ANALYSIS ON MLB PITCH DATA THROUGH PCA AND PITCH-TYPE PREDICTION

Preston Keskey, Derek Wang
University of Minnesota
Minneapolis, MN 55455, USA
{keske037, wan00923}@umn.edu

1 INTRODUCTION AND PROBLEM STATEMENT

Baseball analytics has advanced rapidly since the *Moneyball* era demonstrated that rigorous quantitative insight can turn marginal advantages into wins (Lewis, 2004). Today every Major League Baseball (MLB) club mines pitch-tracking systems such as Statcast, yet translating dozens of raw kinematic variables into coach-friendly knowledge and doing so quickly enough to influence strategy remains a practical challenge for teams.

For our project we use MLB Pitch Data dataset, covering the 2015–2018 seasons. We combine four excel sheets: `pitches`, `at_bats`, `games`, and `player_names` into a single file of roughly 2.9 million pitches and 67 variables, with less than 0.01% missing values. Twenty-five of those variables capture the physical motion of each pitch, such as velocity components, spin rates and axes, release extension, and ball movement at home plate.

Our project addresses two intertwined goals and research objectives

1. **Understanding the pitching landscape.** Use principal-component analysis (PCA) and clustering to condense the 25 motion variables into a handful of interpretable axes (such as, "power/ride" versus "run/sweep"). This low-dimensional view should reveal how pitch types and individual pitchers group together, providing scouts and coaches with an quick understanding of pitching profiles and the variables that influence certain pitch types.

2. **Predicting the next pitch.** Feed the kept PCA scores with in-game context such as count, inning, and recent pitch sequence into machine-learning models that estimate the probability of each forthcoming pitch type. We will train on 2015–2017 data and evaluate on the 2018 season, comparing metrics such as accuracy, log-loss, and Brier score for different models.

Overall, our goal is to keep a small set of orthogonal components that explains around 80% of the variance in pitch-motion data while offering intuitive baseball interpretations. Additionally, we want to use and display that utilizing PC scores in a predicting model for predicting pitch type is an efficient and usable method for MLB teams. Achieving both will demonstrate a workflow that compresses high-dimensional pitch physics, accelerates model training, and delivers actionable forecasts for possible real-time decision making for MLB teams.

2 Data Preprocessing and Cleaning

We start with the four Statcast-derived CSVs released on Kaggle: `pitches`, `at_bats`, `games`, and `player_names`. We were able to combine these CSVs by the following variables: `g_id` (the game identifier), `pitcher_id`, `batter_id`, and `ab_id`. The resulting conbimed CSV yields a data table of 2,903,204 rows and 67 columns. Each pitch now contain ball-tracking physics, pitch outcome (`pitch_type` and batting result from the pitch), and contextual game information (count, inning, weather, home/away).

Missing-Value Imputation ($<$ 0.01 % of total data): We found that only seven variables exhibit any missingness; none exceeded roughly 0.7 % of observations for a specific variable. Because pitch physics differ systematically across pitch types, median imputation stratified by `pitch_type` outperforms a global fill. Therefore, we applied this idea to numeric columns (velocity components, break, spin) that contained missing inputs.

Reasoning for Median Imputation by Pitch-Type We noticed that several motion variables are skewed and pitch-type specific. The following Figure 1 visualizes the *spin-rate* density for the ten most common pitches.

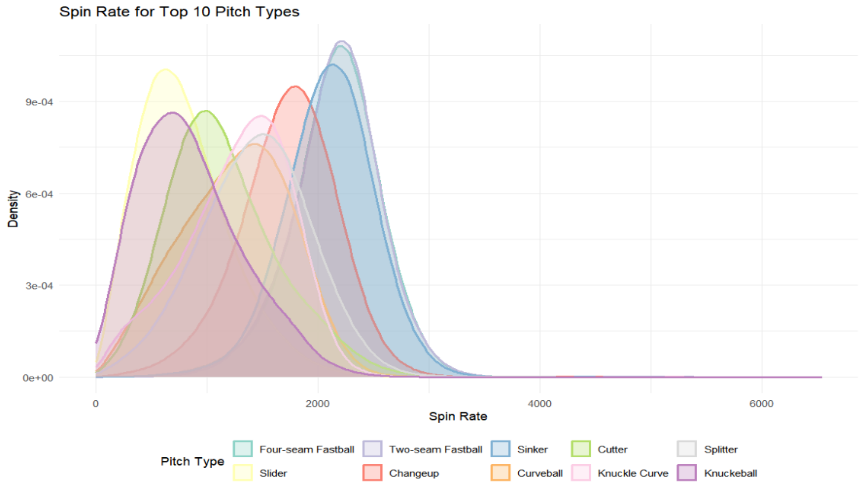

Figure 1: Spin Rate by Top 10 Pitch Types

Four-seam fastballs peak around 2300 rpm, while sliders cluster lower with nearly no spin, and knuckleballs span a broad, low-spin tail. Similar patterns appear in release speed and horizontal break, underscoring the importance of pitch-type conditional imputation by median.

