Modeling Hitter Swing/Take Decisions & MiLB Prospect Swing/Take Analysis

Ruslan Davtian

Problem 1

Data Description:

The 2021-train.csv and 2021-test.csv data sets represent pitch-by-pitch level data for many batters and pitchers across professional baseball for parts of the 2019 season. The data provides information such as pitch speed, pitch location, pitch break, pitch release, pitch type, result of play, etc. The training data set contains a total of 1,078,637 pitches and 37 columns while the testing set contains 370,283 pitches and the same number of columns as the training set. The goal of this analysis is to build a model on some subset of the data (training set) to predict whether or not a pitch in the test data set resulted in a swing or take. Below is an output of the first five rows of the training data set.

date	level	pitcher_id	pitcher_side	batter_id	batter_side	stadium_id	umpire_id	catcher_id	inning
2019-04-30	MLB	5081ca93	Right	5bf361ca	Right	402559d3	4ff102e5	65ac286a	1
2019-04-15	MLB	0bf5d3c5	Left	7c67a595	Left	d0d69f32	51a1c7ee	5e710b9e	7
2019-05-26	MLB	0bf5d3c5	Left	de9d396f	Left	0c59f5af	3007964d	44924919	9
2019-05-27	MLB	af735dc4	Left	073c2b16	Right	a3f610ed	af66b76d	b05114c7	7
2019-05-11	MLB	07d9667f	Right	4f03de7c	Right	402559d3	9d34b92a	016c0582	1

top_bottom	outs	balls	strikes	release_speed	vert_release_angle	horz_release_angle	spin_rate	spin_axis	tilt
2	0	3	2	97.55	-2.59	-4.05	2561.21	217.52	1:15
2	2	0	0	87.31	-1.18	4.83	2247.10	177.75	11:15
1	2	2	0	87.97	-0.23	4.34	2248.64	156.80	11:00
1	1	1	0	82.57	0.45	1.91	1663.20	161.95	11:00
1	2	0	1	94.04	-2.25	-3.89	2012.12	190.31	12:30

rel_height	rel_side	extension	vert_break	induced_vert_break	horz_break	plate_height	plate_side	zone_speed	vert_approach_angle
5.79	2.03	6.70	-9.08	19.32	13.31	2.60	-0.54	88.98	-4.34
5.64	-3.49	5.20	-30.06	7.27	-0.12	2.00	1.01	81.05	-6.67
5.50	-3.81	5.34	-33.18	3.83	-1.35	2.47	0.23	80.81	-6.35
5.96	-2.43	6.51	-37.21	2.88	0.07	3.27	-0.70	75.44	-6.60
6.56	2.54	5.78	-16.57	15.05	2.99	3.02	-0.85	86.34	-5.34

Missing Values and Data Manipulation:

After investigating, I found that 17,878 rows in the training data set are missing spin rate and 528 rows are missing pitch location coordinates. The values seem to be missing at random and with a very large sample size, I felt it was appropriate to filter out those rows. Next, I removed the column named y55 because the values are the same for every observation and filtered out a few rows with switch hitters as the batter side. Finally, I filtered out rows where the pitch type is missing.

I created a count column by concatening number of balls and strikes as well as the is_swing response variable column based on pitch call. A swing happens when pitch call equals in play, strike swinging, or foul ball and no swing otherwise. Furthermore, I added two more columns. First, I created a binary level variable that is minors if a player's at bat is in A, A+, AA, or AAA and majors if a player's at bat is in MLB. Second, I created an in zone indicator variable that is 1 if the pitch crossed the plate inside the strike-zone and 0 if the pitch crossed the plate outside of the strike-zone.

Analysis Approach:

My approach is to build a few machine learning binary classification algorithms to predict swing/no swing in the test set and compare each one's performance. I will choose the model with the highest cross validated prediction accuracy as my final model. I chose to compare three algorithms (penalized logistic regression, random forest, and xgboost) using 5-fold cross validation each time on the training set for consistency between the algorithms. Also, I compared each algorithm by computing overall accuracy and kappa (accuracy normalized to account for imbalance of classes) statistics on the training set. I will select the model that maximizes cross validated train accuracy in predictions. Due to the volume of training data that results in long run times, I decided to fit about 20% of the training data to the models. Since there are over one million rows, I am still using over 100,000 observations to build models which are plenty to draw conclusions from.

Balance of Classes:

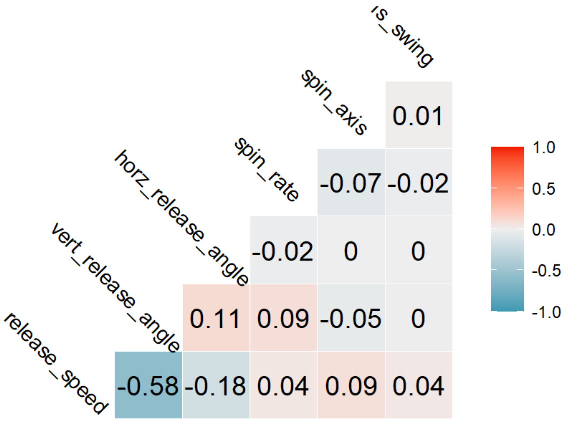
To understand the proportion of pitches resulting in swings or takes, I created a table below displaying the proportions. About 47% of the pitches resulted in swings which means a naive, basic model of randomly guessing swing or no swing would result in about 47% accuracy in predictions. The goal here is to find factors that can explain some of the variation in swing decisions such that when making predictions, the model has an accuracy above 47%.

