Ketel Marte xwOBA Analysis

By Taylor R. Pubins, Prepared Especially for DBACKS BBOPS

I. Introduction & Background

"There are rich teams and there are poor teams. Then, there's 50 feet of crap. And then there's us." – Billy Beane

The Arizona Diamondbacks are slightly more valuable than the Oakland Athletics but are indeed the least valuable franchise in the NL West (per Statista market research [1]). Even with a decrease of intra-division games (from 76 down to 52) for the 2023 season, a third of the games are still played against division rivals. Constructing and optimizing a depth chart around frequently encountered opponents using machine learning (ML), has at a minimum, a chance to increase division series wins, and a best-case scenario of improving playoff-chase leverage.

Sabermetrics were popularized in the 1980s by Bill James's iconic work known as his Pythagorean Expected Win Ratio and then made mainstream by the book & movie *Moneyball*. Sabermetrics can be described as statistical analysis beyond the rote descriptive statistics [2] and combines simple observations to generate more meaningful insight. Applying ML to Sabermetrics has the opportunity to identify and exploit extremely powerful patterns.

Tom Tango & co.'s book, *The Book*, is a comprehensive review of Sabermetrics that dives into the field of predictive analysis. He claims attempting to estimate a player's *true* talent is akin to making a prediction [3]. The primary methodology this project will focus on is regression modeling. In the context of this project, pitchers will be grouped together into architypes based on various Statcast metrics, similar to *The Book's*, *la famiglias*—Italian for families. Certain players are grouped together based on observed and statistical characteristics and then performance can be evaluated against a family which reduces variation implicit to small sample sizes.

The specific hypothesis this paper will explore is the hypothesis that not all hitters perform the same against all *types* of pitchers (e.g. lefty, soft-contact, ground-ball, good-command). Tom Tango refutes the ability to draw this type of conclusive analysis and implies regression towards the mean is as sure as death and taxes. This means that if a batter *dominates against* a family of pitchers for as many as 60 plate appearances (PAs), the likelihood that his next 60 PAs are closer to his average, is far more likely than his previous PAs against that family. His conclusions are based on observations of past occurrences but does not adjust for year over year increases/decreases in ability and skill. This is an analysis problem most suited for ML.

This project attempts to answer the questions:

- Using Sabermetrics and MLB Statcast data, how well can classical ML and neural networks (NN), predict a player's divisional statistics; specifically expected *weighted on-base average* (xwOBA)? This stat is explained further in the *Data Understanding* section.
- Are there other players around the league that could thrive in our test environment? [for a future research effort].

This methodology for this project will utilize a mélange of Statcast categorical and numerical data that will group opposing pitchers into similar abilities (pitch type speed & movement) and outcomes. A naïve model (past four season's performance) will provide a baseline, then classical regression models will attempt to show improvement, then NN models will attempt to improve predictive ability beyond the classical models without overfitting the data. While not in the scope of this project, a recurrent NN (RNN) has the potential to exploit the linear/time-series nature of a baseball season to further predict performance within hot and cold streaks. According to the great Yogi Berra, "Baseball is 90% mental; the other half is physical". While there is no Sabermetric that can quantify a player's mental state, a RNN can potentially identify signs of a streak.

This project will succeed if one or more ML models can predict player performance better than a naïve model. Success will also be counted if the ideas contained in this project spurn enhancements and creativity to the discipline sports ML.

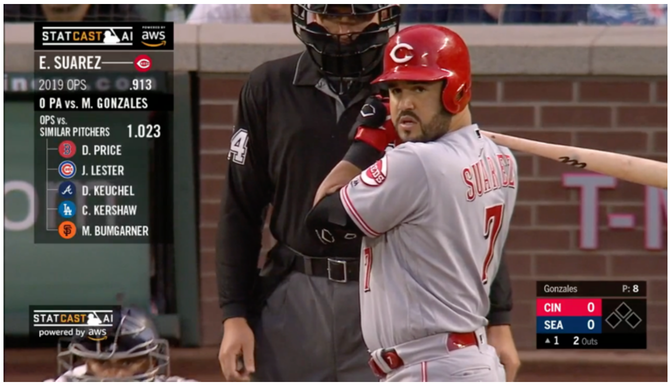

Fig. 1. Screenshot from an MLB game displaying Statcast ML-generated data (left-side) used to highlight the OPS (On-base Plus Slugging) based on *la famiglia* classification for a pitcher of whom the batter has never faced.

