Predicting Undervalued Major League Baseball Players

by Clay Winder and Tommy Skodje

Abstract

This project aims to determine how to effectively find undervalued Major League Baseball players using machine learning. The dataset used is from Statcast, a camera system implemented in every major league ballpark that tracks data about players and the baseball itself. Weighted On-Base Average (wOBA) is predicted through the use of Statcast statistics. Pearson correlation was used for feature selection, and k-nearest neighbors, decision tree, gradient boosted trees, random forest, and linear regression were used as models to predict wOBA. The best performing model (k-nearest neighbors) produced a MAE of about 0.0440, a RMSE of about 0.0618, and a R-squared value of about 0.8849.

Introduction

Major League Baseball (MLB) and sports in general have recently become more statistics-focused. The recent trend in sports statistics aims to answer the question of how valuable an individual player is in helping their team win games. Although pitching and defense are also critical to a team's success, this paper focuses specifically on what makes a hitter valuable to their team.

The traditional measure for quantifying how good a hitter performs is in their basic counting and rate statistics. Counting statistics are integer values that represent how many times a player achieved a certain outcome [1]. Examples of counting stats include home runs, walks, hits, and strikeouts. The other category of baseball statistics is rate stats. Rate statistics usually represent the number of times a player achieved a certain outcome divided by the number of chances they had to achieve that outcome [1]. For instance, batting average is a rate statistic represented by the following formula.

$$\frac{Hits}{At\ Bats} = Batting\ Average\ (bounded\ between\ 0\ and\ 1)$$

Basic statistics have traditionally been the benchmark of how valuable a player is to their team. Players were usually paid in accordance to how good their counting and rate statistics were compared to the league average. However, recent trends in sabermetrics (the statistical analysis of baseball) have exposed the shortcomings of basic stats. Despite usually being fairly accurate at recording a player's contributions to their team, there are times when basic statistics do not fully capture a player's contributions on the baseball diamond. For example, a player who collects many singles may have a large amount of hits, but those singles are ultimately not worth very much if they do not translate into runs for the team. A player who collects an equal number of hits, but collects hits of higher value such as doubles, triples, and home runs has more objective value for their team. This problem is confounded further by the fact that a double is not exactly twice as valuable as a single, and a home run is not twice as valuable as a double. In other words, metrics such as slugging percentage (a rate statistic of how many bases a player generates per at-bat) or total bases (a counting stat that records the total number of bases a player generates) do not accurately account for how valuable an individual hit is.

The need for a more complete assessment of the value of an individual hit led to the creation of Weighted On-Base Average (wOBA). Weighted On-Base Average is an advanced statistic that is derived from basic counting and rate stats. As the website Fangraphs explains, “wOBA is based on a single concept: Not all hits are created equal” [2]. Due to this, wOBA uses a formula where walks, singles, doubles, triples, and home runs are weighted with coefficients representing how valuable they were during a particular season. The formula for wOBA for the 2013 season was

$$wOBA = \frac{0.690 \times uBB + 0.722 \times HBP + 0.888 \times 1B + 1.271 \times 1B + 1.271 \times 2B + 1.616 \times 3B + 2.101 \times HR}{AB + BB - IBB + SF + HBP}$$

[2] (explanations for acronyms within the formula can be found in reference 2). The coefficients within this formula may change from year to year to represent the varying values for each kind of outcome based on the season.

Weighted On-Base Average has become a dependable benchmark for which to measure a hitter’s effectiveness in the major leagues. It has proven to be more accurate at evaluating a player’s performance than traditional statistics and is a useful “catch-all” metric to evaluate a hitter’s value to their team. Due to wOBA’s usefulness, it has become a critical metric to predict when constructing a roster. If a general manager can predict a hitter’s wOBA, they will be able to predict how valuable that hitter will be to their team’s chances of winning. Furthermore, wOBA makes it easier to identify undervalued players. This is because wOBA is a fairly accurate measure of a player’s offensive value, so if a player has a high wOBA but is not paid a similarly high salary, it can be said with reasonable certainty that the player is being paid below their value to the team. Discovering undervalued players can help teams stay competitive without a high payroll. Furthermore, if a team is able to find players who are likely to perform better than expected, they can get even more value from a player. **Finding these undervalued players is the main motivation behind this project.**

