**Using Machine Learning to Predict Yearly Average Pitching Statistics in Baseball**

Nathaniel Hermann

**Introduction:**

Of the major team sports, baseball lends itself most easily to statistical analysis. We can broadly sort the major sports along a gradient corresponding to ease of quantifiability. At the difficult end of this spectrum would be sports like soccer and hockey, fast paced and fluid, and at the other end would be baseball. Baseball is broken up into a series of events (innings, at bats, individual pitches) that have been catalogued, meaning that we can analyze any of these events in depth. Because of this, baseball was the first of the major sports to be subjected to statistical analysis, with the beginnings of theory starting in the 1970s, and analysis becoming commonplace in the early 2000s1.

The need for statistical analysis in baseball (and indeed, in all sports), comes from the need for management to make smart decisions both with player signings and use of players. For small market teams, the implementation of advanced statistics was a way to level the playing field with richer teams by exploiting market inefficiencies. However, in the past decades, the market has developed such that the teams that invests most heavily in analytics are wealthy teams (*e.g.,* the New York Yankees) who have applied their financial advantage to give them an even greater advantage in player acquisition. To this end, baseball analytics are now big business, with hires typically coming out of the Ivy Leagues, or universities of similar stature.

Here, we will be examining only one facet of baseball: pitching. A pitcher in baseball is responsible for keeping the opposing team from scoring runs by keeping runners off the basepaths. Two important statistics to consider for pitchers are earned run average (ERA) and fielding independent pitching (FIP). ERA is calculated by dividing the number of earned runs (ER) surrendered by the number of innings pitched (IP), normalized to 9 innings (the length of a game):

FIP estimates a pitcher’s quality independent of the defense behind him, using only home runs (HR), strikeouts (K), walks (BB), and hit-by-pitches (HBP), all of which a pitcher is responsible for2:

The FIP constant brings FIP to an ERA scale and is year dependent (depending on league averages):

**Aim:**

In this project, I aim to build a model which can predict both a pitcher’s ERA and FIP based on prior performance. Pitching statistics dating back to 1871 are publicly available (I used a set available at seanlahman.com). For each pitcher, we will look back at previous years’ statistics (one, three, and five years) and compare our modelled ERA/FIP to their actual performance.

**Methods/Results:**

The data available in the Sean Lahman database categorizes players by a unique ID, as well as identifying them by the year they played and the team they played for. For each unique combination, many statistics are available. These statistics are wins (W), losses (L), games played (G), games started (GS), complete games (CG), shutouts (SHO), saves (SV), outs pitched (IPouts), hits surrendered (H), earned runs (ER), home runs surrendered (HR), walks (BB), strikeouts (SO), opposing batting average (BAOpp), ERA, intentional walks (IBB), wild pitches (WP), hit-by-pitches (HBP), balks (BK), batters faced (BFP), games finished (GF), runs allowed (R), sacrifice flies allowed (SF), sacrifice hits allowed (SH), and batters grounded into a double play (GIDP). From these, I additionally calculated FIP for each entry.

The next step was to modify the ERA and FIP for each batter by the park factor of the park they played home games in each season. Park factor is a scaling factor, also available in the Sean Lahman database, that scales ER (and therefore ERA), by a factor that corresponds to how above or below league average a ballpark is in run scoring (this is necessary as each baseball park is idiosyncratically designed). After this normalization step, I made three datasets composed of vectors. Each vector has a pitcher’s ERA and FIP for a year, followed by their stats from previous years. These three datasets contained one, three, and five previous years of data respectively.

The first attempt I made at understanding the data was to perform some simple PCA and TSNE analysis to see if there was any correlation in the reduced dimensions with the known ERA and FIP of pitchers. I did this using the PCA and TSNE functions in scikit-learn with both two and three dimensions. I then color mapped the scatter plots to correspond to the known ERA and FIP of pitchers. This was performed for the one-, three-, and five-year datasets. This yielded little interesting, as there did not seem to be any recognizable pattern that the known ERA and FIP followed.

