Do You Really Need Trash Cans?

February 26, 2020

The 2019–2020 MLB off-season was dominated by the revelation that the Houston Astros stole c
looking at a live camera feed from center field in an area just off the dugout and communicating t
predict what pitch was coming next? Do you even need to be stealing signs, or just a computer?”


★ OFFICIAL ★
MAJOR LEAGUE BASEBALL
COMMISSIONER

Photo by Lesly Juarez on Unsplash

The Astros Scheme

The Astros scheme has been very well covered by now. I won't delve into the specifics of their ch
learning. First, they only seemed to bang the trash can if the next pitch wasn't going to be a fastb
the pitch was going to be off-speed. Second is that Rob Arthur from Baseball Prospectus determi
the time.

> "By and large, the Astros tended to get the signals right, but it was hardly perfect. They were mo
> of the time and they were wrong seven percent of the time. … Based on Adams' data, the Astros
> silent."

(Source: https://blogs.fangraphs.com/the-most-important-bangs-of-the-astros-scheme/)

So, in order for our machine learning model to be comparable to the Astros' scheme, it must pred

The Data

The great thing about applying analytics and machine learning to baseball is that there is a wealt
Python package, which provides a wrapper for Statcast data, which has entries of every pitch thr
count is. Also included are what hand the pitcher throws with and the hitter's stance. Crucially, St
and previous pitch velocity column.

Once the data was pulled from Statcast and massaged a bit, I split it into a training and testing s
fastball, and therefore there were many more cases of pitchers throwing a fastball then there wa
achieved very good accuracy by predicting the next pitch was always going to be a fastball. Obvi
data, leaving us with a balanced training dataset between all different types of pitches.

In this analysis, we will use two different datasets, one consisting of Jose Berrios' pitches from t
in every MLB game between April 1st, 2019 and April 7th, 2019. These two datasets will allow us
deciding what pitch to throw in a given situation.

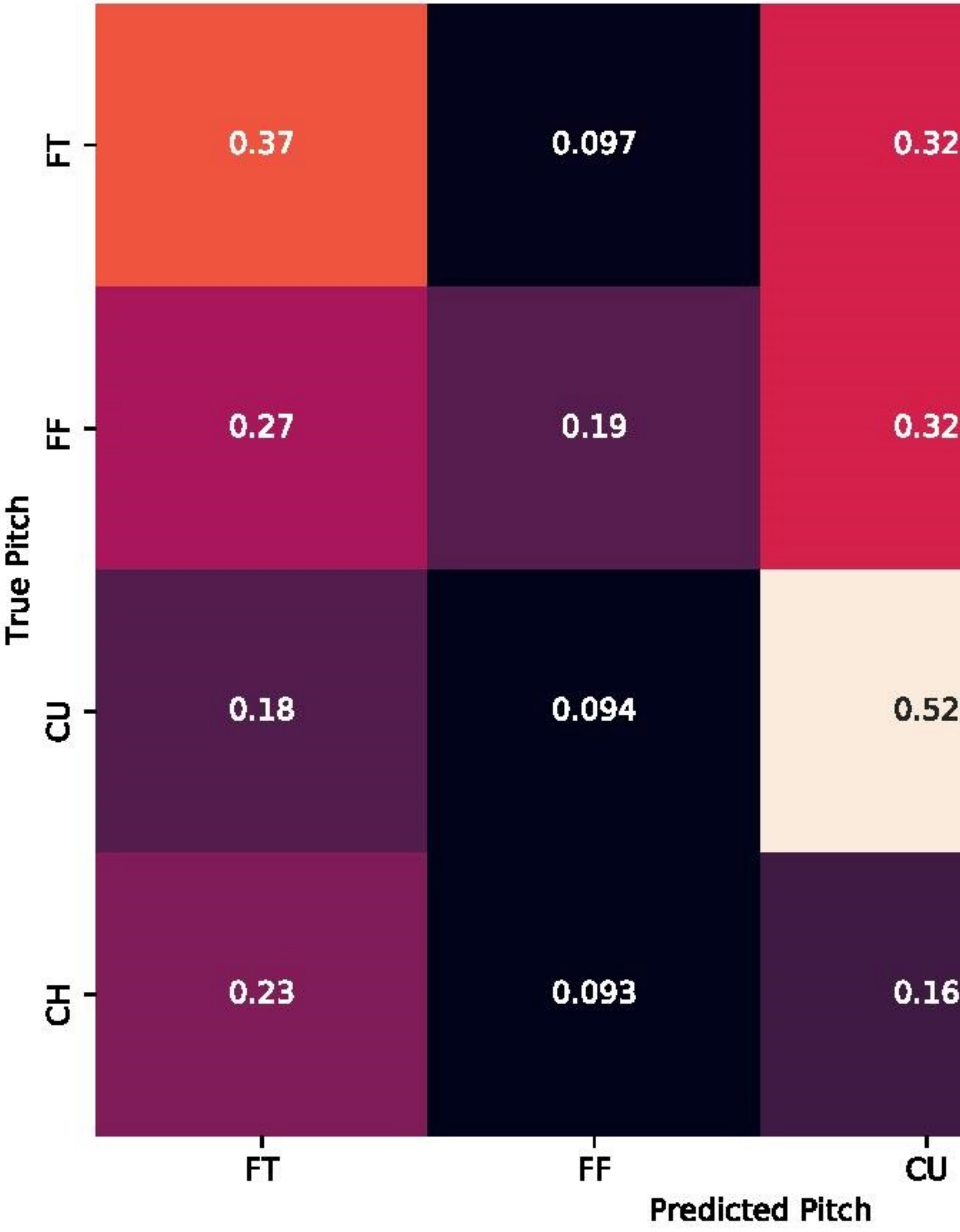
With these two different datasets, I also created a version of each that only contains if the pitch t
two-seam fastballs, and cutters as "fastballs" and everything else as a non-fastball. I am taking a