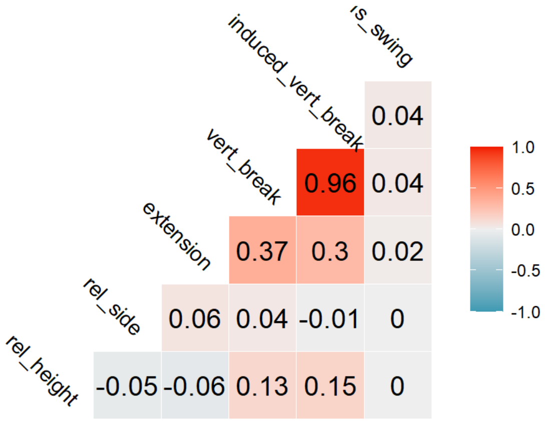
Overall Take%	Overall Swing%
0.534	0.466

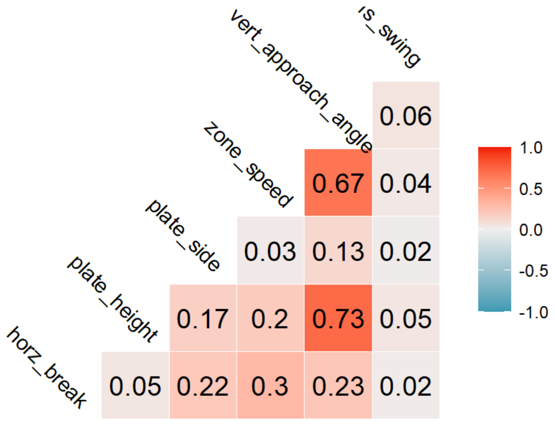
Correlations:

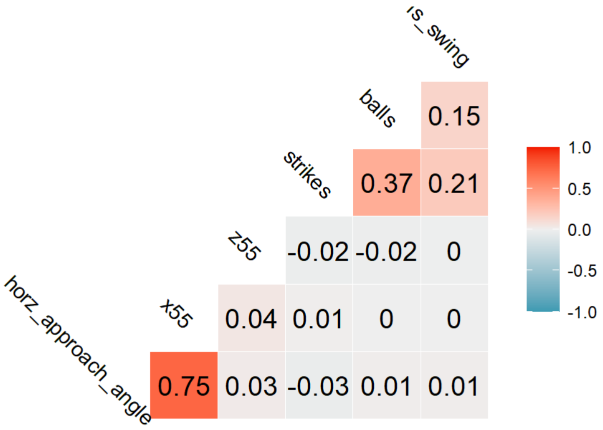
To find which variables are good candidates to include as explanatory variables, I looked into linear correlations among each of the continuous variables against is_swing. Below, is a set of four correlation matrices each with the response variable in the far right corner. To find the correlations between the response variable of interest (is_swing) and any one of the explanatory variables, one needs to traverse down the far right column of each plot. All other cells to the left compare correlations among explanatory variables with each other. This is useful for finding evidence of multicollinearity among the predictors but I will address that issue later.

We can see that most of the candidate variables such as pitch velocity, pitch location, release points, angle and length of breaks, etc. have almost no linear correlation with a batter's swing decision. However, that does not necessarily mean there is no relationship between those candidate variables and the response since uncorrelated does not imply independence. We should not eliminate those variables from consideration when building models. Additionally, the bottom right correlation plot shows that the number of strikes and the number of balls have some positive correlation with the is_swing variable so count may be a strong predictor of a batters swing decision.