As seen in the video frame above (Fig. 1), five pitchers were identified as being similar to the current pitcher. The batter had never faced the current pitcher but had faced the five other pitchers in previous PAs. The average OPS for those PAs was the generated statistic, and with a requisite knowledge of OPS distributions, one could imply (predict) an expected quality of at-bat (AB).

Table 1. Prior machine learning analyses in baseball [4]

Description of work	Method used	Model Score	Ref
Using ML algorithms to identify undervalued baseball players – Ishii	Multi Classification	.857 MSE	[5]
Analyzing and predicting patterns in baseball data using machine learning (ML) techniques	Binary Classification	0.7 R^2	[6]
Data mining career batting performances in baseball	Random forest	.842 GM	[7]

A. *Data Acquisition*

All data for the ML models in this project have been retrieved from baseballsavant.mlb.com; the exact subsets/queries of data is available at Github: trpubz/DBacksAnalysis. Some basic data used in the naïve model was gathered from the mlb.com/stats/ subpages.

A comprehensive analysis, useful for a front office, would provide detailed models for each rostered player against all divisional opponents (batters vs. pitchers) and similar opponent profiles. This paper will demonstrate the capabilities of ML with only one player (Ketel Marte, 2B/CF) vs. National League West pitching staffs. 2019-2022 data will be used for this analysis.

B. *Data Understanding*

A moderate fidelity model (compared to Tom Tango's *eyeball* categorization *famiglias*) will require pitch type, spin rate, and velocity among others. These features are a good foundation to further categorize pitch types for ML modeling insights: rising fastball vs low-90s, sharp curve vs looping curve, circle change vs straight change.

This study will use xwOBA as the primary response variable. This metric synthesizes two, more commonly used but incomplete stats—on-base percentage (OBP) and slugging percentage (SLG). xwOBA was chosen as the output metric for its wide-recognition in the Sabermetrics community for being one of the most important and comprehensive offensive statistics. xwOBA differs from wOBA in that it uses the launch speed and launch angle as the referential measures to provide a probability of similarly hit balls. This removes defensive variations Additionally, wOBA is easily converted to *weighted Runs Above Average (*wRAA) which is a scalar alternative to wOBA, allowing for quick comparisons against league average. For a full explanation, visit the Statcast glossary [8].

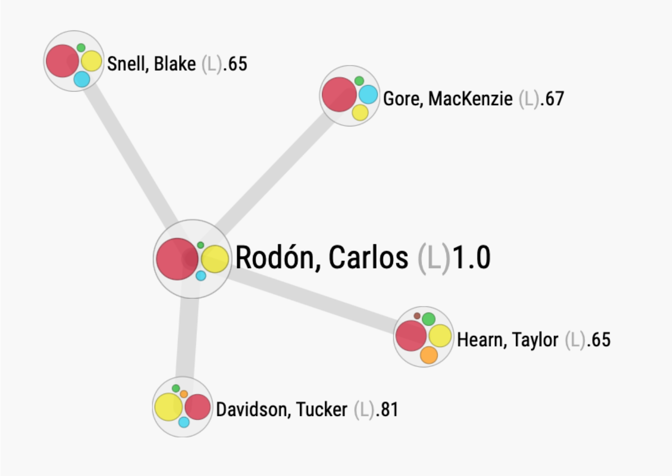

Fig 2. Screen capture of the Player Similarity Scores widget on *baseballsavant* that uses a Euclidean affinity score (details are beyond the scope of this project) to find players with similar batted ball outcomes. This list of players would form the pool of similar players to draw observations from, increasing observation count and tightening variation bounds.

