Predicting the Offensive Performance of MLB Position Players

Susana Benavidez, Stephanie Brito, Derek McCreight, Peter McEvoy {sbenavid, sbrito, dmccreig, pmcevoy} @stanford.edu; *Mentor: Leo Mehr*

Introduction

- Major League Baseball (MLB) position players are those who hit as opposed to pitch.
- The rise of data-driven baseball as popularized by the early 2000s Oakland A's, has lead to a sharp increase in the number of advanced metrics that can be used to evaluate offensive player performance.
- **OPS+** is one of these new metrics that effectively captures the offensive value of MLB position players.
- Baseball player evaluation is uncertain at best; can machine learning improve the accuracy of performance predictions?
- Namely, can a given position player's previous *n* years of hitting statistics produce a reasonable prediction of that player's OPS+ in year ***n+1***?

Dataset

- We used a collection of **annual** per-player offensive statistics published by baseball-reference.com dating back to 1871.
- Intended for historical statistical analysis rather than machine learning, the dataset was simply a repository of various statistics.
- We chose OPS+ as our label--a normalized measure of the frequency with which a player reaches base plus the average number of bases the player records per plate-appearance.

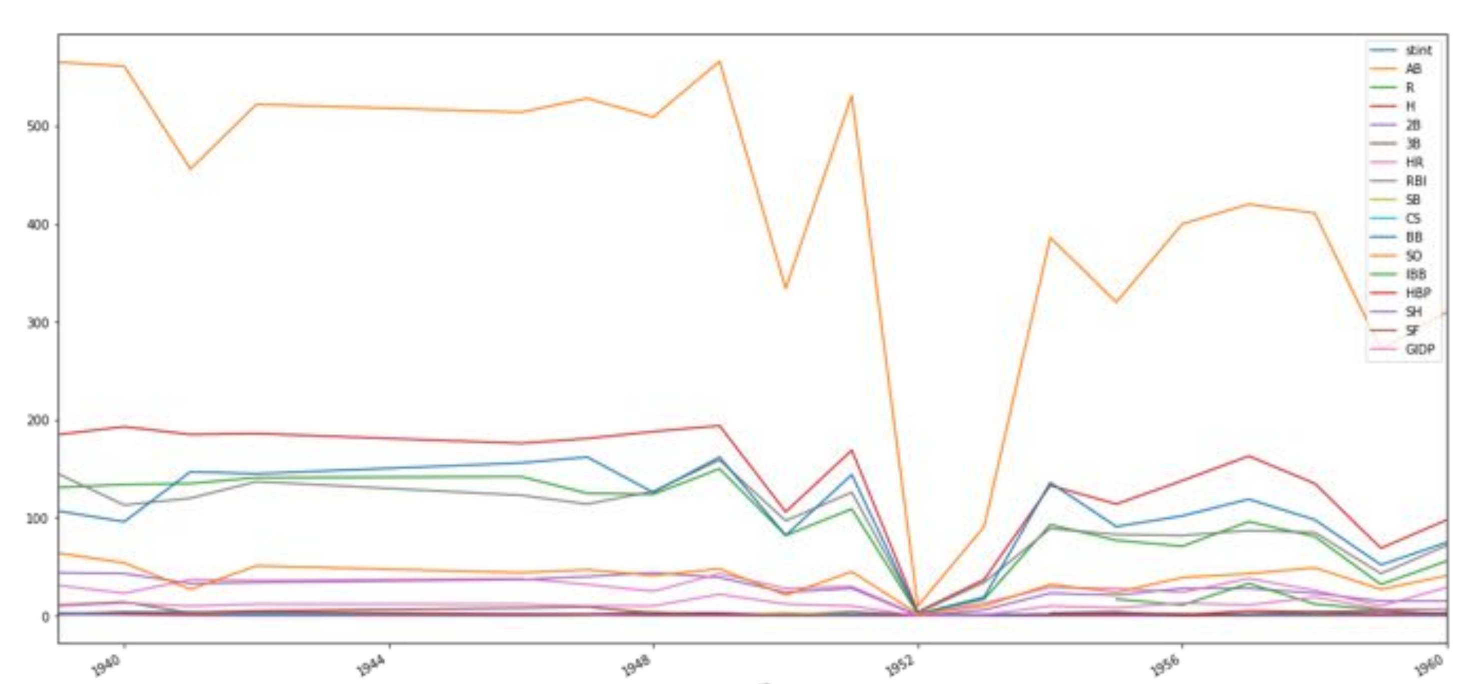

Figure 1: annual statistics over the course of a player's career (Ted Williams).

