Final Report: Predicting MLB Player Performance

Mike Ogrysko

Loyola University Maryland

DS 796: Data Science Project

Dr. Bu Hyoung Lee

April 20, 2023

Abstract

Major League Baseball and its teams are relying on data now more than ever to build their on-field product. This project focused on the player performance aspect of data analytics in Major League Baseball. Specifically, it used available player data from Stathead Baseball and Fangraphs from 1980 onwards to examine three aspects of player performance related to the Wins Above Replacement metric.

First, for the purposes of feature engineering and limiting results, exploratory data analysis (EDA) was completed to determine the estimated peak player age based on WAR at each position. The peak age of players ranged from 26 to 30, with designated hitter as an outlier at age 35. Next, a binary classification using the Random Forest machine learning algorithm was completed to determine whether the players that played at their peak age in 2022 will exceed their 2022 WAR metric in 2023. Twenty out of 45 batters and 21 out of 36 starting pitchers were predicted to exceed their 2022 WAR values in 2023. Finally, this project evaluated two regression models – multiple linear and lasso regressions – built to predict 2023 WAR for the players that played at their peak ages in 2022. Lasso regression reduced a subset of 37 batting features to 28 while explaining 50.99058% of the variation of the response values of the training data – showing that it may be a good model to build upon to predict WAR in the future.

Contents

List of Figures

Project Motivation

Major League Baseball (MLB) is a multi-billion dollar-business, nearing $11 billion in gross revenues in 2022 (Brown, 2022). Its 30 teams have player payrolls that range from $61 million to $282 million with an average of $163 million (Spotrac, 2022). The average value of an MLB team is $2 billion (Ozanian & Teitelbaum, 2022). Given the enormity of these revenues, payrolls, and team values, MLB and its teams are relying on data now more than ever to build their on-field product.

How are these teams using the data they collect? Some teams are using this data as a development tool to measure performance and adjust technique as necessary. For example, pitching coaches for the Boston Red Sox used video and data collected from high-speed cameras and ball trackers to diagnose biomechanical delivery flaws of two of their relief pitchers (Schrage, 2019). Players are using analytics to market themselves during free agency. In the winter leading up to the 2020 season, outfielder Marcell Ozuna sent a 43-page booklet to teams highlighting metrics where he excelled like the exit velocity of his hits and his Expected Weighted On-base Average (xwOBA) (Davidoff, 2019), which takes exit velocity, launch angle, and sprint speed into account to create a measure of a player's offensive contributions (MLB, 2022). Teams use data about where a player hits the ball to position their defensive players on the field. In 2013, the Tampa Bay Rays and Milwaukee Brewers were estimated to have saved their teams between 10 and 15 runs over the course of the season just by shifting their players to specific areas on the field based on data they evaluated for each hitter (Common, 2014). Teams are using data to project the results of individual pitcher and hitter matchups. The Yankees estimate hitters' performances against their pitchers' specific pitches – using measured pitcher velocity and spin as well as a hitter's typical bat path – to simulate an expected result. This allows the team to map out specific scenarios and plans for each game (Carig, 2018). Teams also

use this data to generate metrics to predict player performance and facilitate the creation of their season rosters.

A crucial metric for measuring player performance is Wins Above Replacement (WAR). The general idea behind WAR as a metric is placing a value on a player that accounts for the number of team wins that he alone represents when compared to a replacement player at the same position (Castrovince, 2020, p. 189). As an example, a team comprised of only Minor League players is estimated to win 48 games. If a player from that team was replaced with a free agent, and the team's win total jumped to 54 games, the new player would have a 6 WAR (Castrovince, 2020, p. 191). Interestingly, there is no single calculation for a player's WAR. The individual teams have their own WAR calculations, factoring in different public inputs and other considerations such as scouting reports, budget, and other market-specific information (Castrovince, 2019). Public facing statistical reference sites, Baseball-Reference, FanGraphs, and Baseball Prospectus have their own calculations for WAR that use different metrics in their computations (Baseball-Reference, 2022).

There are commonalities among all WAR calculations. First, all calculations employ different methods for positional players and starting pitchers. In addition, despite the different calculations, the scale is widely agreed upon (see Table 1).

Table 1

WAR Categorization for Positional Players and Starting Pitchers

Quality of Player	WAR
MVP Material	8 or higher
Superstar	6-8
All-Star	4-6
Solid Regular	2-4
Role Player	1-2
Bench Material	0-1
Triple A Material	0 or below

Note. Reprinted from *A Fan's Guide to Baseball Analytics* (p. 192) by A. Castrovince, 2020, Sports Publishing. Copyright 2020 by Anthony Castrovince.

WAR is not used as a viable metric for relief pitchers because they generally do not accrue enough innings pitched to generate a WAR that aligns with this scale (Castrovince, 2020, p. 192).

In general, a WAR model for positional players sums the following elements (Weinberg, 2014):

- Runs contributed through hitting relative to average,
- Runs contributed through baserunning relative to average,
- Runs contributed through fielding relative to average,
- Runs contributed based on position played (e.g., Shortstop, Centerfield, Catcher, etc.) relative to average,
- Runs contributed based on league adjustment relative to average, and
- Runs relative to a replacement-level player.

This sum represents the total number of runs contributed by the player beyond what a replacement player would have contributed (Baumer & Zimbalist, 2014, p. 73). It is divided by

an estimate of the number of runs a team needs to score to add a single win to their total (Weinberg, 2014).

Baumer and Zimbalist (2014, pp. 72-73) provide an example using former Mets third baseman, David Wright's 2008 season and data from FanGraphs as an example:

- Contributed 44.0 runs above average to the Mets in the 2008 season,
- Cost the team 4.5 runs as a baserunner,
- Contributed 5.1 defensively,
- Contributed 2.3 runs as a third baseman, and
- Is credited with being 24.5 runs better than the average replacement player.

Wright's total number of runs beyond what a replacement player would contribute was 71.4. Baumer and Zimbalist found FanGraphs' league adjustment to be negligible, so this was left out of the total. For this estimate of WAR, they estimated 10 runs to produce a single win. Dividing Wright's 2008 total by 10, provides a WAR of 7.1 – meaning that he was worth 7.1 wins above a replacement player for the Mets in 2008.

For starting pitchers, FanGraphs sums the following metrics to calculate WAR (Weinberg, 2017):

- League Fielding Independent Pitching (FIP) less the pitcher's FIP divided by the pitcher's specific runs per win,
    - FIP focuses on the "events a pitcher has the most control over – strikeouts, unintentional walks, hit by pitches, and home runs (Castrovince, 2020, p. 109)."
    - This difference is the number of runs above average per nine innings relative to a specific league.

- Difference between an average pitcher and a replacement-level pitcher multiplied by innings pitched divided by 9, and
- League Correction.

Weinberg (2017) provides an example of starting pitcher Marcus Strohman's 2016 season using data from FanGraphs:

- American League FIP was estimated at 4.56 less Strohman's FIP at 4.12 is equal to 0.44 divided by Strohman's runs-per-win of 9.61 provides a wins-per-game-average at 0.046,
- The difference between an average pitcher and a replacement-level pitcher is estimated at 0.12 added to Strohman's wins-per-game-above-average of 0.046 is 0.166 multiplied by his 2016 innings pitched of 204 divided by nine innings gives an initial WAR of 3.76, and
- League correction is estimated at -0.14.

Summing 3.76 and -0.14 together produces a 2016 WAR of 3.6 for Marcus Strohman.

This project focused on the player performance aspect of data analytics in Major League Baseball. Specifically, it used available player data to examine three aspects of player performance related to the WAR metric. First, for the purposes of feature engineering and limiting results, exploratory data analysis (EDA) was completed to determine the estimated peak player age based on WAR at each position. Next, a binary classification using the Random Forest machine learning algorithm was completed to determine whether the players that played at their peak age in 2022 will exceed their 2022 WAR metric in 2023. Finally, this project evaluated two regression models built to predict 2023 WAR for the players that played at their peak ages in 2022.

