Predicting Major League Baseball Batting Averages Using Machine Learning and Statcast Data

Joey Capps
Washington State University
Pullman, WA 99163

Abstract

Predicting MLB stats is a hard problem that fans, scouts, and fantasy sports players take great interest in. Most of the highest performing models are hidden behind paywalls, and their implementation details are often proprietary. Batting average has historically been a stat that is extremely hard to predict due to the amount of luck involved. However, with the introduction of advanced Statcast data, significant improvements can be made. In this paper we aim to dissect what exactly goes into creating these models, and we also build our own that has shown to rival and even beat some of the top models out there today. Using Statcast data is not exclusive to batting average, and this formula can be applied to many other stats in future work.

1 Introduction

Batting average in baseball is derived by simply dividing the number of hits H a player gets by how many at bats AB they have:

$$BA = \frac{H}{AB}$$

As the Major League Baseball's glossary explains, an at bat is when a batter reaches base via a fielder's choice, hit, an error, or when a batter is put out on a non-sacrifice [12]. So already things are not very straight-forward with the batting average stat. If a player gets hit by a pitch, walks, or does a sacrifice hit to advance his teammate(s), that doesn't count as an at bat. This is in contrast to the much simpler plate appearences (PA) stat, which is just incremented each time the batter steps up to the plate. If a player performs strongly one season, the next season could lead to that player receiving many intentional walks, or coaches choosing to only pitch to him when they have a very strong pitcher. Also, the line between getting a hit and getting out is very thin, so luck plays a large role in both batting average and other stats. As explained in [1], the well known and proprietry system PECOTA (Player Empirical Comparison and Optimization Test Algorithm) has been known to produce batting average predictions with mean absolute errors close to 20 points. 20 points here will be defined as 0.02, so if a player is predicted to have a batting average of 0.250 in a season, a common situation would be for the PECOTA algorithm to pick 0.270 or 0.230. Some seasons are more predictable than others however, for example our models performed much better for the recent 2025 season than the 2024 one.

Stat prediction for baseball falls under the umbrella of Sabermetrics, which is defined as the statistical analysis of baseball data [13]. Stat predictions for the next season are of intereset to some fans, as they would like to see what their favorite players are statistically likely to do, and what they can expect. They are also of interest to general managers, as they need to decide how much to pay their players. More recently, with the rise in popularity of fantasy sports, people want to know which players they

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

should pick to maximize their scores. Sites like DraftKings also offer fantasy sports where users can win real money, so getting an estimate on a player's stats can be crucial. Since the supreme court struck down the federal sports betting ban in 2018, more and more states are legalizing sports betting [11]. In fact, 30% of sports bettors claim to have gambling-related debts. People can make all kinds of bets now on different stat outcomes in a variety of sports, including MLB batting averages. Due to the perceived value of predictions, it is rare to find an MLB stats prediction system that doesn't have proprietary implementation details, and the raw data is often hidden behind paywalls. However there are clues that be used to hunt down the valuable features that can be fed into a machine learning regression model, which is what we aim to capitalize on in this paper. First, we must clearly define the problem that we are addressing and exploring.

2 Problem Overview

The problem is defined as predicting batting averages for players over the span of one MLB season. There are a breadth of factors that play into this, such as which stats to pay attention to and how many seasons to look at before the season that is to be predicted. Since the MLB has been going since 1876 and each team plays 162 games per season, there is a high volume of statistical data that can be used as features in a predictive model. Many stats have been tracked since that first 1876 season, such as age, home runs, and of course batting average. Since time moves in one direction, training sets must be represented as older data than the test sets to avoid data leakage. An easy way to do this is to make a cut at a certain year, and say that data before that is for training, and all data after the cut is for testing. While some stats have been tracked since the inception of Major League Baseball, many other types of stats have been developed in more recent years that have been shown to hold significant predictive power. Standard stats can be defined as something that is trivially computed from the stat counts in a given game. Examples of standard stats include things such as batting average, hits, and the number of times a player gets hit by a pitch. Standard stats can be found on the official MLB website, or on sites such as Baseball Reference. Advanced stats are defined as being more complex, and they may be presented as a combination of standard stats, or something that is harder to measure than standard stats [7]. One example of an advanced stat that we used as a predictive feature in our models is BABIP, or batting average on balls in play. BABIP is defined by the following formula:

