Pitch Prediction

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

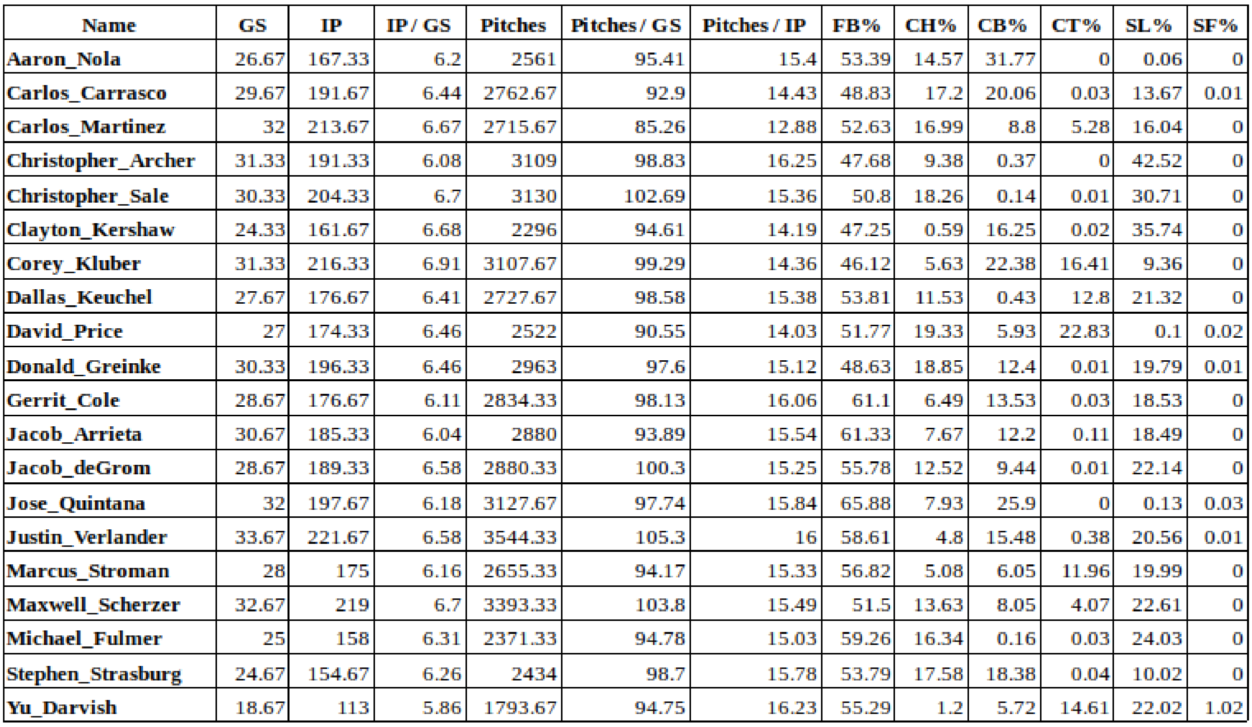
Capstone I Project: In-Depth Analysis
Springboard Data Science Career Track

Mark Rojas

January 2019

1. Major League Baseball (MLB) Pitchers

Each pitcher may exhibit some tendency to use a certain pitch based on what they previously threw. Because of this, it was necessary to analyze the data for each pitcher separately.



Name	GS	IP	IP / GS	Pitches	Pitches / GS	Pitches / IP	FB%	CH%	CB%	CT%	SL%	SF%
Aaron_Nola	26.67	167.33	6.2	2561	95.41	15.4	53.39	14.57	31.77	0	0.06	0
Carlos_Carrasco	29.67	191.67	6.44	2762.67	92.9	14.43	48.83	17.2	20.06	0.03	13.67	0.01
Carlos_Martinez	32	213.67	6.67	2715.67	85.26	12.88	52.63	16.99	8.8	5.28	16.04	0
Christopher_Archer	31.33	191.33	6.08	3109	98.83	16.25	47.68	9.38	0.37	0	42.52	0
Christopher_Sale	30.33	204.33	6.7	3130	102.69	15.36	50.8	18.26	0.14	0.01	30.71	0
Clayton_Kershaw	24.33	161.67	6.68	2296	94.61	14.19	47.25	0.59	16.25	0.02	35.74	0
Corey_Kluber	31.33	216.33	6.91	3107.67	99.29	14.36	46.12	5.63	22.38	16.41	9.36	0
Dallas_Keuchel	27.67	176.67	6.41	2727.67	98.58	15.38	53.81	11.53	0.43	12.8	21.32	0
David_Price	27	174.33	6.46	2522	90.55	14.03	51.77	19.33	5.93	22.83	0.1	0.02
Donald_Greinke	30.33	196.33	6.46	2963	97.6	15.12	48.63	18.85	12.4	0.01	19.79	0.01
Gerrit_Cole	28.67	176.67	6.11	2834.33	98.13	16.06	61.1	6.49	13.53	0.03	18.53	0
Jacob_Arrieta	30.67	185.33	6.04	2880	93.89	15.54	61.33	7.67	12.2	0.11	18.49	0
Jacob_deGrom	28.67	189.33	6.58	2880.33	100.3	15.25	55.78	12.52	9.44	0.01	22.14	0
Jose_Quintana	32	197.67	6.18	3127.67	97.74	15.84	65.88	7.93	25.9	0	0.13	0.03
Justin_Verlander	33.67	221.67	6.58	3544.33	105.3	16	58.61	4.8	15.48	0.38	20.56	0.01
Marcus_Stroman	28	175	6.16	2655.33	94.17	15.33	56.82	5.08	6.05	11.96	19.99	0
Maxwell_Scherzer	32.67	219	6.7	3393.33	103.8	15.49	51.5	13.63	8.05	4.07	22.61	0
Michael_Fulmer	25	158	6.31	2371.33	94.78	15.03	59.26	16.34	0.16	0.03	24.03	0
Stephen_Strasburg	24.67	154.67	6.26	2434	98.7	15.78	53.79	17.58	18.38	0.04	10.02	0
Yu_Darvish	18.67	113	5.86	1793.67	94.75	16.23	55.29	1.2	5.72	14.61	22.02	1.02

