**Predicting Pitch Type**

The purpose of the project is to build a model with the eventual goal of predicting the probability of a fastball, slider, etc., In a real-time environment. The Data contains all pitches from the 2011 season and the associated metadata describing the data.

I divided the project into Four parts:

1. Exploring and Cleaning the Data.
2. Feature Engineering
3. Choosing and building the Model
4. How to further improve the Model

**Exploring and Cleaning the Data:**

First I downloaded the zip file on my local machine and extracted the csv file and uploaded it to my anaconda and now I will read to make sense of the data While that may look like I am throwing away a large portion of the dataset by dropping all of these columns, many of these columns were almost completely null values. The remainder were either so sparse they contributed very little data or were in my view unrelated to the current prediction task.

**Feature Engineering:**

Using LabelBinarizer form scikit learn, I have replaced the categorical (object type) data into columns of binary integers, this will preserve the information of these columns while still allowing them to be processed by the random forest. Now I will drop the original columns from the dataset and replace them with the numpy arrays generated by Label Binarizer.

**Building the Model:**

now it's time to choose a model for our dataset. I had several thoughts on which model I should choose, since I am concerned with overfitting and since Random Forest Random forest is an ensemble method in which a classifier is constructed by combining several different Independent base classifiers. I decided to go with Random Forest. We can see that it overfitted on the training data, but we were able to get 93.5% accuracy on the test set, there are some things I would incorporate in future models as I deploy them in the real world.

**How to further improve the model:**

To further improve upon the model of predicting the next pitch we can also perform grid search using ROCscore as the metric to find optimal parameters for our model. We can compare results with other classification models. It is also possible to add last year’s data for each player. We can try to combine data from specific batter-pitcher matchups to fine-tune the predictions Try to pay attention to the sequence aspect present in real games so that the model can be aware of what the pitcher threw in the past 2 or 3 pitches as pitchers are known to set up pitches