Predi-Pitch: Using Random Forests to Predict Pitch Types

David Scolari, Harrison Snell, Brandon Williams

5/09/2022

Introduction

Hitting a baseball has been shown to be one of the most difficult tasks in all of sports. Anybody who has gone to a batting cage and tried the fastest machine for fun knows how hard it can be to hit a normal fastball. That does not even include the possibilities of breaking balls and off-speed pitches. Batters have a fraction of a fraction of a second to recognize a pitch, determine if that pitch will be a ball or a strike, and set their swing into motion. Considering the human body can only move so fast to get the bat over the plate, the batter has to make their decision moments after the ball leaves the pitcher's hand. To give themselves the best chance of making good contact, professional baseball players will often attempt to predict whether a changeup, fastball, or a curveball is coming before the ball leaves the pitcher's hand.

Hitters will use a variety of information to inform their prediction of an upcoming pitch, including scouting reports that teams build coming into a game, situational context, memory of the previous few pitches, and game specific tendencies of a pitcher. In this project, we attempt to model this decision making process with a random forest pitch type classifier. We use pitch level data from the 2015-2018 seasons to predict pitch types for 30 different pitchers who played during that type period.

Our model aims contributes to the baseball world in the following ways. First, by testing how different sets of features perform at classifying out of sample pitch types, we see that some features are more important for certain pitchers than they are for others. This will help teams focus on the important information when building scouting reports going into a game against a certain pitcher.

Second, we aim to make the results of our pitch type predictor accessible to baseball fans in hopes that it can augment their viewing experience. For many enthusiasts, strategizing along side the players from pitch to pitch, guessing both what the pitcher is going to throw as well as what the batter's strategy will be, is the most enjoyable part of the game. However, among more casual fans, baseball has the reputation for being "boring" to watch, largely due to the pauses between every pitch, during which players sometimes take 30 or more seconds to relay signs and perform ritualistic jersey adjustments, bat taps, and other manor of baseball superstition. To make this aspect of baseball more palatable to fans, we built an R Shiny web application that allows them to input game situations and call our model's pitch type predictions in real time, allowing them to take part in the "game within the game" of baseball.

This report proceeds as follows. First we analyze the physical attributes of the pitches that our 30 selected pitchers throw, making the case that it is indeed in the batters best interest to have an idea of how a particular pitch is going to move once it leave the pitcher's hand. Second, we show some evidence that pitchers do adopt different strategies for different game situations. We then describe the rich, pitch level data set we use for this analysis. Finally, we discuss our random forest classifier and the results it produces.

Reproducibility

While much of our analysis can be reproduced directly from the RMarkdown file, there are several scripts in our Github that are called at various times. In order to have 100% reproducibility, the following load order can be used:

The following scripts allow the writeup to be reproduced: - dugout.R (similar to a include.R) loads the necessary libraries and establishes the file path - **import.R** converts the enormous dataset into a more manageable subset of MLB pitchers - **pitchers.R** pre-processes the data so that it is easily fed into the predipitch script - **predi_pitch.R** takes the subset of pitchers and creates a predictive model for each one (**NOTE:** Be advised, this will take upwards of 30 minutes to run) - performance.R creates a table that shows the performance of the models - kershaw_sequence.R looks at one pitcher and shows sequencing trends throughout the game in different scenarios - situation.R allows the manual entry of at-bat scenarios (balls, strikes, etc.) to create a prediction - predipitch.Rmd creates the final write-up with visualizations

Due to the size of the datasets at hand, running the scripts in **bold** can take a long time. However, they output .RDs which can referenced directly from the Github so you may skip these scripts and run only the other scripts of interest. Just note, dugout.R should be run in either case.

Finally, due to the size of datasets, we gitignore all .csv files. You can find some of the data in archive.zip. However, download pitches.zip from one of the links in the appendix to complete your local repository if you wish to use the import.R script.

Why Do We Care About What Pitch is Coming?