3 Exploratory Data Analysis (EDA)

Our exploratory data analysis serves to illuminate the dominant patterns, anomalies, and feature relationships in our cleaned data table, further guiding both PCA dimensionality choices and the subsequent selection of predictive-model inputs.

Pitch-Type Frequency: Figure 2 shows that fastballs account for greater than 60 % of all MLB pitches during 2015–2018, followed by sliders (around 14 %), change-ups (around 9 %), cutters, curveballs, and sinkers. The heavy class imbalance demonstrated motivates incorporating label smoothing during network training and using Top-3 along with Top-1 accuracy as an evaluation metric.

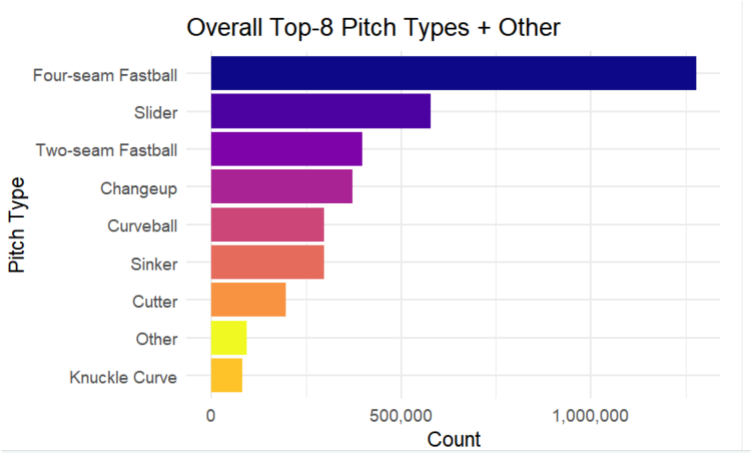

Figure 2: Top 8 Pitch-Types + Other

Correlation Structure of Motion Variables: The correlation map in Figure 3 reveals two notable clusters:

- **Velocity / back-spin block**. `release_speed`, `pfx_z`, and spin-axis variables positively correlate, reflecting the "ride" of high-velocity four-seamers.
- **Side-spin / horizontal-break block**. `pfx_x`, `spin_tilt`, and gyro components cluster together and exhibit negative correlation with the first block, capturing sweeping sliders and cutters.

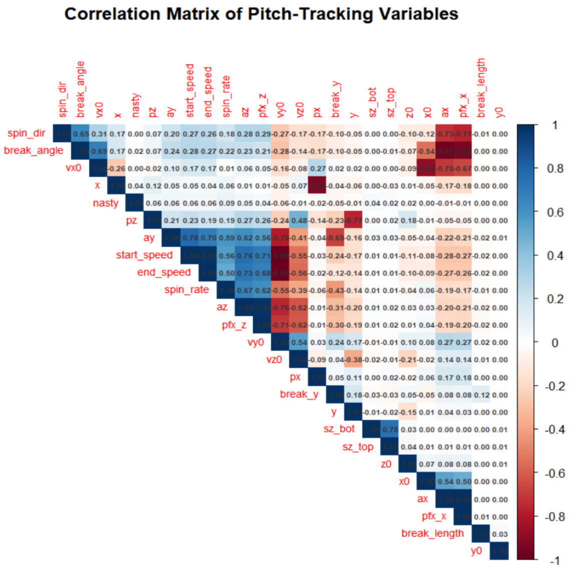

Figure 3: Correlation Matrix of Pitch-Tracking Variables

We believe that these exhibited correlations suffice for performing PCA. Furthermore, these patterns foreshadow PC1 (power/ride) and PC2 (run/sweep) later discovered by PCA.

Count-Based Tendencies: Figure 4 and Figure 5 confirm baseball heuristics: on fastball counts (2–0, 3–0, 2–1) pitchers throw a fastball $> 80\,\%$ of the time, whereas in two-strike counts (0–2, 1–2) they diversify to off-speed offerings. Such conditional probabilities justify conditioning the predictive model on the current count.