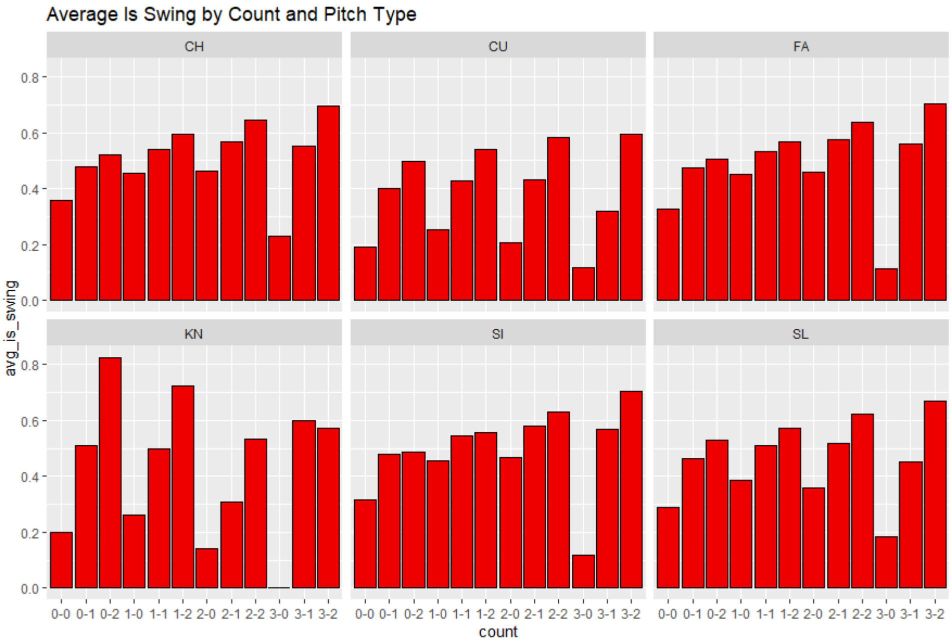




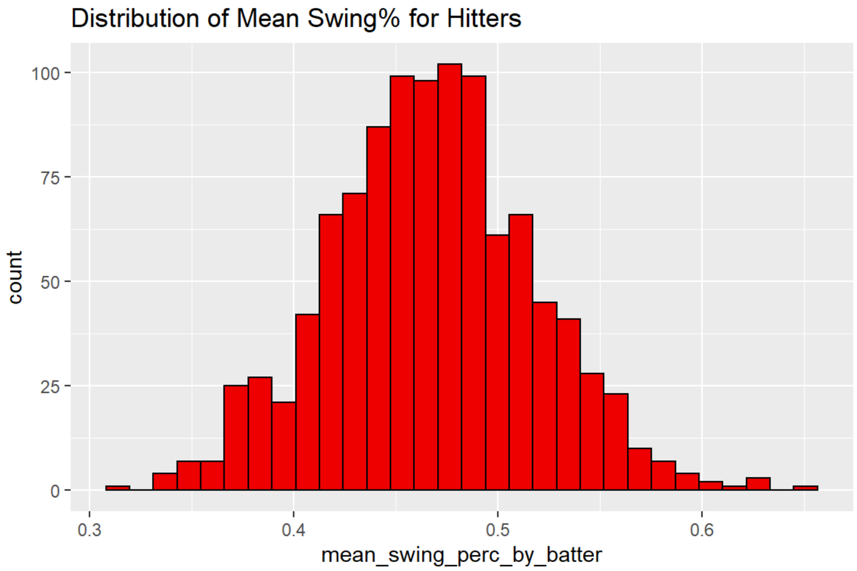
Data Visualizations:

Let’s look more closely at how swing percentage changes with different counts separated by pitch type. Below, are six bar charts where the x-axis represents each of the twelve possible counts and the y-axis represents average swing percentage. Each of the six bar charts represent different pitch types.

Regardless of pitch type, hitters are very reluctant to swing at a 3-0 count which is expected. Also, hitters are typically more patient on the first pitch of the at bat as well as when they are ahead of the count with no strikes. Hitters are most aggressive anytime they have two strikes on them. Clearly, there is an effect of count on a hitter’s swing decision. Taking into account pitch type, there seems to be some effect of pitch type on swing/take decisions. Hitters tend to swing more on fastballs and changeups while swinging less often on curveballs. These plots show that count and pitch type should be included as categorical variables into the models.



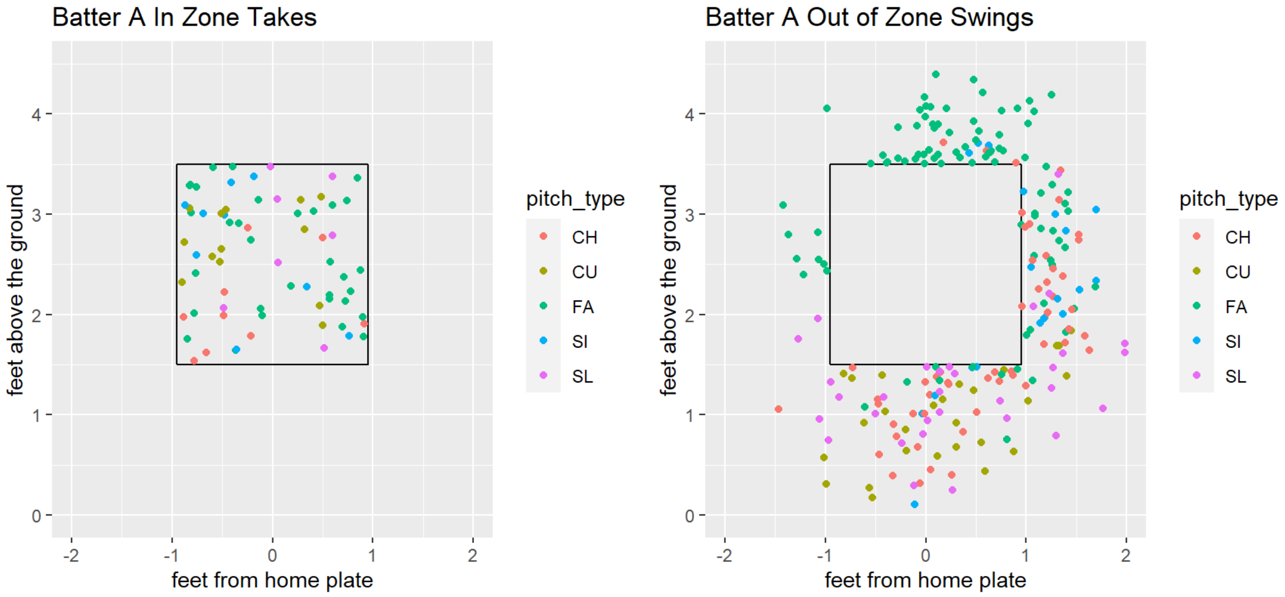
Below, is a distribution plot of average swing percentage per batter. We observe that the distribution appears normally distributed with mean around 0.5 and the variability of swing percentages ranging from about 0.3 to 0.65. Not all hitters in baseball are the same. There exist different types of hitters which results in different types of swing percentages. If there was no variability among swing percentages and all hitters plate decisions were similar regardless of level, then it would be more difficult for models to make good predictions.

