JOURNAL OF
EMERGING INVESTIGATORS

Article

Predicting baseball pitcher efficacy using physical pitch characteristics

Tejas Oberoi[1], Sam Saarinen[2]
[1] Los Altos High School, Los Altos, California
[2] Brown University, Providence, Rhode Island

SUMMARY

The efficacy of baseball pitchers can be predicted from prior pitching data using machine learning (ML) models. Previous ML studies relating to baseball have primarily involved predicting outcomes of baseball games and a thrown pitch. This paper is the first work that uses 16 game-independent features, which describe a pitcher's set of thrown pitches, to predict pitcher efficacy metrics, like walks/hits allowed per inning (WHIP), batting average against (BAA), and fielding independent pitching (FIP). We hypothesized that these 16 "physical features," measured by sensors, can explain greater than 50% of the variance while predicting pitcher efficacy. We applied neural network (NN) models to predict the efficacy metrics using all 16 features, while we used linear regression (LR) models to analyze the individual impact of each feature for predicting the efficacy metrics. We observed from the NN and LR models that the "ballFrequency" feature was the most impactful in predicting the WHIP for any pitcher. For the BAA and FIP metrics, the LR models showed that none of the features, including the pitch velocity and types of pitches thrown, were statistically significant; however, our NN model did improve the prediction of the BAA and FIP metrics. Based on our evaluations, the ML models could not prove our hypothesis, as the results accounted for less than 50% of the variance when predicting the pitcher efficacy metrics. Professional scouts can still use the results of our feature analysis to select better pitchers who have never played a game at the professional level.

INTRODUCTION

In baseball, professional teams rely heavily on advanced statistics related to the performances of batters and pitchers to maximize their success. Traditional metrics like earned run average (ERA) have been complemented by more detailed metrics, like batting average against (BAA), fielding independent pitching (FIP), and walks and hits per innings pitched (WHIP) (1). BAA gauges a pitcher's proficiency in preventing opposing players from getting hits (1). FIP assesses a pitcher's outcomes independent of the team's defense (1). WHIP measures a pitcher's ability to prevent opponents from reaching base (1). For all these metrics, lower values indicate better pitching efficacy, leading to fewer runs and more innings pitched. The creation and analysis of these advanced statistics in baseball is called sabermetrics (1).

Researchers have used machine learning (ML) models to predict different aspects of baseball using sabermetrics. For instance, Lee *et al.* and Hickey *et al.* used ML models to predict a thrown pitch's outcome (2, 3). Furthermore, another study by Bock used sabermetrics and ML models to predict pitchers' short-term and long-term efficacy on their particular teams (4). In addition, other researchers have implemented different types of ML models to predict baseball game outcomes based on a specific player's performance and other in-game statistics (5, 6). Finally, researchers have also applied ML models to predict the efficacy of Major League Baseball (MLB) batters for the next year based on their performances during the current season (7). All the above studies incorporated "non-physical features," like ball-strike or on-base percentages or batting averages, to predict only a single outcome metric. Unlike these previous studies, this study evaluates a single or a combination of "physical features" of a pitch. These features can be described qualitatively or measured using sensors as the input data to the ML models to predict multiple output metrics.

"Physical features" use advanced sensors or the human eye to measure/describe the feature of a pitch thrown either in a game or non-game setting; however, a "non-physical feature," such as the ball-strike count, must be measured in a game setting with or without the use of sensors. Therefore, using these "physical features," scouts and recruiters can evaluate pitchers who have never played a game in the MLB. Accurately predicting sabermetrics like WHIP, BAA, and FIP could be crucial for determining a pitcher's future success. A lower value of these metrics would imply more efficient innings with fewer base runners and runs allowed (8). Even if a pitcher seemed enticing because of his high velocity and diverse set of pitches, he would be ineffective in games if he had high values for these metrics. Knowing these statistics for a pitcher before they are selected or pitch in their first game at the professional MLB level could be pivotal for professional team scouts and managers. In addition, with the knowledge of which features impact a pitcher's efficacy, scouts could emphasize the important features while evaluating a pitcher.

In our study, we tested our hypothesis that physical pitch characteristics can predict greater than 50% of the variance, defined by the term 'r^2', in the efficacy of a pitcher. We developed neural network (NN) models to test this hypothesis and predict the three output efficacy metrics, WHIP, BAA, and FIP, using all 16 "physical features." We also created 16 linear regression (LR) models to analyze the individual impact of each feature for predicting these metrics. When predicting the metrics, the models did not account for more than 50% of the variance (r^2). However, the NN models for the WHIP and FIP metrics still provided statistically significant results. Additionally, when we added a 'non-physical feature' like WHIP to the input space, the NN model accounted for

DOI: https://doi.org/10.59720/22-270

more than 50% of the variance when predicting the BAA. Surprisingly, we observed that most of the features, such as how hard a pitcher throws and the types of pitches, were unable to significantly predict the efficacy of a pitcher. Our findings are contrary to popular belief among baseball scouts and recruiters who place a strong emphasis on these two characteristics while determining the efficacy of a pitcher (9). Consequently, professional scouts should not solely rely on these characteristics as the basis for evaluating a pitcher. Though, they could still use our analysis of the 16 "physical features" when selecting talented pitchers who have not yet played at the professional level.

RESULTS

To test our hypothesis that physical characteristics related to pitching could predict over 50% of the variance in a pitcher's effectiveness as measured by WHIP, BAA, and FIP, we developed NN and LR models using the 16 "physical features" as the input.

