Analysis of Breaking Ball Velocity Differential with Primary Fastballs

I. Problem Identification & Context

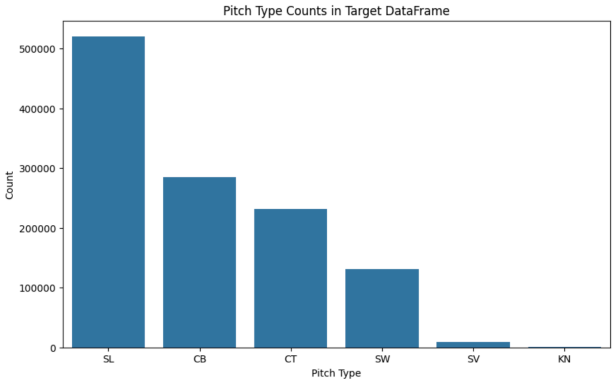
Javier Baez attempts to hit a Sam Long curveball, 2021

In baseball, variations in pitch shape and velocity are the primary tools pitchers use to deceive hitters and generate strikeouts. The inherent difference in velocity between a fastball and a breaking ball is tied to the trajectory of the pitch. By understanding how pitch characteristics interact with each other, we can create a model designed to predict the velocity differential between a pitcher's primary fastball (defined as an average of their four-seamers, two-seamers, and sinkers) their breaking balls. Establishing a per-pitcher fastball velocity baseline and engineering features representing the differences in pitch characteristics (e.g., break, release point, spin) between these fastballs and breaking balls, a we can develop said model. Identifying which engineered pitch characteristic differentials (such as vertical break difference) most significantly influence velocity separation is crucial for guiding pitchers in optimizing their repertoire and enhancing deception by disrupting hitter timing and choosing specific pitch sequences.

II. Data Sources and Exploration

The dataset used consisted of over three million individual pitches from 2020-2024, pulled from the Statcast library. The analysis began with comprehensive data exploration before implementing careful cleaning and organizing to optimize model input.

To keep it simple, I dropped all columns that weren't directly related to the pitch characteristics, only leaving inning, date, and pitcher name/id as non-numeric predictors. To ensure we were only looking at pitchers, I filtered anybody who had both less than 300 pitches *and* never threw a pitch exceeding 86 mph. While not 100% perfect, this filtered out nearly all position players pitching in unserious game situations. While only a fraction of the dataset, this removed about 157 unique players who pitched (9% of total pitchers). I then averaged the velocities of four-seamers, two-seamers, and sinkers per pitcher to establish each pitcher's baseline fastball



velocity. Using that baseline, I calculated the velocity differential of all breaking balls on a per-pitcher basis. I chose to isolate cutters in their own category because they sit in a weird spot between fastballs and breaking balls, but they were still added as a target variable. The distribution of the targeted breaking balls after the cleaning (above) shows that sliders were most common, followed by curveballs and cutters. It should be noted that cutters have a unique high velocity profile with less vertical movement when compared to the other breaking balls.

Finally, the metrics for velocity, break, location, release point, and approach angle all had roughly 500 NaN values; those rows dropped, given they only constituted 0.01% of the dataset. Spin rate and spin axis had roughly 21,000 NaN values each, which were imputed with median values to keep the cleaning simple. Had the amount of missing values for a column exceeded 2% of the total, I would have opted for a kNN imputation to maintain data quality.

III. Modeling and Analysis

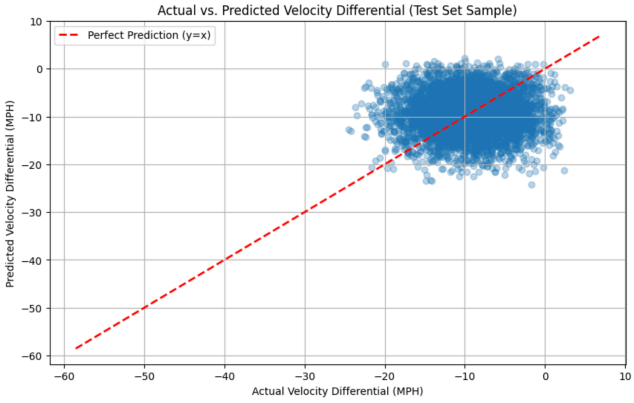
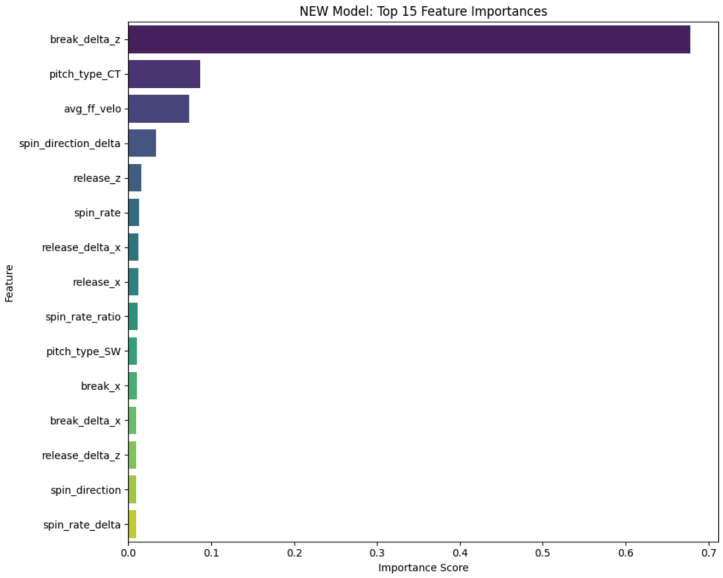