The main challenge encountered in the completion of this project was the use of WAR as the metric to measure overall performance. For position players, WAR measures the performance through the production of runs through hitting, baserunning, fielding, the position played, and the league played in, and is relative to the runs produced by a replacement-level player (Weinberg, 2014). For starting pitchers, WAR measures performance through events the pitcher can control like strikeouts, unintentional walks, hit by pitches, home runs, and the league played in, and is also relative to replacement-level pitcher performance (Weinberg, 2017). For position players and pitchers, these individual performance indicators represent a tremendous amount of variability. Consider two players with similar careers in terms of position and games played, team success, and seasonal WAR. In the next season, suppose the two players diverge. A new player joins Player A's team allowing Player A to see better pitches and have his best statistical offensive season. He maintains his current baserunning and defensive performances. Player A's season WAR increases as a result. On the other hand, Player B's team loses many of its players to free agency. He is forced to change positions and his defensive metrics decline. He suffers an injury and misses half of the season. In addition, his home stadium changes the dimensions on the outfield walls by pushing them back. As a result, his offensive statistics decline, and his season WAR is the lowest of his career. Some factors, like the home stadium dimensions could be accounted for as a predictor; however, many factors cannot reasonably be accounted for. Is it possible to account for injuries, personnel shifts, and game strategies? If not, does that make WAR one of the most difficult metrics to predict?

Managing the Data Science Analysis and Research Platform

Overall, this project was divided into 5 parts:

- Data gathering and preliminary EDA,
- Analysis of the peak player age at each position,
- Binary classification for exceeding 2022 WAR,
- Regression analysis for the WAR prediction, and
- Evaluation of results, report writing, and presentation creation.

MLB player data is publicly available from several sources. This project relied on the data compiled at Stathead Baseball (https://stathead.com/baseball/) and FanGraphs (https://www.fangraphs.com/). The data used was from 1980 onwards. All data was exported from Stathead Baseball and Fangraphs as a series of CSV files, which were then merged, cleaned, and transformed using Python in Jupyter Notebooks.

Python was used for the EDA exercise of identifying the peak age at each position and the binary classification for exceeding 2022 WAR in 2023. Notable Python libraries used included: Pandas, Numpy, Matplotlib, Seaborn, and Sklearn. Within Sklearn, the RandomForest algorithm was used for the binary classification. R was used for the evaluation of the multiple linear and lasso regressions for the WAR prediction analysis.

For the binary classification, Linear SVM was the initial algorithm used. However, when Linear SVM was fitted with the training dataset, the classification accuracy topped out at 60%. Given this accuracy, Gaussian Naïve Bayes, Logistic Regression, and Random Forest were tried. Random Forest provided the greatest accuracy on the training dataset.

Relevant Python and R documents have been uploaded to Github and are available for review. See Appendix A for links.

Related Work

Hakes and Turner (2011) examined peak performance ages using On-base Percentage Plus Slugging Percentage (OPS) as the measure of performance for position players. They found that the best players live in the extreme right-hand tail of an ability distribution curve and tend to peak two years later than marginal players. In terms of OPS as a measure of performance, they believe that it is "simple to calculate and an accurate predictor of team output (wins)." It also measures production agnostic of playing time and, as a result, "provides a conservative estimate of within-career variation." In their study, they also found that there were large differences in OPS across defensive positions.

Schulz, Musa, Staszewski, and Seigler (1994) also examined the peak ages of position players and pitchers. In their study, they examined a series of offensive and pitching statistics for a group of players and identified mean peak ages for each statistic. For all statistics evaluated, they found that the peak mean age fell between 27 and 30 years. Much like Hakes and Turner (2011), they found that the better players consistently peaked at older ages when looking at the entire careers of the players. Elite players tended to continue to play well several years longer than lesser players. When looking at the distribution of peak performance ages by specific statistics, they noticed that fielding percentage and walk rate improved from the early 20s to the late 30s. Pitchers' strikeouts and hitters' stolen bases were more likely in players' early 20s seasons.

Like Schulz, Musa, Staszewski, and Seigler (1994), Bradbury (2009) examined how player performance improved and declined with age by evaluating performance against individual player statistics. He used a multiple regression analysis technique to identify a peak age by skill or statistic. He found that hitters tended to peak in terms of batting average and the rate at which they hit doubles and triples at around age 28, while players peak at age 30 and 32

for On Base Percentage (OBP) and walk rate, respectively. For pitchers, he found that they have their best strikeout rates in their age 23-24 season, while they have their best walk rates in their age 32-33 season. In both cases, this makes sense. Hitters tend to have more speed in their younger years and older players are more patient in their at-bats and will walk more. Pitchers tend to have greater velocity when they are younger, while older pitchers probably rely more on control for successful outcomes and thus are walking fewer batters.

Ng (2017) modeled player performance based on age, experience, and talent. He identified peak physical age using WAR. He felt that WAR was the appropriate metric to use for examining performance as the sum of the league's WAR each season is equal to one thousand, allowing players to be compared between different eras. A player's WAR increases and decreases based on playing time. Ng suggested that this measure of opportunity is another reason that WAR can be used to measure performance because it can be assumed that teams play their best players. Ng argued that teams should use WAR to calculate player salary and contract values. He offered a calculation for salary figures, Expected WAR * Salary/WAR, and suggested that "the salary per WAR can be easily calculated by dividing the total MLB payroll for a particular year by 1,000, which is the WAR in MLB in any given year." He found that the peak age for hitters is 26.6 years, while the peak age for pitchers is 24.5 years.

Though more practitioner based, Marchi, Albert, and Baumer (2019, pp. 189-193) developed a quadratic regression model to estimate career trajectories. Much like Hakes and Turner (2011), they based their performance measurement on OPS using a player's individual season data. They used this trajectory data to determine the peak age of players based on their positions. Due to the variability in the player trajectories – one player peaking in his mid-thirties,

while another peaking in his late-twenties – they found a high variability in the player peaks as well.

Sun, Lin, and Tsai (2023) focused on predicting individual players' home run totals by using Long Short-Term Memory (LSTM) neural network models. They developed five LSTM models and found that they were able to predict home run totals with low Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) in two seasons, 2018 and 2019. They compared their results to other machine learning algorithms and found that a basic multiple linear regression model faired comparably to the LSTM results in 2019.

Baumer, Jensen, and Matthews (2015) suggested a new WAR metric for evaluating overall player performance. They highlight two issues with WAR as a statistical measure of player performance: "a lack of uncertainty estimation and a lack of reproducibility." They suggest that WAR is often "misrepresented in the media as a known quantity without any evaluation of the uncertainty in its value" and criticized services like Baseball-Reference.com and FanGraphs.com for not publishing uncertainty estimates along with their WAR values. They also suggested that WAR calculations often use proprietary data and methods and, as a result, are not reproducible by the general public. As an alternative, they proposed a conservation of runs framework, called openWAR. This metric uses public data to allocate offensive runs generated and defensive runs saved to the players to measure their performance.

Hypotheses, Experiments, and Data Analysis

This project focused on the player performance aspect of data analytics in Major League Baseball. Specifically, it used available player data to examine three aspects of player performance related to the WAR metric. First, for the purposes of feature engineering and limiting results, exploratory data analysis (EDA) was completed to determine the estimated peak player age based on WAR at each position. Next, a binary classification using the Random Forest machine learning algorithm was completed to determine whether the players that played at their peak age in 2022 will exceed their 2022 WAR metric in 2023. Finally, this project evaluated two regression models built to predict 2023 WAR for position players.