Table 1: List of Starting Pitchers

For each starting pitcher, the table includes the average number of Games Started (GS), Innings Pitched (IP), Pitches Thrown (Pitches) over the last three years (2016-2018). Also shown, the average percent of pitch types (Fastball: FB, Changeup: CH, Curveball: CB, Cutter: CT, Slider: SL, and Split-finger: SP) thrown.

Not shown, the average percent of Non-pitches, Intentional Balls, and Unknown pitches.

2. Train / Test Split

The target data consists of the following six (6) pitch-types:

(**FB**) *Fastball* (**CH**) *Change-up* (**CB**) *Curveball*
(**CT**) *Cutter* (**SP**) *Split-finger*(**SL**) *Slider*

The type of pitch thrown will vary from pitcher to pitcher. To predict the “next” pitch, multi-class classification will be required. As seen in **Table 1** above, the target data is imbalanced. For these 20 pitchers of interest and most MLB starting pitchers, on average, 46% - 66% of all pitches thrown are Fastballs. This is between 8X to 5000X greater than at least one of the other pitch types (CH, CB, SL, CT, SP) they throw as part of their arsenal. Attempts to use Synthetic Minority Over-sampling Technique (SMOTE) and stratify the training data to establish a balanced data set did not improve predictive accuracy. A copy of the target data for each pitcher were also binarized for plotting Precision-Recall curves for each class rather than using a single ‘micro’-average score.

3. Feature Selection

3.1. Single Unique Features

A feature with only one unique value cannot be useful for machine learning because this feature has zero variance. Such single unique features were removed. For example, the feature, 'is_pitcher_righty' will not change. If a pitcher is right-handed, the feature will always be 1.

3.2. Correlated Features

Identified and removed features (columns) that had a positive or negative correlation of 95% or greater. A total of 21 unique features (shown in **Table 2**) were removed from 1 or more pitchers data set due to high correlation.

Feature Removed	Feature Description	# of Pitchers
o_id_nPCH	**Pitch outcome:** Non-pitch	20
prev_2_pitches_NP_NP	**Encoded:** Non-pitch followed by Non-pitch	20
IB_cum%	**Cumulative % Pitch-type:** Intentional Ball	14
h_type_NH	**Hit outcome:** No Hit	8
prev_2_pitches_IB_IB	**Encoded:** Intentional Ball followed by Intentional Ball	8
PI_cum%	**Cumulative % Pitch-type:** Pitch Out	7
o_id_oPO	**Pitch outcome:** Pop Out	7
prev_2_pitches_FB_UN	**Encoded:** Fastball followed by Unknown Pitch	5
UN_cum%	**Cumulative % Pitch-type:** Unknown	4
CT_cum%	**Cumulative % Pitch-type:** Cutter	4
prev_2_pitches_FB_PI	**Encoded:** Fastball followed by Pitch Out	2
o_id_bPO	**Pitch outcome:** Pitch Out	1
SF_cum%	**Cumulative % Pitch-type:** Split-finger	1
prev_2_pitches_CH_IB	**Encoded:** Change-up followed by Intentional Ball	1
prev_2_pitches_CH_PI	**Encoded:** Change-up followed by Pitch Out	1
prev_2_pitches_CT_PI	**Encoded:** Cutter followed by Pitch Out	1
prev_2_pitches_FB_SF	**Encoded:** Fastball followed by Split-finger	1
prev_2_pitches_SL_UN	**Encoded:** Slider followed by Unknown Pitch	1
prev_2_pitches_CH_UN	**Encoded:** Change-up followed by Unknown Pitch	1
prev_pitch_grp_code	**Encoded:** One of three groups (Fastballs, Change-up, Curveballs) previous pitch falls into	1
KN_cum%	**Cumulative % Pitch-type:** Knuckleball	1