We performed these experiments to determine if the NN and LR models could accurately predict the metrics enough to satisfy the hypothesis. We defined the NN accuracy for all three output metrics as the percentage of predicted values within 10% of the respective output metric data range away from the actual value. We utilized the correlation coefficient (r) to examine the relationship between each feature and the LR analysis output metric. To evaluate the hypothesis, we used the variance (r^2) measure to predict the variance between the observed and predicted values for both the NN and LR models. We also used the average root mean squared error (RMSE) computed across five-fold cross-validation to assess the statistical significance of the NN and LR models' predictions of the output metrics to make conclusions regarding the quality of the predictions. We calculated the measure of variance (r^2) using the following,

$$RMSE = \sqrt{(1-r^2)} * SD \tag{1}$$

where RMSE is the root mean squared error of the model, Due to the NN model producing slightly different results for every model iteration using identical parameters, we trained the NN model using five-fold cross-validation with at least 50 epochs for each fold (well past the point of convergence). We reported the accuracy and the RMSE as the average of all five iterations. We also used an intercept-only model with the mean value of the respective efficacy metric in our validation dataset to establish the "baseline" RMSE and accuracy values for the three output metrics. Following the baseline prediction, we ran the NN and LR models to obtain the predicted efficacy metrics. For each model-produced RMSE value, we then ran an F-test to determine if the variance of the model was significantly different from the baseline results, implying that the model fit the output data better than the baseline model.

Predicting WHIP Metric

We observed that the intercept-only model yielded 48.2% accuracy and an RMSE of 0.158. Following the baseline prediction, we ran the best NN model, which yielded an average accuracy of 54.6%, an RMSE of 0.140, and an r^2 value of 0.215. The NN model was significantly better than the intercept-only model at predicting WHIP (p = 0.0464, F-test).

In addition, we trained and tested the 16 LR models to determine the RMSE for the relationship between each feature and the WHIP. Then, we ran F-tests comparing the RMSEs of each input feature with the variance of the WHIP to obtain the p-values. All features, except for the "ballFrequency" feature, produced statistically insignificant results (p > 0.05, F-test). We observed that the "ballFrequency" feature, which represents the proportion of pitches thrown by a pitcher that were not strikes, was statistically significantly correlated with WHIP (p = 0.024, F-test; r = 0.498; **Figure 1a**). For the "ballFrequency" LR model, the testing accuracy was also 55.8%, and the RMSE was 0.137 **(Figure 1a)**. The other scatterplots did not show any correlation due to the randomly scattered data points for the non-statistically significant LR models we evaluated using the other 15 physical features **(Figure 1b)**.

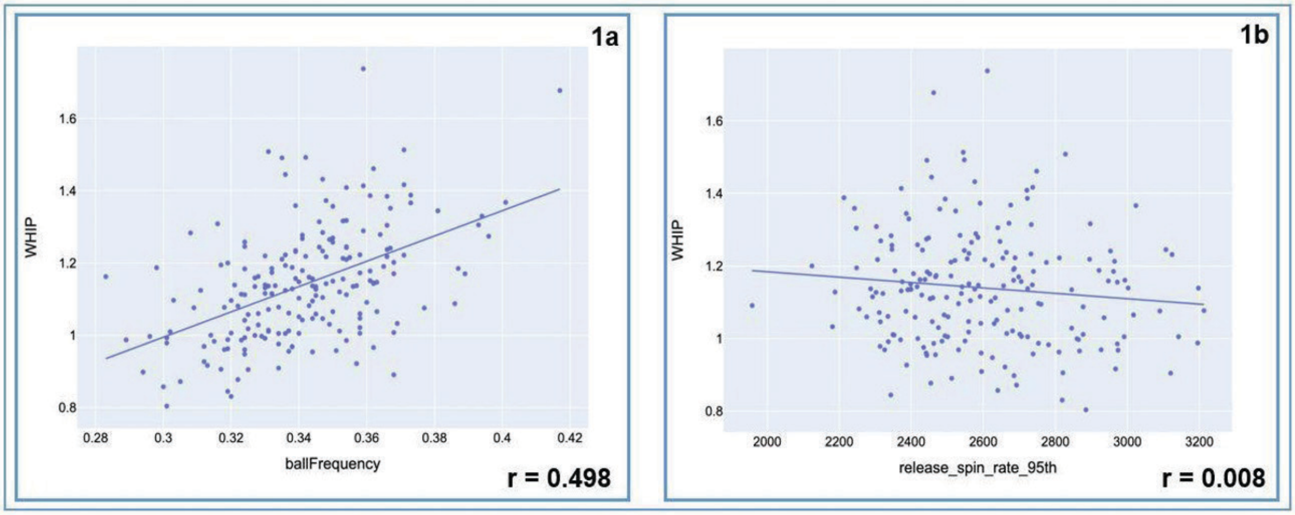

Figure 1: The "ballFrequency" feature is a strong predictor of WHIP. Both scatterplots were created from 1D LR models that attempted to analyze the relationship between each of the 16 physical features and WHIP. **(a)** Scatter plot of the validation dataset when the LR model was run for the "ballFrequency" feature and the WHIP pitcher efficacy metric (p = 0.024, F-test; r = 0.498). **(b)** Scatter plot for release_speed_95th (one of the other 15 "physical features"), and the WHIP metric when the LR model was run (p = 0.465, |r| < 0.1).