Estimated Peak Player Ages by Position

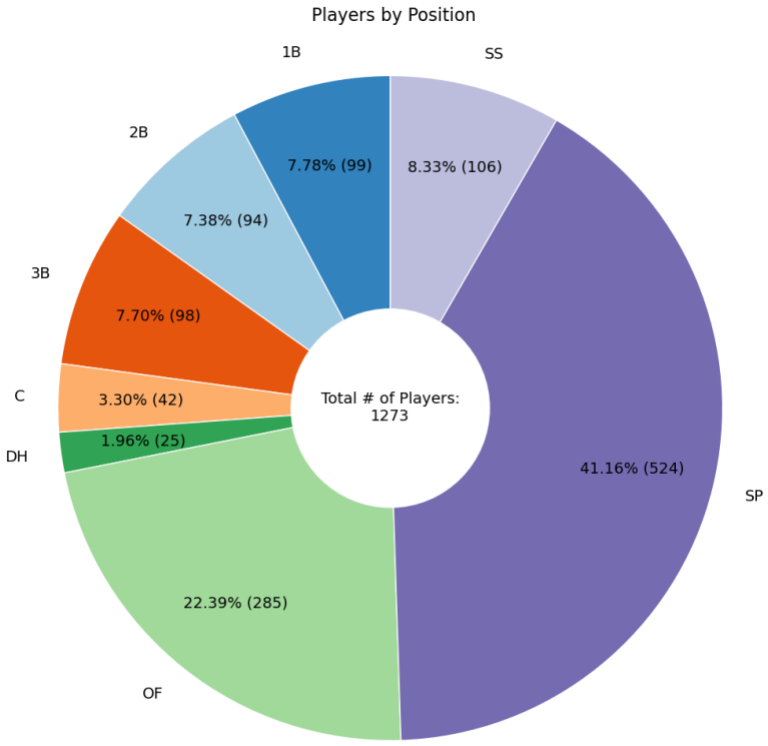
To estimate player peak ages based on position, this project focused on single-season player data from 1980 to present matching the following criteria:

- Regular season only,
- Played specific position for at least 51% of games (positional players),
- Plate appearances are greater than or equal to 500 (positional players),
- Started 60% of games pitched in (starting pitchers), and
- Started at least 20 games (starting pitchers).

This represented 5627 seasons for positional players and 3800 seasons for starting pitchers. The data was then filtered to include only season data for players who played 3 or more seasons at their positions. Filtering resulted in season data for 749 position players and 524 starting pitchers. Of the 749 position players in the dataset, 285 were outfielders, 106 were shortstops, 99 were first basemen, 94 were second basemen, 98 were third basemen, 42 were catchers, and 25 were designated hitters (see Figure 1).

Figure 1

Pie Chart: Players by Position in Dataset

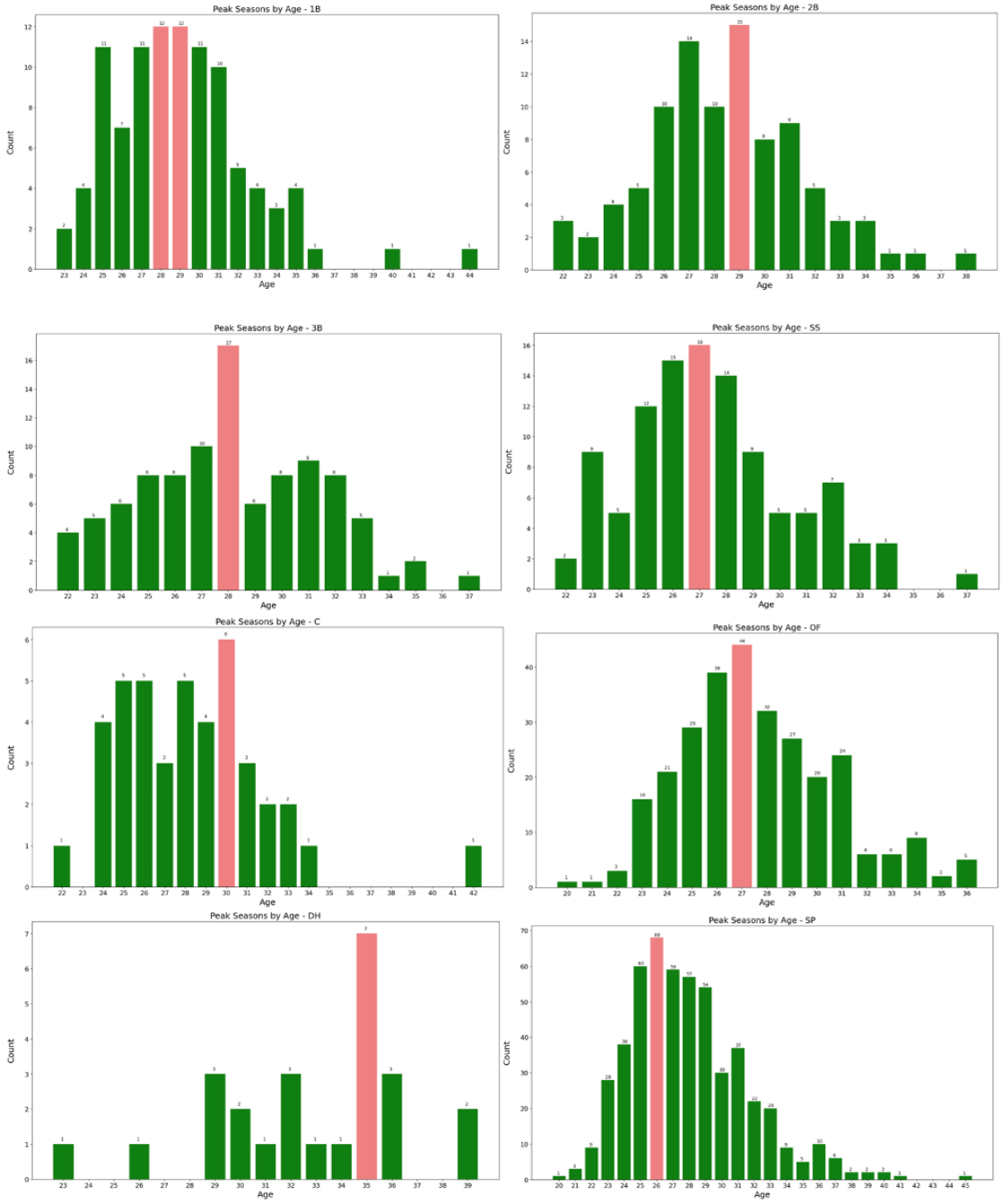


Note. This pie chart shows the counts of players by position in the dataset.

Peak ages were determined by identifying the ages at which players achieved their maximum WAR values in their careers. This provided a frequency for each age at each position. The age with the highest frequency represented the estimated peak age for a given position. Reviewing the frequency distributions (see Figure 2), the peak age for the shortstop and outfield positions was 27. Age 28 was the peak age for third basemen. Ages 28 and 29 were the peak ages for first basemen; however, age 29 was used as the peak age for first basemen for the remaining questions in the project. Age 29 was the peak age for second basemen. The peak ages for catchers and the DH position were older at 30 and 35, respectively. The peak age for starting pitchers was 26.

Figure 2

Frequency Distributions: Peak Seasons by Age and Position



Note. The age with the highest frequency is shown in coral.