Table 2: List of Correlated Features Removed

3.3. Feature Importance

Features with zero-to-low importance for each pitcher were removed. Feature importance was determined using XGBoost Classifier. To avoid overfitting, multiple evaluation metrics (**merror** = *multi-class error rate*, **mlogloss** = *multi-class negative log-likelihood*) were averaged over 10 iterations to identify when early stopping should occur. Figure 1 shows results for both metrics for the first iteration with a 57.41% accuracy for pitcher, Chris Archer.

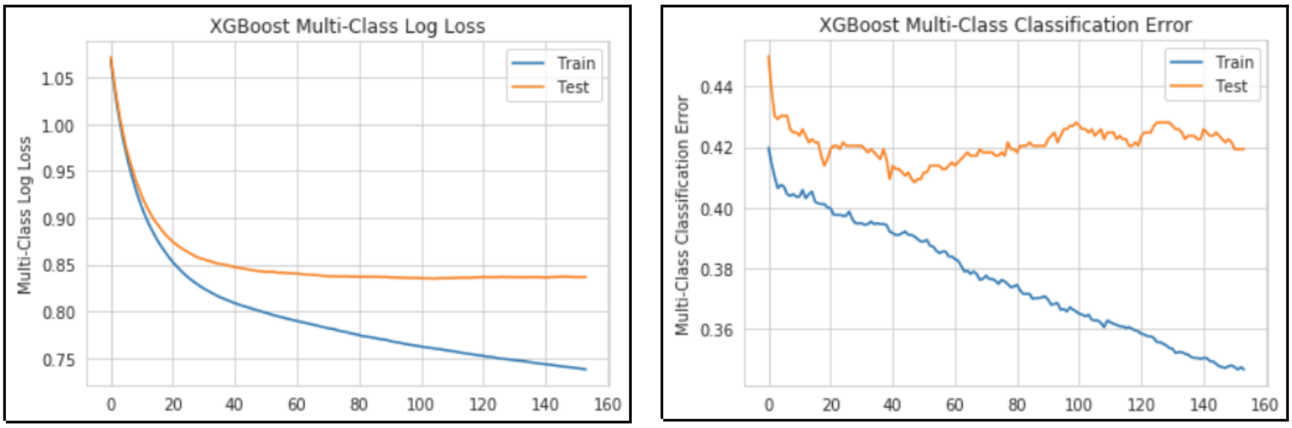


Figure 1: First Iteration Evaluation Metrics Comparison for Chris Archer

<u>Left</u>: Multi-class Log Loss Evaluation using XGBoost Classifier. <u>Right</u>: Multi-class Error Rate (#wrong cases /# all cases) Evaluation using XGBoost Classifier. In this case, early stopping can be applied around the 45-50 epoch.

The most important feature for every pitcher was the Pitcher's Pitch Count (p_p_count) which is the number of pitches the pitcher has thrown in a game. The top 20 features (Figure 2) are illustrated for Chris Archer.

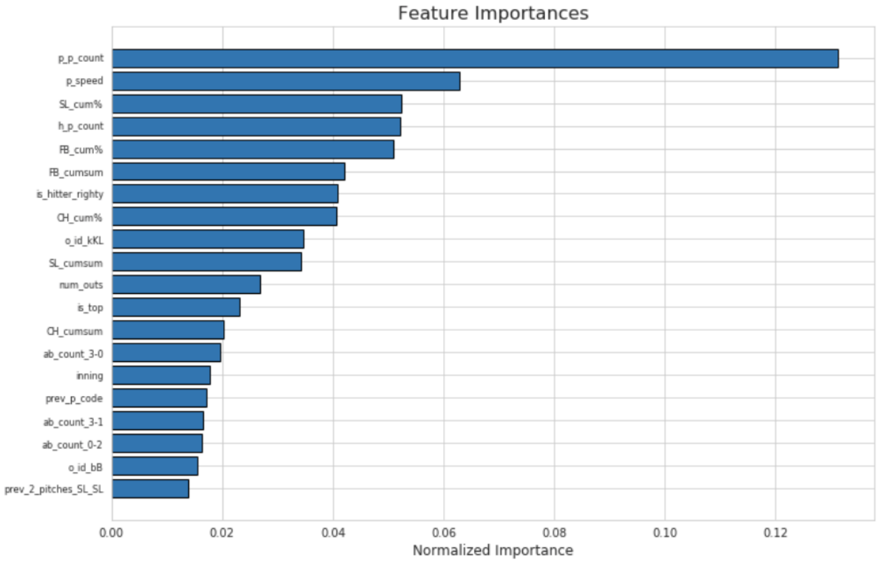


Figure 2: Top 20 Important Features for Chris Archer

With features importance sorted, only the features that contributed to 99% of cumulative importance were kept. For Chris Archer, only 67 of the 100+ features were required for 99% cumulative importance as shown in Figure 3.