DOI: https://doi.org/10.59720/22-270

Predicting BAA Metric

We observed that the intercept-only model yielded 52.8% accuracy and an RMSE of 0.0245. Following the baseline prediction, we ran the best NN model, which yielded an average accuracy of 57.1%, an RMSE of 0.0231, and an r^2 value of 0.121. Finally, we ran an F-test on the RMSE of the NN and the SD of the BAA values that showed the NN model was not significantly better than the intercept-only model at predicting BAA (p = 0.207, F-test).

In addition, we trained and tested the LR models to determine the RMSE for the relationship between each feature and the BAA. Then, we ran F-tests comparing the RMSEs of each input feature with the variance of the BAA to obtain the p-values. We observed that no features were statistically significantly correlated with BAA (p > 0.05, F-test). However, we still analyzed the LR model results for the feature with the lowest p-value, "release_speed_95th." We obtained a testing accuracy of 55.8% and an RMSE of 0.0233 **(Figure 2a)**. The low correlation coefficient implies that no strong correlation existed between the 95th percentile of a pitcher's release speed and the BAA value **(Figure 2a)**. Since no feature produced statistically significant results and the scatterplots did not show any correlation, we could not conclude which LR models fit the data better than the baseline model for the BAA metric **(Figure 2)**.

Predicting FIP Metric

We observed that the intercept-only model yielded 53.3% accuracy and an RMSE of 0.856. Following the baseline prediction, we ran the best NN model, which yielded an average accuracy of 57.6%, an RMSE of 0.712, and an r^2 value of 0.309. Similar to the WHIP experiment, this NN model turned out to be significantly better than the intercept-only model at predicting FIP (p = 0.0053, F-test).

We also trained and tested the LR models to determine the RMSE for the relationship between each feature and the FIP. Then, we ran F-tests comparing the RMSEs of each input feature with the variance of the FIP to obtain the p-values. We observed that no features were statistically significantly correlated with FIP (p > 0.05, F-test). However, we analyzed the LR model results for the feature with the lowest p-value, "release_speed_95th". We obtained a testing accuracy of 53.3% and an RMSE of 0.797 **(Figure 3a)**. The low correlation coefficient implies that no strong correlation existed between the 95th percentile of a pitcher's release speed and the FIP value. Since no feature produced statistically significant results and no scatterplots showed any correlation, we could not conclude which LR models fit the data better than the baseline model for the FIP metric **(Figure 3)**.

Using WHIP/BAA/FIP as Input Features

We performed F-tests comparing the RMSE of the predicted values of a particular output feature when the input data was one of the other output features with the SD of that specific feature. The resulting p-values were used to determine if the two output features were significantly correlated with each other. Both the WHIP vs. BAA and FIP vs. BAA F-tests produced significant results (p = 0.0003 and p = 0.0005, respectively, F-test). Therefore, we decided to add WHIP to the input space to analyze how much the NN model improved in predicting BAA and also to evaluate if a non-physical feature, like WHIP, could help the NN model account for more than 50% of the variance in its predictions. With these added features, we ran the NN model on 100 epochs instead of 50 because it took longer to train the dataset. In addition, we tested the NN with added dropout layers of p = 0.3 between the hidden layers of the model.

We observed that for predicting BAA using the existing physical features and adding WHIP to the input space of our NN model, the average accuracy increased to 74.7% with the dropout layers (72.4% without), while the RMSE decreased to 0.0160. Additionally, we performed an LR analysis of WHIP vs. BAA, and we observed that the accuracy decreased to 67% with an RMSE of 0.0190. With the WHIP metric as an