$$BABIP = \frac{H - HR}{AB - K - HR + SF}$$

The formula starts as the batting average formula from earlier, but also includes the standard stats HR (home runs), K (strikeouts), and SF (sacrifice flies). According to [5], BABIP can be described as providing more context than standard batting average, as it removes outcomes that aren't affected by the opposing defensive fielders (home runs and strikeouts). Since sacrifice flies are a standard stat that only started being counted in 1955, BABIP is an example of a stat that we can only obtain for more recent seasons. The third and newest main category of MLB stats is Statcast data. According to the MLB glossary, Statcast data is defined as "state-of-the-art tracking technology that allows for the collection and analysis of a massive amount of baseball data, in ways that were never possible in the past" [3]. This data is collected via tracking systems installed in every MLB stadium starting in 2015, so this is by far the newest of the three stat categories. Statcast stats include metrics such as player sprint speed, expected batting average, and even the spin rate of pitches. Statcast data is easily obtainable at the MLB-owned Baseball Savant website, and it provides public APIs to grab raw CSV data. Statcast stats can be as granular as per-pitch data, or a custom date range can be set. Machine learning models can use standard, advanced, and Statcast data as features in models to predict a variety of future stats, and we will be focusing on batting average for this paper.

3 Related Work

Most of the related work for predicting MLB stats comes in the form of proprietary models. However, we can gain a lot of insight into how to predict batting averages from what information is known about these models, and strategies people have attempted in the past. These models come from people within the Sabermetrics community, and they usually predict a wide variety of stats that aren't just batting average. In terms of just predicting MLB batting averages, there are only a couple examples of research on this.

3.1 Just Predicting Batting Average

There are two detailed examples of methods people have tried for predicting batting averages that come in a somewhat academic form. One comparable school project was for a machine learning course at Northwestern University from 2016, where students aimed to predict season-long batting averages using only stats from the previous year [2]. They utilized stats from 1990-2014, although it is unclear which season(s) were used for testing. In terms of testing, they were able to obtain a mean absolute error (MAE) as low as 0.056 (56 points) after testing a variety of models. They used exclusively standard stats as features in their models, such as batting average, year, salary, and strike-outs. More recently in 2020, researchers from Canadian universities looked into augmenting the PECOTA predictions using Statcast data [1]. They used Statcast data and PECOTA predictions from 2015-2016 to build a model to predict 2017 batting averages. Using the combined predictor of 2017 PECOTA predictions + their trained model, they were able to lower the MAE of PECOTA predictions from 0.0209 to 0.0208, a 0.1 point improvement. Some examples of Statcast data they used included exit velocity, launch angle, and distance the ball was hit. They then plugged these into a logistic regression model to estimate the probability of a hit, which they used as a feature. Another significant point is that they limited training data to only include batters who had at least 200 at-bats in the 2015 season. This excludes some players who might have gotten lucky / unlucky and skewed the training process in the wrong way. While 0.1 point isn't a huge improvement, it shows the potential for future work to augment model accuracy using Statcast features.

3.2 Steamer

According to [4], Steamer is a system that was developed by a high school science teacher and two of his former students. Also, it is "widely regarded as one of the most accurate predictors in the industry". The model uses past performance, aging trends, and pitch-tracking data to make predictions for a variety of stats, including batting average. The FanGraphs website (`https://www.fangraphs.com/projections`) shows Steamer predictions for the next season, for example right now it has 2026 stats predictions. The exact details of the algorithm aren't available to the public. Also, exporting raw data or viewing predictions from previous years costs $15 a month. In [8], they compared the accuracy results for different models in the year 2024. Out of 14 models, it came in 6th in terms of hitting accuracy in general. In terms of batting average specifically it was slightly weaker, coming in 8th place. This makes Steamer a good model to test results against to see whether or not a competing model is "above average".

3.3 The Bat X

According to [9], The Bat X is an innovative model that incorporates some Statcast data in its predictions. The FanGraphs website also hosts The Bax X. When developing the model, over 150 Statcast variables were considered. Besides Statcast data, also uses weighing multiple years of data, age curves, etc. The Bat X was created by Derek Carty, who also created The Bat and The Batcast. The Bat doesn't use Statcast data (most likely standard and advanced stats), while the Batcast uses exclusively Statcast data. The Bat X is "a combination of The Bat and The Batcast" that was shown via back-tests to be the best of the three systems. This is supported by the 2024 rankings in [8], where The Bat X was shown to have the highest overall accuracy for hitting out of 14 models. It also came in first place by far for batting average specifically. This proves the intuition of [1] that incorporating Statcast features can lead to a significant accuracy boost for models. This makes The Bax X a good model to test results against to see whether or not a competing model is "excellent".