Models

- **Player-agnostic linear regression**, including unregularized, lasso, and ridge variants.
- **Player-specific linear regression**, including unregularized, lasso, and ridge variants.
- **Player-specific support vector regression**, using GridSearch for hyperparameter tuning.
- **Recurrent neural network** with long short-term memory **(LSTM).**
- We chose to **stack multiple recurrent states** with memory cells because it allows the model to determine more complex abstractions from the various offensive metrics.

Features

- Most statistics in the dataset are modern statistics (i.e. wins above replacement) designed to more accurately capture a player's contribution in a given area than classical statistics (i.e. batting average).
- Many baseball statistics are **highly correlated** i.e. the number of plate appearances and the number of games played.
- Example statistics:
  - Wins Above Replacement
  - Wins Above Average
  - Total Bases
  - Runs Produced

Linear Regression

- Player-specific
- Used unregularized, lasso, and ridge regression
- Trained on all years available for a given player
- Given small number of years and high variation among OPS_plus scores across years
- Gaps in data also contributed to poor performance

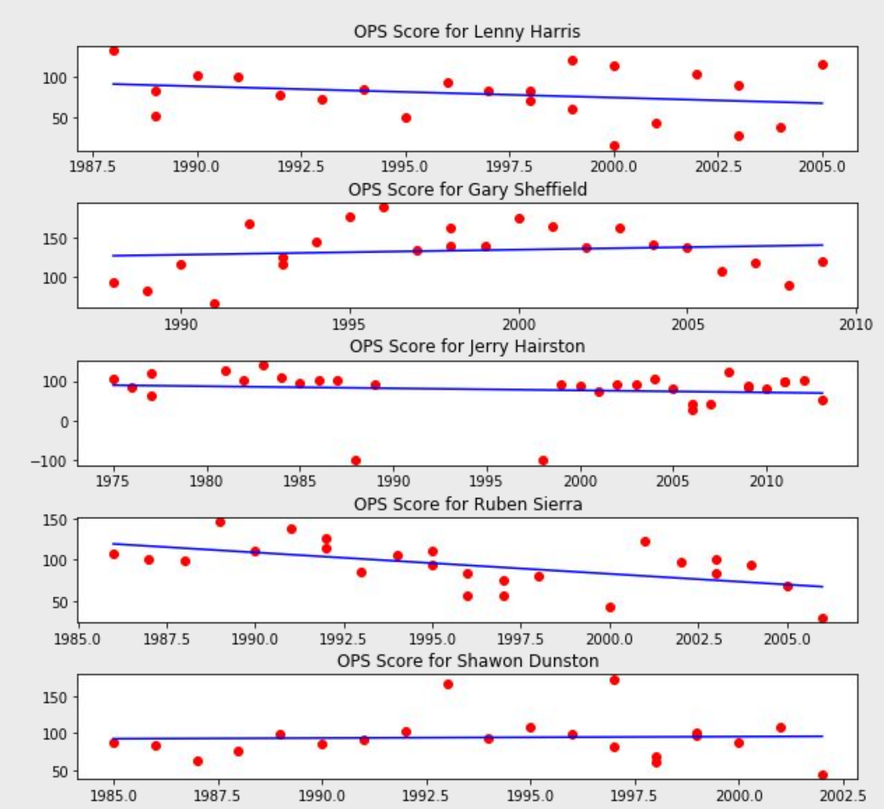

Figure 2: Linear Regression for 5 players

Support Vector Regression

- Player-specific support vector regression:
  - Trained on **14 years** of data
  - Tested on extrapolating the 15th year's OPS_Plus