Players Exceeding their 2022 WAR metric in 2023

A binary classification model was created to predict whether players who played at their peak ages in 2022 will exceed their 2022 WAR in 2023. Training data was compiled by exporting individual season data from two sources, Stathead Baseball and Fangraphs, from 1980 to 2022 with no limits on player appearances. After merging the two data sources together, there were 120 features for each player season (see Appendix B, Table B1). The features included basic batting statistics (e.g., batting average, home runs, stolen bases, etc.) and advanced metrics (e.g., WAR, wOBA, wRC, etc.) from both sites. Using the ages from the peak age analysis, each player's peak season was found and added to each season record. For example, the frequency distribution showed that outfielders tended to peak in their age 27 season. For a player like Barry Bonds, who played outfield throughout his career, the statistics from his age 27 season were added to each of his 22 player seasons. In addition to the basic and advanced statistics, the following features were engineered for each player season:

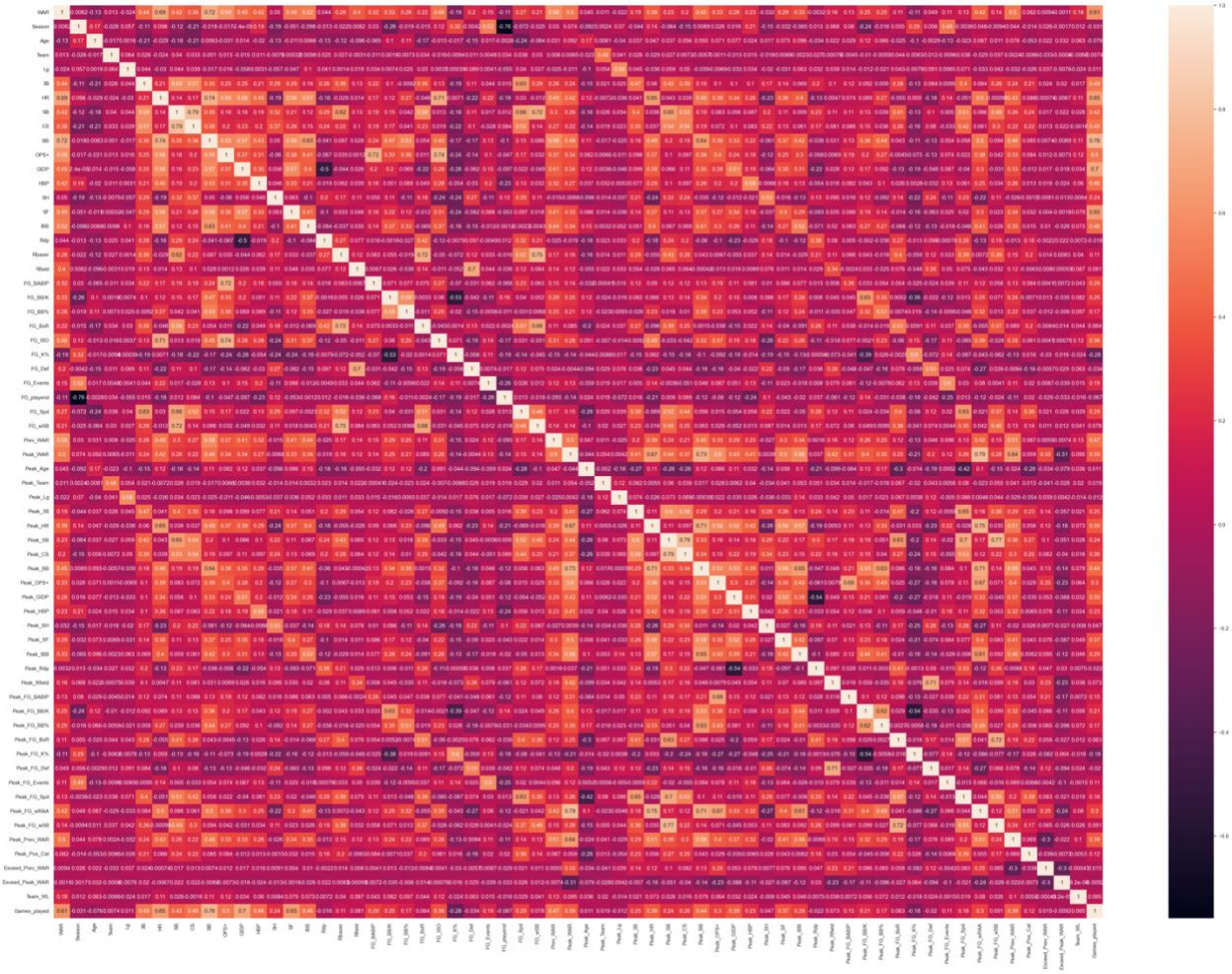
- Next_WAR: the player's WAR value from the next season
- Prev_WAR: the player's WAR value from the previous season
- Peak_Prev_WAR: the player's WAR value from the season prior to the peak season
- Peak_Next_WAR: the player's WAR value from the season after the peak season
- Peak_Pos_Cat: the position played during the player's peak season
- Exceed_Prev_WAR: whether the player exceeded the previous season's WAR value; 0-No, 1-Yes
- Team_WL: the player's team's win-loss percentage for the season
- Games_played: the percentage of games played based on the number of games the team played

- Exceed_Peak_WAR: whether the player exceeded his next season's WAR value; 0-No, 1-Yes; this was the dependent variable

Because of the number of features and the knowledge that many of these features must be highly correlated, many of these features were removed after reviewing the correlation among the features.

Figure 3

Final Correlation Matrix – Batters



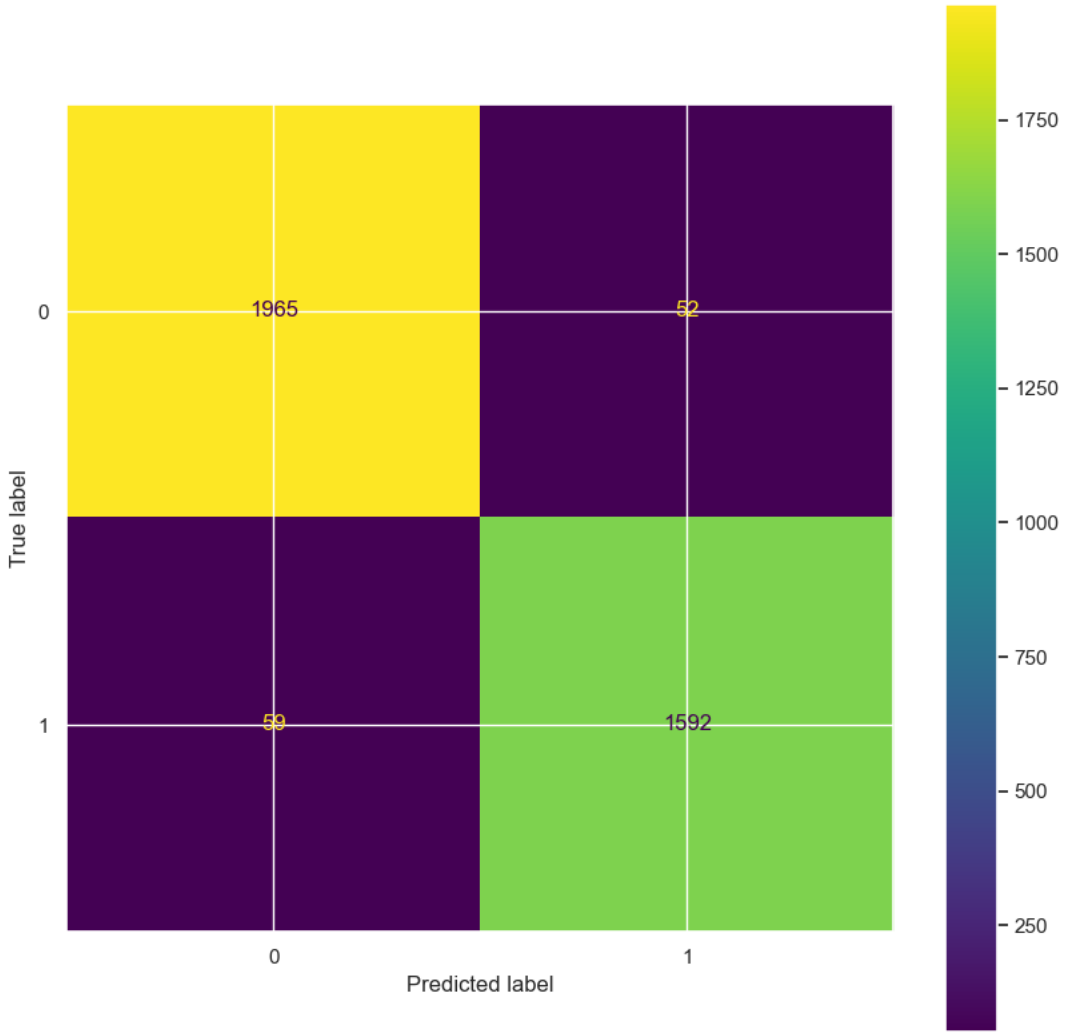
Note. The correlation matrix shows the 62 features used for the batter analysis. Correlations above 0.80 and below -0.80 were removed.