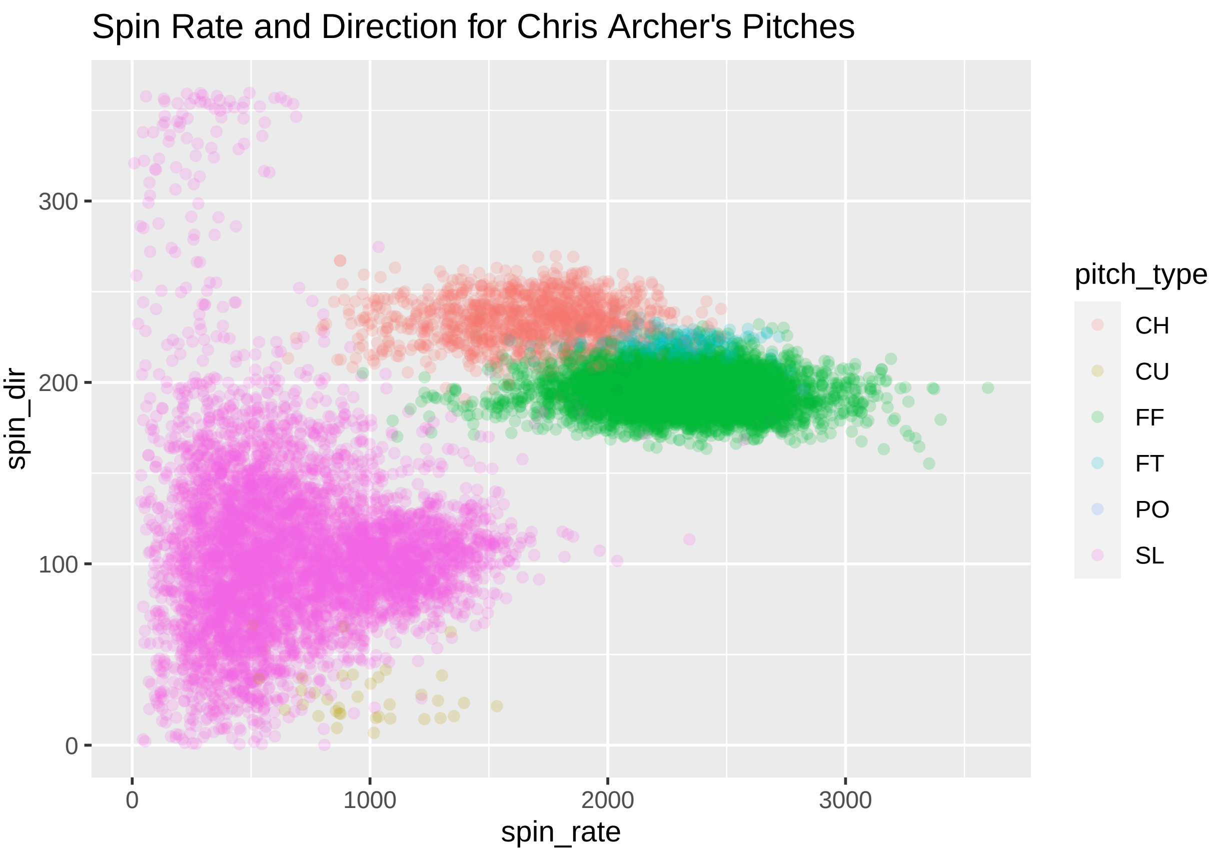
The difference Between Pitch Types

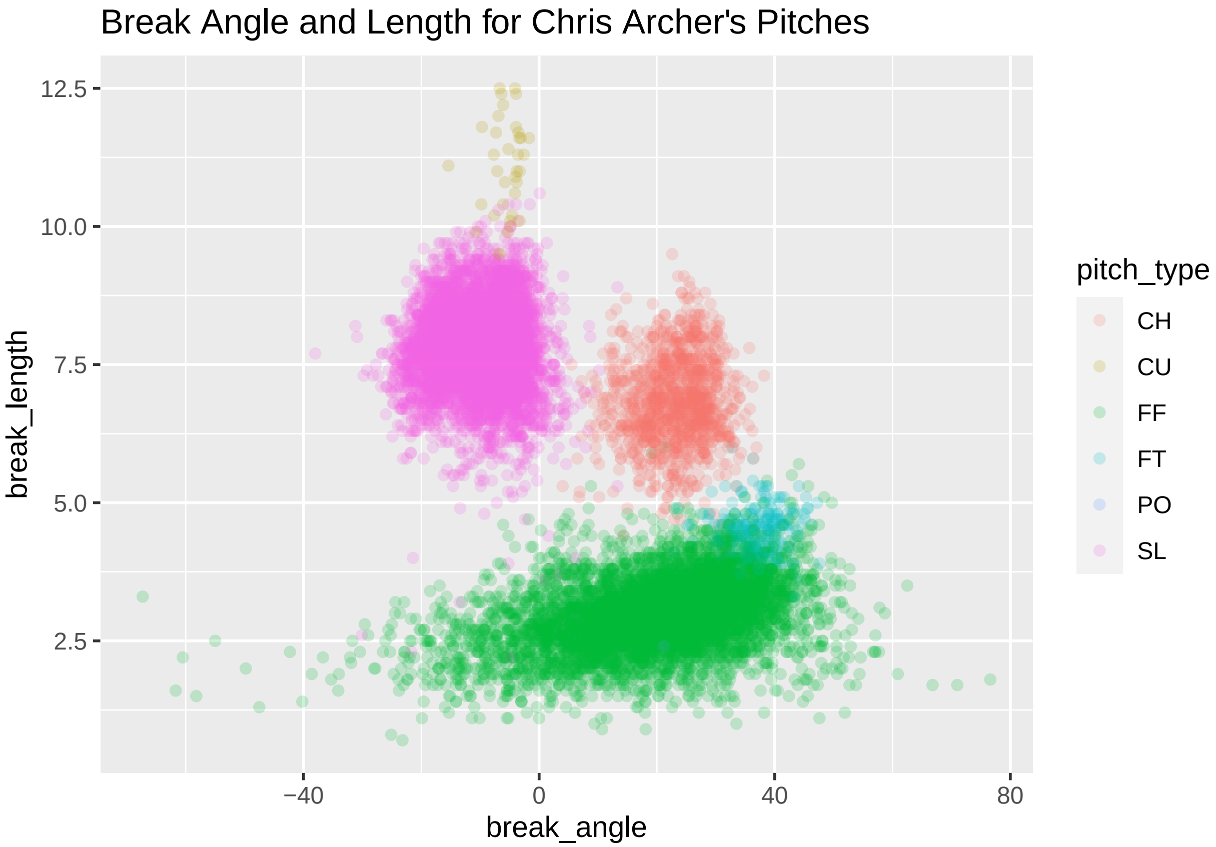
Not all pitches are made the same. From sliders to cutters to knuckleballs, each pitch tends to have its own characteristics that define it.

Table 1: Averages of Different Pitch Characteristics

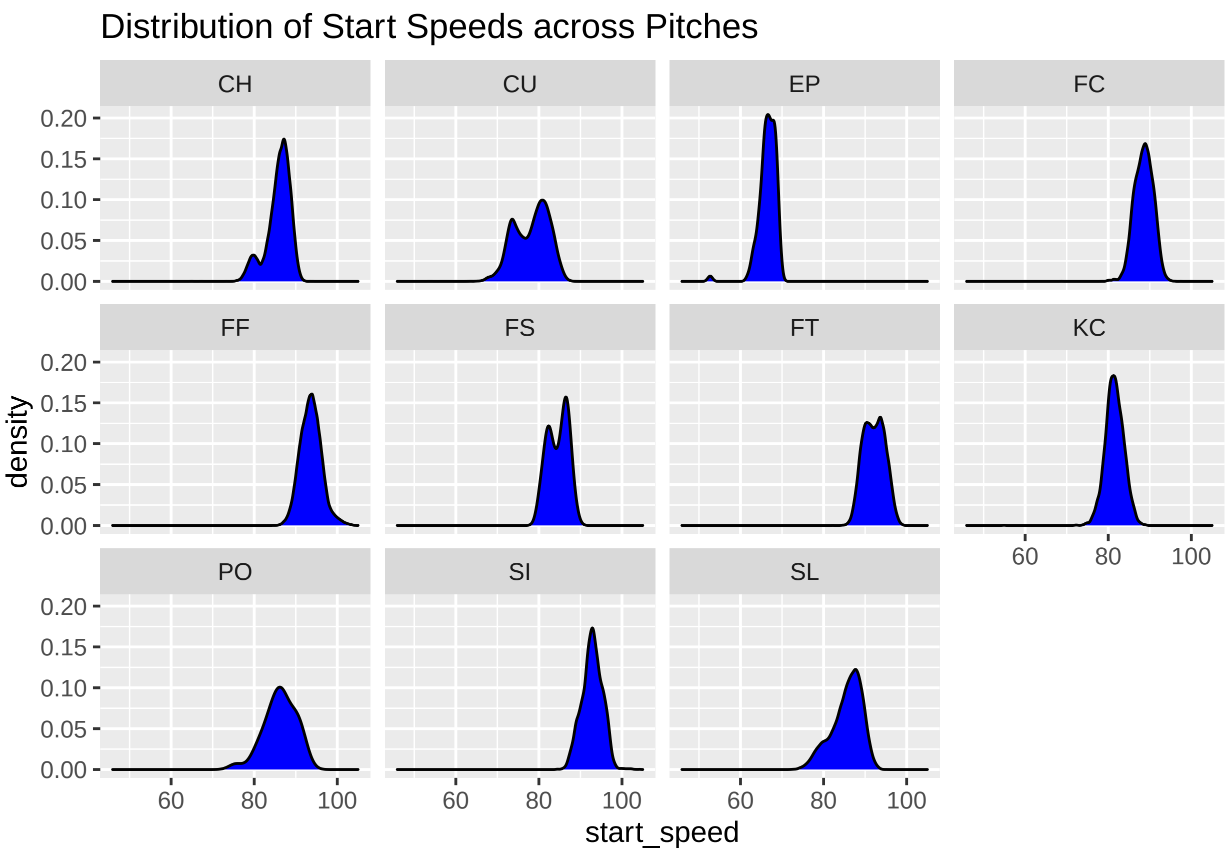
pitch_type	px	pz	start_speed	end_speed	spin_rate	spin_dir	break_angle	break_length
CH	-0.200	1.859	85.814	79.193	1734.540	212.372	8.859	7.695
CU	0.076	1.802	78.164	72.294	1446.625	89.216	-9.949	12.569
EP	0.183	2.022	66.516	61.323	1381.736	44.096	-9.033	16.908
FA	-0.184	1.768	88.900	80.600	2448.751	148.710	-7.067	7.500
FC	0.221	2.310	88.543	81.992	1018.410	164.455	-5.357	6.093
FF	0.007	2.638	93.628	85.901	2175.728	193.631	13.491	3.721
FS	-0.377	1.916	84.568	78.603	1483.198	222.074	17.104	6.653
FT	-0.023	2.351	92.151	84.678	2129.669	169.489	-5.558	5.750
KC	0.084	1.710	81.507	75.232	1318.344	89.548	-6.516	11.551
PO	0.968	4.114	86.217	79.080	1928.408	210.204	21.037	5.171
SI	-0.132	2.309	92.813	85.419	2008.583	216.362	21.268	5.880
SL	0.232	1.867	85.588	79.352	901.077	162.506	-3.271	8.079
UN	3.488	3.531	41.867	39.800	823.262	218.886	-1.167	31.800