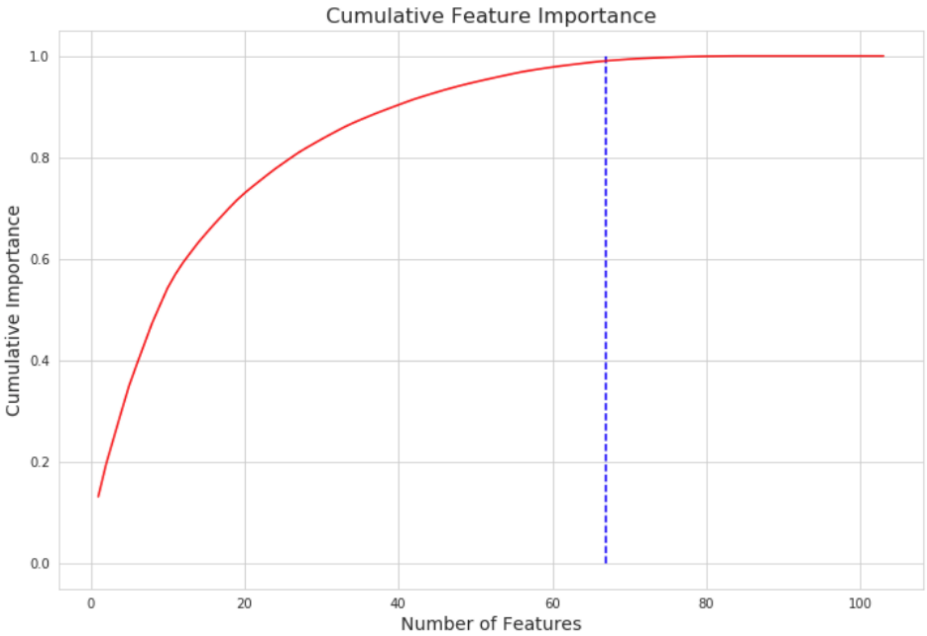


Figure 3: Cumulative Feature Importance Curve for Chris Archer

4. Compare Models

4.1. Naive Approach

The naive approach to predicting the next pitch-type would be to only consider the previous pitch. There are two ways to apply the naive approach.

1. **Consider the distributions for “first” pitches**
    a) At the beginning of every inning, there is a “first” pitch – when there is not a previous pitch
2. **Consider the distributions for two-pitch sequences**
    a) Previous Pitch → Next Pitch

When we consider the distribution of “first” pitches for Carlos Martinez, shown in Figure 4, we see that he throws a Fastball more than 60% of the time. The next probable pitch could be either a Change-up or Curveball at 12%.

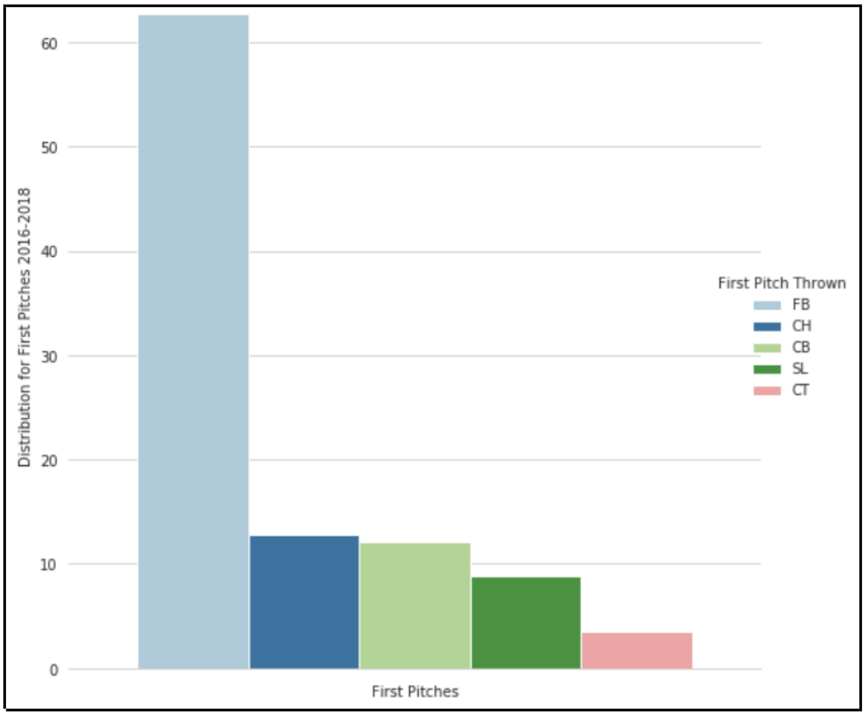


Figure 4: Distribution of First Pitches for Carlos Martinez (2016-2018)

