1. Problem Statement

- The goal of this project is to build a model that attempts to accurately predict when a
- pitcher will throw a fastball. The targeted clients for this model are professional,
- semi-professional, and collegiate baseball teams, including the players, managers, and
- other personnel. The organizations could use this data to inform their batters to
- anticipate fastballs under certain circumstances, which in turn should lead to the team
- hitting more home runs.

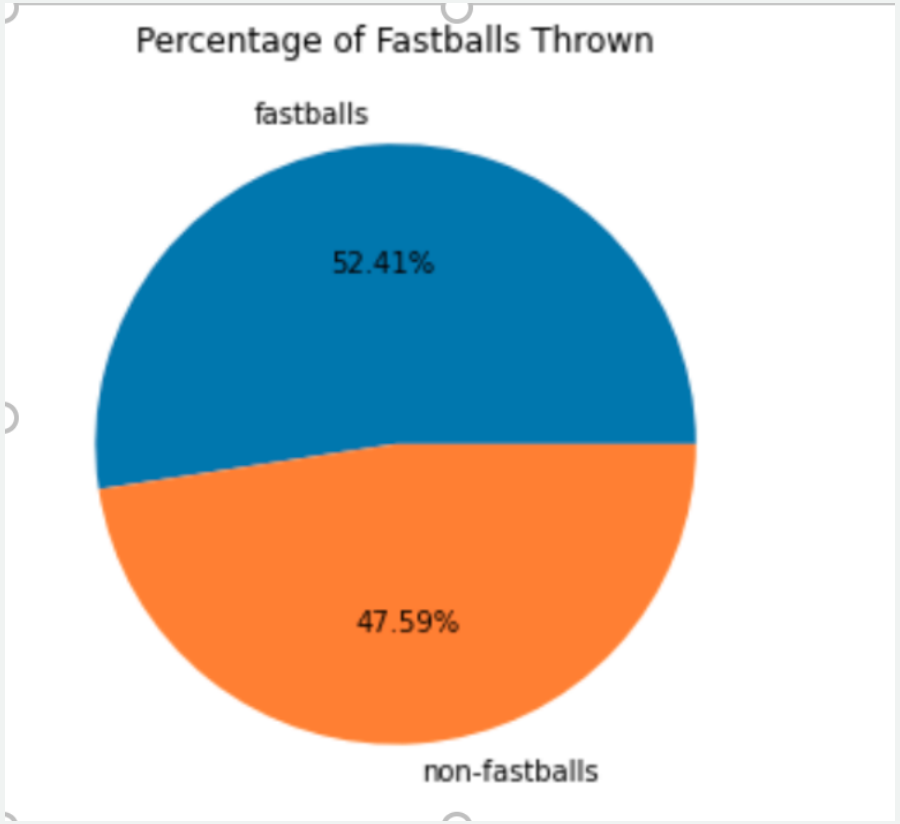
II. The Datasets

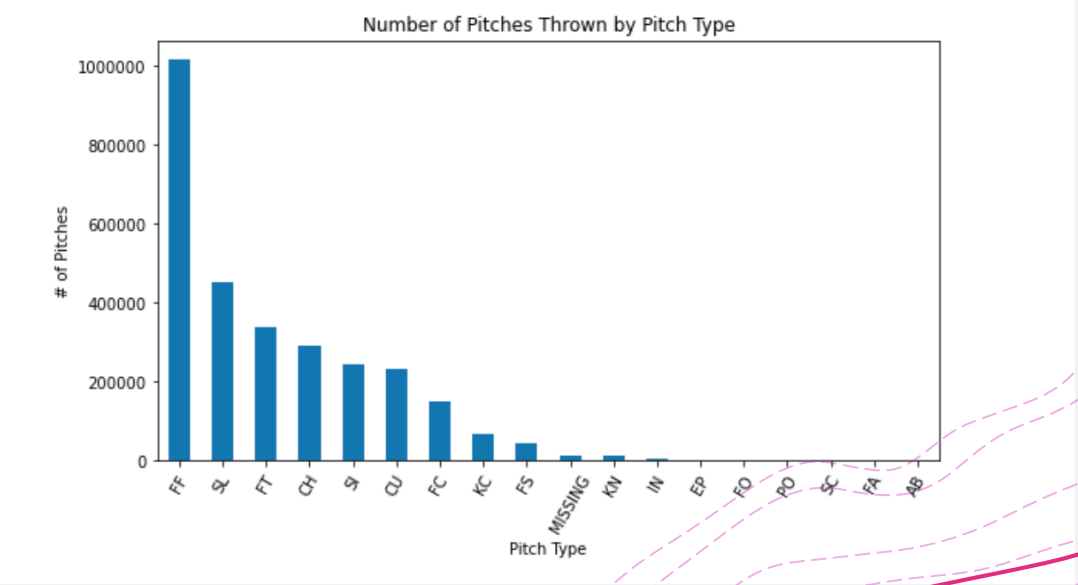
- This project uses the MLB Pitch Data 2015-2018 dataset that is publicly available on
- The first file, pitches.csv, charts various data for each pitch thrown during each of the four seasons from 2015 through
- The second file, atbats.csv, contains various static data for each at-bat from each of the four seasons from 2015 through
- The dataset
- create a new binary variable to classify each of these three types of fastballs as a