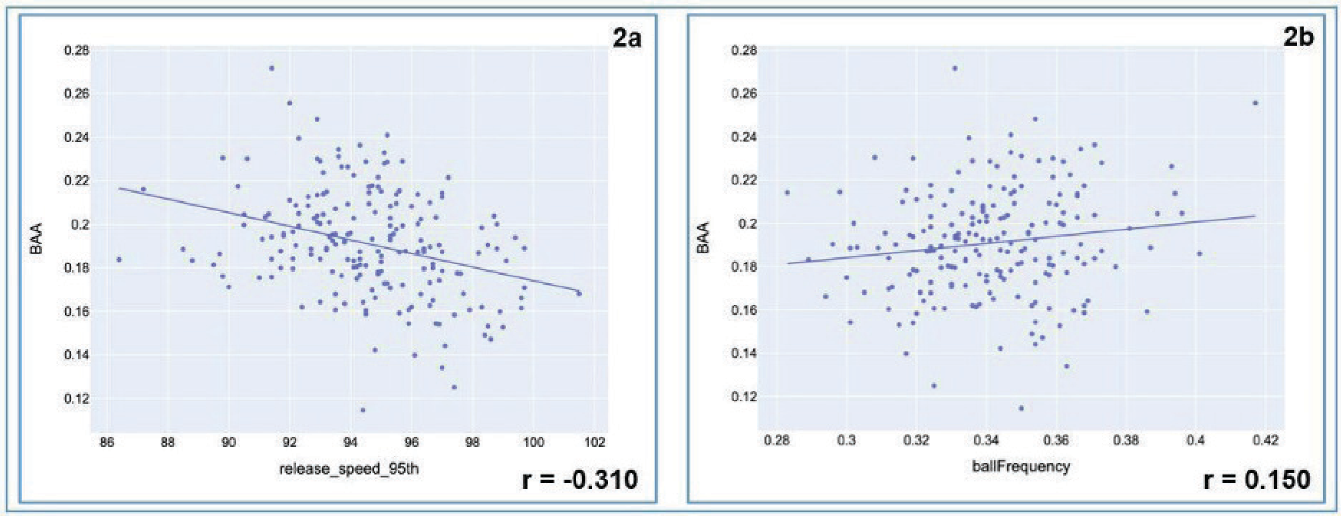

Figure 2: None of the 16 physical features individually are strong predictors of BAA. Both scatterplots were created from 1D LR models that attempted to analyze the relationship between each of the 16 physical features and BAA. **(a)** Scatter plot of the validation dataset when the LR model was run for the "release_speed_95th" feature and the BAA pitcher efficacy metric (p = 0.226, F-test, r = -0.310). **(b)** Scatter plot for an LR model run on another non-correlated and non-statistically significant feature, "ballFrequency" with the BAA metric (p = 0.437, F-test, r = 0.150).

DOI: https://doi.org/10.59720/22-270

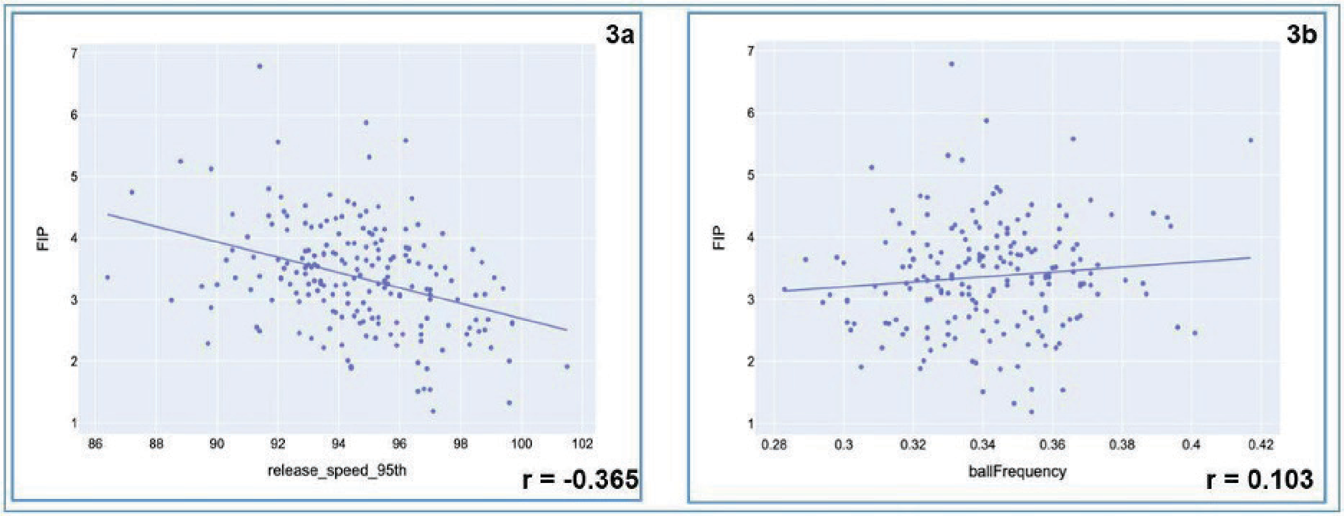

Figure 3: None of the 16 physical features exhibited a strong correlation with FIP. Both scatterplots were created from 1D LR models that attempted to analyze the relationship between each of the 16 physical features and FIP. **(a)** Scatter plot of the validation dataset when the LR model was run for the "release_speed_95th" feature and the FIP pitcher efficacy metric (*p* = 0.152, F-test, r = -0.365). **(b)** Scatter plot for an LR model run on another non-correlated and non-statistically significant feature, "ballFrequency" with the FIP metric (*p* = 0.464, F-test, r = 0.103).

added input feature, we ran another F-test comparing the RMSE of this new NN model and the SD of the BAA values that yielded a *p*-value of 0.00002, which implied that the NN model significantly improved the accuracy for predicting BAA when we added the WHIP metric into the model's input space **(Figure 4)**.

DISCUSSION

We used the RMSE metric for our research because of its usefulness in significance tests to make statistically backed solutions. When we analyzed the WHIP output metric, we observed that the LR model's RMSE with the "ballFrequency" feature and the NN model's RMSE yielded similar *p*-values of 0.024 and 0.0464, respectively. These results imply that the NN model did not improve predictions compared to simply using an LR model with only the "ballFrequency" feature. Despite the relatively low r^2 values (<0.3), the *p*-values still established statistically significant findings. This event probably occurred due to the sample size of the testing data (195) because as the sample size increases, the *p*-value decreases at a given F-value.

We also decided to run the NN using only the "ballFrequency" feature. This modified model yielded similar RMSE and accuracy values to the LR model with only "ballFrequency." Based on the similar RMSEs and *p*-values from the F-test for the NN and LR models, the original NN model did not use the other 15 features to its advantage and most likely implemented a linear regression-like function using the "ballFrequency" feature instead of a more sophisticated function using multiple features. Furthermore, the low correlation values and high *p*-values for the other 15 features imply that they do not add any meaningful value to predicting WHIP. These results directly contrast with the popular belief among professional scouts and coaches who consider that "physical" features such as "release_speed" (velocity of a pitch thrown), "pitchTypeEntropy" (mixing up pitch types frequently), and "release_spin_rate" (the amount a pitch spins when thrown) better indicate a pitcher's efficacy than other "physical" features.