To find which players are undervalued, it is first necessary to understand how player performance is predicted in Major League Baseball. Statistics only account for a player’s past performance and do not contain any information about how a player will perform in the future. This is where Statcast becomes a useful supplement to statistics to predict a player’s future performance. Statcast is a system of cameras installed at every MLB stadium that records data about players and the baseball itself [3]. The data collected ranges from things like player sprint speed to the launch angle of a ball hit off of a bat. Statcast statistics are collected for every major league player and are used to calculate expected performance for players. This project focuses on the Statcast statistic xwOBA (Expected Weighted On-Base Average). xwOBA calculates the expected wOBA value for each ball a player hits based on similar batted balls. Each batted ball is given an "expected" wOBA value based on its exit velocity, launch angle, and sprint speed of the player hitting the ball [4].

Similar to wOBA, xwOBA has become an important metric for assessing a player's hitting performance. According to Major League Baseball, "xwOBA is more indicative of a player's skill than regular wOBA, as xwOBA removes defense from the equation. Hitters, and likewise pitchers, are able to influence exit velocity and launch angle but have no control over what happens to a batted ball once it is put into play" [5]. Players with higher xwOBA are likely to be more valuable to their team.

Since xwOBA is such an important aspect in assessing a player's hitting value, finding inefficiencies in the statistic is a way in which general managers can find an advantage over their competition. Players such as Nolan Arenado consistently have higher wOBA values than xwOBA values, showing that the statistic does not account for some aspects of his offensive value [5]. If general managers can understand where xwOBA falls short, they can more easily find undervalued players and increase their chances of winning. This project aims to understand why certain players are undervalued, and where xwOBA falls short.

This project will use Statcast data gathered from Baseball Savant [5] from the 2015 through 2023 seasons. The data includes both traditional statistics as well as Statcast data for each player who played in any one of those seasons. The label to be predicted is a player's change in wOBA value from season to season (not to be confused with predicting xwOBA). The percent change in wOBA is then applied to a player's wOBA in the previous year to get their predicted wOBA value for the upcoming season. Please note that the error metrics in the below sections are relative to the actual wOBA values, and not the percent change in wOBA.

Background

Previous machine learning work related to baseball performance predictions has concerned predicting pitcher and player performance. However, this is usually done using traditional stats such as hits, walks, and home runs for hitters or strikeouts, walks allowed, and earned run average for pitchers. Recently, more research has been performed involving predicting undervalued pitchers based on undervalued pitches [6] and predicting the wOBA for a player in a future season, relying in part on Statcast data [7]. However, in the case of the wOBA model, very few of the multitude of advanced stats available from the Statcast data were considered, and no explanation for why was provided. In the interest of exploring the differences the other statistics could offer, we tried to develop our own models to predict future wOBA that take into account the plethora of advanced stats available.

Methods

The models that were chosen for this project were:

- K-Nearest Neighbors
- Decision Tree
- Gradient Boosted Tree
- Random Forest
- Linear Regression

K-nearest neighbors is an algorithm that works by calculating the mean of the nearest "neighbors" to a point. In the case of this problem, the k-nearest data points to a player's wOBA during a certain season are found and then averaged to calculate that player's predicted wOBA value for the next season. The group thought this would be a good algorithm for this particular problem because its downside of being slow is mitigated by having a small number of rows in the Statcast dataset. It is also very interpretable, as it is a very simple algorithm. Interpretability is important in this problem because the goal of the project is not only to find undervalued but to understand why they are undervalued as well. An additional reason why k-nearest neighbors was chosen was that it may act similarly to baseball scouting procedures. When baseball players are drafted, they are often given a "draft comparison" to a current player in the league. K-nearest neighbors provides an objective "draft comparison" to similar players in terms of wOBA.