Figure 1

Given the objective to minimize residuals between predicted and actual velocity differentials, I opted to use a Random Forest Regressor (RFR) model. RFRs are ideal for structured tabular datasets and they do well with nonlinear relationships, which I anticipated I would run into given how complicated the movement profiles of individual pitches can be. While other powerful models like Gradient Boosting Machines (e.g., XGBoost, which was later used for validation) or Support Vector Regression could be considered, RFR offered a



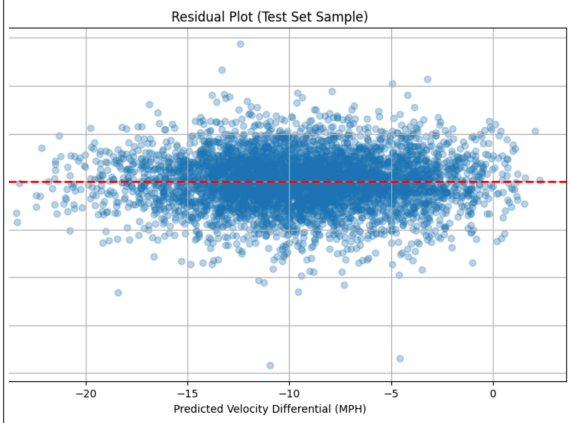
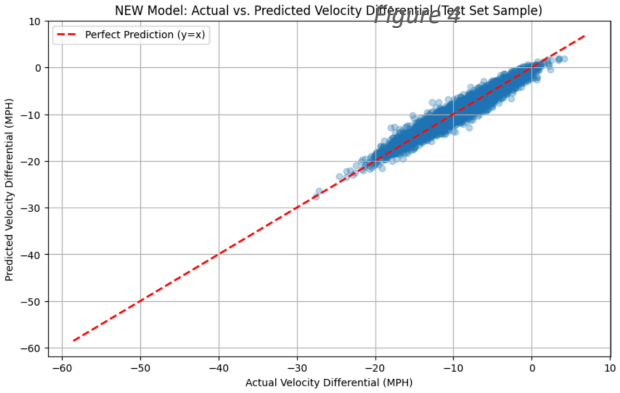


Figure 2



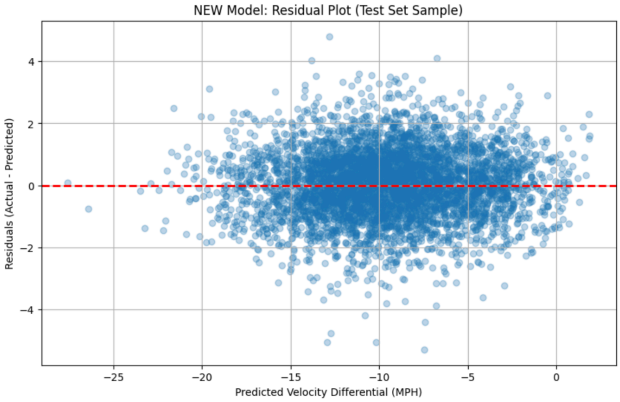


Figure 3

strong balance of performance, interpretability, and relative ease of implementation for this project.

I trained the first RFR model strictly on the raw pitch characteristics. The results were initially promising with an out-of-bag R^2 score of 0.9. However, the actual predictions (Fig. 1) appeared more clustered and nebulous, not following the perfect linear relationship, y=x.

The predictions and the residuals (Fig. 2) indicated a need for feature engineering to give the model additional context. Instead of looking at raw pitch data, the model should be looking at the difference in pitch data values between the fastball baseline and the breaking balls. This includes differentials for break, release point, spin rate, and spin axis. After creating those features and adding them back into the data frame, I re-trained the RFR and received much better predictions (Fig. 3).

Both models revealed that the vertical break differential between breaking balls and primary fastballs emerged as the dominant predictive factor (Fig. 4). This finding lines up with basic logic where curveballs have the most drop and sliders drop less but are often thrown much harder.