The Models

I started with some “basic” machine learning models such as Decision Trees, SVMs and k-Neares
to feed the data into two different neural networks: one for each version of the dataset.

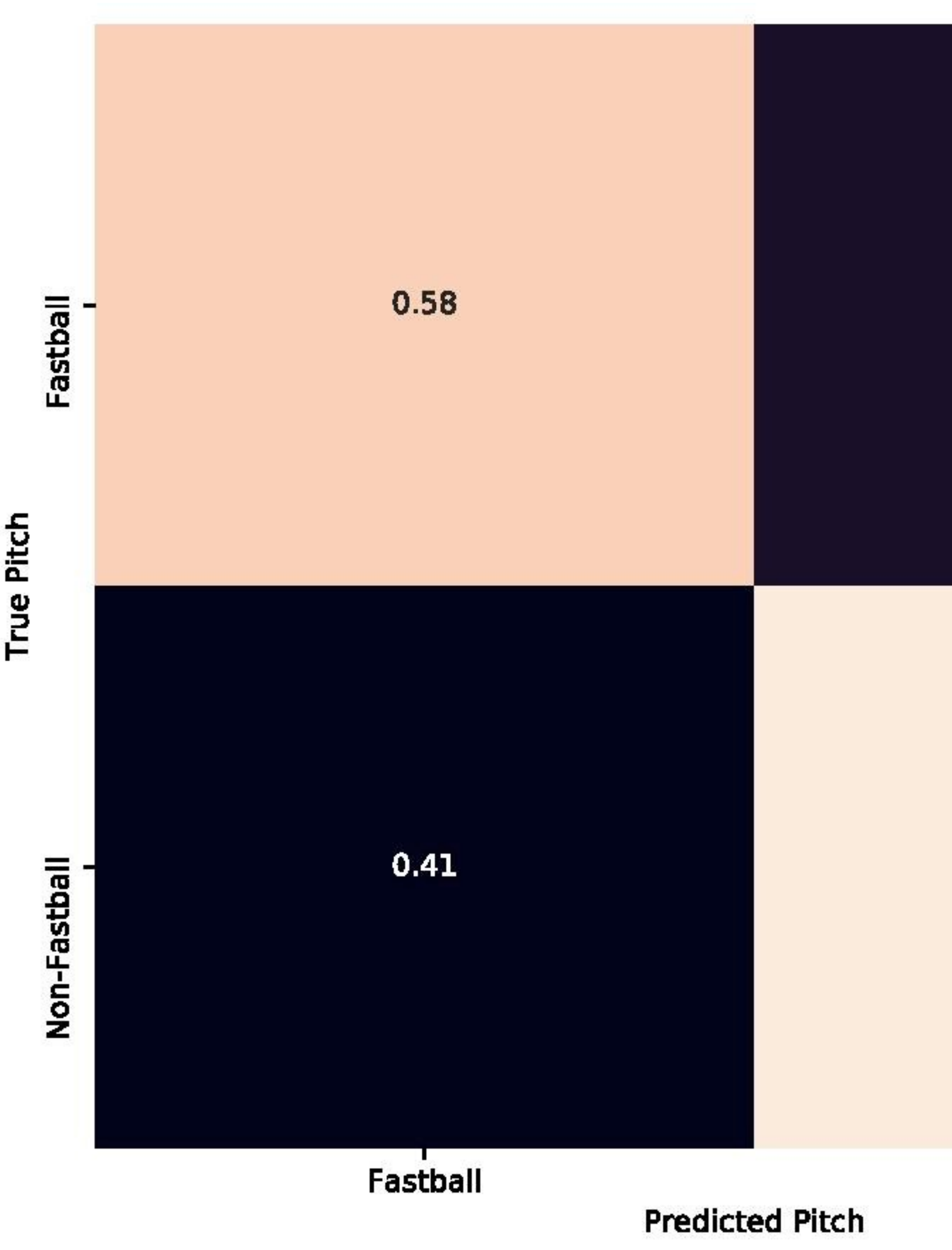
The Jose Berrios Models

These models achieved better accuracy than our other approaches, with the multi-pitch classifier
accuracy. It is unclear to me why there is such a discrepancy between these two classifiers, but le



True Pitch
FT
FF
CU
CH
0.37
0.097
0.32
0.27
0.19
0.32
0.18
0.094
0.52
0.23
0.093
0.16
FT
FF