III. Exploratory Data Analysis

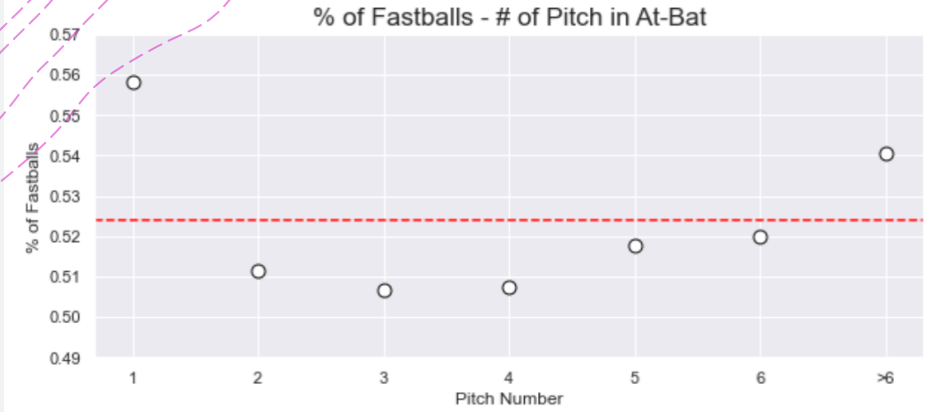
Pre-Pitch Categorical Variables

- The second type of features in the dataset are pre-pitch categorical variables
- continuous variables discussed above; they are known prior to the pitch being thrown
- categorical data variables
- The most significant influence on fastball usage is whether there is at least one runner
- Fastball usage decreases depending upon the number of outs