However, unlike the WHIP metric, we observed that the BAA and FIP metrics had much lower RMSEs for the NN model than for any of the LR models. In addition, the FIP NN model predictions displayed a statistically significant result for the F-test that we ran between the NN model's RMSE and the SD of the FIP values (*p* = 0.0053). Since the NN model that predicted the FIP did not contain any statistically significant input features, the NN model probably created a function that used a combination of the "physical" features to significantly improve its performance for the FIP predictions (*p* > 0.05, F-test).

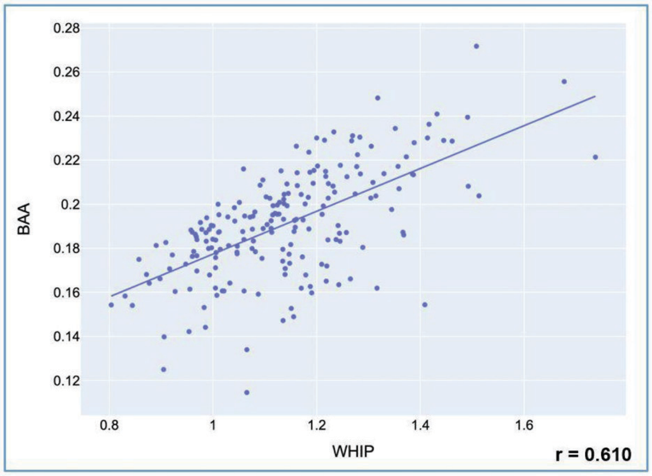

Figure 4: WHIP and BAA are closely correlated with each other. The scatterplot was created from an 1D LR model consisting of all the BAA and WHIP data points from the validation dataset. The model predicts the BAA using the WHIP (*p* = 0.0003, F-test, r = 0.610). Since the *p*-value is < 0.05, adding WHIP to the input space of our neural network model should significantly decrease the RMSE of the model's predictions on BAA.

DOI: https://doi.org/10.59720/22-270

Based on all the experiments conducted, we concluded that the current "physical" feature pool and the dataset lack sufficiency in explaining more than 50% of the variance of all three pitcher efficacy metrics ($r^2 < 0.5$ for all models). However, when we extended the feature pool to include a non-physical feature, WHIP, in the input space, we observed that the NN model's predictions accounted for more than 50% of the BAA's variance ($r^2 = 0.574$). The WHIP's influence on the model did not surprise us because of its high correlation with BAA compared to the other input features, and it essentially measures a similar property: the pitcher's efficacy **(Figure 4)**.

Additionally, we concluded that the "ballFrequency" feature plays the most important role in determining WHIP as it obtained the lowest *p*-value in the F-test and exhibited the highest linear correlation with WHIP. This finding makes sense since WHIP considers the number of walks a pitcher allows per inning. Pitchers who throw a lower number of balls will generally allow fewer walks, which results in a lower WHIP. Even though "release_speed_95th" had the lowest *p*-values in both F-tests with the BAA and FIP output metrics, we observed that these *p*-values did not reach the level of statistical significance. This result implied that throwing a high percentage of strikes helps determine a pitcher's efficacy. Furthermore, the current trend among scouts and coaches of using pitch velocity and a wide variety of pitches to determine a pitcher's efficacy, though crucial attributes, may not be as important because these MLB batters consist of the best in the world and therefore can hit pitches of any speeds and spin rates.

Because our target application for this work involves helping scouts model the performance of new pitchers based on their physical pitch characteristics before MLB play, we do not leverage any game history for our predictions. Because of the self-imposed constraint mentioned above and because of our relatively small sample sizes, we used a feedforward network rather than a sequential model, such as a recurrent neural network, which has become popular in efforts to predict MLB matchup outcomes.

However, some limitations do exist in the experiments conducted. For instance, a lack of pitcher data (only 777 pitchers) could have contributed to an NN model's inability to find a pattern more sophisticated than one exhibited by an LR model. A popular rule of thumb is to have a training dataset at least 10x (ideally more) the size of the number of parameters in an ML model. In our case, we had 16 input features and 2 hidden layers, which made the total number of parameters in the ML model about 200. Therefore, a dataset of more than 3,000 distinct pitchers (with ~2,000 pitchers in the training datasets and 500 for validation and testing datasets) could potentially improve the results. Acquiring this much data would require decades of MLB pitch data with tens of millions of thrown pitches because, with 5 years' worth of data (three million pitches), we could only find 777 distinct pitchers who threw enough pitches (over 1,000) in the MLB.

Because we only selected game-independent "physical" features for this study, the models only knew about the properties of the thrown ball and nothing about the "non-physical" features like a batter's batting average, game score, ball/strike count, fielder positions, and more. We decided only to use these features because coaches can use them to directly evaluate a pitcher's potential success in a non-game setting, like a tryout, before they sign the pitcher to play for a professional team. In prior sabermetric studies, many experiments involved knowing some of these "non-physical" features, especially the opposing batter's statistics against a particular pitcher, which provided useful information that determined the likelihood of the pitcher allowing a hit or getting an out, which can directly be used to predict a pitcher's efficacy. We did not include these in-game factors as they defeat the objective of our study. Coaches would need to observe the pitcher's performance in multiple real-life games to capture patterns in how they perform with these non-physical attributes to predict their efficacy in future games accurately. Due to the existing statistically significant results with the WHIP and FIP metrics with the current dataset and feature pool, more data and features could potentially result in a significant correlation with BAA and improve the RMSE of the FIP and WHIP NN models as well.