CU
Predicted Pitch

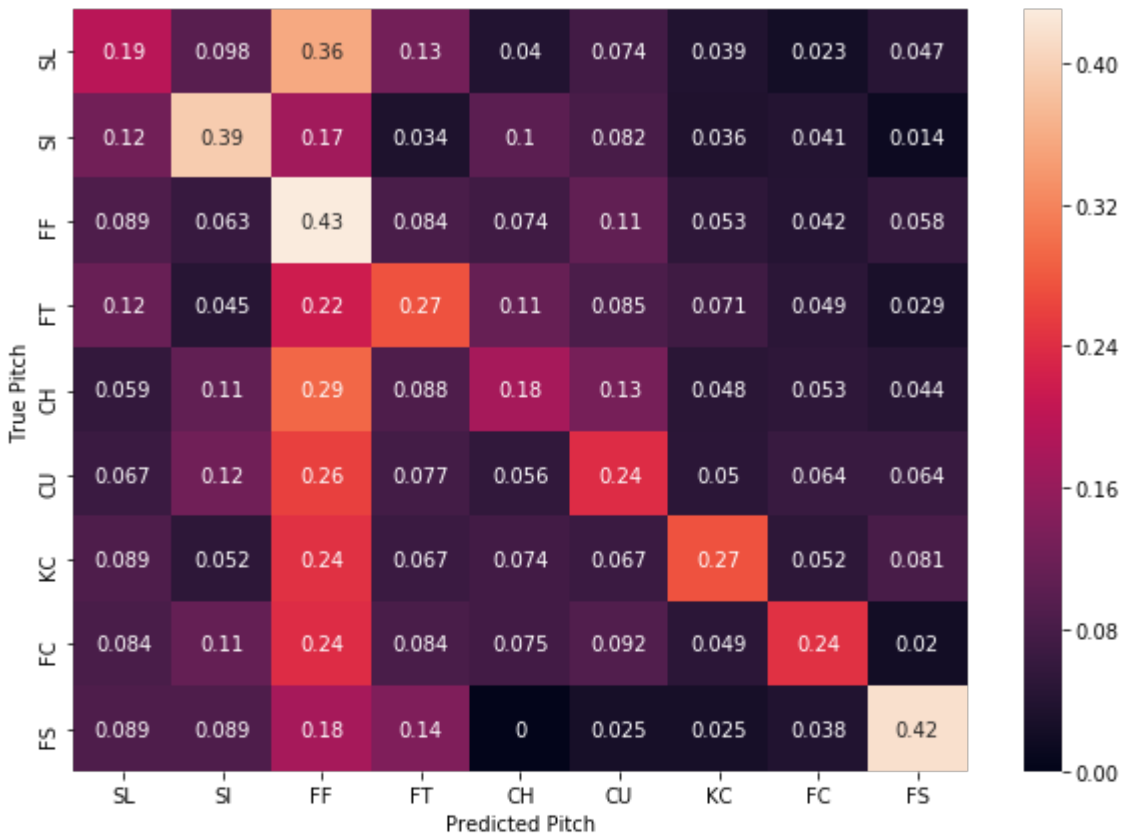
Looking at the confusion matrix above, we can see how many of the pitches of a given type matc
predicted 52% of the curveballs thrown. Looking across the rows of the matrix tells you what perc
(labelled “CU”), we can see that 52% the model predicted successfully, while 20% of the time it pre
predicted at 18% of the time, followed by a four-seam fastball at about 10% of the time. If you turn
accuracy of this model was pretty good, turning its predictions into actionable results would be d