4 Methods

4.1 Data Collection

To build a strong model to compete with other top models, a wide variety of data would have to be collected. As mentioned earlier, there are three different main categories of MLB stats. These include standard, advanced, and Statcast data types. Since The Bat X model saw impressive results from using all three types, we will have to do the same in order to compete on batting average accuracy. For standard stats, these were collected from the website Baseball Reference from the years 1955 to 2025, but we would later drop pre-2015 stats in order to join with

Stat Name	Stat
player	Shohei Ohtani*
key_bbref	ohtansh01
Age	30
WAR	6.6
G	158
PA	727
AB	611
H	172
2B	25
3B	9
HR	55
RBI	102
BB	109
SO	187
BA	0.282
OBP	0.392
SLG	0.622
OPS	1.014
OPS+	179.0
rOBA	0.423
Rbat+	175.0
HBP	3
SH	0
SF	2
IBB	20
year	2025

Figure 1: Shohei Ohtani's Standard Stats From 2025

Statcast data, which started in 2015. An example year of 2025 can be seen viewed at (`https://www.baseball-reference.com/leagues/majors/2025-standard-batting.shtml`). For each unique [year, playerID] combination we collected 24 predictive stats / features. Some of these stats like BA (batting average) are averages, and some are totals, such as HBP (count of times hit by pitch). Example stats for the [year, playerID] combo of 2025 Shohei Ohtani can be seen in figure 1. A wide variety of standard stats were collected, as we could use feature engineering and other techniques to determine which features yield the highest predictive power for batting average. For Baseball Reference stats, each player comes equipped with a unique player id, which is called "key_bbref". This is helpful because it guarantees uniqueness of players, even if they have the same first and last name as another player. It should be noted that the "Player" column in figure 1 was only added for visualization, and wasn't used for player identification. In terms of advanced stats, we later use these standard stats to compute a few different advanced stats, such as BABIP which was mentioned earlier.

Next, Statcast data would have to be obtained. Our strategy for this would be similar to the standard stats one, and that is to collect a wide variety of features. All Statcast data was collected from Baseball Savant. Statcast data is split into different categories, one of which is Expected Statistics, which can be found for 2025 stats at (`https://baseballsavant.mlb.com/leaderboard/expected_statistics?type=batter&year=2025`). We collected a vast number of Statcast stats for each [year, playerID] combo from 2015 - 2025. The categories used were Expected Stats, Batted Ball, Sprint Speed, Exit Velocity & Barrels, and Batting Run Value. In total, 87 predictive Statcast features were collected. The method used was to collect data from each of the five categories and then perform an inner join on them for matching [year, playerID]. The Statcast player id is different than the Baseball Reference one, and it is called "key_mlbam". Statcast also gave access to player positions and team names, which were added using one-hot encoding so that ML models could effectively train on them. These didn't end up being very predictive in the end, but it was worth it to try a variety of features to see which ones might have surprising correlative results.

One essential thing that needed to be done was to see what opposing models had predicted in the past, so we could compare batting average predictions. Unfortunately, most of the top performing models mentioned in [8] require money to obtain past predictions in raw form, and the budget for this research paper was $0. Luckily we can use the Wayback Machine (`https://web.archive.org/`) to see what past FanGraphs models predicted. The default FanGraphs predictions url pulls up Steamer predictions, so the batting predictions from this model were collected from right before the 2025 season. Assuming Steamer performed similarly in 2025 as in 2024, we can tentatively declare our model as "above average" if it can beat these. The real test would be going up against The Bat X, which was the top performer by far in 2024 for both hitting in general and batting average out of 14 assessed models. Luckily there was one timestamp for these predictions saved on the Wayback Machine from right before the 2024 season started, so we were able to collect these batting average predictions. Now that we had a variety of predictors along with predictions from competing models, we would be able to both train models and compare results to others.