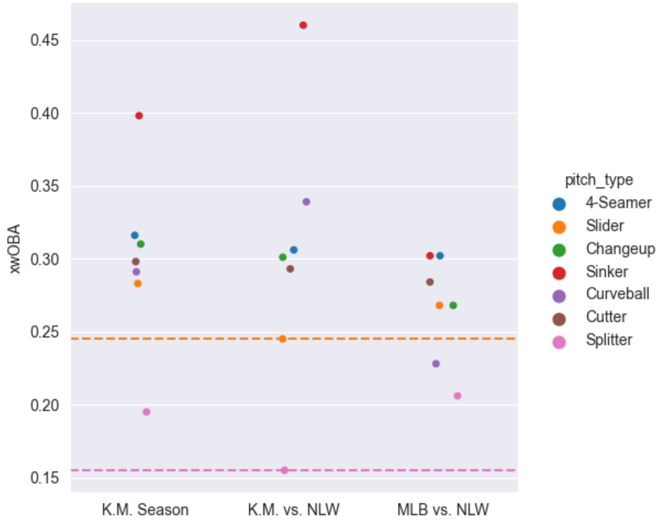

Fig 3. Categorical plot with referential lines based on the second column of data (K. Marte vs NL West). Slider and Splitter are referenced to highlight the deltas between Marte and league average for those types of pitches. The implication can be stated—identifying correlating pitch metrics for those types of pitches and finding other hitters around the league that succeed against those pitch metrics could yield offensive improvements for games played against the NL West opponents.

Table 2. Response variable xwOBA

```
          xwOBA
count  805.000000
mean     0.305965
std      0.359511
min      0.000000
25%      0.022000
50%      0.177000
75%      0.482000
max      1.954000
```

Table 3. Non-categorical Features

```
       release_speed  release_spin_rate   spin_axis        zone       pfx_x  \
count     805.000000         805.000000  805.000000  805.000000  805.000000   
mean       89.656770        2237.119255  179.073292    7.346584   -0.106410   
std         5.598907         362.007749   66.002004    3.912651    0.814202   
min        70.000000         873.000000    4.000000    1.000000   -1.810000   
25%        85.400000        2076.000000  137.000000    4.000000   -0.780000   
50%        91.000000        2283.000000  197.000000    7.000000   -0.180000   
75%        94.100000        2476.000000  221.000000   11.000000    0.510000   
max       101.500000        3291.000000  357.000000   14.000000    1.720000   

            pfx_z       pfx_v  
count  805.000000  805.000000  
mean     0.706025    1.219316  
std      0.712983    0.439451  
min     -1.650000    0.020000  
25%      0.280000    0.944034  
50%      0.820000    1.334054  
75%      1.280000    1.562370  
max      1.870000    2.064800  
```

Table 4. Feature Summary (non-exhaustive)

Input Variable	Data Distribution
Spin Rate	~Log-Normal
Velocity	~Log-Normal
Pitch Type	Categorical
Pitcher Handedness	Binary
Release Angle	~Log-Normal
Zone	Categorical

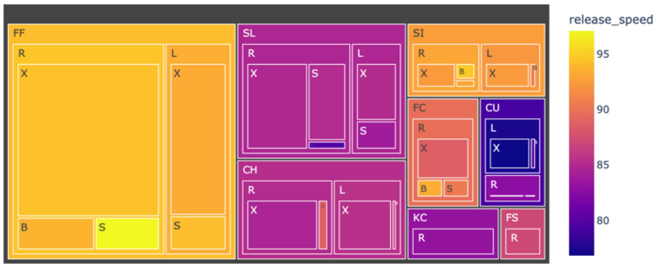

Fig 4. Tree map categorizing pitch type, pitcher handedness, and outcome. Pitch speed informs the color overlay. B = walk, S = strikeout, X = in play. One can easily see the speed difference in strikeouts compared to walks within the Fastball tree.