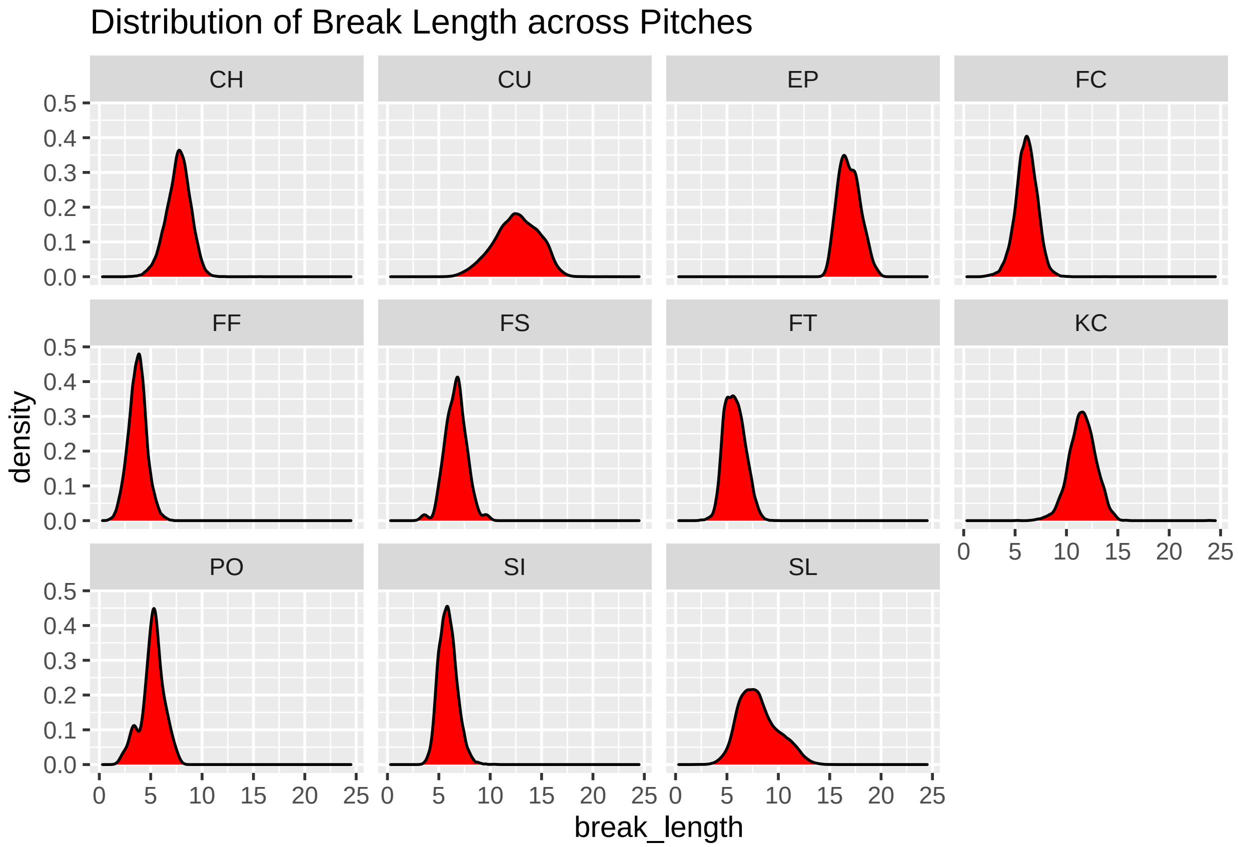
From the table, we essentially see the obvious that is pitches are different. On average we find that fastballs start at almost 94 mph and changeups start at 86 mph. The average break length of a fastball is about 4 inches, which is not too bad, the point that a batter sees a fastball leave the pitcher's hand will most likely cross following a straight line. Compare that to a curveball with almost 13 inches of break and we start to see why knowing the next pitch can be helpful.


Spin Rate and Direction for Chris Archer's Pitches
spin_dir
spin_rate
0
100
200
300
0
1000
2000
3000
pitch_type
CH
CU
FF
FT
PO
SL



The plots of Chris Archer's pitches give us a glimpse of where one type of pitch becomes another. For Archer, we see that his slider tends to have a relatively low spin rate, with his fastballs having higher spin rates and his changeups in between. As we would expect, Archer's fastballs have little break. His slider however, has a large but variable amount of break and always tends to break in the same direction. These visuals help us understand where a pitch may be going. If we can predict the next pitch is a slider, then we can start to understand the direction and magnitude of break in the pitch.


Distribution of Start Speeds across Pitches
CH
CU
EP
FC
FF
FS
FT
KC
PO
SI
SL
density
0.00
0.05
0.10
0.15
0.20
60
80
100
start_speed



Mostly as an exposition into our data, we found the graphs of distributions of break length and start speed to be interesting. They provide a more throrough comparison of pitch types because every pitcher is slightly different. Some pitchers may have a lot more movement on their sinker than others. Similarly, Aroldis Chapman throwing 105 mph fastballs is certainly not the norm in the league. Getting to see the variability of break length and speeds begins to paint the picture of why some pitchers are better than others.

The Difference Between Situations

Naturally, a pitcher will have a generally well established "arsenal" if pitch types that he throws on a regular basis, so a batter will have an idea of types of pitches he is likely to face and the relative frequency at which he will face them. However, by being aware of the game situation, a batter might have a better idea of the type of pitch he is likely to see next. The following section illustrates how this works in practice.

The below table lists the distribution of pitch types that Clayton Kershaw throws conditional on a few different game situation. We observe that Kershaw throws about 48% four seam fastballs across all at bats in our data set. However, if we only look at the first pitch, we see that he throws about 68% four seam fastballs. Following fastballs, his frequency of sliders and curves increases significantly. In two strikes situations, he tends to throw a lot of sliders and relied more on his curve relative to other counts. With three balls, Kershaw relies on four seam fastballs and sliders (probably his two most reliable pitches). To the "heart" of the order, or the three and four hitters in the batting order, he ups the frequency of sliders and curves and throws less four seamers. To the bottom of the order (eight and nine hitters), he throws more four seamers. He also throws a lot of fastballs to lefties, probably because lefties hit off left handed pitchers with a lower batting average than righties do.