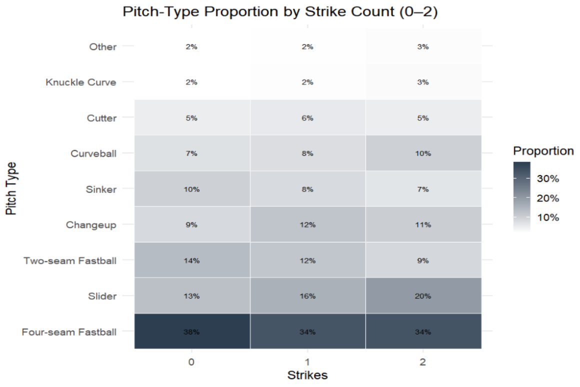

Figure 4: Pitch-Type Proportion by Strike Count

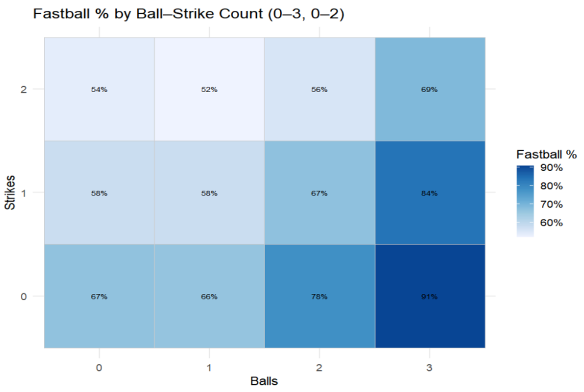

Figure 5: Fastball Percentage by Strike Count

Overall, our hope for our predictive model is to have the heuristics embedded in the predictions demonstrated by our exploratory data analysis.

We found that our EDA validates three modeling assumptions used for PCA and our predictive model:

1. Motion variables contain high internal correlation.
2. Pitch selection is *not* i.i.d.; it depends on count and recent history.
3. Fastballs' dominance creates class imbalance that must be treated explicitly in model training.

4 Applying Principal Component Analysis (PCA)

To compact the 25 raw pitch–tracking metrics seen in Figure 3 into a compact set of orthogonal, interpretable axes, we performed PCA on the 2015–2018 dataset. When performing PCA, we centered and scaled each variable because several variables were expressed in different units (such as mph, inches, rpm) and many variables did not share similar distributions. The scree plot in Figure 6 shows an "elbow" roughly at the sixth component; we found that the first eight PCs capture roughly 80% of the total variance and are therefore retained for all subsequent analyses and as inputs to the predictive model.

Interpreting the first two components: Figure 7 overlays 95 %-probability ellipses for the eight most common pitch types on the PC1–PC2 plane, together with the variable loadings. The loadings reveal two physically intuitive directions:

- **PC1 (27 % variance):** dominated by velocity measures (*start_speed*, *end_speed*) and back–spin indicators (*pfx_z*, *az*). High scores correspond to hard–thrown pitches with vertical ride.

- **PC2 (15 % variance):** driven by side–spin and horizontal movement (*pfx_x*, *spin_dir*, *break_angle*). Positive scores imply arm–side run; negative scores imply glove–side sweep or top–spin.

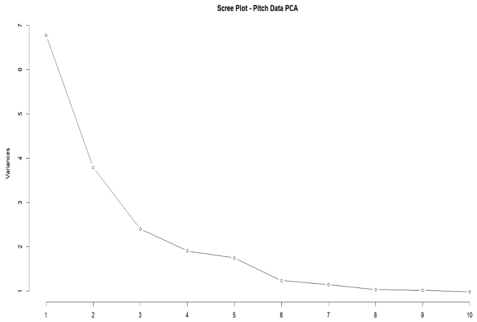

Figure 6: Scree plot for the pitch–tracking PCA.

Where pitch types live in PC space: Further examining figure 7, distinct clusters emerge:

- **Four-seam fastballs** (cyan ellipse) occupy the high-PC1, near-zero PC2 region This represents high velocity and strong back-spin.
- **Curveballs** and **knuckle curves** lie low on PC2 and slightly low on PC1, reflecting lower velocity and top-spin.
- **Sliders** and **cutters** are slightly left on PC2 (strong glove-side sweep) while maintaining moderate PC1 values.
- **Changeups** and **sinkers** cluster just below the origin: reduced velocity with arm-side fade.