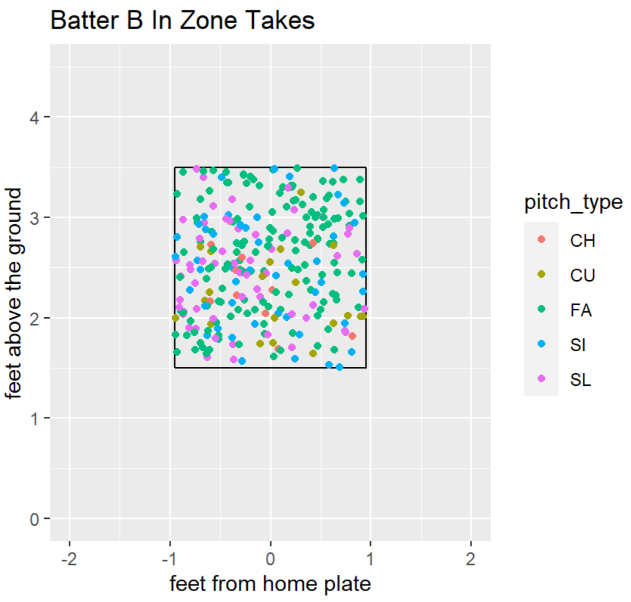


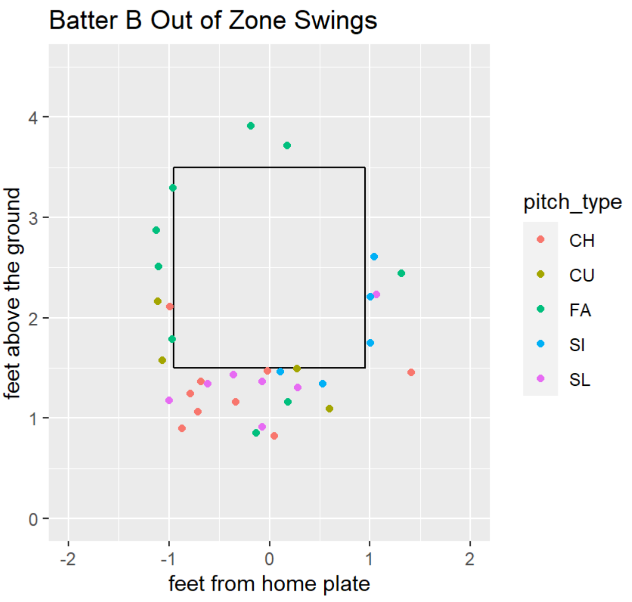
To better illustrate how type of hitter and level can explain a hitter's approach at the plate, I decided to choose two players on the extreme ends: The hitter with the highest average swing percentage (hitter A) and the hitter with the lowest average swing percentage (hitter B). Hitter A is a high A (A+) level left-handed hitter while hitter B is a major league right handed hitter. Both hitters have seen approximately between 900-1000 pitches in the training data.

Below, the first row of plots represent batter A's "bad decisions" while the second row represents batter B's "bad decisions". A "bad decision" is defined as not swinging at pitches that landed inside the strike-zone (first column of plots) while swinging at pitches outside of the strike-zone (second column of plots).

In choosing the two extremes, we observe clear differences among both batters in regards to swinging on pitches out of the strike-zone. The high A ball hitter is very aggressive compared to the major league hitter which is expected. I expect minor league players to have a higher level of desperation to perform well in order to advance to the major leagues as quickly as possible than already established major league players. Hitter A's aggression seems to be attributing to making bad decisions and he seems to struggle on fastballs above the belt and breaking balls below the zone. Hitter B has swung out of the zone much less than hitter A. However, hitter A still took lots of pitches in the zone, especially fastballs. The decision to not swing at pitches in the zone might be strategic as opposed to incorrect. At the MLB level, the level of competition is better and hitters are more selective in their swing decisions. It is strategic for a hitter to sit on a location and swing at pitches only in his sweet spot while taking pitches in the strike-zone that cannot be barreled up.







Multicollinearity Check:

Before I perform any modeling, I want to check if any of the variables selected as candidates for the models explain similar variation in swing/take decisions. Therefore, I checked Variance Inflation Factors (VIFs) among the variables. Any two variables over 9 reveal severe multicollinearity among them and one of them should be removed. The VIFs are located below, and variables horz_release_angle, vert_break, rel_side, horz_approach_angle, vert_approach_angle, and vert_release_angle all have extreme values that indicate they are all collinear. Some pairs of these variables have extremely high positive correlations with each other so removing one of them is necessary.

strikes	balls	release_speed	vert_release_angle	horz_release_angle	spin_rate	rel_height	rel_side	vert_break
1.22	1.21	28.88	936.33	17874.43	1.19	36.38	316.5	1975.94

induced_vert_break	horz_break	plate_height	plate_side	vert_approach_angle	horz_approach_angle	x55	z55
72.92	8580.64	58.74	65.75	1256.38	10978.74	362.18	34.52

Now, severe multicollinearity is no longer an issue. Below are the quantitative variables that I will include in the model along with count, level, and pitch type.

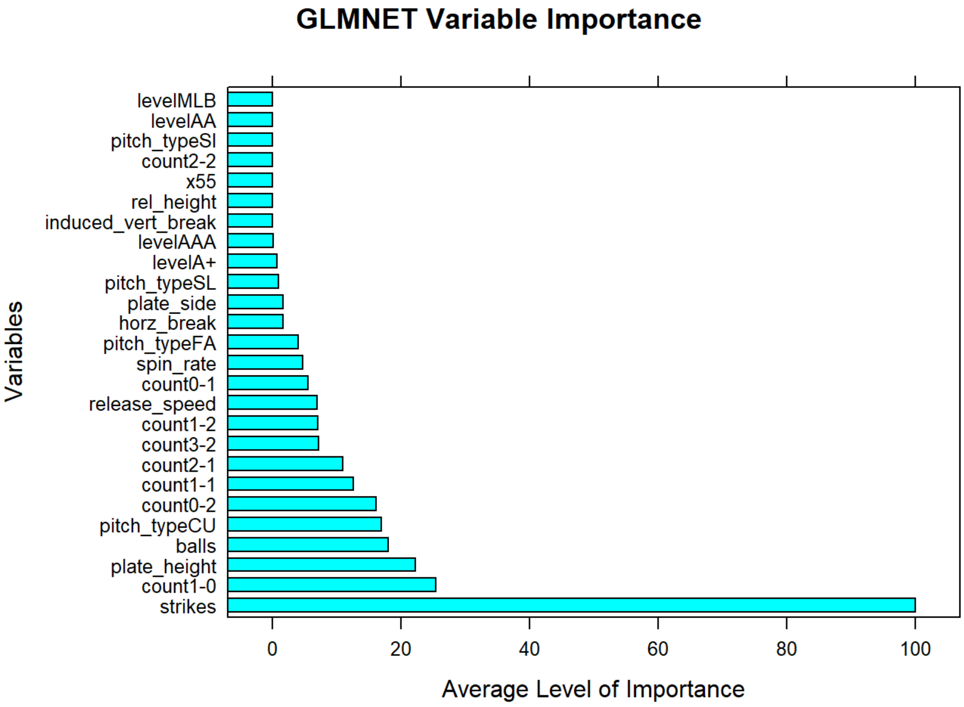
strikes	balls	release_speed	spin_rate	rel_height	induced_vert_break	horz_break	plate_height	plate_side	x55	is_swing
1.24	1.21	2.31	1.13	1.04	2.38	1.55	1.14	1.09	1.36	1.06