Table 2: Clayton Kershaw: Condistional Pitch Arsenal

pitch_type	All	first_pitch	after_ff	two_strikes	three_balls	heart	bottom	lefties
FF	0.475	0.684	0.390	0.355	0.417	0.427	0.550	0.558
SL	0.335	0.246	0.362	0.390	0.422	0.372	0.268	0.279
CU	0.169	0.050	0.229	0.243	0.147	0.176	0.162	0.144
FT	0.015	0.019	0.010	0.011	0.013	0.017	0.018	0.018
CH	0.005	0.001	0.010	0.001	0.001	0.008	0.003	0.000

We can further improve the specificity of our distribution by interacting elements of a situation. For example, the below table looks at what Kershaw throws with his first pitch after a walk to the clean-up hitter (fourth in the order) and the eight and ninth place hitters. We see that after a walk, if Kershaw is facing the bottom of the order, he does not mess around very much with off speed pitches and relies heavily on the four seam. However, to clean-up hitters, Kershaw still mixes in a large share of sliders, clearly balancing the need to throw strikes after a walk with respect for the offensive threat of the clean-up hitter. These kinds of interactions are exactly what our random forest aims to leverage to make it's predictions.

Table 3: Clayton Kershaw: First Pitch After a Walk

pitch_type	4 Hitter	8-9 Hitters
FF	0.529	0.762
SL	0.412	0.238
FT	0.059	-

The Data

The data involved in this project consists of every pitch from the 2015 through the 2018 Major League Baseball (MLB) seasons. These four seasons contained nearly 3 million observations, and the data include categorization of 16 different types of pitches, the spin, speed, and location of each pitch, the game situation (score, runners on base, balls and strikes, etc.), information about the pitcher and batter, and the result of the at-bat.

From this data set, we engineer a few important features. Since each pitch is identified by an at bat id and a pitch number, we can add the previous pitch types as features to each pitch. For this model, we add categorical features for the previous two pitches. We also add the previous at bat's result as a feature.

Although information on each batter might be helpful for this modeling problem, we did not have easy access to this kind of hitter data. As an alternative, we add a factor variable indicating the position in the batting order the pitcher is facing. A team's best hitters tend to bat 3rd or 4th in the order, and a team's worst hitter tends to bat 8th and 9th, so batting order may be able to tell our model a lot about what pitch types a pitcher will throw.

Lastly, we add features for the game specific share of a pitcher's total pitches that each pitch types makes up. This set of features is meant to capture the fact that a pitcher's arsenal of pitch types may change from game to game. Most games a pitcher might throw mostly four seam fastballs, but there might be game specific "fixed effects" that lead him to rely more on his secondary pitches. We'd like our model to be able to use this information in making its predictions.

Methodology

Given how difficult it is to get hits consistently at the major league level, having an idea an idea of what the opposing pitcher is about to throw would confer a significant advantage to the hitter. We endeavor to create a model that successfully predicts the next pitch a pitcher will throw in an at-bat, given the circumstances of the at-bat, the tendencies of the pitcher, and the progress of the game up to that point. Using three distinct random forest models, we derive a predictive approach that generally outperforms guessing that the pitcher will throw their most common pitch (often called "sitting on a pitch"), and in most cases, significantly exceeds this "sitting on a pitch" approach.

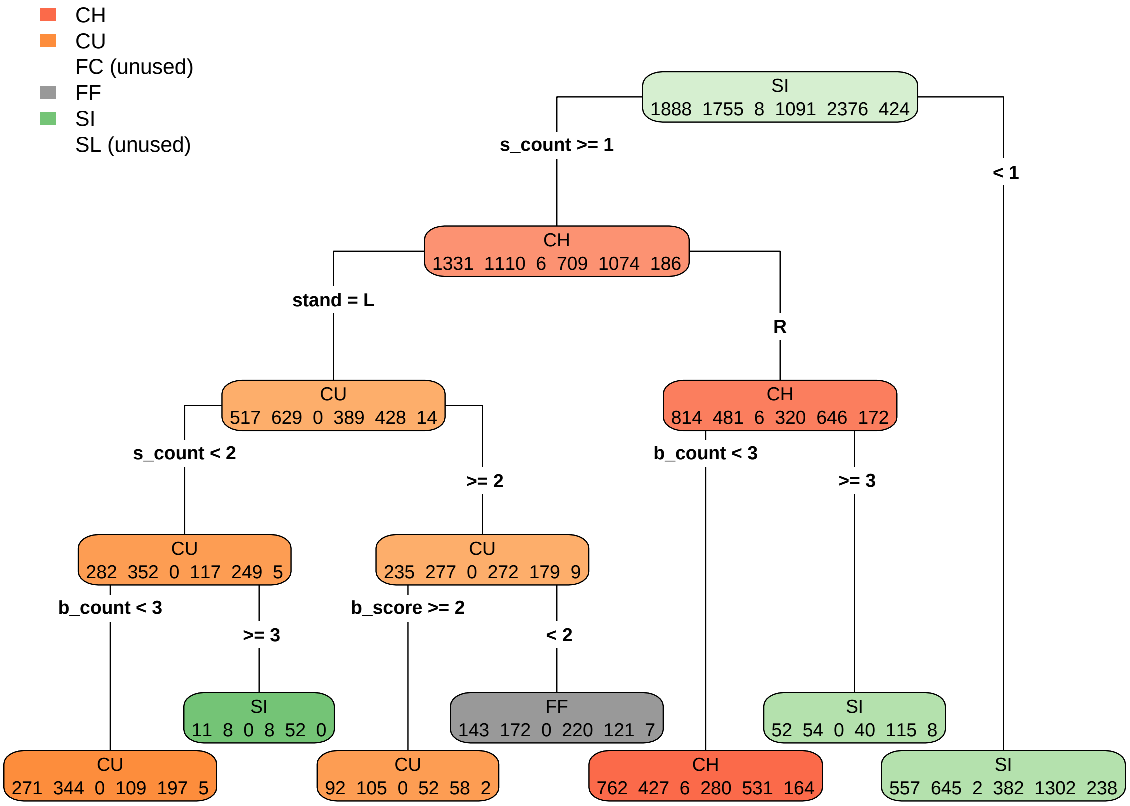
The models are generated with a train/test split and then evaluated for the out-of-sample performance against the testing set. Models are generated per pitcher, in such a way that be beneficial in application to a baseball manager or hitter, given the situation in the game.

Our Random Forests

Random forests, much like actual forests, are an aggregation of individual trees. Tree models make predictions through a series of binary decisions based on a selection of features that sort the data into groups of most likely outcomes. While trees are intuitive, a well performing model will add some layers of complexity that reduce over fitting. Bootstrap aggregating, or "bagging" involves taking B bootstrapped samples of the original data and fitting a tree model to each one. Predictions are generated using a summary of the B tree models. For a categorical outcome, each tree model contributes one vote and the outcome is classified by majority rule.

Random forests extend bagged trees by only allowing each individual tree to use m of the total number of features, p. So bagging is equivalent to a random forest with $m = p$. Restricting the each tree to a subset of the total features decorrelates the individual tree models, reducing variance in the predictions and making it less likely that the model will over fit the data. For our random forest models, we use a common choice of feature size, $m \approx \sqrt{p}$.