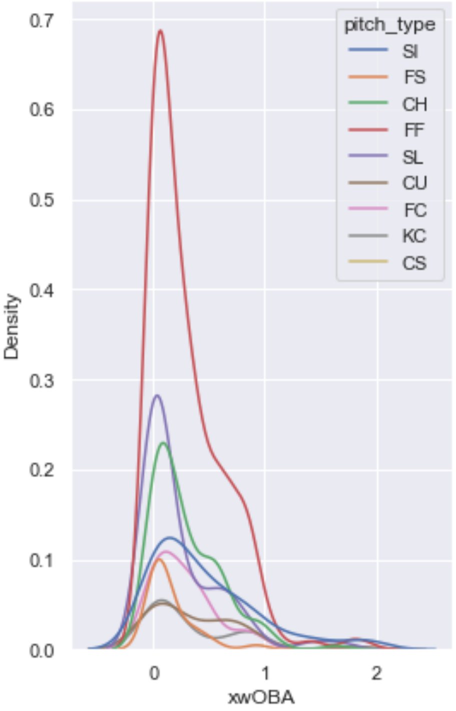

Fig 5. KDE plot shows his best pitch to hit is the Sinker (blue line).

II. Method

All necessary data was retrieved from a baseballsavant search query using the following arguments:

PA Result: [all options except for errors, interferences, steals, and HBP/Intentional BB], Opponent: [NL West], Include Stats: [wOBA, xwOBA], Player: [2022-AZ Marte, Ketel], Player Type: Batter

A. Data Preparation

The data were preprocessed by filtering out sacrifice events (sac-bunt, sac-fly); only 0.23% of PAs. An additional column labeled ‘pfx_v’ was computed as the Euclidean distance or vector magnitude of the pitch’s horizontal and vertical movement. Additionally, any *NaN* occurrences in response variable was converted to the event’s ‘woba_value’. Strikeouts and walks produce a *NaN* xwOBA because there is no ‘observed’ batted ball event; a strikeout has a theoretical xwOBA of 0.0 and a walk of 0.7—however a player’s ability to lay-off a ball is not likely correlated to that pitch’s metrics but rather the player’s plate discipline. For this reason and assumption, a total of 62 walks (7.2% Pas) were dropped from the data set.

This data set combines numerical and categorical features. Categorical features were converted non-ordinally, using pandas.get_dummies(). Categorical features include 'pitch_type', 'zone', & 'p_throws'. The numerical features were normalized and scaled using sklearn.preprocessing PowerTransformer. xwOBA, the target variable, was normalized using the MinMaxScalar transformer from sklearn.

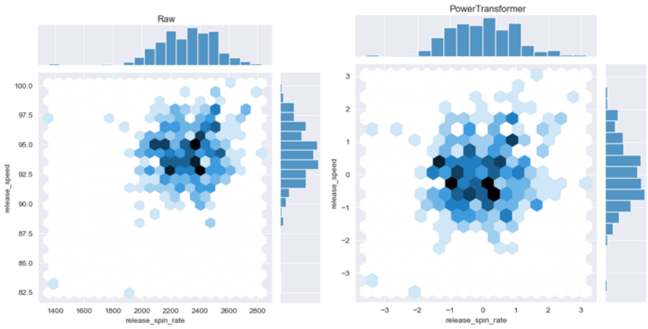

Fig 6. Two features, Velocity and Spin, shown graphed as raw distribution (left) and then normalized and scaled using PowerTransformer (right).

B. Metrics

For this analysis, the regression metrics of root mean squared error (RMSE), R^2 stat, and Akaike information criterion (AIC) are used to measure the performance of the models. Minimizing the RMSE was the primary focus as it provides a scale for how far away the errors in prediction are from the true values. RMSE is more sensitive (than MAE) to outliers and since the target variable is bounded by a theoretical min/max, outliers are not a concern [9]. R^2 is useful in that it explains the proportion of variance accounted for in a model [10]. Lastly, AIC is the last metric used to relatively compare models as it is good at penalizing models with more parameters and rewards the models with better fits [9].