4.2 Feature Engineering

After collecting standard / Statcast stats it was time to prepare our data for some machine learning models. The strategy we used was to simply combine our standard and Statcast data into single rows where [year, playerID] matched on an inner join. That way, each row would give us access to a large amount of both standard and advanced data that we could train on per player per year. However, in terms of player ids the standard and Statcast stats used a different format. For standard stats from Baseball Reference, they used "key_bbref", while Statcast uses "key_mlbam". While these are nice unique identifiers, their formats look nothing alike, so we can't just inner join on those keys. Luckily there is something called the "Chadwick Basweball Bureau Persons Register" available at (`https://github.com/chadwickbureau/register`), and we will refer to this as the "Register". This contains csv files where each row is a player, and it contains the associated "key_bbref", "key_mlbam", along with "key_fangraphs". For The Bat X and Steamer the competitors we are back-testing against, they are both hosted on FanGraphs, which of course uses its own custom player identifier as well. The Register fixed the player ID alignment issue, and we were able to have each row hold both standard stats and Statcast stats for each [year, playerID] combo. This was done by a series of inner joins that used the Register. After this we had a total of 2454 rows of stats that provided batting data for players ranging from 2015 to 2025.

While we had both standard and Statcast data, the advanced stats were still missing. We calculated a few of these using formulas involving standard stats. Some of these included BABIP, K% (strikeout rate), and ISO (isolated power). ISO is mathematically defined as:

$$ISO = \frac{1 \times 2B + 2 \times 3B + 3 \times HR}{AB}$$

Where 2B are doubles, and 3B are triples. According to the MLB, ISO measures the raw power of a hitter, and it can serve as a metric for how much raw power a player has [6]. Combination of standard stats like this have the potential to have more statistical significance than their parts, so we wanted to add in a few advanced stats to see how true that was in the case of batting average. Another stat we added to each row was the team average of a player's team per season, and we decided to just focus on batting average here. Adding team average batting average might help in that the model can see whether or not a player is outperforming his team, and it can learn to better predict future batting averages based on this.

Next, we engineered some features that represented the cumulative average of a players' career leading up to the year that each row represented. The features we decided to take cumulative means for included BA (batting average), xBA (expected batting average), and BBE (batted ball events). We chose these because of their high F-score in relation to batting average. An example for batting average cumulative means would be the following. If we are trying to predict a player's batting average for 2025, their row would hold a "CareerAvg_BA" feature. This represents their cumulative average batting average from the years 2015-2024, because adding 2025 to that would mean data leakage and cheating. Cumulative averages can be powerful for batting average prediction because players often times have outlier seasons where their batting average is way above or below what they are normally capable of. The cumulative average naturally regresses the prediction to the mean, which is proven later to be a very significant predictor.

Besides cumulative means, we also introduced lagged features. Lagged features are like cumulative means but without the means part. For example a "BA_lag1" feature for a player in 2025 represents the batting average they recorded in 2024, and "BA_lag2" would be their batting average for 2023. Also if a player played in 2023 but not 2024, the "BA_lag1" would just use their stats from 2023 or whenever they most recently played. We tested a variety of year counts to lag back to, and it seemed that using the past 3 years of data was most effective for batting average prediction. It should be noted that cumulative averages considered the player's entire career here, not just the last 3 years. In terms of which columns we lagged, we chose all of the predictive columns except position one hot vectors, team one hot vectors, and career averages. For a player's rookie year they had no previous years values to get the cumulative average or lagged features from, so we decided to not included rookies in our data set. Imputation could have been done to predict on rookies, but we decided to not include them since we did not include minor league data in training. Rookies often have minor league experience before going to the MLB, and minor league stats are available that some other models train on. Since we had not collected minor league stats, we decided our model would be at a disadvantage if we included them. Also, Statcast data isn't available for minor league players and we wanted to focus on the power of Statcast predictors combined with non-Statcast predictors for this paper.

The features we included for inputs / 'X' values included cumulative career averages and lagged features. Age-related features were also included because those don't result in data leakage, since someone's age for any season can be determined using their previous age. The output feature / 'y' value was obviously just the batting average for a player in a given year. After dropping rookies and removing NA values, our total rows went from 2454 to 1845 from the span of 2015-2025. The final column count for selected columns was 236, which is arguably too high and may result in curse of dimensionality issues.