However, when we consider the distribution for two-pitch sequences for Carlos Martinez, shown in Figure 5, we now see that after he throws a Fastball, more than 30% of the time, he follows it up with another Fastball. It is also interesting to see that when Carlos throws a Slider or Cutter, he almost never follows it up with a Curveball.

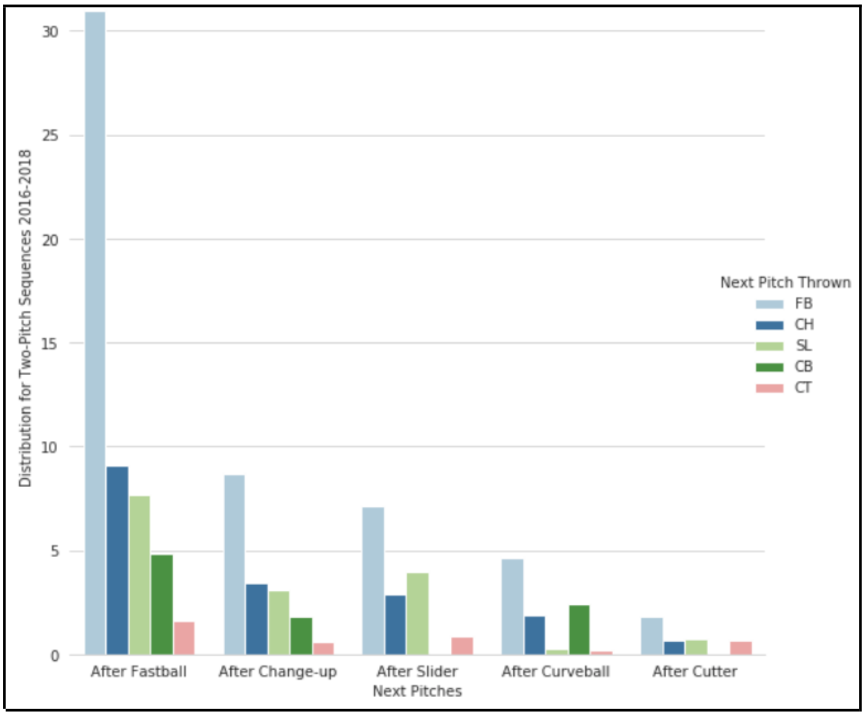


Figure 5: Distribution for Two-Pitch Sequences for Carlos Martinez (2016-2018)

4.2. Test ‘Out-of-box’ Models

To determine the “best” model to use for predictive analysis, I first looked at eight “out-of-box” models by implementing a pipeline to normalize feature vectors into a representation more suitable for the downstream estimators and run 5-fold cross-validation. For Jake Arrieta, when considering the average test score, we can see which of the estimators performed the best:

```
Player 12:  Jake Arrieta
---------------------------------
Logistics Regression : 60.97
SVC - linear : 61.45
SVC - rbf : 61.45
K-neighbors : 57.58
Decision Trees : 45.71
Random Forest : 57.16
Gaussian NB : 11.3
Gradient Boosting : 60.02
Adaboost : 60.64
XG Boost : 61.18
-----------------------------------
```

Based on the average accuracy test scores, I decided to go with **Logistic Regression**, **Support Vector Machine** using *Radial Basis Function*, **Random Forest Classifier**, **Gradient Boosting**, and **Decision Trees**. I chose to keep Decision Trees to include as a “weak” classifier in an Ensemble Voting Classifier.

4.2. Hyper-parameter Tuning

For each of the six (6) estimators selected, I performed an exhaustive search over specified parameter values specific for each estimator using GridSearchCV. The “best_estimator” for each estimator was saved to .pkl file.

4.3. Compare Models

Using an Ensemble Voting Classifier with all six (6) of the “best” estimators, I was able to compare the average class probabilities computed by each estimator against the combined voting classifier. For pitcher, Marcus Stroman who uses five (5) pitches as part of his arsenal, we can see slight differences between the classifiers in Figure 6.

While we see that the proportion of class probabilities is similar across all classifiers, this is not always the case. For some pitchers, we do see where some pitch-types have a higher probability in one classifier than in others.

