Elijah Kruse
10/15/2024

Technical Report: Pitch Type Classification Model

Overview
This report summarizes a machine learning model developed to classify pitch types in baseball, specifically distinguishing fastballs (FB), breaking balls (BB), and off-speed pitches (OS). The model utilizes historical pitch data and player/game characteristics, employing a Random Forest Regressor optimized via GridSearchCV. The primary goal is to classify pitch types for each batter and evaluate the model's performance through mean absolute error (MAE) comparisons between actual and predicted pitch type distributions.

Data Overview
The dataset includes pitch-level information, such as game context (e.g., outs, balls, strikes) and player identifiers (e.g., batter and pitcher IDs). Irrelevant columns were removed through careful selection based on domain knowledge and correlation analysis. Key preprocessing steps included:

- Dropping irrelevant columns (e.g., game dates).
- Filtering out rows with missing target variables (PITCH_TYPE).
- Mapping pitch types to three categories: FB, BB, and OS.
- One-hot encoding categorical features (e.g., batter and pitcher handedness).
    - Note: Typically, Random Forest algorithms can handle categorical features. However, in practice, many machine learning libraries (like Scikit-learn) expect all features to be numerical.
- Binarizing base runner presence for occupancy features.

The final dataset was split into training (80%) and testing (20%) sets.

Model
A Random Forest Regressor was selected for its ability to manage non-linearity in the data while allowing for parallel processing, significantly speeding up computational time. The model predicts the probability distribution of different pitch types for each scenario, aggregating predictions based on Batter ID to create a comprehensive pitch mix profile.

Key Model Components

- **Features**: The model utilizes a comprehensive set of player and game statistics, including BATTER_ID, PITCHER_ID, game context (INNING, AT_BAT_NUMBER), and fielding alignments.
- **Target**: Predicting the probability distributions for pitch types: Breaking Ball (PITCH_TYPE_BB), Fastball (PITCH_TYPE_FB), and Offspeed (PITCH_TYPE_OS).
- **Cross-validation**: A 5-fold KFold cross-validation strategy was employed using GridSearchCV to optimize the model parameters, specifically the number of trees in the Random Forest.

Results
The model's performance on the test set resulted in a Mean Absolute Error (MAE) of 0.3508, reflecting its overall predictive accuracy. However, this performance suggests the model is facing some challenges and could be improved.

Limitations and Reflections
Several limitations impacted the modeling process:

1. **Feature Selection**: I dropped columns I believed would not contribute effectively to pitch prediction, including pitch outcomes (e.g., home runs), as incorporating these would have increased model complexity. Given the 8-hour time constraint for this assignment, I opted to simplify the model.
2. **Computational Complexity**: The GridSearchCV process was computationally expensive, and future iterations may benefit from a reduced grid search space. While 250 estimators was identified as the optimal number, this is likely due to memory limitations encountered during trials with 750 and 1000 estimators, which resulted in out-of-memory errors.
3. **Limited Data**: The predictions were based solely on the small 20% of data that was reserved for testing. In retrospect, a more effective approach would have been to train and test on the actual dataset (as we did here), but also to generate a variety of hypothetical at-bat scenarios for the batter to predict the pitch mix. However, this would require an additional algorithm or model to predict when and how likely those combinations of features would occur, which adds complexity to the process.

In hindsight, I believe a boosting algorithm would have been a better choice for its error-correcting mechanisms. The Random Forest model did not effectively learn from the data, while boosting could capture patterns more accurately. Moreover, I considered implementing base learner splits at the batter ID level, tailoring predictions to specific batters for enhanced pitch mix insights.

Conclusion
This Random Forest model demonstrates reasonable accuracy in predicting pitch type distributions. However, improvements in feature selection, data imbalance handling, and computational efficiency could further enhance predictive power. Future work could explore more sophisticated modeling techniques, such as gradient boosting, to refine these predictions.