Decision tree is another very simple and interpretable model. It works by selecting an attribute and value that results in maximum information gain (the group used Gini index to represent information gain in this project) and then splitting the data at that point. The data is continuously split until certain stopping criteria are met (in this project, maximum depth and minimum leaf samples were used). The group thought that decision tree would be a good algorithm to use because it explicitly defines which attributes are the most relevant predictors of wOBA.

Gradient boosted tree is a model that works by combining many smaller decision trees, and assigning a weight to each one. The mistakes of decision trees that are created earlier are corrected with later models. The group decided to use gradient boosting because it is a simple way to reduce bias of decision trees and achieve better results without altering the model too much. The downside is that it makes the decision tree less interpretable.

Similarly, random forest works by considering only a subset of variables from a decision tree for each iteration. The samples used to build each tree are bootstrapped. This leads to a decision tree that has reduced variance. This is another way to improve decision trees without much change to the original model, however, it also makes the model less interpretable, just as gradient boosting does.

Finally, linear regression is a model that assumes the label can be predicted as a linear combination of the attributes. The group decided to use this model because it generally works well for problems where the data is not assumed to have complex relationships (i.e. the data is likely able to be modeled by a linear combination of attributes). It is also a simple and interpretable model, just like the other models that were discussed.

All descriptions for the models in this section were inspired by lectures from CS 334: Machine Learning at Emory University [10].

Experiments/Results

As mentioned in the introduction section, data for this project was obtained from Baseball Savant [5]. Data from all hitters from 2015 up through 2023 were used to have the largest sample of players possible. The minimum number of plate appearances needed to be included in the data was 100 in a given season. Players with less than 100 plate appearances in a given season were not included. The data contains 3920 samples of players and 70 features. The features contain both traditional statistics such as batting average along with Statcast statistics such as launch angle and exit velocity. Data of player name, player ID (a unique integer identifier for each player), that player's wOBA, and the year of each data point was recorded in a file containing the labels. During preprocessing, the labels were replaced with the change in wOBA for that player compared to the previous year. For instance, if a player's wOBA increased from .300 to .330 from 2015 to 2016, their label would be 1.1 because their wOBA increased by 10 percent.

After the data was gathered, data preprocessing was performed. Preprocessing began with finding the first year that every player appeared in the data. This was needed because the label that is to be predicted is the change in wOBA during a period of at least two years. Therefore, the first year a player appears in the dataset needed to be omitted. After the appropriate rows were removed, a scikit-learn standard scaler was applied to the data. "The standard score of a sample x is calculated as: $z = \frac{x-u}{s}$ where u is the mean of the training samples … and s is the standard deviation of the training samples" [8]. The purpose of this scaler is to "standardize features by removing the mean and scaling to unit variance" [8]. It was also necessary to perform standard scaling in order to perform Pearson correlation. This was because some of the attributes represented rate statistics, while other attributes represented counting stats. Standardizing the data allowed every attribute to be evaluated on the same scale. Pearson correlation was performed with the pandas.DataFrame.corr method [9]. Attributes with a Pearson correlation coefficient less than 0.25 with respect to the label were removed, as these were deemed not very significant in predicting the change in wOBA. Attributes that had a Pearson correlation coefficient greater than 0.95 with respect to another attribute were also removed, as these attributes were deemed too similar in the information they represented. The attributes that were selected from this process were xba, xslg, xobp, xiso, wobacon, xwobacon, bacon, xbacon, xbadiff, xslgdiff, batting_avg, slg_percent, on_base_percent, and babip. Please see reference [5] for explanations of each attribute.