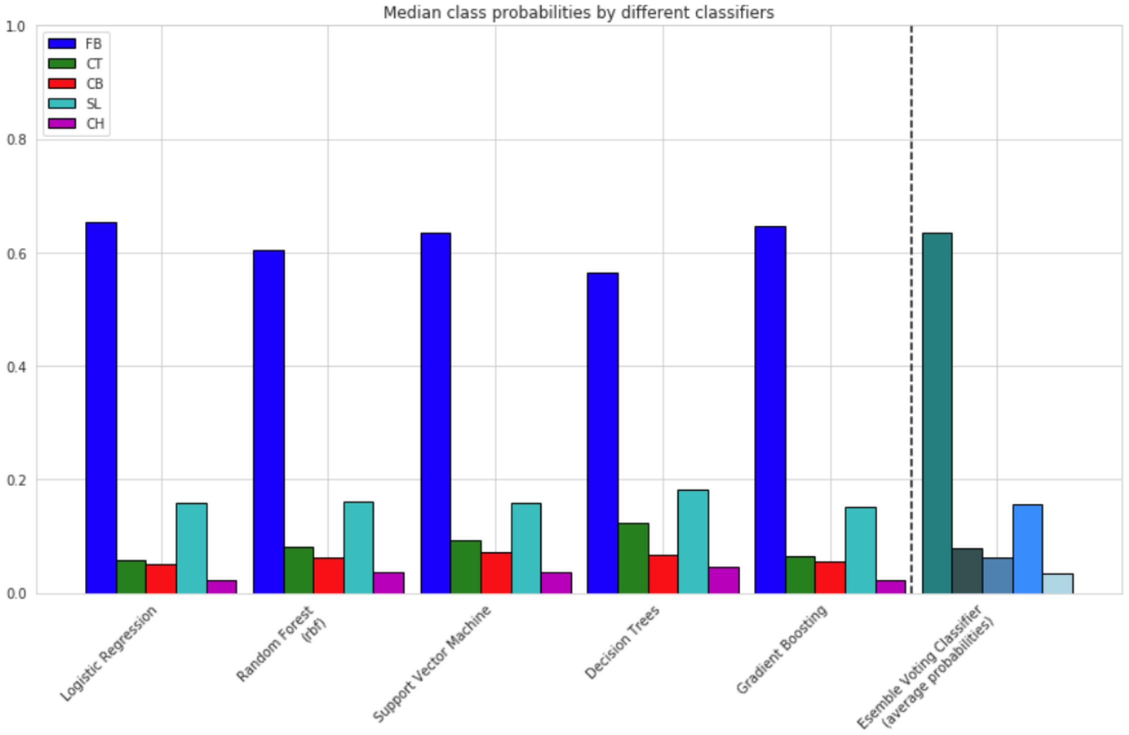


Figure 6: Median class probabilities for each classifier for Marcus Stroman

5. Measurements

To determine if a predictive model is considered “good”, we need to first define the purpose of the model and what is considered success criterion.

- For this project, the purpose of the classification model is to predict the next pitch-type thrown by a starting Major League Baseball pitcher.
- The model can be considered “successful” if it is able to predict the next pitch-type with greater accuracy/probability than the naive approach.

While most classification machine learning models can be validated by accuracy estimation techniques, this is not the case for this project. Because the data is imbalanced, we cannot use accuracy as best indicator. Also, while ROC curves summarize the trade-off between the True Positive Rate and False Positive Rate at different probability thresholds, they are more appropriate for when the observations are balanced. Because this data is imbalanced, Precision-Recall curves are better suited as they summarize the trade-off between the True Positive Rate and the Positive Predictive Value at different probability thresholds.

5.1. Confusion Matrix

To describe the performance of the voting classifier model on the pitcher data set where the true classes (pitch-types) are known, a confusion matrix was used. For Clayton Kershaw, we see in Figure 7 that most of the pitch-types predicted are Fastballs (FB) with Sliders (SL) being the second most predicted pitch-type. As a result, the model is unable to effectively distinguish between FB’s, SL’s, or CB’s.