0.58
0.41
Fastball
Non-Fastball
True Pitch
Fastball
Predicted Pitch

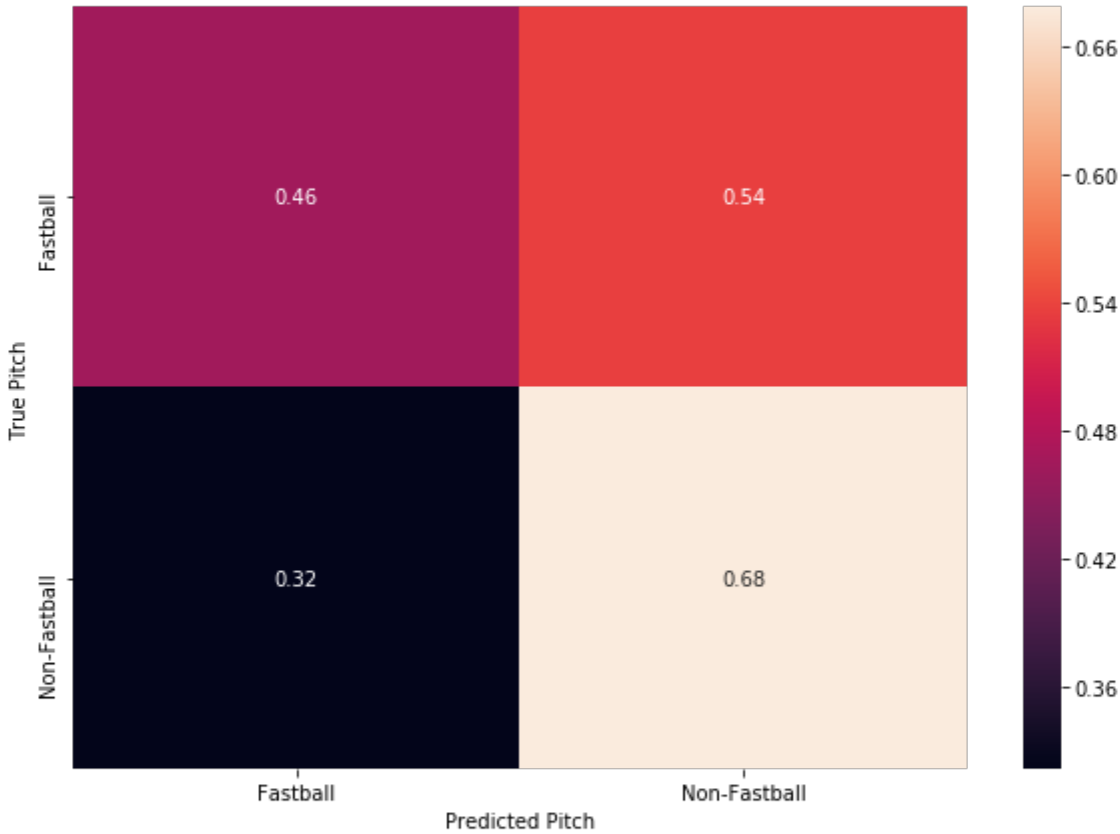
Above is the confusion matrix for the fastball/non-fastball classifier. As you can see, our true pos multi-pitch classifier, the outcome was relatively balanced. This is a good thing, as we want to rec worse outcome for our trash can banger: only about 60% of the time they would be correct. Mayb

The League-Wide Models

The league-wide classifier for all pitches was 88% accurate on the testing set, but as before, let's



As you can see, there isn't that much improvement in terms of improving the trash-can bang's acc is going to be thrown 43% of the time it actually is, with it predicting that a curveball or a slider is correctly predicting that the pitch is going to be either a four or two-seam fastball is only about 5 the fastball/non-fastball classifier will produce better results this time?



So, obviously this isn't a great outcome either. The test set accuracy was 57%, and as the confusi would bang 54% of the time there was a fastball coming, when it is only supposed to bang on a n but still had a false negative rate of 32%. This means that the trash can wouldn't be banged 32% o

Conclusion

Coming into this project, I thought that machine learning might be able to predict what the next pi the case. Even with some models producing an accuracy of greater than 75%, the false positive a architectures of neural networks and more data surrounding each pitch could possibly produce b know what the next pitch will be is to cheat.

Checkout the code here: https://github.com/parkererickson/baseballDataScience/blob/master/n