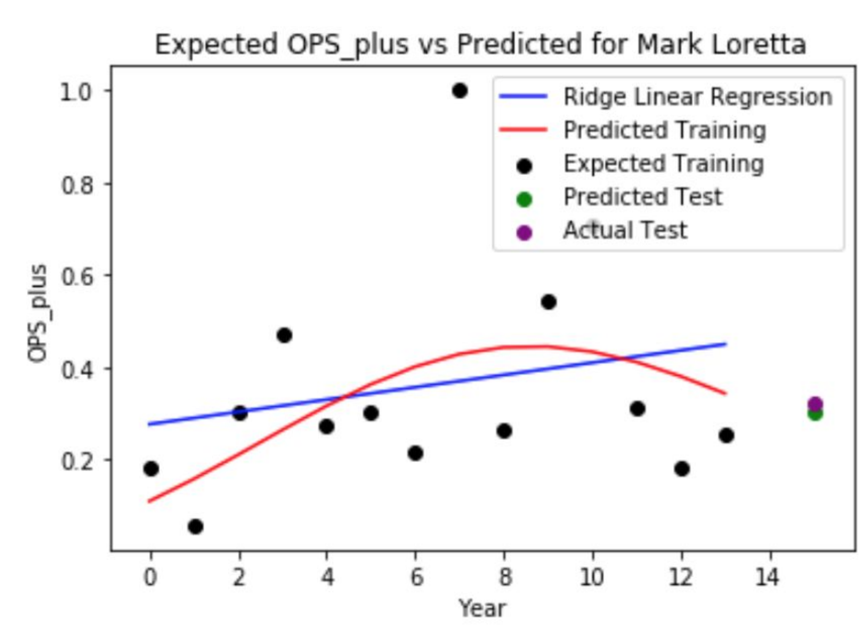

Figure 3: Example of a player on which regression performs well

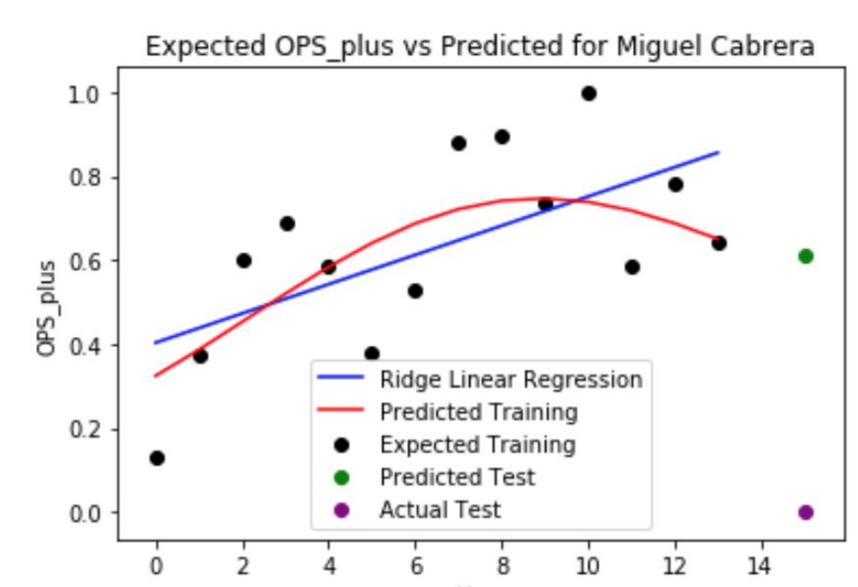

Figure 4: Example of a player on which regression performs poorly. Miguel Cabrera was injured in that year.