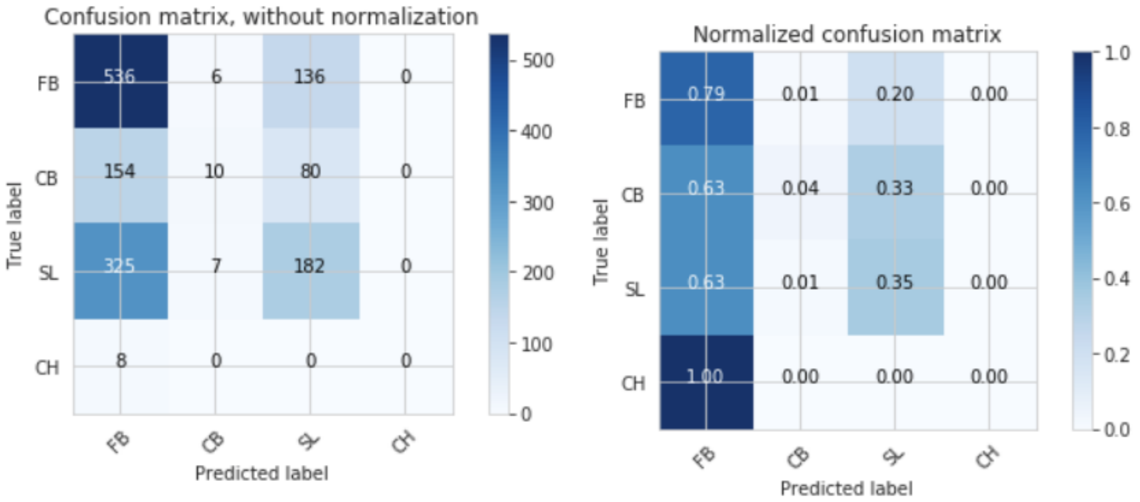


Figure 7: Confusion Matrix for Clayton Kershaw

Left: Raw predicted vs true values per class. Right: Normalized predicted vs true values per class.

5.2. ROC curve

While ROC curves are not appropriate for this data set due to being imbalanced, using the One vs. All classifier model, we see in Figure 8, for Carlos Martinez the ROC curves for each class (pitch-type), including the micro- and macro- averages. Because the data is imbalanced, micro-average ROC curve should be considered.

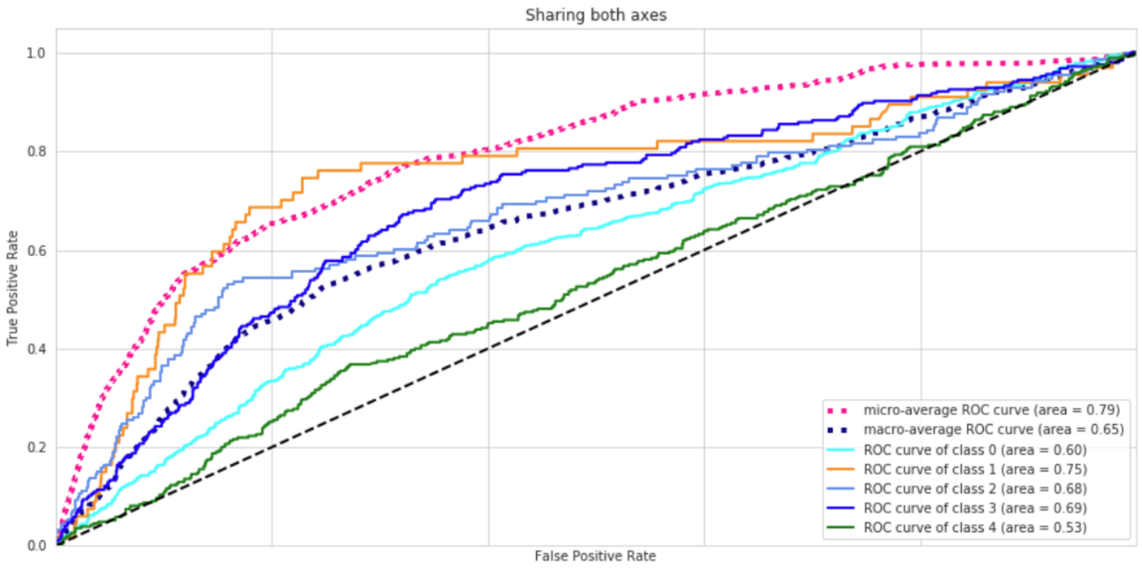


Figure 8: ROC Curve for Carlos Martinez

Pitch-type – class associations are as follows:

Fastball (FB) = 0, Cutter (CT) = 1, Curveball (CB) =2, Slider (SL) = 3, and Change-up (CH) = 4

5.3. Precision-Recall

In Figure 9, the Precision-Recall curve for Jose Quintana shows the micro-average – quantifying score on all classes jointly. We see that the micro-average is 0.66.