Instead of attempting to predict a pitcher efficacy metric using a training dataset of professional baseball pitchers, researchers could analyze each pitcher individually. For instance, one could break up the statistics of each pitcher by each pitch that they throw. With each thrown pitch, one can use 1D LR analysis and hypothesis testing for each physical feature as the input and the pitch result (i.e., hit or no hit) as the output to determine if these features convey useful information pitch-by-pitch level for that individual pitcher.

In addition, one could explore more robust types of ML models and techniques with more pitcher data (i.e., 20 years instead of 5 years). Doing this might yield more accurate results for predicting pitcher efficacy metrics using only "physical pitch characteristics."

One could also extend this pitcher efficacy evaluation using ML models for players at the high school and collegiate levels because of the higher variation in skill among the players at these levels. Potentially, velocity and other physical pitch features may better predict a pitcher's efficacy in these settings. In such a setting, an NN model could potentially yield more accurate results for predicting the pitcher efficacy metrics. However, we could not find publicly released data (no equivalent version of "Statcast") for the amateur levels because most teams only collect statistics for their own information and do not get paid as much as professional teams do to release data to third-party companies that release them to the public.

Overall, our study conveys that pitcher traits like high velocity, spin rate, and throwing many different types of pitches are not defining factors of a pitcher with high efficacy due to the statistically insignificant results of the models. In addition, most importantly, a pitcher who consistently pitches within the strike zone has significantly lower WHIPs (a measure of efficacy) than other pitchers.

MATERIALS AND METHODS

Dataset Preprocessing

The dataset used for this research was obtained from Statcast, which used an advanced camera-driven tracking system that was installed in every MLB stadium to extract advanced features for each pitch, such as its velocity, spin rate, exit velocity, pitch movement, pitch location, and more (10). The Statcast dataset is comprised of the pitch-by-pitch data from 2017 – 2021 from the Kaggle website. It was combined into one Pandas data frame that yielded 3,149,505 rows of pitch data and 92 columns of pitch features (11). The

DOI: https://doi.org/10.59720/22-270

dataset was initially modified by deleting pitches that resulted in extremely rare outcomes (such as pickoffs), and only data from 777 pitchers that had pitched at least 1000 pitches over the 5 seasons was used. As a result, the number of rows was reduced to 2,835,562 (90% of the original data). The number of columns was also reduced from 92 to 83 by removing 9 deprecated columns.

The modified dataset consisted of an "events" column, which described the outcome of an at-bat, and a "description" column, which described if the resulting pitch was a ball, strike, foul, or hit into play. These two columns were combined into a new and more detailed "description" column that contained the results of the at-bat from the "events" column and the results of the other pitches from the original "description" column. Additionally, some of the infrequently occurring variables in the new "description" column, a subset of a more commonly occurring result, were combined to obtain the modified final dataset.

The modified dataset was used to extract the 16 "physical" features, as defined below, that were used as input for our ML models from our current assortment of 83 columns **(Table 1)**. The "pitch_type" feature determined the type of thrown pitch and was classified using the one-hot encoding method as "0" if it was not the pitch type thrown and "1" if it was the pitch type thrown. Five categories for the "pitch type" feature – fastball, curveball, change-up, slider, and other pitches – were created (12). The four pitch types selected were the most commonly thrown pitches, while the "other_offspeed" category comprised all other pitches (knuckleball, forkball, etc.) that were rarely thrown. The "zone" feature determined the location of the thrown pitch, whether it was thrown "high" (above the batter's waist) or "low" (below the batter's waist). The "release_ extension" feature measured the horizontal extension in feet of the pitcher's arm before the ball was released (called the "release extension" of the pitch). The "release_spin_rate" and "release_speed" features described the spin rate and the velocity of the thrown pitch, respectively (13). The "p_throws" feature was used to classify right-handed pitchers as 0 and left-handed pitchers as 1. Unlike the above features based on a single-thrown pitch, the "ballFrequency" and "pitchTypeEntropy" features were based on all the pitches thrown by a specific pitcher from the dataset of 777 pitchers. The "ballFrequency" of each pitcher was calculated using the following,

$$ballFrequency = \frac{pitches\ thrown\ as\ balls}{total\ pitches\ thrown} \quad \textbf{(2)}$$

"PHYSICAL" FEATURES	EXPLANATIONS
pitch_type_FF (PTFF)	fastball (FF)
pitch_type_SL (PTSL)	slider (SL)
pitch_type_CH (PTCH)	curveball (CH)
pitch_type_CU (PTCU)	change-up (CU)
pitch_type_other_offspeed (PTOS)	off-speed pitches (OS)
zone_high_zone (HZ)	% of high zone pitches (above waist)
zone_low_zone (LZ)	% of low zone pitches (below knee)
release_extension_5th (RE5)	5th percentile of pitch release
release_extension_95th (RE95)	95th percentile of pitch release
release_spin_rate_5th (RSR5)	5th percentile of pitch spin rate
release_spin_rate_95th (RSR95)	95th percentile of pitch spin rate
release_speed_5th (RSPD5)	5th percentile of pitch speed
release_speed_95th (RSPD95)	95th percentile of pitch speed
p_throws (PT)	left or right-handed pitchers
ballFrequency (BF)	frequency of balls
pitchTypeEntropy (PTE)	distribution of pitches