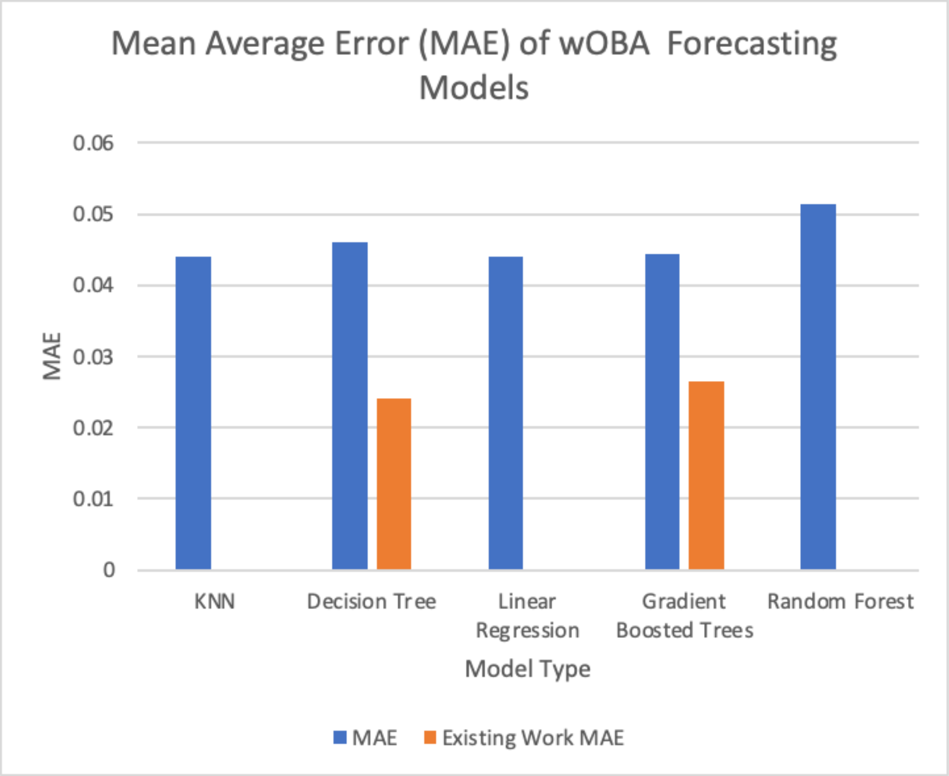
After data preprocessing, a train-test split was done. The ratio for this split was 78-22, with 22 being the holdout. The significance of the number 22 is that 22 percent of the data constitutes just the 2023 data. The data from 2015 through 2022 was used as the training data, and the 2023 data was the test data. This means that our models used the 2015 through 2022 seasons to predict the 2023 season. It is the hope that this represents a similar situation to using the 2015 through 2023 data to predict the upcoming 2024 season.

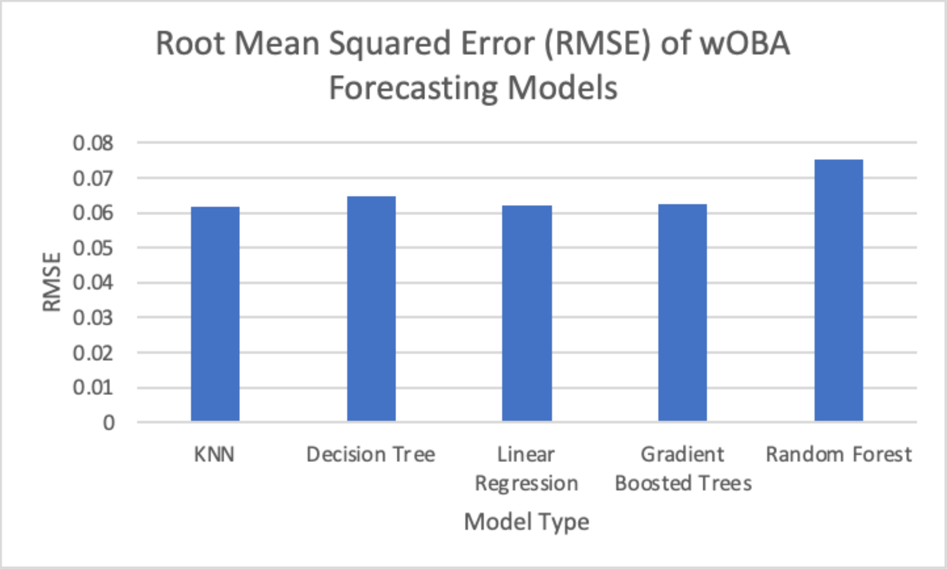
As discussed in the methods section, the models chosen were k-nearest neighbors, decision tree, gradient boosting, random forest, and linear regression. For explanations of why certain models were chosen, please refer to the methods section.

