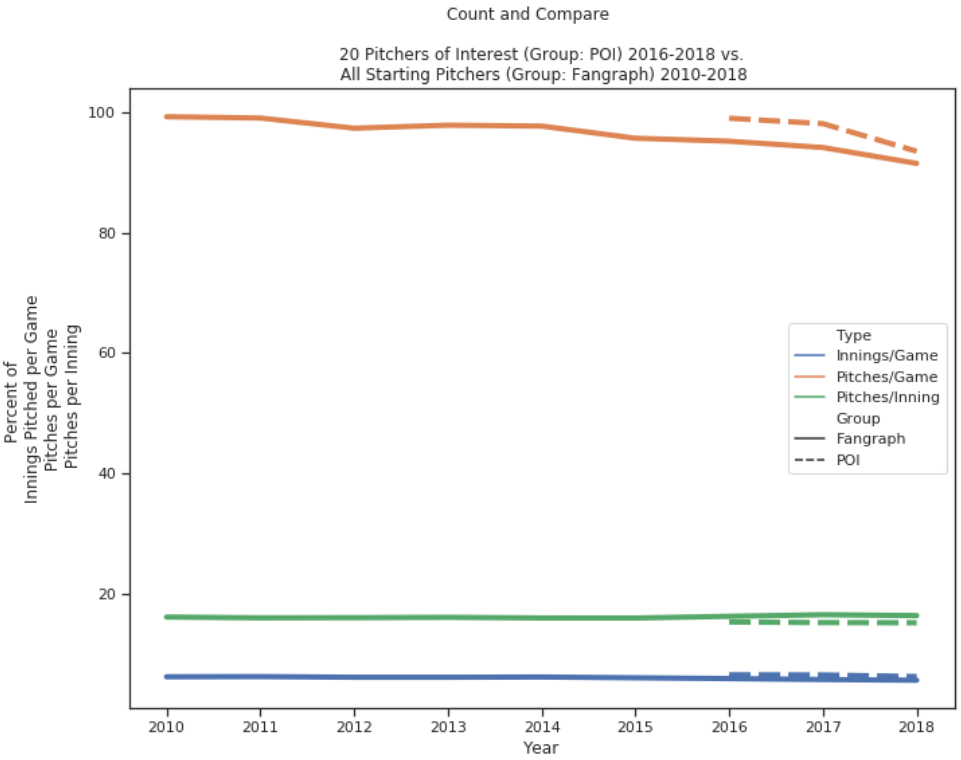


Figure 1: *20 MLB Starting Pitcher's of Interest Statistics vs. All MLB Starting Pitchers*

To identify possible trends in the data, I first took a look at the percent of pitch types thrown per year. I also compared the 20 POI's against the other MLB Starting Pitchers to see if the percent

of pitch types were similar. Results shown in Figure 2.

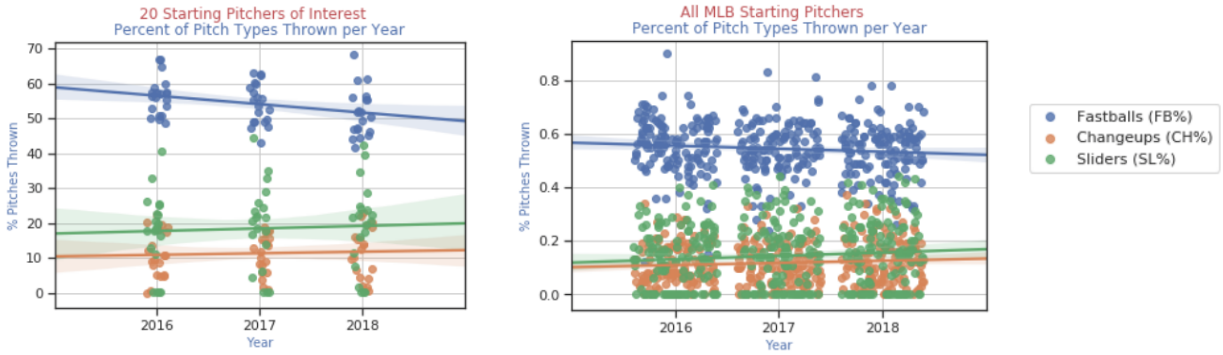


Figure 2: *Percent of Pitch Types Thrown per Year*

What we observe for both groups is the percentage of Fastballs (FB) appear to be decreasing over the past 3 years (2016-2018) and the percentage of Sliders (SL) and Changeups (CH), however, appear to be increasing over the same time span. It is possible that this trend is due to changes/improvements in the pitch classification system. Sliders are breaking pitches that are thrown with less velocity than Fastballs but faster than Curveballs. It is possible that Fastballs and/or Curveballs are now being more accurately classified as Sliders, thus the increase.