Pre-Processing and Machine Learning:

First, I split the overall train data set into a 70/30 train, test split. For each model, I implemented the standard five fold cross validation on the 70% subset of the training data. Within the 70% train split, data is split into five subsets where the first four subsets are used for training and the fifth is used as the testing set for making predictions. This process is repeated until each of the five subsets has been used as a testing set. This ensures that the model will not overfit the data by evaluating the model's average performance on the five hold-out folds. I standardized the input variables for only the penalized logistic regression. Tree-partitioning algorithms such as random forests and xgboost do not estimate coefficients so standardizing will not have interpretation benefits. Since the binary class is_swing is approximately balanced (47% vs 53%), I felt it is appropriate to use accuracy and kappa metrics to evaluate the models. If the classes were unbalanced, I would consider minimizing the log-loss cost function or looking at precision and recall metrics from a confusion table. Lastly, I computed predictions on the 30% test split from earlier and computed accuracy from the confusion table. I expect the best cross validated model to also have the highest accuracy on the 30% test split. For each algorithm, I provided a sorted horizontal barchart displaying the importance rank of each variable for the final model. The most important variable will have the highest value on a scale from 0 to 100 and all variables not included will have a value equal to 0.

Penalized Logistic Regression:

Instead of manually inputting variables and iterating through every possible combination of variables, I chose to run a penalized logit model that imposes a penalty to the logistic model for having too many variables. There are two parameters to optimize (alpha and lambda). The alpha parameter ranges from 0 to 1 which controls the mix or weight of performing lasso regression (alpha = 0, shrinks insignificant beta coefficients to exactly 0) and ridge regression (alpha = 1, shrinks insignificant beta coefficients towards 0). The lambda parameter controls the size of the penalty (relaxed vs strict). I tested five equally spaced alpha parameters and for each alpha parameter, five unique values for lambda. The best model was chosen such that the set of parameters formed the largest mean cross validated accuracy score. The variable importance chart from the best model and the confusion matrix from that model on the 30% testing set are seen below. The overall accuracy on the test set is 60%.

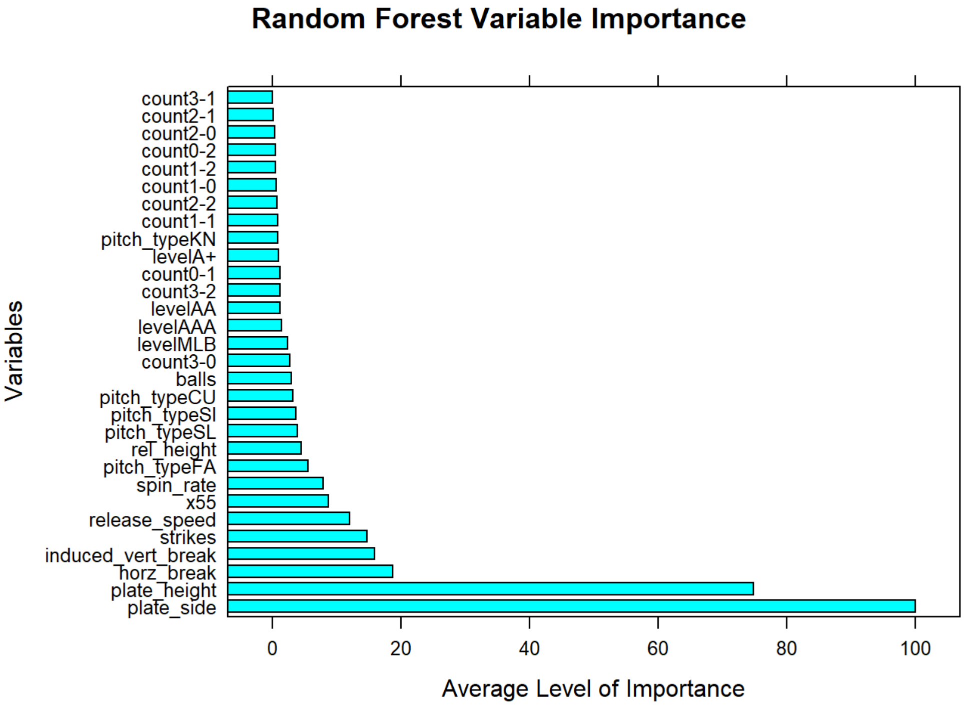


	Predicted	
Actual	21752	13274
	10276	14698

* Confusion Table on Test Set for Best Model

Random Forest:

Instead of creating one decision tree and basing results off one iteration of splits on the same data, random forests create n number of trees bootstrapping the training data set each time. The n decision trees form n predictions for each observation and the algorithm takes the class majority as the overall prediction for the observation. This ensures that the model does not over fit the training data and averaging out n predictions reduces the variability in the predictions. The only parameter to optimize is the number of features considered as candidates for splitting at each node. I fixed number of trees n equal to 500. The variable importance chart from the best model and the confusion matrix from that model on the 30% testing set are seen below. The overall accuracy on the test set is 76%.