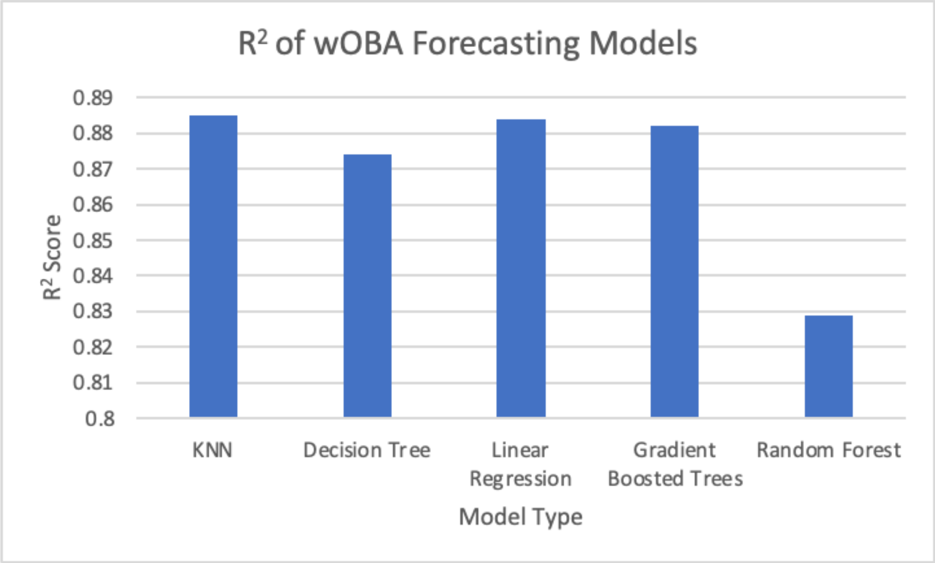
The metrics used to evaluate the models were mean absolute error (MAE), root mean square error (RMSE), and R-squared. Optimal parameters for each model were chosen through grid search.

For k-nearest neighbors, the optimal amount of neighbors was found to be 47, which produced an MAE of about 0.0440, an RMSE of about 0.0618, and an R-squared value of about 0.8849. Decision tree was found to produce the best results with a maximum depth of 5 and 50 minimum leaf samples, which produced an MAE of about 0.0463, an RMSE of about 0.0650, and an R-squared value of about 0.8725. Gradient boosting produced the best results when alpha was 0.1 and the learning rate was 0.1 as well. It produced an MAE of about 0.0443, an RMSE of about 0.0624, and an R-squared value of about 0.8825. Random forest produced the best results when max depth was 5, max features was 0.75, and minimum leaf samples was 5. It produced an MAE of about 0.0510, an RMSE of about 0.0755, and an R-squared score of about 0.8281. Finally, linear regression produced an MAE of about 0.0437, an RMSE of about 0.0620, and an R-squared score of about 0.8839.

Previous work on a similar (but not identical) dataset produced an MAE of about 0.0240 for both decision tree and gradient boosted trees. Below are graphs summarizing the results for each model.


Mean Average Error (MAE) of wOBA Forecasting Models
0.06
0.05
0.04
0.03
0.02
0.01
0
MAE
KNN
Decision Tree
Linear Regression
Gradient Boosted Trees
Random Forest
Model Type
MAE
Existing Work MAE


Root Mean Squared Error (RMSE) of wOBA Forecasting Models
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
RMSE
KNN
Decision Tree
Linear Regression
Gradient Boosted Trees
Random Forest
Model Type



Discussion

One of the main takeaways from the results gathered from this experiment was the surprising lack of variance in the error metrics between each type of classifier. Although k-nearest neighbors and linear regression tended to perform the best according to MAE, RMSE, and R-squared, the differences in error values between each classifier were minimal. An explanation for this could be that the data has quite simple patterns in how certain attributes affect wOBA, but there is a level of irreducible error inherent in the dataset. This would explain why simpler classifiers performed at the level of (or in this case, exceeded) the models that were more complex.