Finally, I wanted to also look at the percent frequency of pitch types per inning. Because Fastballs with a higher velocity are more effective at striking out a hitter it makes sense that a pitcher may decrease the number of times they throw a Fastball as the game progresses due to arm fatigue. Figure 3, confirms the percent of Fastballs (FB) thrown decreases as the game progresses. What wasn't expected was that the decrease happens rapidly from inning 1-3 and leveling off around 50% from innings 4-6. There is a little jump at inning 9 which actually makes sense considering if the starting pitcher makes it to this point, they are attempting to complete the game and are probably leading the game in runs, thus a good reason for the pitcher to give it all they got to complete the game.

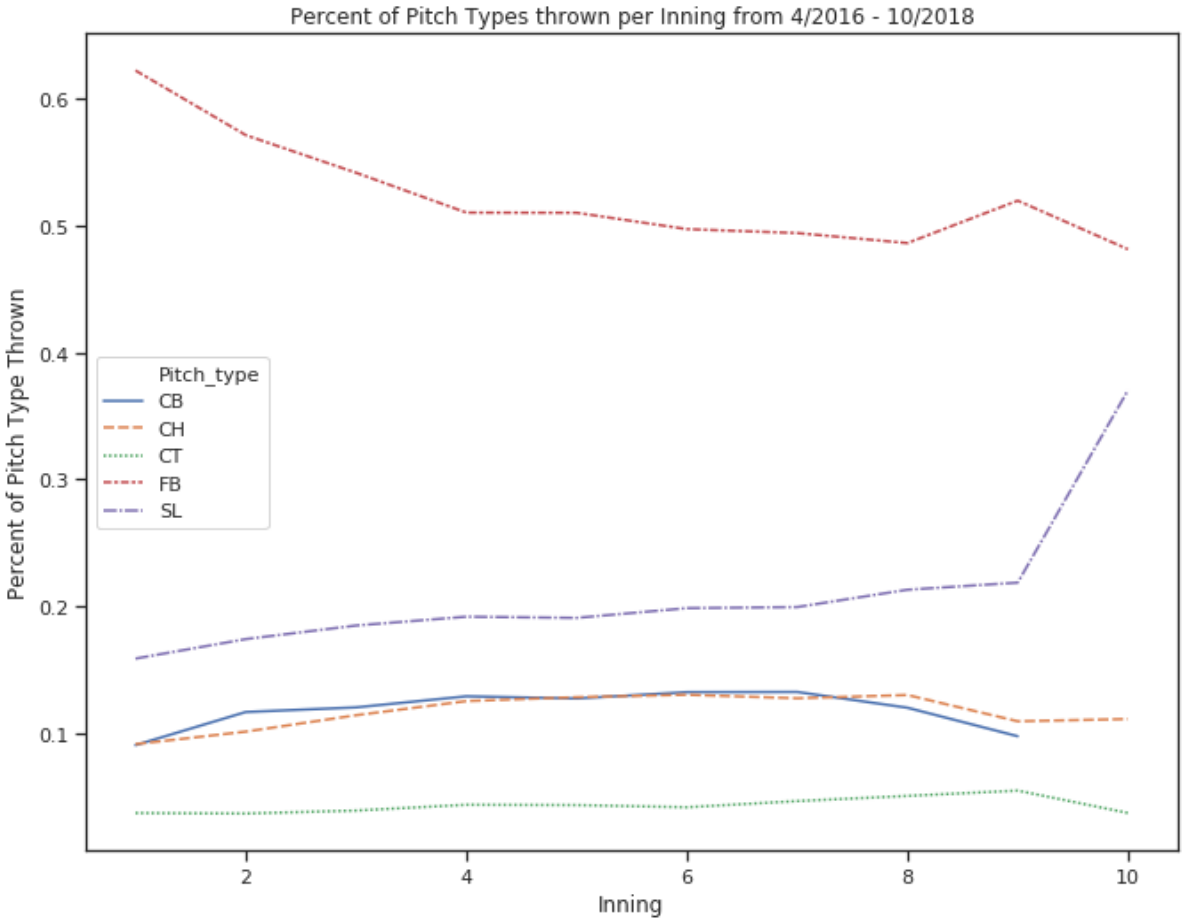


Figure 3: *Percent of Pitch Types Thrown per Inning from 2016 - 2018*

IV. Next Steps

The next steps will be to apply additional hypothesis testing to identify correlations between pitch types and outcomes. It will also be important to eliminate uninformative variables and focus only on those that contribute to predicting the next pitch.

So far, with the exception of looking at percent of Fastball pitches thrown by each pitcher individually, I have analyzed different observations of all pitchers of interest collectively.

It will be interesting to see if predictive models are more successful for different pitchers as well as for different pitch types.