The Situational Model The first of our three random forest models uses information readily available in the at-bat to predict the upcoming pitch. The features involved here include the ball-strike count, opposing batter's stance, inning, how many pitches thrown in the at-bat so far, the game score, and the runners on base. This is the most interpretable model, as it is composed of the factors that are generally considered most relevant and well-known by players and coaches in the moment. As random forests are an aggregation of individual trees, it can be illustrative to look at a single tree to get a sense for how decisions are being made at various nodes. Consider, for example, the following tree as an example of what the random forest is doing for pitcher Felix Hernandez.



Hernandez is a pitcher with a diverse arsenal of pitches. As one can see, the predictive model analyzes factors about the game situation to indicate the next pitch. It considers, for example, the count on the batter and whether the batter stands left or right. After running through the branches and nodes with binary decisions based on the in-game situation, a prediction about the next pitch is made at the bottom of the tree. The random forest takes a bootstrap aggregate of trees like this one, but a single tree helps us see an example of the predictive process.

The Lagged Model The second random forest builds upon the features selected in the Situational Model and supplements them with information about the previous two pitches and the event the last at-bat. This allows for the model to incorporate lagged information that might directly influence the next pitch. Did the pitcher just give up a home run on the curve? Maybe it's a steady diet of fastballs from here on out. Did the pitcher just give the batter two straight fastballs to study? Perhaps it's time for an off-speed pitch like a change-up.

Trash Can Model The third and final random forest uses nearly every feature from the dataset to control for all possible scenarios and variations in the feature matrix. This means that it not only considers all the factors included in the other two models, but also includes what the pitcher's game has looked like so far; that is, it takes into account the pitch choices as a percent of the overall pitches in the game up to that point. (Is this a correct reading?) So if the pitcher is leaning heavily on the slider that day, this model will incorporate that pattern.

Pitch Prediction

Now that we have our models constructed, how does it work? Let's return to Clayton Kershaw in a real game situation to find out. Let's say the count is one ball, one strike, and there is one out. Kershaw is facing a right-handed batter. What does the Trash Can model say that Kershaw will throw here?

```
##   pitch_type prediction     percent actual actual_percent
## 1         SL        110 0.372881356    106    0.359322034
## 2         FF        106 0.359322034     98    0.332203390
## 3         CU         76 0.257627119     86    0.291525424
## 4         CH          2 0.006779661      2    0.006779661
## 5         FT          1 0.003389831      3    0.010169492
```

The model says that there's about a third chance that he will throw a slider, a third chance he will throw a fastball, and a third chance he will throw a curveball. Obviously, this is a difficult predicament for a hitter, but the model itself did pretty well! We slightly overpredict the chance of a slider and a fastball, and slightly underpredict the curveball.

What if we change the situation again? Now say it's the first pitch of an at-bat with two outs against a lefty.

```
##   pitch_type prediction    percent actual actual_percent
## 1         FF        143 0.72588832    133     0.67512690
## 2         SL         39 0.19796954     46     0.23350254
## 3         CU         11 0.05583756     14     0.07106599
## 4         FT          4 0.02030457      4     0.02030457
```

Recall that we saw that Kershaw loves to throw a fastball on the first pitch, and indeed our model predicts that (at a slightly higher rate than is actually observed.) So if you are a left-handed hitter coming into the first pitch of an at-bat, it's probably a good idea to look for a fastball.

These predictions can be generated for any pitcher, and we invite you to explore in-game situations with our pitchers in the Shiny app (link here?).

Results

In [TABLE/APPENDIX], we publish our models' results for 30 pitchers from Major League Baseball. These pitchers are generally considered some of the best in the business, but there are some pitchers from outside the top tier for completeness. We also note that the model is easily adaptable to any pitcher who threw between 2015 and 2018. In [TABLE/APPENDIX], we present the pitching profiles of the 30 pitchers based on the style of their pitches and the frequency that they are thrown.

Table 4: Overall Performance

pitcher_last	pitcher_first	rate_sitONE	rate_trashcan1	rate_trashcan2	rate_trashcan3
Archer	Chris	0.474	0.559	0.580	0.566
Arrieta	Jake	0.471	0.474	0.477	0.470
Boxberger	Brad	0.631	0.644	0.651	0.644
Britton	Zach	0.896	0.894	0.899	0.899
Bumgarner	Madison	0.283	0.314	0.367	0.409
Chapman	Aroldis	0.748	0.732	0.738	0.731
Cole	Gerrit	0.497	0.482	0.491	0.469
Darvish	Yu	0.367	0.389	0.412	0.453
Davis	Wade	0.487	0.474	0.459	0.452
deGrom	Jacob	0.428	0.407	0.419	0.410
Gray	Sonny	0.321	0.394	0.401	0.421
Gregerson	Luke	0.433	0.513	0.492	0.560
Greinke	Zack	0.417	0.430	0.436	0.411
Hernandez	Felix	0.314	0.380	0.389	0.368
Kershaw	Clayton	0.475	0.506	0.522	0.522
Keuchel	Dallas	0.467	0.470	0.483	0.476
Kluber	Corey	0.322	0.330	0.343	0.338
McHugh	Collin	0.332	0.360	0.397	0.398

pitcher_last	pitcher_first	rate_sitONE	rate_trashcan1	rate_trashcan2	rate_trashcan3
Melancon	Mark	0.588	0.608	0.608	0.596
Miller	Andrew	0.574	0.583	0.601	0.581
Norris	Bud	0.411	0.467	0.457	0.451
Price	David	0.323	0.322	0.322	0.338
Rosenthal	Trevor	0.752	0.726	0.740	0.704
Sale	Chris	0.332	0.370	0.381	0.463
Scherzer	Max	0.528	0.530	0.526	0.484
Strasburg	Stephen	0.482	0.467	0.461	0.435
Street	Huston	0.427	0.672	0.700	0.645
Tolleson	Shawn	0.614	0.598	0.603	0.554
Wacha	Michael	0.517	0.516	0.514	0.483
Zimmermann	Jordan	0.510	0.503	0.499	0.479

To measure our models' effectiveness, we compare them to the most-common pitch thrown by the pitcher (called "sitONE") and to each other. The out-of-sample performance of each model is presented in "trashcans" 1 to 3 (our sincerest apologies to Astros fans for the name).

Most models improved on the guess of the upcoming pitch when compared to the "sitting on the pitch" as seen in [TABLE/APPENDIX]. Consider the case of Chris Sale. Sale is a perennial All-Star, and finished top-5 in Cy Young (MLB's most valuable pitcher award) votes each year of our data. If a batter were to look for his most common pitch (the two-seam fastball), he would would only be right about one third of the time. However, Sale becomes increasingly more predictable as the model incorporates more features. The Situational Model and the Lagged Model predict his next pitch at 37% and 38% respectively. Still, the Game Fixed Effects Model predicts his next pitch with an out-of-sample accuracy of nearly 46%, a jump of about 13%. There isn't a hitter in baseball who wouldn't want to know Sale's next pitch with a 13% increase in accuracy!