4.3 Exploratory Data Analysis

Now that we prepared our data, we decided to look at some of the features to see which ones were strongest by themselves at predicting batting average. The top 25 features with highest F-score are schematically represented in figure 2. There are a variety of features represented, but expected batting average and batting average are the most common at the top (expected batting average is called est_ba). The highest score is actually cumulative expected batting average instead of normal batting average. This is a significant result, because it shows that expected batting average may actually be a better predictor of future batting average compared to the actual recorded batting average, but what exactly is expected batting average? It is defined in [10] as a Statcast metric that measures the likelihood of a batted ball becoming a hit. This value is computed by looking at how often comparably hit balls result in a runner getting on base. Expected batting average is such a powerful predictor because it strips away the luck factor, and just looks at how high quality the contact on the ball is. The top two F-scores are cumulative expected batting average, followed by the lag1 expected average, then the lag1 batting average. This shows that looking at career averages might be better than just looking at the previous season, because it regresses to the mean and controls for outlier seasons. Looking at the F-scores gives us intuition into how significant the use of Statcast data can be for future stat predictions.

4.4 Models

The models we used to predict batting average included linear regression, lasso regression, XGBoost, and a neural network. Since our feature count was so massive at 236, we decided to use PCA to cut down on this and reduce noise / curse of dimensionality. We decided to have PCA pick a component count that explains around 90% of variance, as this is empirically what produced the best results. We found that the linear models consistently yielded the lowest mean absolute error (MAE) compared to XGBoost and neural network. This could be because our best features such as previous BA or previous xBA may be highly linear to future BA.

Feature Name	F-Score
CareerAvg_est_ba	292.32
CareerAvg_BA	286.4
est_ba_lag1	243.1
BA_lag1	215.63
est_ba_lag2	191.31
est_ba_lag3	181.09
BA_lag2	162.9
BA_lag3	160.77
bbe_lag2	134.56
bbe_lag1	134.56
CareerAvg_bbe	134.56
bbe_lag3	134.56
H_lag1	134.43
K%_lag1	133.63
competitive_runs_lag1	133.32
2B_lag1	118.91
K%_lag2	113.72
ev95plus_lag1	108.25
rOBA_lag1	106.28
K%_lag3	103.8
OBP_lag1	102.97
attempts_lag1	101.92
bip_lag1	101.92
WAR_lag1	101.73
woba_lag1	100.28

Figure 2: Top 25 F-scores of Selected Features

5 Results

5.1 2023 - 2025 Test Set

To gauge the general performance of the models, we had a training set of 2015 - 2022, and a test set from 2023 - 2025. These results are represented in figure 3, and it included a test set of 971 examples. The best model was just basic linear regression, with a MAE of 0.0208, or around 20.8 points. This is a pretty good result and almost exactly the same as the 2017 PECOTA MAE mentioned earlier, but next we will compare it with competing models using the exact same set of players on each.

5.2 2025 Steamer

At the time of writing this paper the most recent MLB season was the 2025 one. Since we were able to collect the Steamer predictions from FanGraphs on the Wayback Machine right before the 2025 season started, we can compare batting average predictions. To do this we edit our training set to be 2015 - 2024, and our test set to be just the 2025 season. For this one we had a test set of 360 batters from the 2025 season, and we compared our predictions to those of Steamer on the same 360 batters. The results are described in figure 4. Our best result was again linear regression with a 0.0201 MAE this time. Steamer's MAE was higher at 0.02194, and all four of our models outperformed it. Our best model provided an MAE improvement of around 1.8 points over Steamer. This shows our model has potential to compete with some averagely performing pay-walled models, but how will it do against the best?

5.3 2024 The Bat X

Since the article in [8] showed The Bat X as performing by far the best in 2024 out of 14 models, we felt the need to compare results. To do this, we re-tuned our training set to be 2015 - 2023, and our test set to just be 2024 (adding 2025 would be time traveling so it's considered cheating / data leakage). This time the test set was 336 batters, and again we compared our MAE to that of The Bat X for the exact same set of 336 batters. As described in figure 5, two of our models managed to beat The Bat X, with linear regression performing the best at 0.0214 MAE. The Bat X was roughly 1 point off at 0.02235 MAE. This is a significant results as it shows that the process described in this paper has the potential to outperform the best models that currently exist in Sabermetrics.