Eliminating the highly positively- and negatively- correlated features reduced the number of features to 62 (see Figure 3, Appendix B, Table B2). The training dataset for this model included 12,224 individual player seasons. This portion of the project focused on players who played at their peak age in 2022 to predict whether they will exceed their 2022 WAR in 2023. This represented 45 players. For this prediction, the dependent variable, Exceed_Peak_WAR was used with 0 representing 'No' and 1 representing 'Yes'.

Several machine learning algorithms were tested with the training dataset. The Random Forest algorithm returned the best results. The Random Forest algorithm is an ensemble of decision trees that takes the average of "multiple (deep) decision trees that individually suffer from high variance to build a more robust model that has a better generalization performance and is less susceptible to overfitting" (Raschka & Mirjalili, 2019, p. 198). For this problem, the algorithm was configured to generate 1000 decision tress with a max depth of 10 and using 8 cores. Using test-train-split from the Sklearn library, 70% of the 12,224 data records were used for training and 30% to test the accuracy. Of the 3,668 player records used for testing, 111 were misclassified. The accuracy returned from the Random Forest algorithm on the training dataset was 97% (see Figure 4).

Figure 4

Confusion Matrix Batting Training Dataset

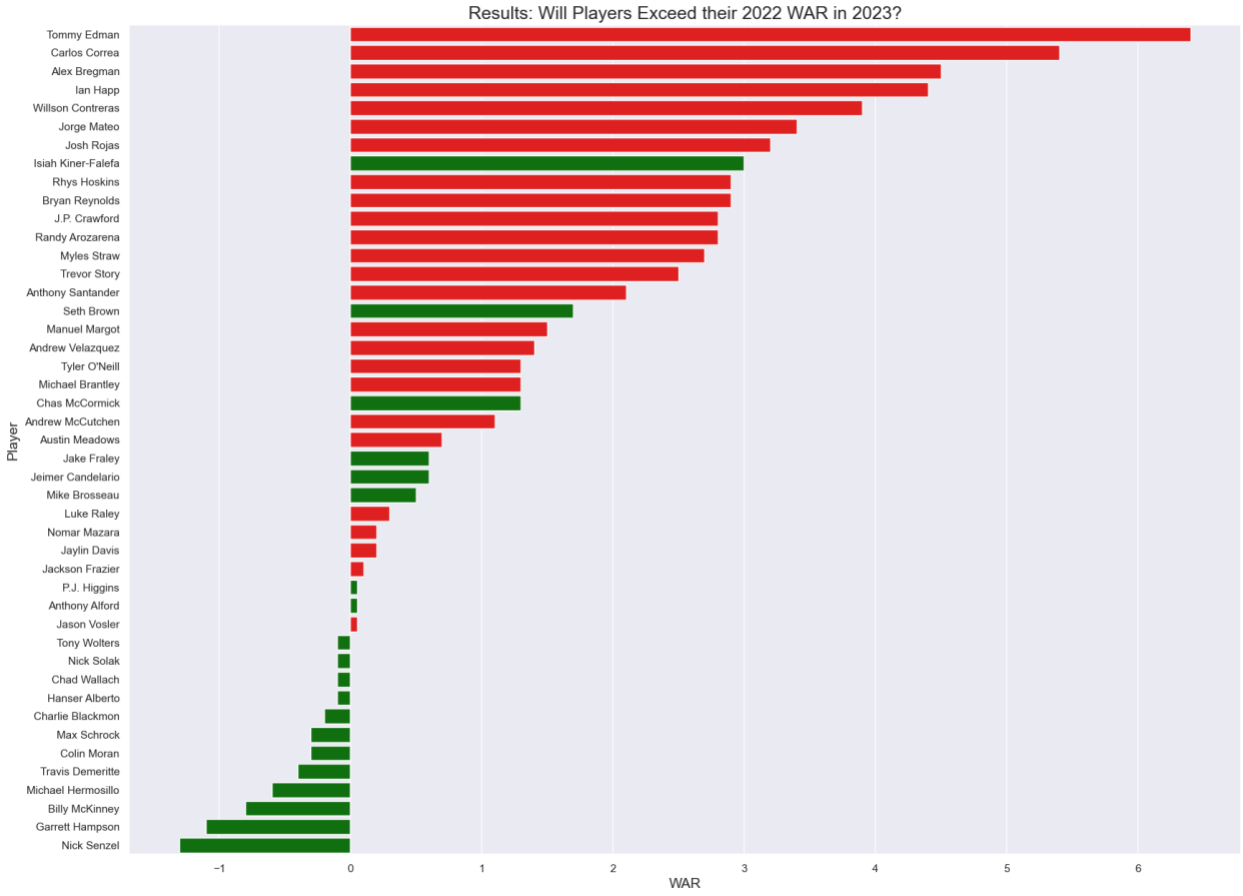


Note. This confusion matrix shows that 111 of the 3,668 batter records were misclassified.

After fitting the data for the 45 players into the Random Forest model, 20 of the 45 players that played at their peak age in 2022 were predicted to exceed their 2022 WAR values. The remaining 25 players were predicted to fall below their 2022 WAR. Figure 5 is a horizontal bar chart that plots the 2022 WAR values for each of the players and highlights in green those that were expected to exceed their 2022 WAR values in 2023. Those that were not expected to exceed their 2022 WAR values are highlighted in red.

Figure 5

Will Players Exceed their 2022 WAR in 2023?



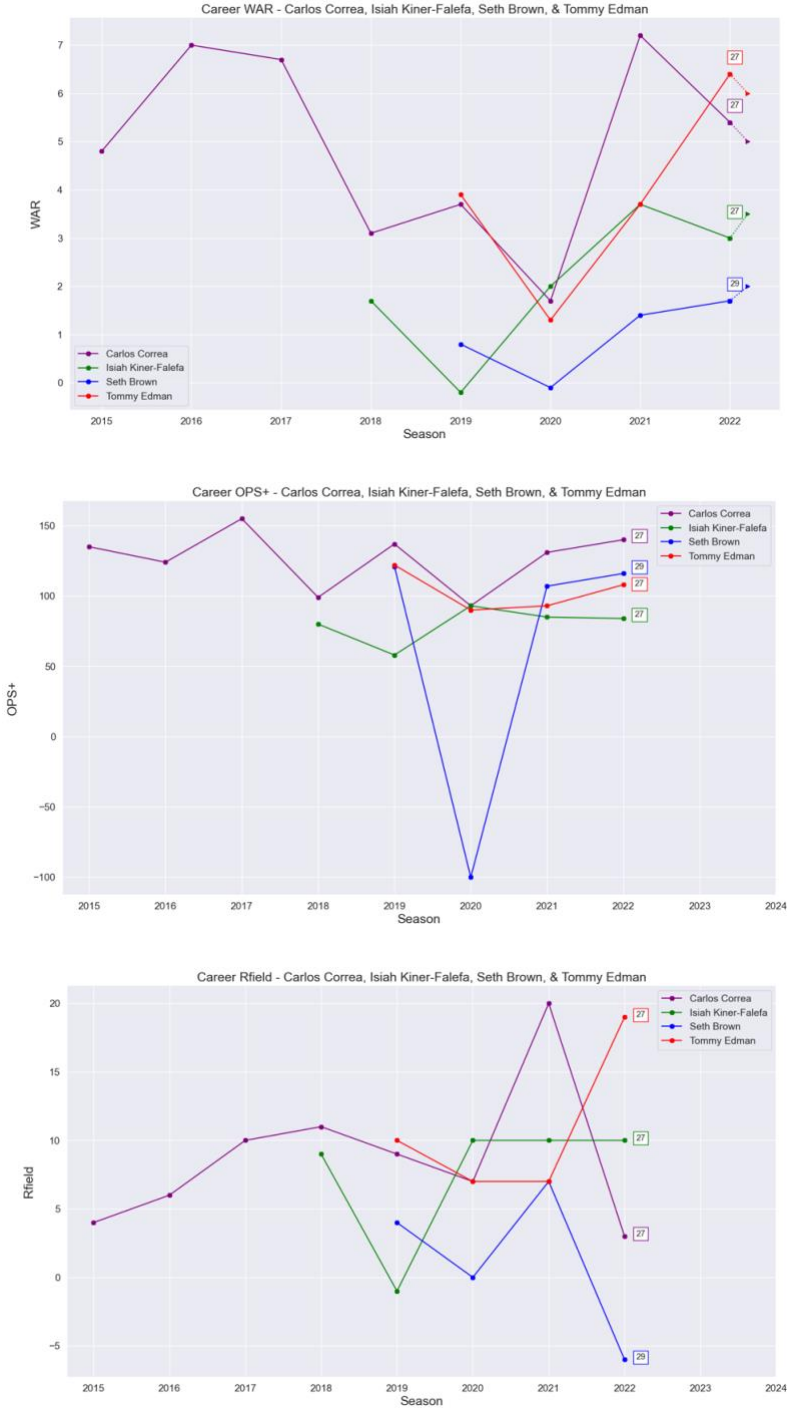
Note. This horizontal bar chart shows the 2022 WAR for each player and highlights in green if they were predicted to exceed their 2022 WAR in 2023. Red bars indicate that the player was not predicted to exceed his 2022 WAR in 2023.