On the other hand, some pitcher profiles grew in accuracy only up through the first or second model, and then decrease as more features are added. For example, Mark Melancon and Huston Street have out-of-sample prediction accuracies that peak in the Lagged Model but fall in the Game Fixed Effects Model. We present this as some evidence of over-fitting for certain pitchers.

Not all pitchers are particularly predictable, however. Corey Kluber, a two-time Cy Young winner (one time in our data window), has such a varied arsenal of pitches that even the most predictive model (.343 out of sample accuracy) barely little information about the next pitch, even if it beats he sit-one rate. Further, some pitchers remained elusive for all three of the models. Trevor Rosenthal, for instance, throws a fastball on 75% of his pitches, and all three models performed worse than this "sitting on the pitch" rate. [Do we have a good reason for why this might be happening?]

One fairly common pattern that we notice throughout the models is as we feed more features to our random forest, the models trade accuracy in the pitchers' most frequently thrown pitch for accuracy in the secondary pitches. Take Yu Darvish as an example. We see in table [REFER TO THE CORRECT TABLE] that the model predicts his pitches with greater overall accuracy with each addition of features. However, if we look at [THE BELOW TABLE], we see that from model 2 to model 3, four seam fastball accuracy actually decreases while several of Darvish's secondary pitches get classified with much greater accuracy, particularly two seam fastballs. Specific to Darvish's case, this might be due to his "streaky" use of the two seam fastball, relying on it heavily in some games and not at all in other. Or perhaps he added this pitch to his arsenal in the middle of our data set span, which only the game specific pitch distributions will pick up. More generally, this primary/secondary pitch accuracy trade off is a common trade off that our models make to achieve better overall performance for many of the pitchers.

Table 5: By Pitch Performance: Yu Darvish

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.744	0.751	0.680
SL	0.324	0.332	0.360
FT	0.144	0.206	0.428
FC	0.080	0.168	0.292
CU	0.061	0.000	0.045
CH	0.000	0.000	0.048
FS	0.000	0.000	0.000
EP	0.000	0.000	0.000

However, whether this trade off is a good one depends on a batter's hitting approach. A power hitter might aim to never miss a fastball, opting to get fooled more often on off speed pitches in exchange for the ability to crush any fastball that the pitcher throws. This hitter might prefer a modeling approach that maximizes fastball performance and that does not make the primary/secondary pitch accuracy tradeoff.

Finally, we note that predictive power does not necessarily equate to hits. Enter Zack Britton, who led the league in saves during the 2016 season. Britton is nearly a one-pitch pitcher, throwing sinkers on 89.7% of his pitches. The Game Fixed Effects model improves this prediction a few fractions of a percent, but in either case, it's fair to say that most hitters know exactly what Britton is about to throw. Nevertheless, Britton put up an otherworldly 0.54 ERA in 2016! That is to say, even if batters knew with near perfect clairvoyance a that sinker was on its way, making contact that leads to a hit is a whole other matter.

what it all means, probably nothing

implications for how hitters should approach thinking about predicting pitches. What it all means: - we've demonstrated that the game situation, previous pitches, previous result, batting order position, and game to game changes in a pitcher's "stuff" all help predict pitches - for some pitcher, all of these features are helpful. for others, using all of the features over fits - this modelling is not dierrctly useful to teams on a pitch to pitch level. it'd probably be considered cheating for an assistant coach to be sitting in the dugout with laptop inputting the situation and having the model spit back predictions. Baseball has already shown it is uncomfortable with certain levels of technology entering the game, adding machine learning would be overstepping no doubt. - rather, what we've provided is a framework for hitters to analyze what kind of information is helpful in predicting a particular pitcher's pitch type. They can focus on the information that is important and ignore the rest. this will help them in building scouting reports on pitchers - as the Astros showed us all in 2017?, this is crucial

(gonna try to make this coherent) Moreover: - where our work might be useful on a pitch to pitch basis is with fan enjoyment experience. baseball has lost popularity in recent years, with Tv ratings on the decline and game attendance faltering as well. Surely, this decrease is due to the games slow pace and lack of entertainment value for the casual fan. after all, the game grinds to a halt after each pitch, pausing for players and coaches to relay signs, adjust equipment, and mentally prepare for the next pitch

but for baseball enthusiasts, this segmented gameplay is part of the fun. After each pitch, it is perfectly observable to all players involved what the situation is. The "game within the game" is figuring out how to best respond to each state of the game. And a big part of that is the batter trying to guess what pitch is coming and the pitcher trying to throw a pitch that the batter might not expect.

our modeling tool can be a tool to let fans watching at home in on the game within the game, which is why we've built our shiny web app. Our app allows a fan to input a situation and get pitch prediction percentages in return. This will vastly improve porch beers with the boys while listening to America's past time on the radio as well as Friday Night baseball on the couch with Dom's

Appendix A

Table 6: Pitch Types: Trevor Rosenthal

pitch_type	count	pct
FF	2206	0.753
CH	432	0.148
SL	255	0.087
CU	32	0.011
FT	2	0.001
FA	1	0.000

Table 7: Pitch Types: Felix Hernandez

pitch_type	count	pct
SI	2956	0.314
CH	2348	0.249
CU	2185	0.232
FF	1393	0.148
SL	536	0.057
FC	10	0.001

Table 8: Pitch Types: Chris Archer

pitch_type	count	pct
FF	6045	0.474
SL	5295	0.415
CH	1125	0.088
FT	250	0.020
CU	36	0.003

Table 9: Pitch Types: Zach Britton

pitch_type	count	pct
SI	2840	0.897
SL	243	0.077
FF	59	0.019
FT	24	0.008

Table 10: Pitch Types: Wade Davis

pitch_type	count	pct
FF	1910	0.487
FC	1133	0.289
KC	788	0.201
FT	86	0.022

pitch_type	count	pct
CH	1	0.000

Table 11: Pitch Types: Dallas Keuchel

pitch_type	count	pct
FT	5442	0.467
SL	2394	0.206
CH	1429	0.123
FC	1203	0.103
FF	1175	0.101

Table 12: Pitch Types: Corey Kluber

pitch_type	count	pct
SI	4012	0.322
CU	2615	0.210
FF	1931	0.155
SL	1637	0.131
FC	1578	0.127
CH	684	0.055

Table 13: Pitch Types: Luke Gregerson

pitch_type	count	pct
SL	1250	0.435
SI	738	0.257
FT	550	0.192
FF	261	0.091
CH	41	0.014
FC	31	0.011
CU	1	0.000

Table 14: Pitch Types: David Price

pitch_type	count	pct
FT	3536	0.323
CH	2359	0.216
FC	2152	0.197
FF	2141	0.196
KC	744	0.068

Table 15: Pitch Types: Max Scherzer

pitch_type	count	pct
FF	7117	0.528
SL	2639	0.196
CH	1856	0.138
CU	1090	0.081
FC	688	0.051
FT	89	0.007

Table 16: Pitch Types: Aroldis Chapman

pitch_type	count	pct
FF	2927	0.749
SL	748	0.192
CH	158	0.040
SI	73	0.019

Table 17: Pitch Types: Clayton Kershaw

pitch_type	count	pct
FF	4873	0.475
SL	3435	0.335
CU	1737	0.169
FT	153	0.015
CH	51	0.005