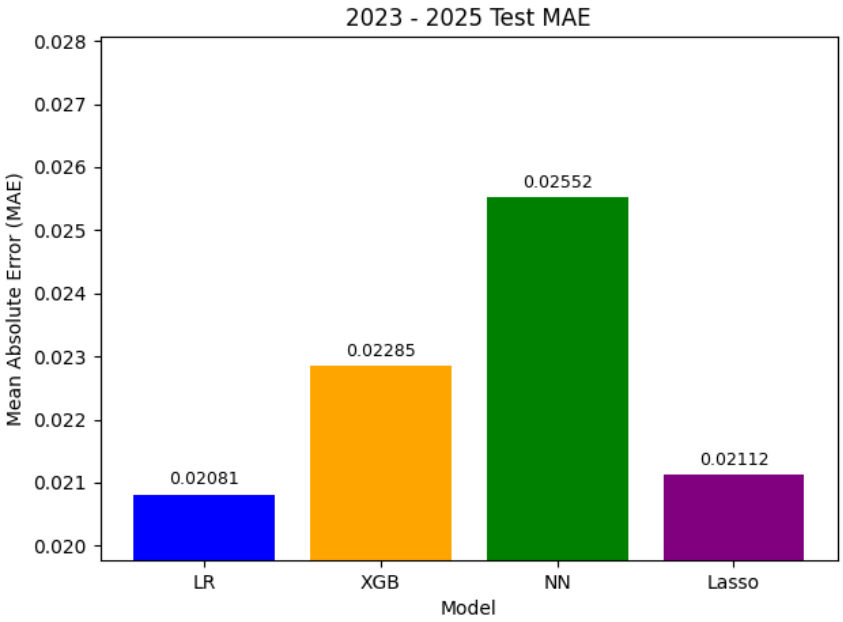

Figure 3: 2023 - 2025 Test MAE

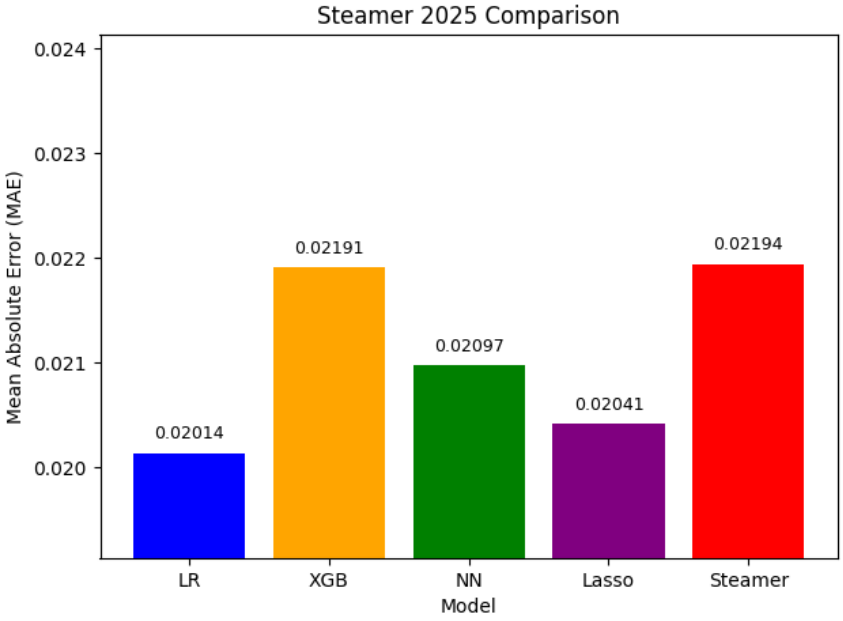

Figure 4: Steamer 2025 Comparison

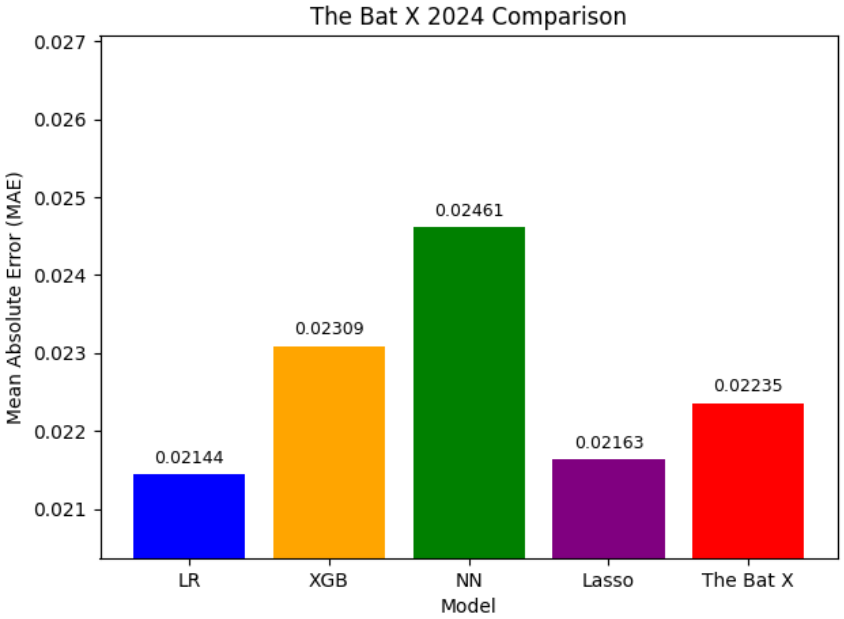

Figure 5: The Bat X 2024 Comparison

6 Conclusion, Limitations, and Future Work

We have shown in this paper that a somewhat simple process has the potential to outperform the best models out there for predicting MLB batting averages. The Bat X was shown to outperform 13 other popular models in 2024 batting average predictions, and ours managed to outperform it. Our process includes using standard, advanced, and Statcast statistics as features. Then, take the cumulative averages for players' careers, along with lagged stats to see what they did in recent previous years. Finally, use PCA to cut down on the large amount of features and capture the bulk of the variance. There are some limitations with this paper though, such as the fact that rookies weren't included. While 300+ players per season of test data is a pretty good representation of the league, all batters must be included to build a more comprehensive model. The future work that can be done for this is to expand this methodology to other stats to see if a very accurate model can be built that predicts a wider range of stats besides just batting average. Since the xBA was such a significant predictor, other stats under the Expected Statistics Statcast category can be considered as features. Also, a more accurate model could possibly be created by creating a model that trains on older data besides just 2015. Statcast data can be imputed years older than 2015, or a model can be created that combines a Statcast-based model with the baseline model. Another strategy for future work that the Statcast website supports is to look at data more granularly than just per-season. The seasons could be split into sections, and machine learning models could learn more detailed patterns in the data. Also to get a better metric of how good our model is, we would need a lot of data from multiple