Looking at two batters that were predicted to exceed their 2022 WAR in 2023 and two that were not, it was interesting to see whether anything could be identified by looking at their career trends for several statistics. In Figure 6, there are three charts plotting career WAR, OPS+, and Rfield for four players. In the career WAR chart, the players with the higher 2022 WAR values were predicted to decline in 2023, while the players with the lower 2022 WAR values were predicted to increase. All four players were very close in their OPS+ numbers throughout their

careers to date with the exception of Seth Brown's 2020 season. It was difficult to determine whether Rfield, the number of runs better or worse than average the player was for all fielding, had an impact on WAR. The numbers fluctuated for all players except for Kiner-Falefa who has been relatively consistent with the exception of one season. Kiner-Falefa and Brown were predicted to improve on their 2022 WAR values in 2023, but their Rfield numbers, respectively, remained level and declined. Edman and Correa's Rfield values went in opposite directions in 2022, but the WAR values for both were predicted to decline in 2023. For future work, it might be interesting to continue to this level of analysis for more of the batter statistics.

Figure 6

Career WAR, OPS+, and Rfield for Selected Players



Note. The charts above show career WAR, OPS+, and Rfield numbers for four players with their ages noted for 2022.

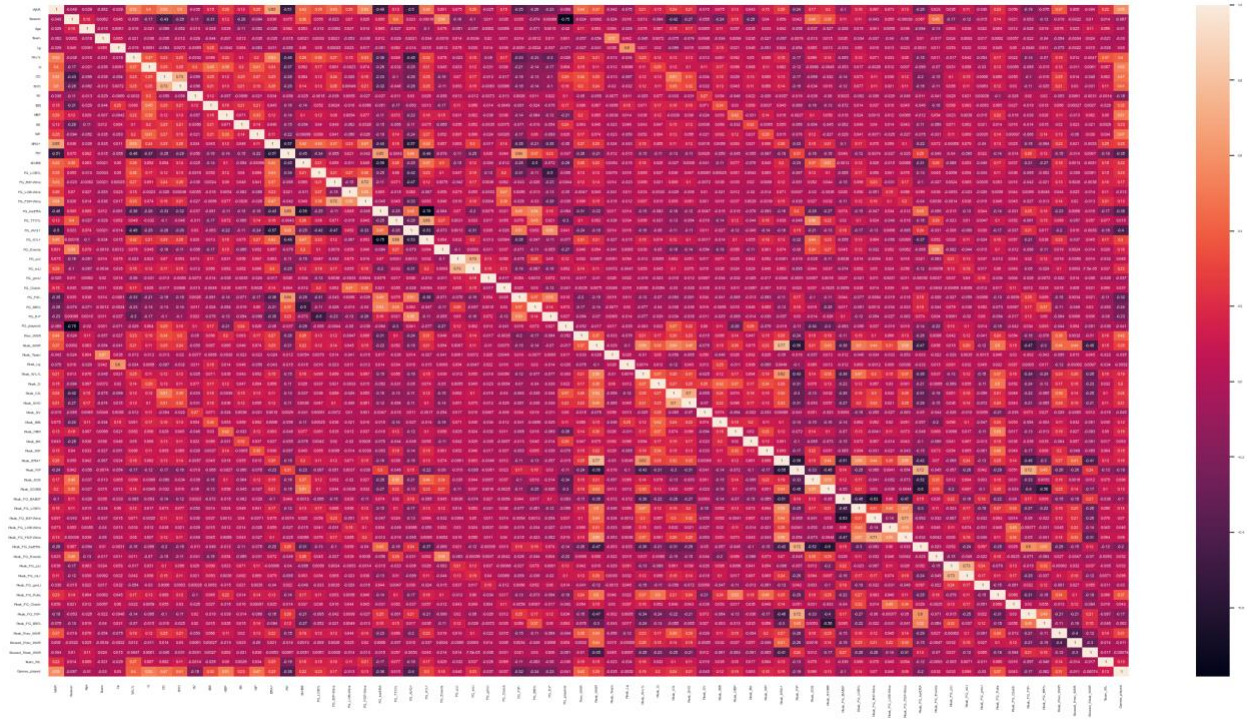
The same analysis was completed for starting pitchers using a different dataset. Again, training data was compiled by exporting individual season data from two sources, Stathead Baseball and Fangraphs, from 1980 to 2022 with no limits on appearances. After merging the two data sources together, there were 148 features for each pitcher season (Appendix B, Table B3). The features included basic pitching statistics (e.g., wins, loses, ERA, strikeouts, etc.) and advanced metrics (e.g., WAR, FIP, RE24, etc.) from both sites. Much like with the batters, the ages from the peak age analysis were used by adding each pitcher's peak season to each of their season records. For example, the frequency distribution showed that starting pitchers tended to peak in their age 26 season. For Roger Clemens, the statistics from his age 26 season were added to each of his 24 seasons. In addition to the basic and advanced statistics, the following features were engineered for each pitcher season:

- Next_WAR: the pitcher's WAR value from the next season
- Prev_WAR: the pitcher's WAR value from the previous season
- Peak_Prev_WAR: the pitcher's WAR value from the season prior to the peak season
- Peak_Next_WAR: the pitcher's WAR value from the season after the peak season
- Exceed_Prev_WAR: whether the pitcher exceeded the previous season's WAR value; 0-No, 1-Yes
- Team_WL: the pitcher's team's win-loss percentage for the season
- Games_played: the percentage of games started based on the number of games the team played
- Exceed_Peak_WAR: whether the player exceeded his next season's WAR value; 0-No, 1-Yes; this was the dependent variable

Again, because of the number of features and the knowledge that many of these features must be highly correlated, many of these features were removed after reviewing the correlation among the features.