	Predicted	
Actual	25196	6832

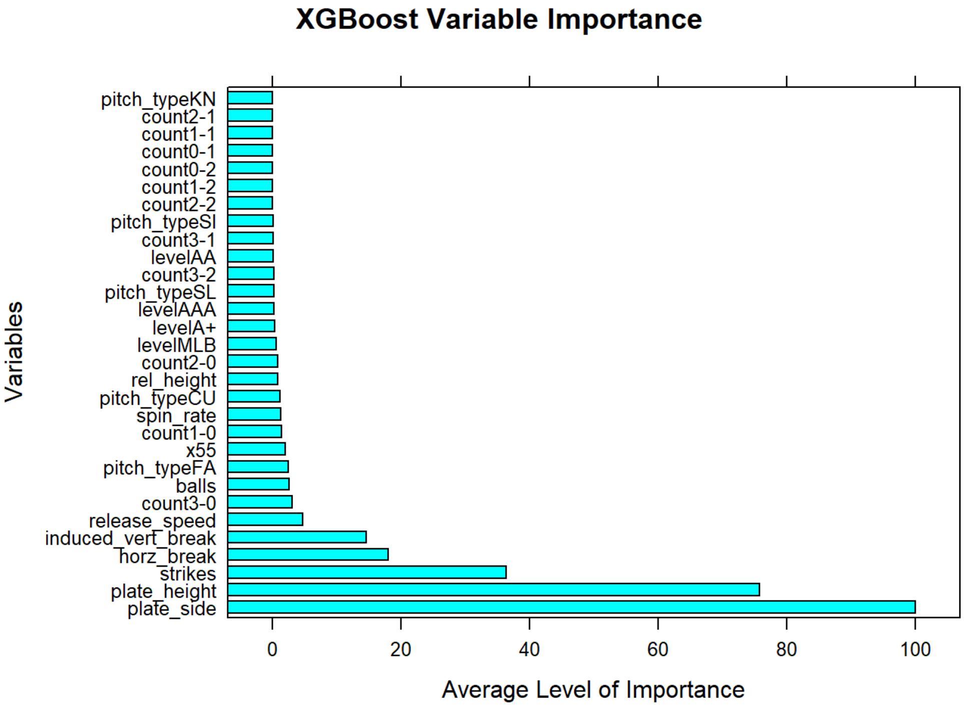
Confusion Table on Test Set for Best Model

	Predicted	
	7017	20955

[*] Confusion Table on Test Set for Best Model

XG Boost:

This algorithm is an improved implementation of the gradient boosting model (gbm), which is focused on fast computation and better model performance. The basic idea of this algorithm is an ensemble method where sequential “weak” tree-based learners are built in which the inputs to each subsequent model are the errors from the previous models. Each model attempts to learn more about the unexplained variation in the response of the previous models. Instead of creating large trees like random forests, boosting algorithms create many tree models with fewer splits and less depth. However, there are more parameters to optimize such as number of trees or iterations, the rate at which the gradient boosting learns, and the depth of the tree but these are all tuned through cross validation. The variable importance chart from the best model and the confusion matrix from that model on the 30% testing set are seen below. The overall accuracy on the test set is 77%.

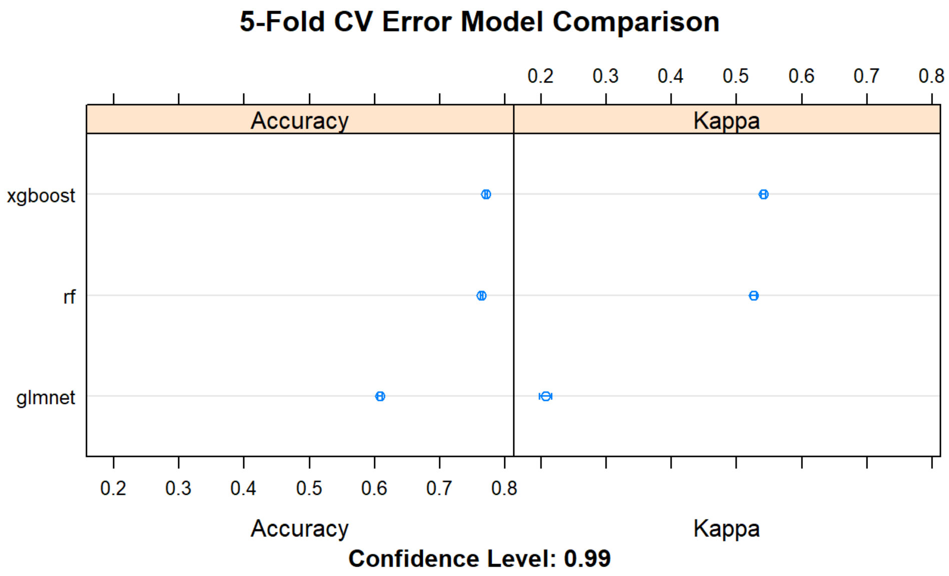


	Predicted	
Actual	24768	7260
	6327	21645

[*] Confusion Table on Test Set for Best Model

Model Comparisons:

After fitting all three models, we observe that the tree based models clearly outperformed the penalized logistic regression model. The xgboost model seems to have performed slightly better than the random forest so in making my final predictions on the test set provided, I will use the xgboost model to make the final predictions. I expect to have approximately a 75% accuracy rate on those predictions.