RNN with LSTM

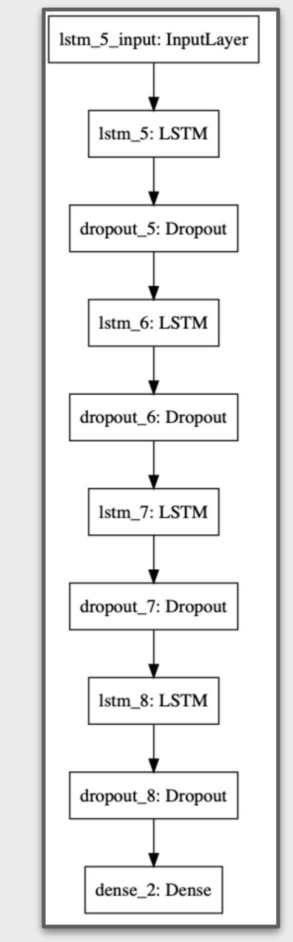

Figure 5: LSTM Layers

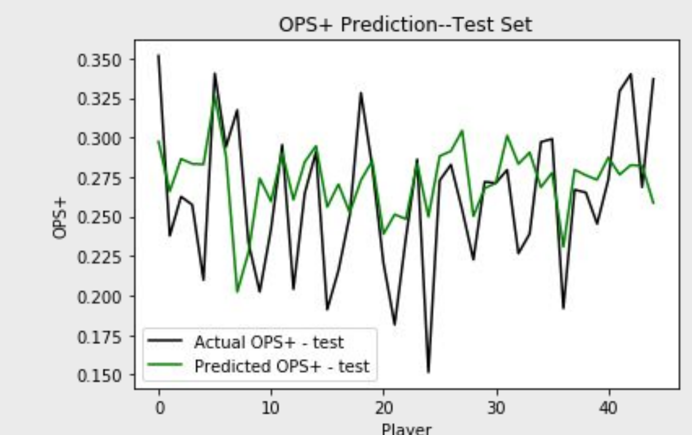

Figure 6: LSTM Prediction on Test Set

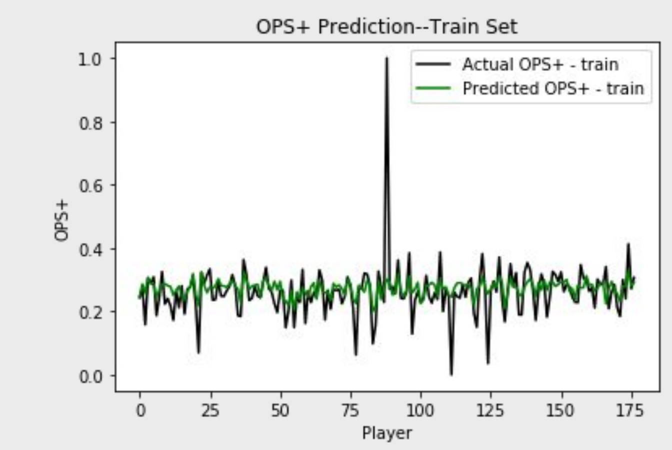

Figure 7: LSTM Prediction on Train Set

- Input shape is (num_players, years, features)
- Output shape is (num_players, OPS_plus)
- Split into train (177 players) and test (44 players)
- Used **dropout layers** to reduce overfitting on the training data
- Trained on 2012-2016 metrics to **predict OPS+ in 2017**

Evaluation

	Mean Squared Error	Average Absolute Error
SVR	0.088	0.232
LSTM (300 epochs)	0.0019	0.033

Figure 8: Core metrics for main models

- The **LSTM performs much better** than the SVR in terms of MSE and AE.
- All of these metrics were acquired from data that was scaled using a **minimax scaler.**
- Although the SVR is trained on more years, the LSTM is more informed because it also incorporates other metrics (such as age)

Conclusion & Future Work

- For players with an OPS+ career trend of a certain shape, our models performed reasonably well.
- We were unable to handle the unpredictability of OPS+ values created by player injury and other off-the-field issues.
- We recently discovered a database of play-by-play outcomes for all MLB games at **Retrosheet**; the incorporation of more finely-grained features may impact model performance.