Table 18: Pitch Types: Madison Bumgarner

pitch_type	count	pct
FF	2985	0.283
SL	2111	0.200
FT	1874	0.178
CU	1822	0.173
FC	1318	0.125
CH	441	0.042

Table 19: Pitch Types: Sonny Gray

pitch_type	count	pct
FF	3193	0.321
FT	2712	0.273
CU	1596	0.161
SL	1537	0.155
CH	712	0.072
FC	182	0.018

Table 20: Pitch Types: Huston Street

pitch_type	count	pct
SI	623	0.428
SL	541	0.371
CH	293	0.201

Table 21: Pitch Types: Brad Boxberger

pitch_type	count	pct
FF	1942	0.634
CH	942	0.308
SL	86	0.028
CU	53	0.017
FC	37	0.012
FT	2	0.001

Table 22: Pitch Types: Zack Greinke

pitch_type	count	pct
FF	5049	0.418
SL	2402	0.199
CH	2200	0.182
CU	1305	0.108
FT	1032	0.085
EP	104	0.009

Table 23: Pitch Types: Shawn Tolleson

pitch_type	count	pct
FF	1113	0.618
CH	291	0.161
SL	291	0.161
FT	64	0.036
FC	43	0.024

Table 24: Pitch Types: Jordan Zimmermann

pitch_type	count	pct
FF	4853	0.510
SL	2547	0.268
CU	1437	0.151
CH	346	0.036
FT	326	0.034

Table 25: Pitch Types: Jacob deGrom

pitch_type	count	pct
FF	5009	0.428
SL	2384	0.204
FT	1687	0.144
CH	1490	0.127
CU	1121	0.096

Table 26: Pitch Types: Gerrit Cole

pitch_type	count	pct
FF	5812	0.497
SL	2253	0.193
KC	1440	0.123
SI	815	0.070
CH	715	0.061
FT	626	0.054
CU	29	0.002

Table 27: Pitch Types: Mark Melancon

pitch_type	count	pct
FC	1913	0.589
KC	864	0.266
FF	443	0.136
FS	30	0.009

Table 28: Pitch Types: Jake Arrieta

pitch_type	count	pct
SI	5663	0.471
SL	2583	0.215
CU	1584	0.132
FF	1368	0.114
CH	830	0.069

Table 29: Pitch Types: Andrew Miller

pitch_type	count	pct
SL	2068	0.575
FF	1446	0.402
FT	81	0.023

Table 30: Pitch Types: Stephen Strasburg

pitch_type	count	pct
FF	4464	0.483
CU	1757	0.190
CH	1487	0.161
FT	776	0.084
SL	762	0.082

Table 31: Pitch Types: Collin McHugh

pitch_type	count	pct
FF	2879	0.332
FC	2447	0.283
CU	2141	0.247
SL	440	0.051
FT	383	0.044
CH	371	0.043

Table 32: Pitch Types: Michael Wacha

pitch_type	count	pct
FF	4822	0.518
CH	1793	0.193
FC	1521	0.163
CU	1085	0.116
FT	93	0.010

Table 33: Pitch Types: Chris Sale

pitch_type	count	pct
FT	4218	0.332
SL	3509	0.276
CH	2660	0.210
FF	2301	0.181
FA	2	0.000
FS	1	0.000

Table 34: Pitch Types: Zack Greinke

pitch_type	count	pct
FF	5049	0.418
SL	2402	0.199
CH	2200	0.182
CU	1305	0.108
FT	1032	0.085

pitch_type	count	pct
EP	104	0.009

Table 35: Pitch Types: Yu Darvish

pitch_type	count	pct
FF	1967	0.367
SL	1261	0.235
FT	969	0.181
FC	684	0.128
CU	328	0.061
CH	104	0.019
FS	37	0.007
EP	5	0.001

Apendix B

Table 36: Overall Performance

pitcher_last	pitcher_first	rate_sitONE	rate_trashcan1	rate_trashcan2	rate_trashcan3
Archer	Chris	0.474	0.559	0.580	0.566
Arrieta	Jake	0.471	0.474	0.477	0.470
Boxberger	Brad	0.631	0.644	0.651	0.644
Britton	Zach	0.896	0.894	0.899	0.899
Bumgarner	Madison	0.283	0.314	0.367	0.409
Chapman	Aroldis	0.748	0.732	0.738	0.731
Cole	Gerrit	0.497	0.482	0.491	0.469
Darvish	Yu	0.367	0.389	0.412	0.453
Davis	Wade	0.487	0.474	0.459	0.452
deGrom	Jacob	0.428	0.407	0.419	0.410
Gray	Sonny	0.321	0.394	0.401	0.421
Gregerson	Luke	0.433	0.513	0.492	0.560
Greinke	Zack	0.417	0.430	0.436	0.411
Hernandez	Felix	0.314	0.380	0.389	0.368
Kershaw	Clayton	0.475	0.506	0.522	0.522
Keuchel	Dallas	0.467	0.470	0.483	0.476
Kluber	Corey	0.322	0.330	0.343	0.338
McHugh	Collin	0.332	0.360	0.397	0.398
Melancon	Mark	0.588	0.608	0.608	0.596
Miller	Andrew	0.574	0.583	0.601	0.581
Norris	Bud	0.411	0.467	0.457	0.451
Price	David	0.323	0.322	0.322	0.338
Rosenthal	Trevor	0.752	0.726	0.740	0.704
Sale	Chris	0.332	0.370	0.381	0.463
Scherzer	Max	0.528	0.530	0.526	0.484
Strasburg	Stephen	0.482	0.467	0.461	0.435
Street	Huston	0.427	0.672	0.700	0.645
Tolleson	Shawn	0.614	0.598	0.603	0.554
Wacha	Michael	0.517	0.516	0.514	0.483

pitcher_last	pitcher_first	rate_sitONE	rate_trashcan1	rate_trashcan2	rate_trashcan3
Zimmermann	Jordan	0.510	0.503	0.499	0.479

Apendix C

Table 37: By Pitch Performance: Trevor Rosenthal

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.952	0.980	0.907
CH	0.069	0.023	0.092
SL	0.000	0.000	0.098
CU	0.000	0.000	0.000
FT	0.000	0.000	0.000

Table 38: By Pitch Performance: Felix Hernandez

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
SI	0.591	0.633	0.566
CH	0.447	0.432	0.406
CU	0.277	0.277	0.238
FF	0.129	0.115	0.197
SL	0.009	0.037	0.083
FC	0.000	0.000	0.000

Table 39: By Pitch Performance: Chris Archer

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.685	0.700	0.650
SL	0.554	0.582	0.585
CH	0.053	0.080	0.124
FT	0.000	0.000	0.200
CU	0.000	0.000	0.000

Table 40: By Pitch Performance: Zach Britton

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
SI	0.995	0.998	0.991
SL	0.000	0.000	0.041
FF	0.000	0.000	0.000
FT	0.400	0.600	1.000

Table 41: By Pitch Performance: Wade Davis

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.770	0.846	0.730
FC	0.264	0.123	0.233
KC	0.114	0.057	0.146
FT	0.000	0.000	0.000