Table 1: Table 1: Input "physical" features. The pitch_type feature defines the type of pitch thrown. The zone feature describes the location of pitch: low" zone means the pitch was thrown below the batter's waist, and "high" zone means the pitch was thrown above the waist. The "release_extension" feature measures the horizontal extension of the pitcher's arm from the starting point. The "release_spin_rate" feature measures the spin rate of a thrown pitch for a single pitcher. The "release_speed" feature measures the velocity of the thrown pitch for a single pitcher. All three of the features mentioned above were computed using the 5th and 95th percentile values for each pitcher (as each pitcher had pitched over 1000 times). The "p_throws" feature classifies right-handed pitchers as 0 and left-handed pitchers as 1. The "ballFrequency" feature describes, for each pitcher, the relative frequency of pitches thrown as "balls." The "pitchTypeEntropy" feature computes the distribution of the different types of pitches thrown (FF, SL, CH, CU, OS) by a particular pitcher.

DOI: https://doi.org/10.59720/22-270

To compute the "pitchTypeEntropy," which described the distribution of the type of pitch thrown for each pitcher, the Shannon Information Entropy value (E(x)) was used on the five one-hot encoded "pitch type" features to estimate a value using the following,

$$E(x) = -\sum_{i=1}^{n} P(x_i) * log_2(P(x_i)) \tag{3}$$

where P(x_i) is the probability of an event xi occurring.

A higher E(x) value implied that the pitcher threw a larger variety of pitches than a small E(x) value, which implied that the pitcher threw fewer pitches.

Creating the Input Features for Each Pitcher Using the Data Given for Each Thrown Pitch

After grouping the rows by the pitchers, a Pandas GroupBy object was created from the original data frame with the one-hot encoded categorical columns ("p_throws," "pitch_type," "zone) from the 777 groups containing all the pitches thrown by a particular pitcher. Next, the 5th and 95th percentiles of each quantitative feature ("release_speed," "release_spin_rate, "release_extension") for each pitcher group in all the pitches were computed. Percentiles were used rather than mean values because two pitchers with similar mean physical features can vastly differ in pitching style. For example, one pitcher may predominantly throw 90 mph fastballs, while another may throw a mix of 95 mph fastballs and slower offspeed pitches at 70 mph. Both pitchers would have the same mean release speed but different 5th and 95th percentile values. For each of the one-hot encoded qualitative features, relative frequencies were computed of their occurrences in the pitches for each pitcher group. As a result, the input features for the ML models were created **(Table 1)**.

Dataset Preprocessing

The WHIP output metric was evaluated for the initial simulations because it is one of the most well-known sabermetrics and is easily computed (14). WHIP can be computed using the following,

$$WHIP = \frac{walks + hits}{IP} \tag{4}$$

where, "walk + hits" is the total number of pitches in the dataset whose description column contained a walk or a hit (single, double, triple, or home run), and

IP = Total innings pitched by each pitcher, which is equal to the total number of outs divided by three.

However, WHIP was not the only metric that measured a pitcher's efficacy, and when paired with other sabermetrics, it can provide more accurate predictions (15). Therefore, two other sabermetrics, BAA and FIP, were also evaluated.

Unlike the WHIP, the BAA measures pitcher efficacy based on each batter's ability and considers only hits and no walks (16). BAA is defined using the following,

$$BAA = \frac{Hits}{At - bats} \tag{5}$$

where, at-bat is defined as any event excluding a walk, a hit by pitch (HBP), a sacrifice, and catcher's interference (17).

To compute the BAA for each pitcher, the same GroupBy object was used. However, instead of summing up the walks and hits, only the hits were summed up by ignoring any value in the "description" column that was a walk, an HBP, a sacrifice event, or a catcher's interference.

For evaluation of the FIP metric, only home runs (HR), strikeouts (K), hit by pitch (HBP), and walks (BB) were considered, as these were completely fielder-independent metrics and were controlled by only the pitcher. FIP is defined using the following,

$$FIP = \frac{13HR - 3(BB + HBP) - 2K}{IP} + C \tag{6}$$

where, IP is the innings pitched and C is a constant that is assigned a value of 3.18, which is the average of all C values from 2017 – 2021.

Overall, all three of these metrics interpreted the same outcome, a pitcher's efficacy, while minimizing the dependence on confounding factors (such as fielders, errors, game situations, and more). The main difference between these three metrics is the events that they considered.

Training/Testing/Validation Data

The dataset of 777 pitchers was split randomly to produce a training dataset of 582 pitchers and a validation dataset of 195 pitchers for the LR models. The NN models were trained and tested on the whole dataset of 777 pitchers using cross-validation.

Neural Network Model

A two-hidden layer NN model was used to evaluate the dataset by varying the hidden layer widths from 4-14 and trying every possible combination of the layers using a linear search with increments of 2 (4, 6, 8, 10, 12, 14). In addition, the leaky_relu activation function was implemented on each hidden layer. The models were developed and run using the PyTorch library in Python, which provides great flexibility in calibrating the properties of each model. The initial results indicated that the accuracies were similar regardless of the width of the hidden layers. So, for the main model, the decision was to set the sizes of the two hidden layers to 8 and 6, respectively.

Linear Regression Model

16 LR models were applied to determine the correlation between each of the 16 "physical" features with each of the three output metrics. The LR and the NN model results were compared due to a suspicion that the NN model was only using a few of the 16 features to make its predictions.