For validation rigor, I implemented an XGBoost model in parallel. The XGBoost model trained on the feature engineered dataset ultimately performed marginally worse than the RFR model while adding complexity and computational cost. While both models demonstrated exceptional predictive accuracy (Fig. 5), the results underscored that thoughtful feature engineering was ultimately the decisive factor in the model success.

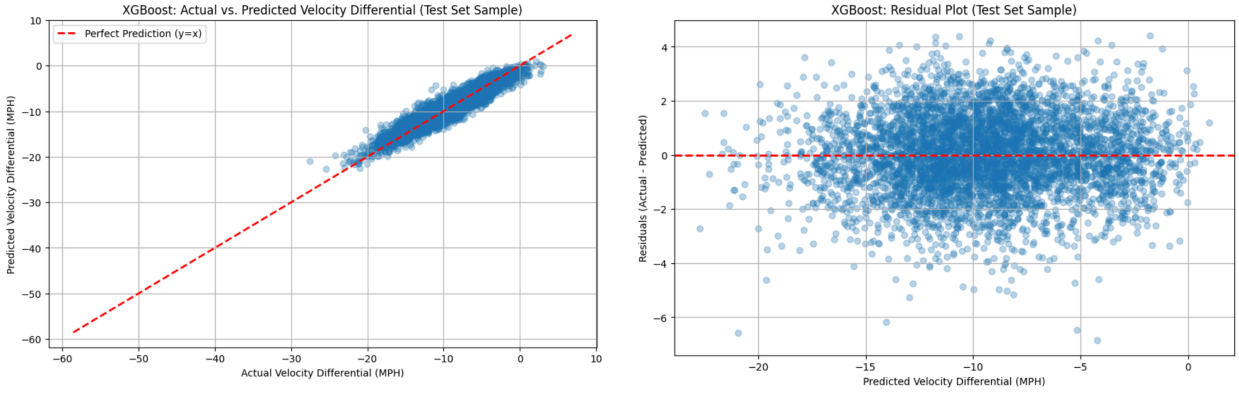


Figure 5

	Mean Absolute Error (ΔMPH)	Root Mean Squared (ΔMPH)	R-squared (R^2)
RFR Basic	0.843	1.1845	0.9371

	Mean Absolute Error (ΔMPH)	Root Mean Squared (ΔMPH)	R-squared (R^2)
RFR Feature Engineered	0.846	1.1938	0.9366
XGBoost	1.1335	1.4447	0.8891

Achieving a high R^2 of approximately 0.94 is promising, but with large datasets and complex models like Random Forest, it can't hurt to consider overfitting. The out-of-bag score (0.9, close to test R^2) provided some initial reassurance during RF training, but more rigorous validation techniques would be employed in a production setting. In future iterations, I would explore incorporating K-Fold Cross-Validation while tuning the RF model's hyperparameters to ensure robust generalization.

Additionally, in a future rendition of this project, I would test a model that more explicitly evaluates year-to-year changes in baseline fastball velocity. I do think that the amount of pitchers losing velocity due to age is counterbalanced by the younger pitchers increasing their velocity through development, but a model trained explicitly factoring pitcher age could show clearer changes in velocity differential from season to season.

IV. Potential Future Directions

Several avenues could further enhance the predictive power and practical utility of this model for the Marlins:

- **Fatigue/Workload Metrics & Injury Forecasting:** This model implicitly captures some effects of fatigue through raw pitch data, but explicit incorporation of more sophisticated workload indicators could be beneficial. To incorporate this effectively, you would need to distinguish between relievers and starters. Pitches thrown in the current inning, rolling averages of velocity/spin over recent appearances, and days of rest could capture nuances in the data. Such features might better explain variations in velocity differentials, particularly late in games or during a strenuous stretch of a full season.

- **Tailored Development:** The model could establish data-driven baselines for expected velocity differentials based on a pitcher's mechanics, and deviations could guide player development efforts. If a pitcher's sweeper has a smaller differential than predicted given their change in break and release, coaches could investigate if there's potential to add more separation or if their current execution is optimal for their arm action. Combining a robust iteration of this model with direct biomechanical measurements (e.g., motion capture of arm speed, joint angles, kinetic chain sequencing) or more Hawkeye/Trackman data (e.g. spin axis decomposition, seam-shifted wake effects) has the potential to markedly improve pitch development and pitcher deception.
- **Scouting Efficiency:** This framework could flag amateur or minor league pitchers with outlier differential profiles (either exceptionally good or surprisingly small given their other metrics) for further in-person scouting to identify undervalued or unique skill sets.
- **Exploring Pitch Sequencing:** Future models could investigate the impact of pitch sequencing by using the preceding pitch movement, location, and velocity differential to optimize the swinging strike rate of breaking balls. This would likely improve the value of pitcher deception and repeatable mechanics, as well as the ability of catchers to call the correct pitch.