Next, I moved on to trying to make regression models to predict ERA and FIP. I used the ridge and lasso regression functions available in scikit-learn for these models. First, I take a dataset and shuffle it, then divide it into training, validating, and testing sets (these are 50%, 25% and 25% of the data respectively). From the dataset, I split off two columns of data corresponding to known ERA and FIP and shuffle these the same. The training data is then used to build the regression model, and the validation data is used to optimize the hyperparameter of the model to minimize the error. With this model, the training data is then used to make predictions. From these predictions, I find the residuals and pull two statistics out of this data. These are the mean average error (MAE) and the standard deviation (SD). The MAE is calculated from the absolute value of the residuals, while the SD is calculated from the residuals. This process was iterated over both regression types (ridge, lasso), the number of years of data (1,3,5) and the output statistic (ERA, FIP), giving us twelve models. We observe that the higher years (3/5) have lower MAE, and ridge performs better generally (Figure 1).

Next, I tried to improve performance by splitting the data between starting pitchers and relief pitchers. I filtered the data simply by looking for GS > 0, taking those and calling them starting pitchers. Splitting the data this way and repeating the modelling gave us twenty-four models. We observe that the lasso model using FIP, 5 years of data, and looking at starters performed the best. Generally, the starter models perform better than relief models. This intuitively makes sense, as it is understood in baseball that reliever performance has greater variance from year to year (Figure 2).

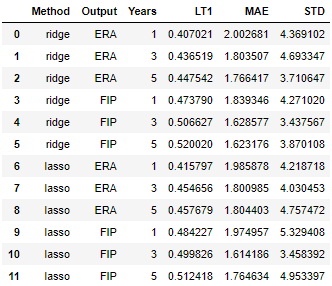
Finally, I tried to improve performance by including yearly normalization. It is known that baseball statistics on average vary from year-to-year, due both to changing rules, material conditions, and strategy. I calculated the average of each dataset statistic in every year and normalized each vector with respect to that year’s averages. I then repeated the above procedure for both starters and relievers. I found, somewhat surprisingly, that this did not improve performance (Figure 3).

The only other modelling attempt I made was to run the data through a neural network. After preprocessing to shuffle and normalize the data, it was fed through a 3-layer network with sigmoidal activation functions over five epochs. I did this to model both ERA and FIP, and constructed networks that give us losses of less than 1.2 runs from the correct value. These worked then around as well as the regression models.

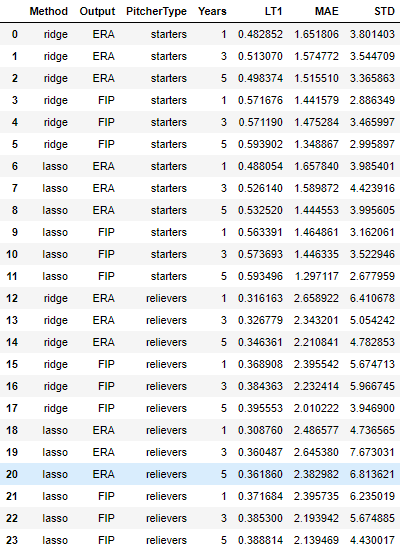
These models can generally be used to make predictions for future seasons. All one has to do is take the model outputted buy the code and use it to fit a vector of data from a pitcher-of-interest to the user. I imagine this model could be made much better through both more careful selection of data used in training (I blindly took all pitchers ever – excluding obvious outliers would certainly help) and inclusion of more data. It often does not look like including more than 3 years of data significantly improves performance, so the bottleneck in modelling will be the quality of data.

**Figures**

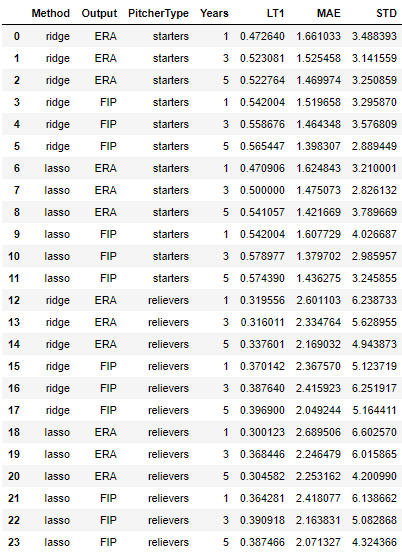
**Figure 1:**



**Figure 2:**



**Figure 3:**



**Citations**

1. James, B. (2003). *The New Bill James Historical Baseball Abstract*.
2. https://library.fangraphs.com/pitching/fip/