Neural Network Model Training and Validation Process

Before training the model, the training and validation input datasets were normalized by using the z-score normalization formula:

$$x_{new} = \frac{(x_i - mean)}{SD} \tag{7}$$

where, x_{new} = transformed dataset value
x_i = initial dataset value
SD = standard deviation

A five-fold cross-validation training loop was run on the 777-pitcher dataset using 50 epochs on a CPU, batch sizes of 4, and an Adam optimizer with a learning rate of 0.001 and a weight decay of 0.01. Each fold's validation (testing) dataset

DOI: https://doi.org/10.59720/22-270

size consisted of 195 pitchers. After the model ran all five folds, the five iterations' average accuracy and the RMSE were computed using the NumPy Python library.

Linear Regression Model Training and Validation Process

The 16 LR models were trained on the training dataset (the independent variable was the respective input feature, and the dependent variable was the output feature). Then, the testing dataset was used on the trained LR models (same independent and dependent variables) to calculate the RMSE. In addition, 16 scatterplots of the physical features in the testing data vs. the actual pitcher metrics were made to visualize any potential correlations **(Figures 1–4)**. The NumPy, Scikit-Learn, and Plotly Python libraries were used to plot the scatterplots for each linear regression model and to compute the statistical values like the RMSE and r^2 values. To do the hypothesis testing, the Sci-Py library was used because it contained an in-built function to compute the *p*-value of our F-test results.

APPENDIX

Code used to run our experiments - https://github.com/toberoi05/BaseballResearch

Received: October 19, 2022
Accepted: May 14, 2023
Published: January 11, 2024

REFERENCES

1. Beneventano, *et al.* "Predicting Run Production and Run Rrevention in Baseball: The Impact of Sabermetrics." *International Journal of Business, Humanities, and Technology,* vol. 2, no. 4, Jun. 2012, www.researchgate.net/profile/Bruce-Weinberg/publication/266344641_Predicting_Run_Production_and_Run_Prevention_in_Baseball_The_Impact_of_Sabermetrics/links/54bab6740cf253b50e2d0563/Predicting-Run-Production-and-Run-Prevention-in-Baseball-The-Impact-of-Sabermetrics.pdf.
2. Lee, Jae Sik. "Prediction of Pitch Type and Location in Baseball Using Ensemble Model of Deep Neural Networks." *Journal of Sports Analytics,* vol. 8, no. 2, Jul. 2022, pp. 115-126, https://doi.org/10.3233/jsa-200559.
3. Hickey, Kevin, *et al.* "Dissecting Moneyball: Improving Classification Model Interpretability in Baseball Pitch Prediction." *Proceedings of the Annual Hawaii International Conference on System Sciences,* Jan. 2020, https://doi.org/10.24251/hicss.2020.031.
4. Bock, Joel. "Pitch Sequence Complexity and Long-Term Pitcher Performance." *Sports,* vol. 3, no. 1, Mar. 2015, pp. 40–55, https://doi.org/10.3390/sports3010040.
5. Heaton, Connor, and Prasenjit Mitra. "Learning to Describe Player Form in the MLB." *Communications in Computer and Information Science,* Apr. 2022, pp. 93–102, https://doi.org/10.1007/978-3-031-02044-5_8.
6. Huang, Mei-Ling, and Yun-Zhi Li. "Use of Machine Learning and Deep Learning to Predict the Outcomes of Major League Baseball Matches." *Applied Sciences,* vol. 11, no. 10, May. 2021, p. 4499, https://doi:10.3390/app11104499.
7. Watkins, Christopher. "Novel Statistical and Machine Learning Methods for the Forecasting and Analysis of Major League Baseball Player Performance." *Chapman University,* 2020, https://doi:10.36837/chapman.000139.
8. Rogers, Michael. "What Is WHIP in Baseball? (Fully Explained)." *Nations-Baseball,* www.nations-baseball.com/whip-in-baseball. Accessed 29 Aug. 2022.
9. Destefano, Christine. "What Scouts Look For." www.baseballakademie.de/wp-content/uploads/2016/02/whats_scouts_look_for.pdf. Accessed 29 Aug. 2022.
10. "Statcast | Glossary | MLB.com." *MLB,* www.mlb.com/glossary/statcast. Accessed 29 Aug. 2022.
11. "MLB Statcast Data." *Kaggle,* www.kaggle.com/datasets/s903124/mlb-statcast-data. Accessed 29 Aug. 2022.
12. "What Are the Pitch Types Generated from MLB Statcast for Speed of Pitch?" *Daktronics,* www.daktronics.com/en-us/support/kb/DD3312647. Accessed 29 Aug. 2022.
13. "Statcast Search CSV Documentation." *Baseballsavant,* baseballsavant.mlb.com/csv-docs. Accessed 29 Aug. 2022.
14. Slowinski, Piper. "Whip." *Sabermetrics Library,* library.fangraphs.com/pitching/whip/. Accessed 29 Aug. 2023.
15. Slowinski, Piper. "FIP." *Sabermetrics Library,* library.fangraphs.com/pitching/whip/. Accessed 29 Aug. 2022.
16. "Batting Average | Glossary | MLB.com." *MLB,* www.mlb.com/glossary/standard-stats/batting-average. Accessed 29 Aug. 2022.
17. Nelson, Steve. "What Is an At-Bat in Baseball?" *Baseball Training World,* baseballtrainingworld.com/what-is-an-at-bat-in-baseball/. Accessed 29 Aug. 2022.