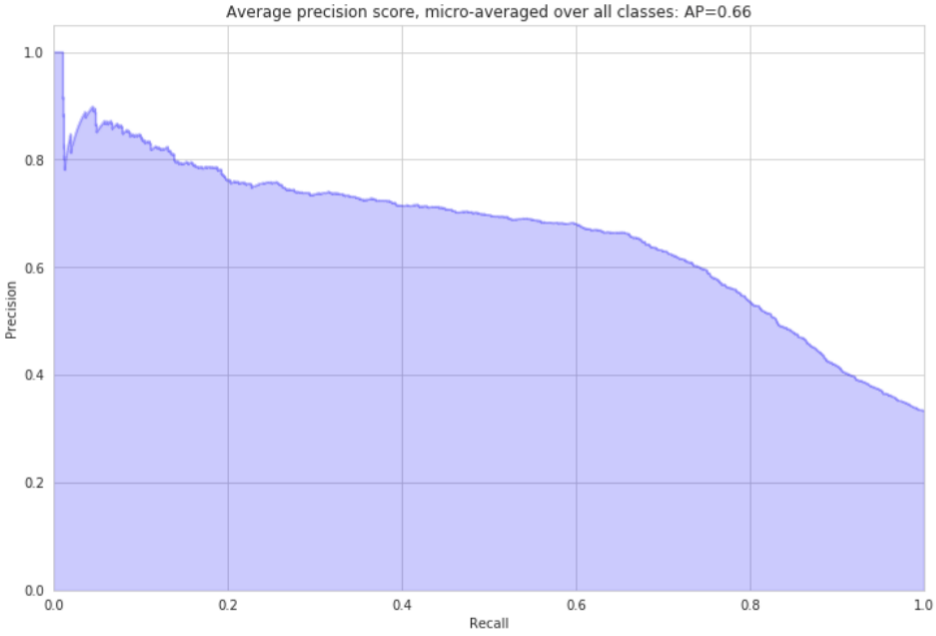


Figure 9: "Micro-average" Precision-Recall Curve for Jose Quintana

In Figure 10, we see the Precision-Recall curve and average precision score for each class (pitch-type) for Jose Quintana. The F1 score was also computed to be 0.54.

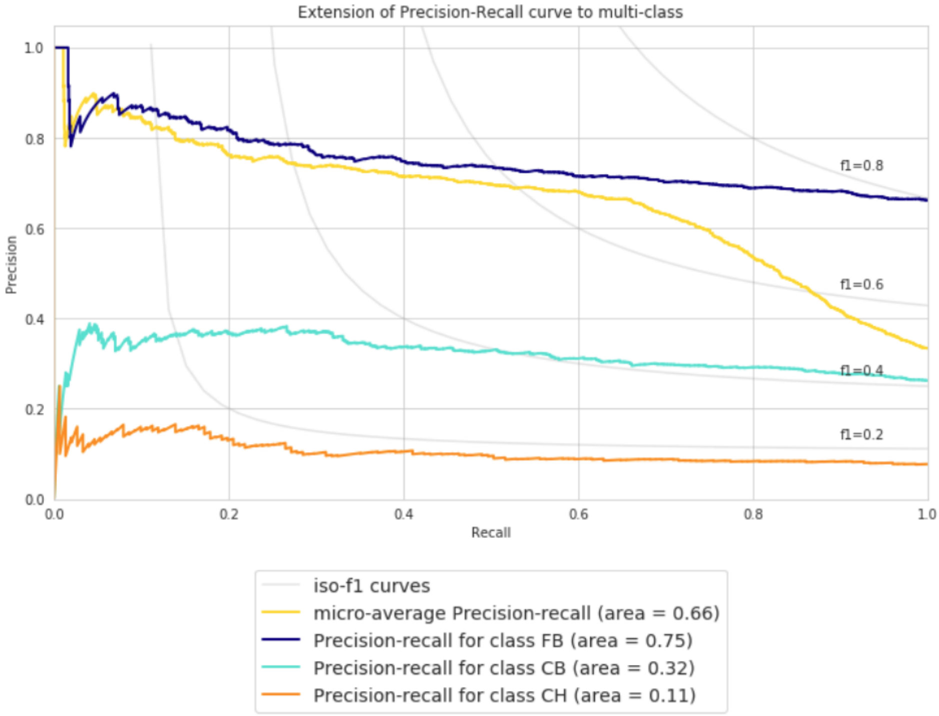


Figure 10: Precision-Recall Curve for each Class for Jose Quintana

7. Summary

Identifying and fitting the different models to the pitch data set for each pitcher was an extremely interesting and learning experience. Based on the sample size of 20 pitchers over 3 years, there are a couple of observations we can take away.

1. It does appear that we can predict the next pitch with a slightly higher probability than the naive approach for Fastballs, but much more difficult for other pitch-types.
2. For some pitchers, we are able to predict the next-pitch more accurately than for others.
   a) This may be due to the number of classes (pitch-types)
      - Pitchers with more pitches in their arsenal make it more difficult to predict non-Fastballs

It is also important to note that only supervised learning approaches were used in this project and that unsupervised and reinforcement learning should be considered.

Some features such a score differential, hitters statistics, weather conditions, and additional season should be included in further analysis to contribute to training process.