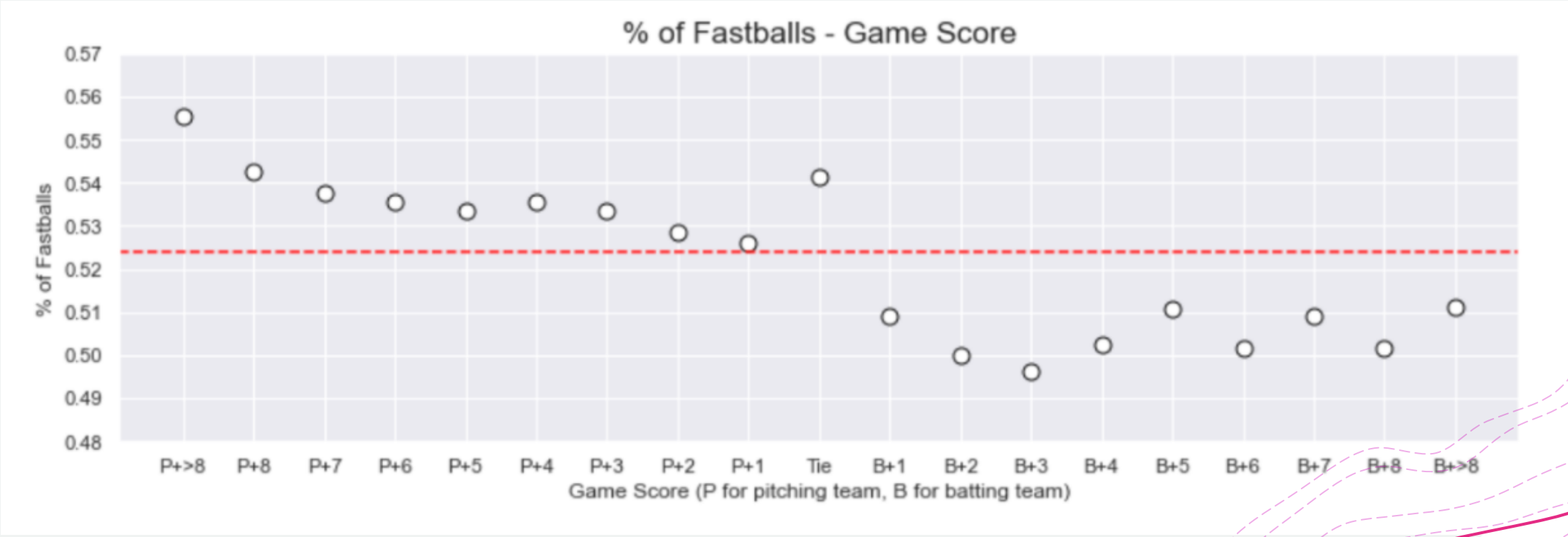
Pitch Sequence

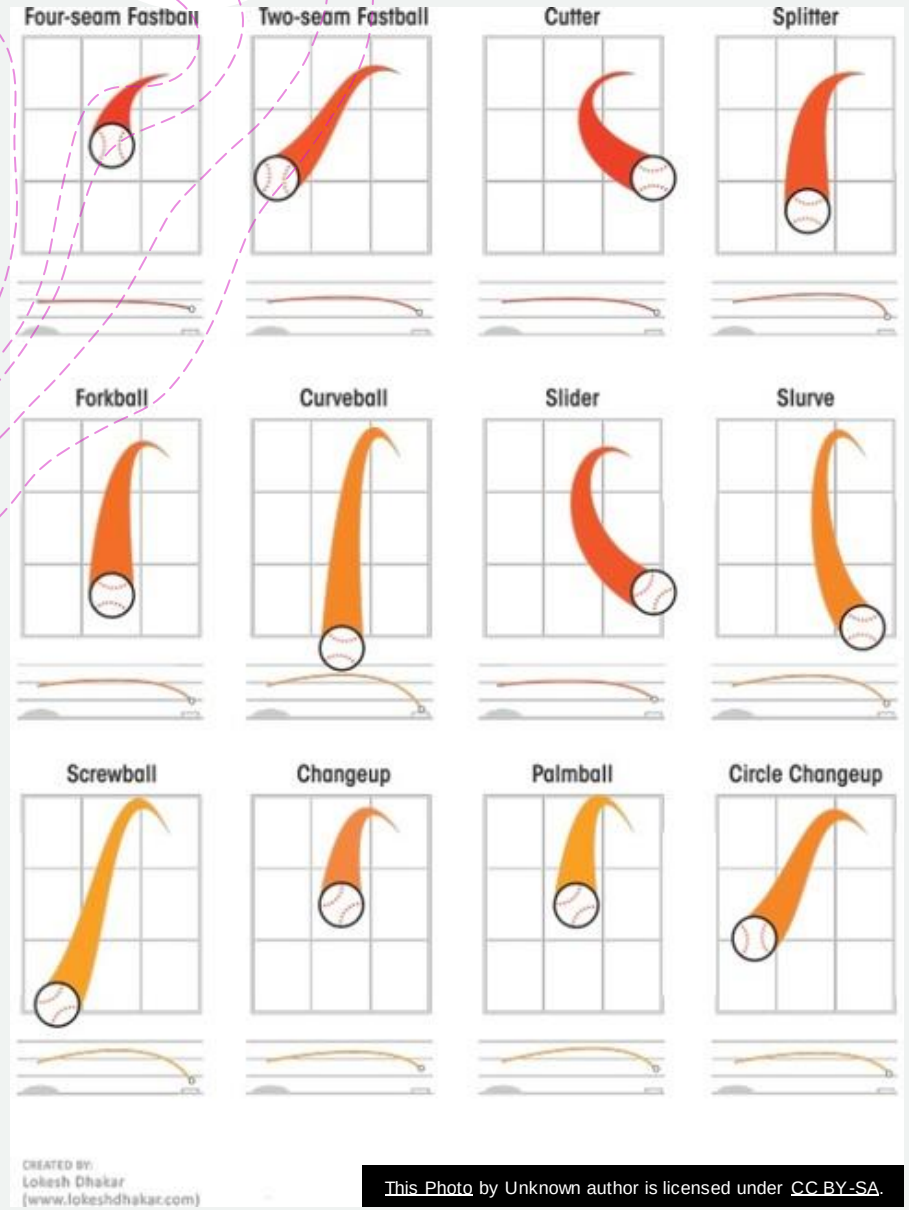


+ Pitchers throw more than average amounts of fastballs by a considerable amount on the
+ first pitch of an at-bat and any pitch after the sixth pitch in a long at-bat
+ Fastballs are thrown higher than average in the early innings and the ninth inning
+ Right-handed pitchers throw fastballs more often than left-handed pitchers regardless of

Game Score

+ Pitchers throw more fastballs than usual when their team is winning, or the game is tied
+ significantly less fastballs when their team is behind
+ game circumstances and the likelihood that a pitcher will throw a fastball on the next