Figure 7

Final Correlation Matrix – Pitchers



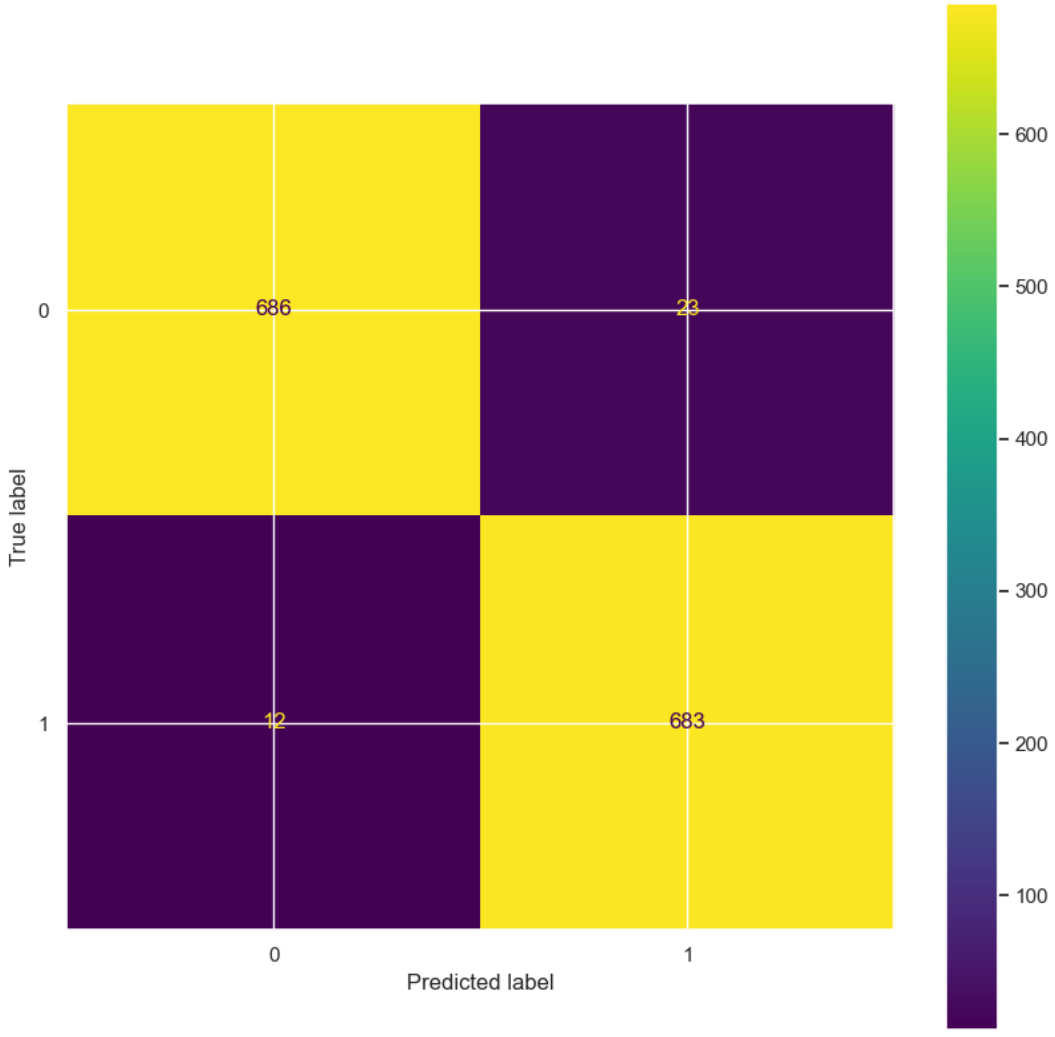
Note. The correlation matrix shows the 68 features used for the pitcher analysis. Correlations above 0.80 and below -0.80 were removed.

Eliminating the highly positively- and negatively- correlated features reduced the number of features to 68 (see Figure 7, Appendix B, Table B4). The training dataset for this model included 4,678 individual pitcher seasons. As with the batter analysis, this portion of the project focused on pitchers who played at their peak age in 2022 to predict whether they will exceed their 2022 WAR in 2023. This represented 36 pitchers. For this prediction, the dependent variable, Exceed_Peak_WAR was used with 0 representing ‘No’ and 1 representing ‘Yes’.

Again, the Random Forest algorithm was used as it returned the best results. As with the batters, the algorithm was configured to generate 1000 decision tress with a max depth of 10 and using 8 cores. Using test-train-split from the Sklearn library, 70% of the 4,678 data records were used for training and 30% to test the accuracy. Of the 1,404 player records used for testing, only 35 were misclassified. The accuracy returned from the Random Forest algorithm on the training dataset was 98% (see Figure 8).

Figure 8

Confusion Matrix Pitching Training Dataset

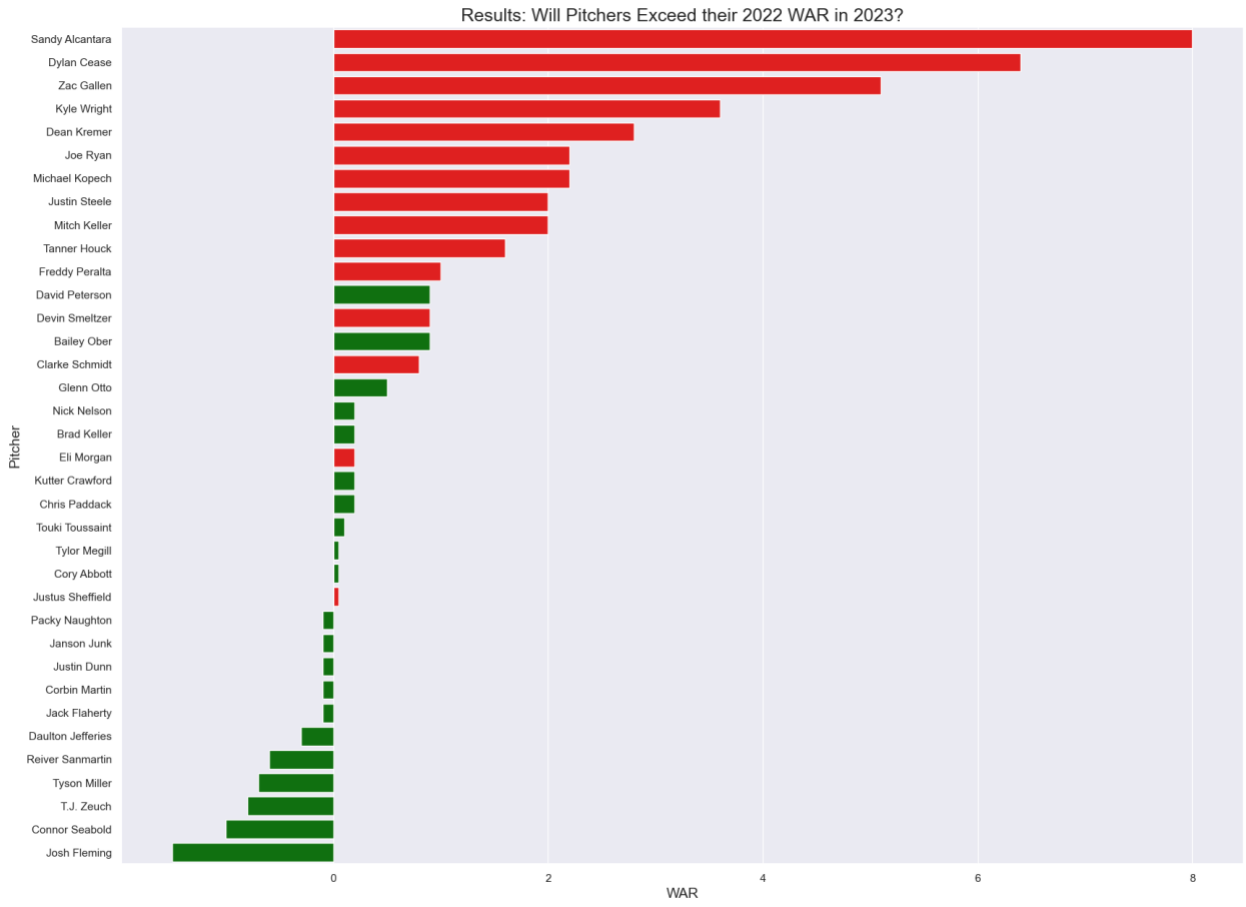


Note. This confusion matrix shows that 35 of the 1,404 pitcher records were misclassified.

After fitting the data for the 36 players into the Random Forest model, 21 of the 36 players that played at their peak age in 2022 were predicted to exceed their 2022 WAR values. The remaining 15 players were predicted to fall below their 2022 WAR. Figure 9 is a horizontal bar chart that plots the 2022 WAR values for each of the pitchers and highlights in green those that were expected to exceed their 2022 WAR values in 2023. Those that were not expected to exceed their 2022 WAR values are highlighted in red.