seasons from the best models, besides just a couple individual seasons. For example we were unable to obtain raw PECOTA data because it is paywalled and not available on the Wayback Machine. Future research projects with higher funding can look at a wider range of models and years to get a better measure of how strong a candidate model is.

References

[1] Bailey, S. R., Loeppky, J., & Swartz, T. B. (2020). The Prediction of Batting Averages in Major League Baseball. Stats, 3(2), 84–93. https://doi.org/10.3390/stats3020008

[2] Todes, M. (2016). Baseball Predictions. Github.io. https://mikhailtodes.github.io/Machine_Learning_Baseball/

[3] Major League Baseball. Statcast | Glossary. MLB.com. https://www.mlb.com/glossary/statcast

[4] Steamer | Glossary | MLB.com. (2025). MLB.com. https://www.mlb.com/glossary/projection-systems/steamer

[5] Batting Average on Balls in Play (BABIP) | Glossary. MLB.com. https://www.mlb.com/glossary/advanced-stats/babip

[6] Isolated Power (ISO) | Glossary. MLB.com. https://www.mlb.com/glossary/advanced-stats/isolated-power

[7] Advanced Stats | Glossary. MLB.com. https://www.mlb.com/glossary/advanced-stats

[8] Herlin, J. (2025, February 4). Most Accurate Fantasy Baseball Projections (2024 Results). FantasyPros. https://www.fantasypros.com/2025/02/most-accurate-fantasy-baseball-projections-2024-results/

[9] Carty, D. (2020, June 11). Introducing THE BAT X. RotoGraphs Fantasy Baseball. https://fantasy.fangraphs.com/introducing-the-bat-x/

[10] Expected Batting Average (xBA) | Glossary. MLB.com. https://www.mlb.com/glossary/statcast/expected-batting-average

[11] Garrison, G. (2025). 2025 Sports Betting Survey: 1 in 4 Sports Bettors Have Missed Bill Payments Due to Wagers. US News & World Report; U.S. News & World Report. https://www.usnews.com/banking/articles/2025-sports-betting-and-debt-survey

[12] At Bat | Glossary. MLB.com. https://www.mlb.com/glossary/standard-stats/at-bat

[13] Neyer, R. Sabermetrics | statistics. Encyclopedia Britannica. https://www.britannica.com/sports/sabermetrics