An interesting observation that was gleaned from the feature selection process was that the most important Statcast statistics for predicting a player's wOBA were their other expected stats, such as xBA (expected batting average) and xISO (expected isolated power). This implies that the most important metrics in predicting a player's wOBA lie in other expected statistics. Perhaps a weighted average of these expected statistics could be used to more accurately predict wOBA.

One aspect of the dataset that may have led to this irreducible error is the fairly low requirement of 100 plate appearances to be eligible for the dataset. In Major League Baseball, 100 plate appearances is considered quite a low number, as there are 162 games in a season and a player needs 502 plate appearances to qualify for certain awards such as a batting title [11]. Players who do not meet this requirement are acknowledged to have too small of a sample size to be considered for awards. This low plate-appearance threshold was used in order to obtain as many samples as possible, as the number of hitters between 2015 and 2023 with over 502 plate appearances was too low to derive meaningful insights.

The MAE value of about 0.0240 achieved by previous work suggests that the number of plate appearances that was used for this experiment was low enough to increase variance in a meaningful way. When players with a low number of plate appearances are included, the number of outliers increases, as this may include players who only play a partial season due to injury, being sent down to the minor leagues, or other conflicting factors.

The good news is that the size of the Statcast dataset increases every season, as more Statcast data is recorded after each game. This means that future studies should be able to produce better results, as larger datasets will be able to be used without lowering the number of minimum plate appearances.

Contributions

Clay and Tommy:

- Acquiring Dataset
- Data Preprocessing
- Training and Testing KNN and Gradient Boosted Tree Regressors Models
- Writing methods section
- Writing discussion section

Clay:

- Training Decision Tree Regressor, Random Forest Regressor, and Linear Regression Models
- Creating Graphs
- Writing background section

Tommy:

- Writing Introduction
- Writing Abstract
- Writing Methods
- Writing Experiments/Results
- Writing Discussion

Code and Dataset

Google Drive Link

https://drive.google.com/drive/folders/1kFV1OblHmCLxtRJZgnEyc47aOpXwd48e?usp=sharing

References

[1] P. Slowinski, “Counting vs. rate statistics,” Sabermetrics Library, https://library.fangraphs.com/principles/counting-v-rate/ (accessed Dec. 13, 2023).

[2] P. Slowinski, “Woba,” Sabermetrics Library, https://library.fangraphs.com/offense/woba/ (accessed Dec. 13, 2023).

[3] “Statcast,” Wikipedia, https://en.wikipedia.org/wiki/Statcast (accessed Dec. 13, 2023).

[4] “Expected weighted on-base average (xwOBA): Glossary,” MLB.com, https://www.mlb.com/glossary/statcast/expected-woba (accessed Dec. 13, 2023).

[5] “Baseball Savant: Statcast, trending MLB players and visualizations,” baseballsavant.com, https://baseballsavant.mlb.com/ (accessed Dec. 13, 2023).

[6]T. Ishii, “Using Machine Learning Algorithms to Identify Undervalued Baseball Players ,” cs229.stanford.edu, Autumn 2016. 2016. https://cs229.stanford.edu/projects2016.html (accessed Dec. 13, 2023).

[7] C. Watkins, “Novel Statistical and Machine Learning Methods for the Forecasting and Analysis of Major League Baseball Player Performance,” digitalcommons.chapman.edu. Spring 2020. https://digitalcommons.chapman.edu/cads_dissertations/10/ (accessed Dec. 13, 2023).

[8] “Sklearn.preprocessing.StandardScaler,” scikit, https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html (accessed Dec. 13, 2023).

[9] “Pandas.dataframe.corr,” pandas.DataFrame.corr - pandas 2.1.4 documentation, https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.corr.html (accessed Dec. 13, 2023).

[10] L. Xiong, “CS 334: Machine Learning,” 2023

[11] “Qualifier,” Qualifier - BR Bullpen, https://www.baseball-reference.com/bullpen/Qualifier (accessed Dec. 13, 2023).