C. Naïve Model

For regular season games from 2019—2022, Ketel Marte's xwOBA for all PAs without walks (805) is .305. This value was used as the predictor for the PAs against NL West opponents. The RMSE equaled .383 telling us the naïve predictor was as good, if not worse, as a random guess at the desired target. An R^2 score of "-0.003" indicates a worse-than mean-equivalent predictability.

D. Classical Modeling

The data was split into a 65/35 train/test segmentation. The model was trained on the larger training data subset and then used to predict the output on the small test subset. The true data was compared to the predicated data which generated the various scoring metrics for the models.

The first model created was a standard ordinary least squares (OLS) regression with the full subset of features. Of note, the p-values for features with presupposed influence such as velo and pitcher handedness indicated weak influence. The full-featured model produced marginal improvement over the naïve model. The second model employed a backward selection methodology by evaluating the p-val stats of the features. After removing features with a p-value < 0.10 in a piecewise manner and remodeling the new feature selections, the only remaining features were interestingly the Sinker & Slider categorical features. Those pitches correspond to Marte's worst xwOBA intra-divison. This model performed worse than the full-featured model. For the third model, a recursive feature elimination (RFE) model was ran over all ~30 features. The best model was interestingly identified as only the Sinker and pitches outside the strike zone. Lastly, a LassoLARSIC (Information Criterion) model was ran. Least-angle regression (LARS) is a regression algorithm for high-dimensional data performed similar to stepwise forward-selection. This method penalized coefficients until the lowest AIC was identified.

E. Neural Network Modeling

Since predicting xwOBA is naturally a regression problem, the loss function chosen was *mean_squared_error*. This allows for incidental comparison to the previous regression models. The Adam optimizer algorithm was chosen for its general applicability and widespread utilization for similar regression problems. The NN models used batch normalization by utilizing z-scaled activations at each layer with the intent to prevent variance shifting during weight assignments. Our problem-set suggests a multi-layer perceptron approach (as opposed to a recurrent neural net RNN) and thus lends itself to the ReLU activation function. And because this is a traditional regression problem, the NN's used a linear activation function for the output layer.

The same train/test split used in the classical modeling was used for the NN analysis. For the higher fidelity models, utilizing the tensorflow callback EarlyStopping class allowed the lowest validation loss value to identify the correct model parameters, minimizing overfitting problems.

A hyperparameter sweep consisting of neurons, layers, batch size, & epochs was conducted. An L2 Ridge regularization method was applied to the best model with notable improvement.

III. Analysis & Results

A. Classical Modeling

Overall, performance was generally weak with marginally better R^2 scores for the four various supervised learning models. The quantifiable improvements are observed by focusing on the RMSE. The models improved the average predicted distance from the true value up to 60%. Using

RMSE as the valued metric: the LassoLars Regression model performed the best.

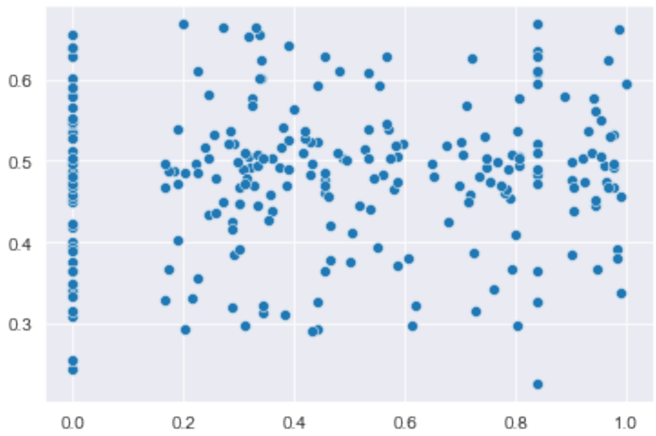

Fig 6. Residual analysis of P-val Selection model. Shows even distribution made possible by normalization to target variable.

