Batter Pitch Mix Predictor - Technical Report
Timothy Clay
October 28th, 2024

To predict the pitch mix that any given batter will see in 2024, I used a series of three regression models. The data I used for this problem was pitch-by-pitch data for every batter who faced 1,000+ MLB pitches between 2021 and 2023. This data included information like pitch type (which I was able to convert to pitch classification), pitch location, and outcome (ball, called strike, swinging strike, etc.).

For this problem, I aggregated the pitch-by-pitch data at the season and batter level to get the percentage of pitches that each batter saw in a given season. In doing so, I also computed a variety of stats about the player's performance in the previous year, including swing rate, whiff rate, o-swing (chase) rate, z-swing rate, and wOBA. Furthermore, these statistics were also computed for each pitch classification (swing rate on fastballs, swing rate on offspeed, etc.). I chose these 5 stats because I thought they best encapsulate a hitter's strengths and weaknesses at the plate, which a pitcher might try to avoid or take advantage of. In addition to these features, I also included the batter's handedness, and their pitch mix distribution from the previous year as the input features for my regression models.

For the models themselves, I decided to use an XGBoost regressor for all three pitch classifications. I chose to use XGBoost since the data for this problem is highly complex, and there are likely interactions between features that the XGBoost model is the most robust at identifying. While I did consider other modeling types, such as a random forest regressor, I ultimately decided that the XGBoost model performed the best. I tuned each of these models separately in three phases using grid search. Each phase tuned a different set of parameters: first n_estimators and learning_rate, then max_depth and min_child_weight, and finally gamma. To tune the models, I used root mean squared error as my scoring metric. After tuning the models, I trained a final version for each on the entire training data, which I used to generate my predictions.

While the models all improved after tuning, their performances still varied between pitch types. Ultimately, the offspeed model performed the best on the testing data (RMSE = 0.0233), the fastball model was in the middle (RMSE = 0.0290), and the breaking model performed the worst (RMSE = 0.0301). The fastball and breaking models both had R^2 scores between .4 and .5, while the offspeed model had an R^2 of nearly .7. Using these trained, tuned, and evaluated models, I ran each independently for each batter to get a percentage prediction for each pitch type. I then scaled these predictions to add to 1, which I saved as the final projections.

Given the complexity of the problem and the data, there are undoubtedly ways that my model and approach could be improved. First, a more thorough feature selection process might yield more important features, or reveal the redundancy of existing features. While I focused on what I thought were the most important features – swing decisions and swing results – there are many other features that I did not even consider. With more time, I would have liked to have evaluated more of these options. Additionally, there are likely other features that impact a batter's pitch mix that were not available entirely. Swing speed and swing path, for instance, would likely play a large factor, as players with slower swings might be attacked with more fastballs, since they would struggle catching up to them. Despite these opportunities for future improvement, however, I believe that the models I created to predict a batter's pitch mix are incredibly valuable and can have meaningful implications.