Table 42: By Pitch Performance: Dallas Keuchel

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FT	0.870	0.869	0.769
SL	0.251	0.263	0.280
CH	0.059	0.122	0.175
FC	0.017	0.033	0.183
FF	0.034	0.043	0.191

Table 43: By Pitch Performance: Corey Kluber

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
SI	0.705	0.702	0.518
CU	0.377	0.396	0.331
FF	0.023	0.041	0.189
SL	0.055	0.088	0.262
FC	0.038	0.060	0.269
CH	0.161	0.146	0.066

Table 44: By Pitch Performance: Luke Gregerson

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
SL	0.696	0.756	0.672
SI	0.527	0.318	0.588
FT	0.400	0.436	0.600
FF	0.000	0.000	0.038
CH	0.000	0.000	0.000
FC	0.000	0.000	0.000

Table 45: By Pitch Performance: David Price

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FT	0.633	0.620	0.514
CH	0.280	0.254	0.231
FC	0.174	0.181	0.220
FF	0.114	0.156	0.385
KC	0.007	0.000	0.040

Table 46: By Pitch Performance: Max Scherzer

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.900	0.884	0.796
SL	0.237	0.254	0.254
CH	0.065	0.067	0.078
CU	0.000	0.005	0.023
FC	0.000	0.000	0.014
FT	0.000	0.000	0.111

Table 47: By Pitch Performance: Aroldis Chapman

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.964	0.962	0.922
SL	0.047	0.093	0.160
CH	0.031	0.000	0.094
SI	0.000	0.000	0.333

Table 48: By Pitch Performance: Clayton Kershaw

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.677	0.712	0.691
SL	0.378	0.357	0.421
CU	0.339	0.379	0.293
FT	0.000	0.032	0.194
CH	0.000	0.000	0.091

Table 49: By Pitch Performance: Madison Bumgarner

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.551	0.652	0.596
SL	0.270	0.312	0.374
FT	0.096	0.181	0.453
CU	0.422	0.266	0.159
FC	0.110	0.333	0.447
CH	0.011	0.011	0.056

Table 50: By Pitch Performance: Sonny Gray

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.501	0.505	0.512
FT	0.470	0.488	0.473
CU	0.169	0.181	0.262
SL	0.494	0.471	0.435
CH	0.021	0.042	0.217
FC	0.027	0.000	0.135

Table 51: By Pitch Performance: Huston Street

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
SI	0.632	0.632	0.592
SL	0.780	0.817	0.817
CH	0.559	0.627	0.441

Table 52: By Pitch Performance: Brad Boxberger

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.913	0.920	0.869
CH	0.222	0.222	0.296
SL	0.000	0.056	0.111
CU	0.000	0.000	0.091
FC	0.000	0.000	0.000
FT	0.000	0.000	0.000

Table 53: By Pitch Performance: Zack Greinke

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.787	0.757	0.674
SL	0.279	0.322	0.351
CH	0.252	0.286	0.255
CU	0.000	0.019	0.057
FT	0.000	0.014	0.077
EP	0.048	0.048	0.048

Table 54: By Pitch Performance: Shawn Tolleson

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.901	0.924	0.807
CH	0.153	0.153	0.102
SL	0.102	0.051	0.186
FT	0.077	0.077	0.231
FC	0.000	0.000	0.111

Table 55: By Pitch Performance: Jordan Zimmermann

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.856	0.849	0.743
SL	0.224	0.218	0.271
CU	0.045	0.045	0.115
CH	0.000	0.014	0.071
FT	0.000	0.015	0.227

Table 56: By Pitch Performance: Jacob deGrom

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.886	0.881	0.738
SL	0.090	0.103	0.172
FT	0.018	0.109	0.234
CH	0.034	0.037	0.158
CU	0.022	0.004	0.053

Table 57: By Pitch Performance: Gerrit Cole

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.934	0.939	0.825
SL	0.086	0.113	0.151
KC	0.007	0.014	0.056
SI	0.000	0.006	0.184
CH	0.007	0.014	0.049
FT	0.000	0.000	0.127
CU	0.000	0.000	0.000

Table 58: By Pitch Performance: Mark Melancon

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FC	0.903	0.914	0.836
KC	0.277	0.249	0.283
FF	0.022	0.034	0.213
FS	0.000	0.000	0.000

Table 59: By Pitch Performance: Jake Arrieta

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
SI	0.909	0.891	0.786
SL	0.099	0.147	0.234
CU	0.155	0.158	0.142
FF	0.004	0.018	0.234
CH	0.054	0.048	0.060

Table 60: By Pitch Performance: Andrew Miller

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
SL	0.727	0.763	0.761
FF	0.410	0.403	0.355
FT	0.000	0.000	0.059

Table 61: By Pitch Performance: Stephen Strasburg

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.908	0.877	0.746
CU	0.060	0.077	0.131
CH	0.097	0.131	0.154
FT	0.006	0.019	0.237
SL	0.013	0.007	0.065

Table 62: By Pitch Performance: Collin McHugh

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.462	0.483	0.497
FC	0.410	0.443	0.443
CU	0.357	0.410	0.343
SL	0.045	0.182	0.318
FT	0.000	0.000	0.104
CH	0.013	0.013	0.067

Table 63: By Pitch Performance: Michael Wacha

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.888	0.880	0.790
CH	0.253	0.262	0.245
FC	0.046	0.046	0.131
CU	0.005	0.009	0.028
FT	0.000	0.000	0.211

Table 64: By Pitch Performance: Chris Sale

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FT	0.518	0.627	0.668
SL	0.415	0.352	0.316
CH	0.244	0.242	0.233
FF	0.176	0.137	0.575
FA	0.000	0.000	0.000

Table 65: By Pitch Performance: Zack Greinke

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.787	0.757	0.674
SL	0.279	0.322	0.351
CH	0.252	0.286	0.255
CU	0.000	0.019	0.057
FT	0.000	0.014	0.077
EP	0.048	0.048	0.048

Table 66: By Pitch Performance: Yu Darvish

pitch_type	rate_trashcan1	rate_trashcan2	rate_trashcan3
FF	0.744	0.751	0.680
SL	0.324	0.332	0.360
FT	0.144	0.206	0.428
FC	0.080	0.168	0.292
CU	0.061	0.000	0.045
CH	0.000	0.000	0.048
FS	0.000	0.000	0.000
EP	0.000	0.000	0.000

Apendix D

Datasets: Due to the size of datasets in this folder, we gitignore all .csv files. You can find some of the data in archive.zip. However, download pitches.zip from one of the below links to get pitches data:

- data is from here: http://inalitic.com/datasets/mlb%20pitch%20data.html
- pitchers data dropbox: https://www.dropbox.com/s/9gyz3ujwx7jsh5j/pitches.zip?dl=0
- data is built from here: https://www.kaggle.com/datasets/pschale/mlb-pitch-data-20152018
- detailed info on the data: https://docs.google.com/document/d/1ztD20pt5K0HUi2EcJHT4SYdOZw9YPYhtLUmi8BpInuA/edit?pref=2&pli=1#heading=h.mnao9thv84r1
- hitting here: https://www.baseball-reference.com/leagues/majors/2015-standard-batting.shtml?sr&utm_source=direct&utm_medium=Share&utm_campaign=ShareTool#players_standard_batting