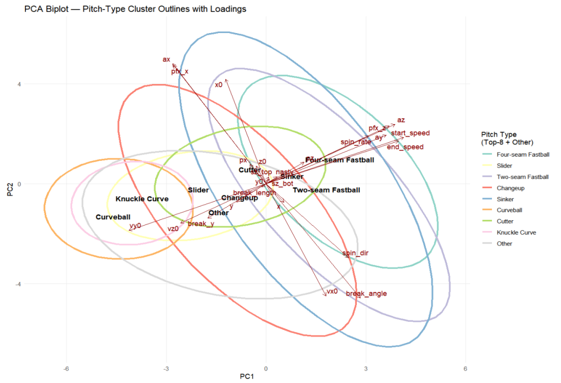

Figure 7: PCA biplot with 95 % cluster ellipses and variable loadings.

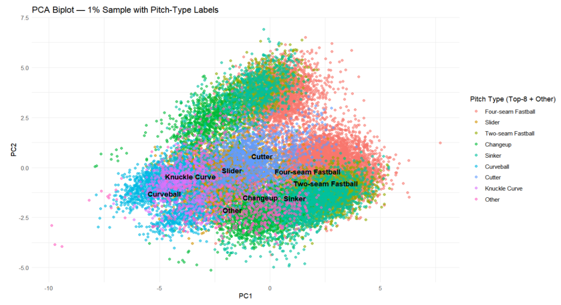

Figure 8: PC1–PC2 biplot of a 1 % sample, colored by pitch type.

Density of the full data cloud: Because individual observations obscure loadings at full scale, we also plot a 1 % sample in Figure 8. The same geometric structure is apparent: fastballs in the upper-right, breaking balls lower-left, further reinforcing that pitch types can be reasonably separated with only the first two PCs.

Why retain eight components? Beyond visual clarity, PCs 3–8 collectively explain an additional $38\,\%$ of the variance and capture subtler patterns (release-point differentials, late break, vertical approach angle) that may improve downstream prediction without appreciably inflating dimensionality. Using these eight scores as inputs will speed model training, mitigate collinearity, and reduce over-fitting while preserving the underlying mechanics embedded in the raw metrics.

In summary, PCA compresses high-dimensional pitch-tracking data into a handful of physically meaningful axes. The resulting scores succinctly encode pitch velocity, spin, and movement characteristics, and they form the feature set for the predictive model developed in the next section.

5 Predictive Model Evaluation

Our goal is to predict the next pitch type given all game information available. That is, for each pitch t, the model will learn a mapping

$$f_\theta : \big(\mathbf{x}_{\text{pitch},t-1},\, \mathbf{x}_{\text{count},t},\, \mathbf{x}_{\text{game}},\, \mathbf{x}_{\text{context}}\big) \longrightarrow y_t \in \{\text{FA}, \text{SL}, \text{CU}, \dots\}.$$

For data splitting, we split the full 2.9 million rows, 67-variable dataset (2015–2018) into *train* (2015–2017) and *test* (2018). A season-level split safeguards against possible look-ahead leakage that would arise if we randomly shuffled the sequential data. All models and metrics are imputed in Python and utilize PyTorch libraries.

Using Principal-component scores: The first 8 principal components, which captures around 80% of total variance, condense 25 raw pitch-tracking variables into an orthogonal, low-collinearity basis.

Incorporating Game-State Variables: Count (balls, strikes), inning, outs, base-state, and score differential are utilized in the model. In the future, we aim to incorperate more variables, such as weather and stadium influence

Treating Pitches as Sequential Data: For sequence models we embed up to the three previous pitch types, $\big[y_{t-1}, y_{t-2}, y_{t-3}\big]$, into 32-dimensional learned vectors, letting the network infer possible setup patterns. Depending on the available computational resources, or perhaps evidence that increasing the previous pitches leads to a better accuracy, this number can be increased in the future.

Our model options for this task include:

- **Multinomial logistic regression:** interpretable linear baseline; no temporal dependence.
- **Vanilla RNN:** single hidden layer, 64 units; captures short-range dynamics but can be susceptible to vanishing gradients.
- **LSTM:** two stacked layers, 64 hidden units each; gated architecture alleviates vanishing gradients and models longer-range dependencies.

All models minimize cross-entropy on the training set. Hyper-parameters are chosen using the 2017 season as an internal validation fold. The RNN and LSTM employ the Adam optimizer and label smoothing of 0.05 to reduce class imbalance.