Table 4. Classical Model metrics comparisons.

Model	*RMSE*	*R^2*	*AIC*
Full	.159	-0.01	-221
P-value Selection	.160	-0.03	-224
RFE	.156	0.02	7
LassoLars	.156	0.03	-237
FF Only	.156	-0.03	-110
Naïve Model	.383	-0.002	--

B. *Neural Network Modeling*

A 1-D simple NN performed surprising well compared to the naïve model. The chosen feature was spin_rate to predict xwOBA.

Performing a sweep of neurons, layers, batch size, and number of epochs produced slightly worse values than the single dimensional NN model. Neurons were swept from 4-16, layers from 1-4, batch size from 8-128, and epochs from 10-1250.

The L1L2 regularization penalty term is the sum of squared coefficients. This drives coefficients to be smaller and since the moderately large number of features, NetElastic is an appropriate regularization. Similar to the Ridge regression model from the classical analysis section, the Ridge regularized variant of the NN performed the best with the smallest RMSE. Roughly an 8.8% improvement in predicting the true outcome.

Table 5. NN Model metrics comparisons.

Model	*RMSE*	*R^2*	*MAE*
1-D: Spin Rate	.158	-0.002	.127
Full Feature	.198	-0.58	.154
L1L2 Regularized	.159	-0.02	.129

REFERENCES

[1] Statista, "Statista dossier on the Arizona Diamondbacks," 2022.

[2] G. Costa, M. Huber and J. Saccoman, Understanding sabermetrics: an introduction to the science of baseball statistics, McFarland, 2007.

[3] T. Tango, M. Lichtman and A. Dolphin, The Book: Playing The Percentages in Baseball, TMA Press, 2006.

[4] K. Koseler and M. Stephan, "Machine Learning Applications in Baseball: A Systematic Literature Review," *Applied Artificial Intelligence,* vol. 31, no. 9-10, pp. 745-763, 2017.

[5] T. Ishii, "Using Machine Learning Algorithms to Identify Undervalued Baseball Players," Standford University, 2016.

[6] W.-I. Jang, A. Nasridinov and Y.-H. Park, "Analyzing and Predicting Patterns in Baseball Data using Machine Learning Techniques," *Sensors,* 2014.

[7] D. Tung, "Data Mining Career Batting Performances in Baseball," *Journal of Data Science,* 2012.

[8] [Online]. Available: https://www.mlb.com/glossary/statcast/expected-woba.

[9] A. Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 3rd Edition, O'Reilly, 2022.

[10] G. Jame, D. Witten, T. Hastie and R. Tibshirani, An Introduction to Statistical Learning with Applications in R; Corrected 7th Printing, New York: Springer Science+Business Media, 2013.

[11] M.-L. Huang and Y.-Z. Li, "Use of Machine Learning and Deep Learning to Predict the Outcomes of Major League Baseball Matches," *Applied Sciences,* vol. 11, no. 10, p. 4499, 2021.

[12] B. Hall, "Artificial Intelligence, Machine Learning, and the Bright Future of Baseball," *The National Pastime: The Future According to Baseball,* 2021.

[13] M. Hamilton, P. Hoang, L. Layne, J. Murray, D. Padgett, C. Stafford and H. Tran, "Applying Machine Learning Techniques to Baseball Pitch Prediction," in *International Conference on Pattern Recognition Applications and Methods*, 2014.

[14] O. Hall Jr and S. Seaman, "Developing winning baseball teams: a neural net analysis," *International Journal of Sport Management and Marketing,* vol. 5, no. 3, pp. 277-294, 2009.

[15] Y. J. Park, H. S. Kim, D. Kim, H. Lee, S. B. Kim and P. Kang, "A deep learning-based sports player evaluation model based on game statistics and news articles," *Knowledge-Based Systems,* vol. 138, pp. 15-26, 2017.