Conclusion:

The main goal of this problem is to be able to build a model to predict swing/no swing more accurately than simply guessing at random which would give an accuracy about 50%. That goal is achieved as the xgboost model chosen can accurately predict whether a batter swings or not 77% of the time, a huge improvement over guessing. Looking back at the variable importance of the xgboost model, it seems that both x, y coordinates of plate location are the most significant predictors of swing/no swing followed by count, pitch movement (horizontal and vertical), pitch velocity, and fastball pitch type. These variables all make sense intuitively to have an effect on a hitter's decision to swing. Lastly, below is a quick summary distribution of swing/no swing on the original training set and my predictions on the test set. The distributions are similar which means my predictions make sense.

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0	0	0	0.466	1	1

[*] Distribution of Training Swing/Take Decisions

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0	0	0	0.481	1	1

[*] Distribution of Test Predictions Swing/Take Decisions

Problem 2

Introduction:

Hi coach,

I am one of the analysts on the team and have been assigned to investigate a baseball related question of interest for you.

Question:

Is there any recency effect of promotion that affects a player's ability to swing at good pitches while avoiding bad pitches?

Goal:

The goal of this research is to compare plate discipline of recent prospects after their promotion to MLB against their plate discipline during their most recent minor league stint. From this comparison, we can quantify an "adjustment period" that players have when being promoted to MLB from the minors.

Method:

I used Statcast data across different levels for the first half of the 2019 season. I came up with a list of hitters who qualify as prospects that got promoted to MLB. This was done by taking all hitters with minor and major league at bats and only keeping players who spent the majority of their at bats in the minors. This eliminates any major league player who made rehab starts in the minors. Also, I created a list of all current major league hitters as a third comparison group.

In order to quantify plate discipline, I created a metric called decision meter that is either 1 for a good decision or 0 for a bad decision. A bad decision is defined as a hitter who did not swing at a pitch that landed inside the strike-zone while also swinging at a pitch outside of the strike-zone. A good decision is the opposite where a hitter swung at a pitch inside the strike-zone and did not swing at a pitch outside of the strike-zone. Averaging the decision for each hitter yields a score between 0 and 1. The higher the score, the better the hitter is at making good decisions. This is the metric that will be used to investigate any adjustment period that prospects may have.

Results:

Below, are quick summary tables that describe the decision meter for batters under three different scenarios. The first table depicts the characteristics of decision meter for all prospects when hitting in the minors. The second summary table describes the decision meter of the same prospects but during their at bats at the major league level. The last table describes the decision meter for current major league players.

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0.586	0.672	0.693	0.693	0.712	0.779

[*] Prospects MiLB Decision Meter Distribution

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0.5	0.655	0.685	0.682	0.717	0.84

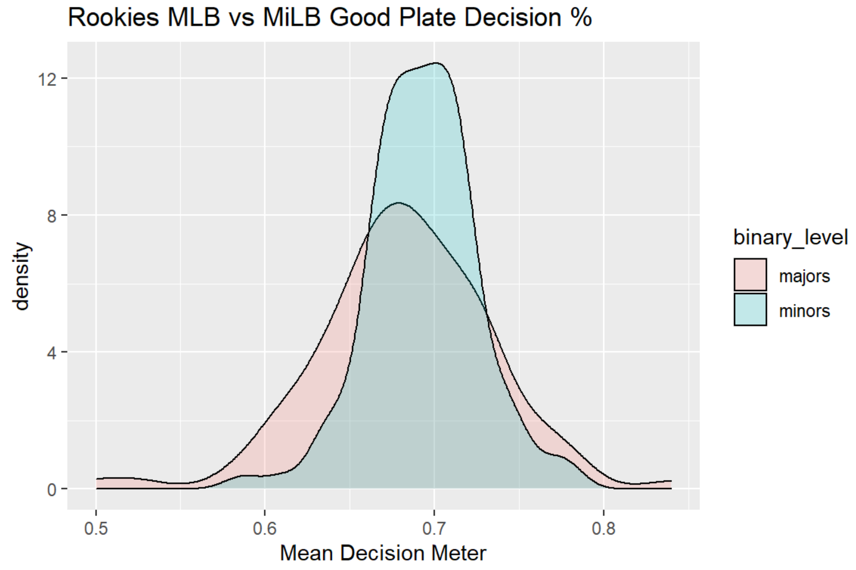
[*] Prospects MLB Decision Meter Distribution

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0.621	0.679	0.699	0.698	0.721	0.773

[*] MLB Pros Decision Meter Distribution

From the mean above, we can conclude that on average, prospects are making worse decisions when they get called up versus their minor league at bats. This makes sense as we expect minor league hitters to face lesser competition compared to MLB pitchers and these hitters tend to have a more aggressive approach at the plate. For minor league players, they are looking to make things happen and produce quickly to position themselves for promotion quicker. So keeping the same approach at the MLB level but face better pitchers will result in making worse decisions.

We can observe this more visually with the plot below. These prospects at the minor league level are consistently making good decisions at a rate centered close to 70%. They would not be called up if they were not performing well. However, after the call-up, the spread or variability among hitters increases. Some hitters adjust well but others take longer to adjust. There are more cases of prospects making good decisions at a below average rate at the major league level than at the minor league level.