The following metrics will be used to evaluate the three model types:

- **Top-1 accuracy (indicator function):** $\frac{1}{N}\sum_{i=1}^{N} \nVdash \{\hat{y}_i = y_i\}$.
- **Top-3 accuracy:** true pitch must appear in the model's three highest probabilities.
- **Log-loss** for sharpness/calibration of probabilities.
- **Brier score** for overall reliability; measures predictive accuracy for probabilistic outcomes.

Table 1 reports performance that we found on the 2018 hold-out season.

Table 1: Predictive performance on 2018 test season

Model	Top-1 (%)	Top-3 (%)	Log-loss	Brier
Logistic Reg.	{48.72}	{76.21}	{1.23}	{0.22}
Vanilla RNN	{55.14}	{82.48}	{1.01}	{0.17}
LSTM	**{57.86}**	**{84.63}**	**{0.94}**	**{0.14}**

Why we think the RNN and LSTM outperform logistic regression. Logistic regression assumes i.i.d. pitches, which we found is not true in our exploratory data analysis. Any temporal signal or dependency must be added manually as lagged features. The RNN learns these dependencies directly, which boosts Top-1 accuracy by around 7 percentage points.

Why we think LSTM beats the vanilla RNN. The gated cells in the LSTM mitigate vanishing gradients, preserving memory over longer pitch sequences (for example, a two-strike "waste" pitch followed by a put-away slider). This translates into an additional 2.5 percentage points gain in Top-1 and better calibration.

Trade-offs. Logistic regression is transparent and fast but blind to sequence. Vanilla RNN is a lightweight sequence model, yet training can be unstable. LSTM offers the best accuracy at the cost of computational resources and reduced interpretability.

Future work. This is a rough idea that assesses how different baseline models perform on pitch-type prediction. These models serve as an introductory to how certain neural network designs perform on pitch modeling and can be vastly improved in the future. In the future, we also want to assess transformer encoders for longer context windows, incorporate pitcher-specific random effects, and benchmark inference latency for in-game deployment pipelines.

6 Conclusion and Future Applications

To conclude, this study shows that dimensionality–reduction can distill the kinematics of ~ 3 million MLB pitches into a handful of physically meaningful signals that both describe and predict on-field behavior:

- **PCA insight.** Eight orthogonal components preserve roughly $80\,\%$ of the variance in 25 raw motion variables while mapping cleanly onto intuitive baseball concepts such as power/ride (PC1) and run/sweep (PC2).
- **Model efficiency.** Feeding those eight scores (plus count and some game context) into an LSTM yields a $57.9\,\%$ Top-1 and $84.6\,\%$ Top-3 accuracy on the 2018 hold-out season,

which outperforms both logistic regression and a vanilla RNN at a modest computational cost. If interpretability matters for a certain application, logistic regression can be utilized with slight performance decrease.

- **Actionability.** Because PCs are unitless, orthogonal, and interpretable, coaching staffs can visualize clusters for scouting, isolate which physical levers move a pitcher's arsenal in PC space, and deploy fast, real-time pitch-type probabilities during a game.

Limitations: Our evaluation focuses on only a four-year Statcast window and treats all pitchers identically; it ignores changes in arsenal over time, stadium/weather changes over years, and hitter tendencies. Furthermore, the LSTM's black-box nature hampers interpretability relative to simpler probabilistic models.

Future work and extensions:

1. **Pitcher-specific adaptation.** Add hierarchical random-effect terms or fine-tune pitcher-level subnetworks to capture individualized sequencing habits.

2. **Long-context architectures.** Benchmark transformer encoders or Temporal Convolutional Networks on sequences of 10–20 prior pitches to assess whether longer memory boosts accuracy.

3. **Streaming deployment.** Work on a service that ingests live Statcast feeds and returns pitch-probability dashboards within sub-second latency for coaches to make real time decisions with our models.

4. **Broader outcome modelling.** Extend the feature set and label space to predict pitch location or expected run value, enabling holistic run-prevention strategy tools for coaches and MLB teams.

By compressing high-dimensional physics into a concise, interpretable basis and demonstrating that those latent factors drive predictive performance, this work offers a blueprint for scalable, real-time analytics that can inform player development, in-game decision making, and front-office valuation for MLB teams.

References

Michael Lewis. *Moneyball: The Art of Winning an Unfair Game*. W. W. Norton & Company, New York, NY, 2004. ISBN 9780393338393. URL https://wwnorton.com/books/9780393338393.