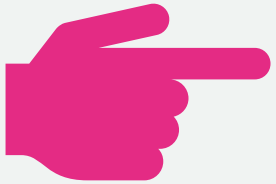


IV. Initial Machine Learning Models Test Run

+ At this point in the project, I decided to feed the dataset into four classifiers to see how they would perform
+ I elected to feed the classifiers approximately 5% of the data. 145,000 pitches
+ The initial models yielded the following accuracy on the test set from
+ the sample data:
+ ● Logistic Regression Classifier: 0.5572
+ ● SGDClassifier: 0.5442
+ ● Random Forest Classifier: 0. 5280
+ ● Gradient Boosting Classifier: 0. 5510

V. Additional Feature Engineering

● Inning Pitch Count - this feature provides how many pitches a pitcher has thrown in a particular inning. I could only tally this efficiently on aninning-by-inning basis (rather than a pitch count for the entire game) given how the data was aggregated in my dataset.



● Previous Pitch Type - this feature provides the pitch type of the immediate previous pitch thrown to the batter using the pitch type labels (Once this data was compiled, I converted it to categorical features using dummy variables)

● Vertical/Horizontal Previous Pitch Location - these two features provide the vertical and horizontal pitch location of the immediate previous pitch to a batter in a particular at-bat. These were built off of the ‘px’ and ‘pz’ continuous variables.

VI. Second Machine Learning Models Test Run

+ Having added the previous pitch features to the dataset, I fed the updated 5% sample data into some models to see if I had improved performance significantly improved models
+ Here are the resulting test set accuracy numbers for each model:

- Logistic Regression Classifier: 0.601517
- SGDClassifier: 0.591917
- Random Forest Classifier: 0.569545
- Gradient Boosting Classifier: 0.601848

VII. Third Machine Learning Models Test Run

+ GridSearchCV is used to optimize our classifier and iterate through different parameters to find the best model. I used this on all our models to see if this would improve the models' accuracy.
+ Here are the resulting test set accuracy numbers for each model:
+ Logistic Regression Classifier: 0.656
+ SGDClassifier: 0.648
+ Random Forest Classifier: 0.62
+ Gradient Boosting Classifier: 0.604

Random Forest Classifier Feature Importance

+ The Random Forest classifier only produces positive “feature importance” values to each model feature on a normalized scale. These values will indicate to us which features are most important to the model.

	coefficient	rfc_feature_importance
inning_10	-1.055783	0.005890
pitcher_lead_4.0	-0.816536	0.006904
pitcher_lead_-9.0	-0.500451	0.000821
pitcher_lead_-6.0	-0.476489	0.002827
outs_1.0	-0.386676	0.018196
pitcher_lead_5.0	-0.375364	0.005536
inning_5	-0.330019	0.013156
pitch_num_10.0	-0.314202	0.000343
s_count_1.0	-0.313586	0.017046
pitcher_lead_-5.0	-0.294217	0.002742
pitch_num_8.0	-0.281542	0.001725
s_count_2.0	-0.265491	0.014646
pitcher_lead_2.0	-0.248157	0.011091
pitcher_lead_-1.0	-0.241768	0.011143
b_count_1.0	-0.241483	0.015446

	coefficient	rfc_feature_importance
pz_prev	-0.062737	0.157753
px_prev	-0.048638	0.156829
pitch_num	-0.032886	0.070032
stand_num	0.018800	0.029257
top_num	-0.068869	0.027969
p_throws_num	0.058830	0.025329
outs_1.0	-0.047877	0.024299
prev_fastball	0.614518	0.023597
outs_2.0	0.023413	0.022506
on_1b	0.109228	0.022212
pitcher_lead_0.0	-0.061119	0.020516
b_count_1.0	-0.015333	0.019855
on_2b	-0.070866	0.019732
s_count_1.0	-0.047185	0.017912
pitch_num_2.0	0.070412	0.016219

IX. Summary/Conclusions

- This project demonstrates that it is possible to analyze pre-pitch circumstances to predict whether a pitcher is going to throw a fastball as the next pitch at a higher accuracy rate than the default frequency percentage of 52.41%.
- Logistic Regression Classifier should be used for most circumstances given its overall superior performance. As explained in detail above, the Logistic Regression Classifier performed substantially better in identifying non-fastballs and generated higher accuracy numbers on most of the model features. It also appears to have more potential to improve through future work.

X. Future Work for Potential Model Improvements

- I believe that I potentially could improve the performance of the final models generated by this project with some additional work like:
  - Adding more previous pitch data as additional features. I was only able to add certain previous pitch features to the dataset given how much time/computational power it took to build them
  - Adapt the model so it can predict pitcher specific pitches.