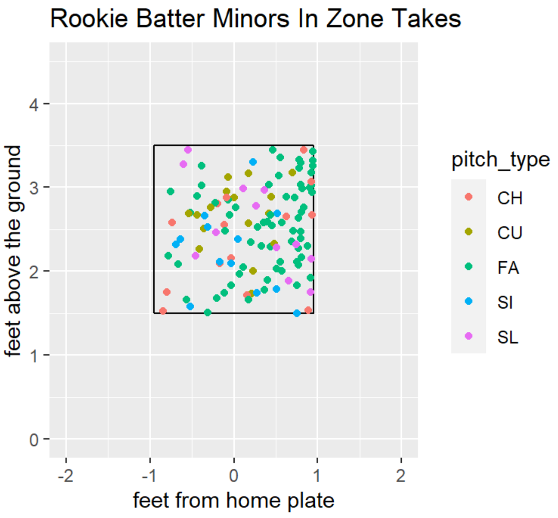


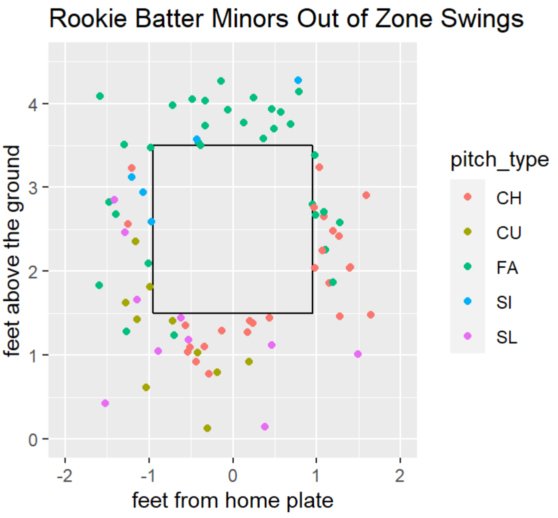
Let’s look at specific examples. I have listed below the top ten players with the largest decision meter drop at the major league level compared to their minor league performance. On average, these ten hitters have seen only about 70 pitches at the MLB level but its clear that these hitters are going through an adjustment period. They are making worse plate decisions in the first few weeks of MLB experience which is crucial to know during a playoff run late in the season.

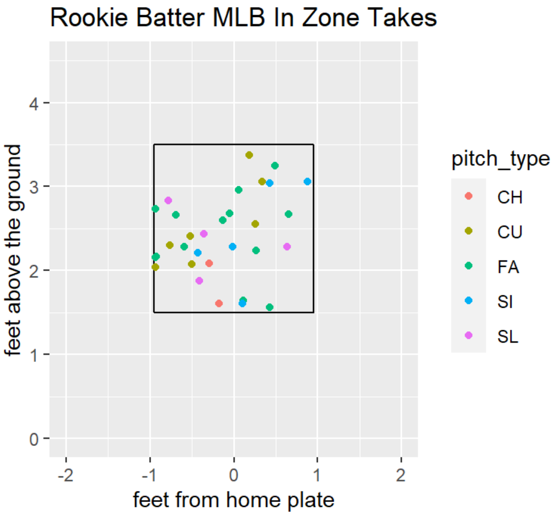
batter	MLB_Decision_Meter	MiLB_Decision_Meter	adjustment
c92851df	0.5000000	0.6828685	-0.1828685
8f8ab5af	0.5263158	0.6825153	-0.1561995
e3dde457	0.5987654	0.7035971	-0.1048317
2d75ea3e	0.5777778	0.6709870	-0.0932093
8afbe891	0.6862745	0.7790698	-0.0927953
c413ed8b	0.6060606	0.6979446	-0.0918840
664ced26	0.6081081	0.6953191	-0.0872110
49fe8e10	0.6250000	0.7098166	-0.0848166
0f037a54	0.6074074	0.6916342	-0.0842268

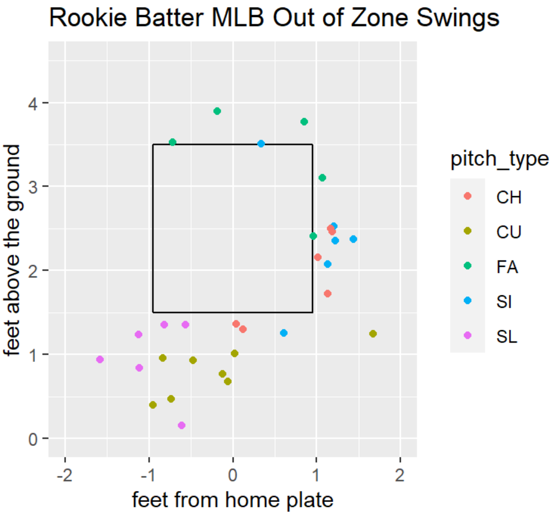
batter	MLB_Decision_Meter	MiLB_Decision_Meter	adjustment
e78f405e	0.6393443	0.7175295	-0.0781853

For example, let’s choose the third player on the top ten list (batter id “e3dde457”). Below, the first row of plots represents the hitters bad plate decisions before the call-up while the second row represents the hitter’s bad decisions after the call-up. This particular batter saw 695 pitches in AAA and 162 pitches in MLB. Even with the sample size discrepancies, we are still seeing a large number of swings for pitches way out of the strike-zone at the MLB level.


Rookie Batter Minors In Zone Takes


Rookie Batter Minors Out of Zone Swings


Rookie Batter MLB In Zone Takes


Rookie Batter MLB Out of Zone Swings

Conclusion:

As a recap, we are interested in investigating whether or not an adjustment period exists for prospects after being called up. From the analysis above regarding specific examples of prospects as well as prospects in general, there exists evidence that prospects on average endure some kind of adjustment period of worse decision making than in their minor league at bats. We should not expect prospects in their early major league at bats to be able to make good plate decision at the same rate as they performed in the minor leagues. In other words, we should expect a slight drop in good decision making percentage at the plate for any prospect called up during their adjustment period.