Figure 9

Will Pitchers Exceed their 2022 WAR in 2023?

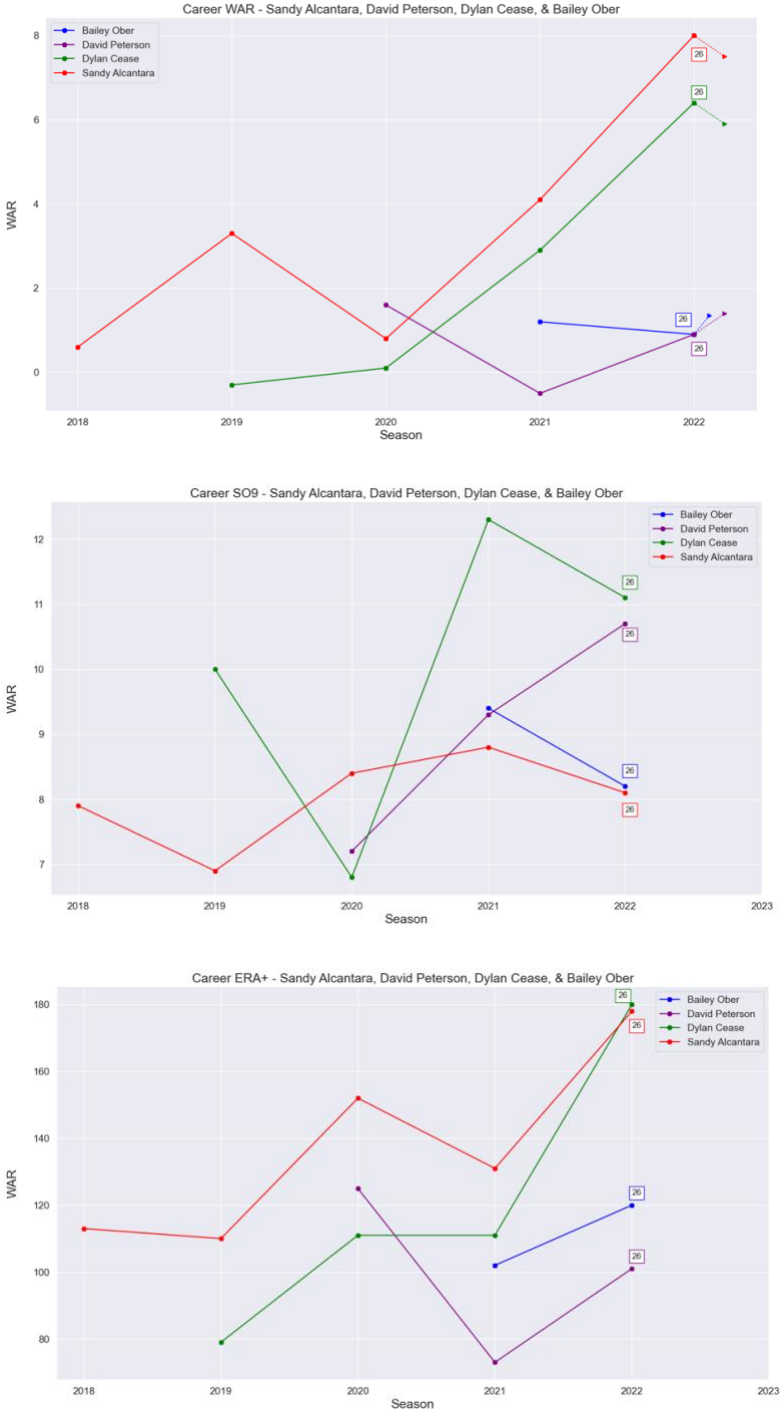


Note. This horizontal bar chart shows the 2022 WAR for each starting pitcher and highlights in green if they were predicted to exceed their 2022 WAR in 2023. Red bars indicated that the pitcher was not predicted to exceed his 2022 WAR in 2023.

Looking at two starting pitchers that were predicted to exceed their 2022 WAR in 2023 and two that were not, it is interesting to see whether any notable points could be identified by looking at their career trends for several statistics. In Figure 10, there are three charts plotting career WAR, SO9, and ERA+ for four players. In the career WAR chart, much like the batters, the players with the higher 2022 WAR values were predicted to decline in 2023, while the players with the lower 2022 WAR values are predicted to increase. Three of the four pitchers' SO9 values decreased from their previous season's numbers in 2022. Interestingly, all pitchers' ERA+ values increased in 2022. Alcantara and Cease had nearly identical ERA+ numbers in 2022 and both were substantially higher than the ERA+ numbers for Ober and Peterson. Alcantara and Cease were predicted to decline in WAR value in 2023, while Ober and Peterson were predicted to increase. As with the batters, it might be interesting to continue to this level of analysis for more of the pitching statistics for future work.

Figure 10

Career WAR, SO9, and ERA+ for Selected Starting Pitchers



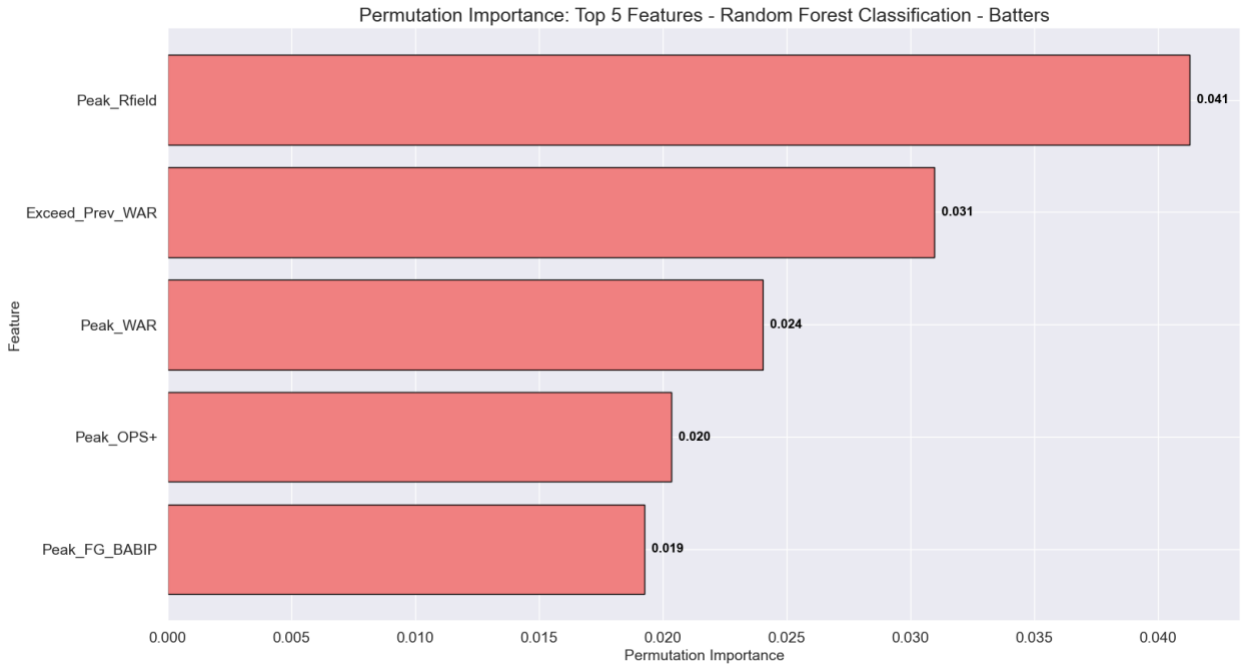
Note. The charts above show career WAR, SO9, and ERA+ numbers for four players with the age included on the 2022 season.

A permutation-based feature performance was completed to determine the features that impacted the Random Forest models the most. For the position player data, the following features were found to have the greatest impact: (see Figure 11):

- Peak_Rfield: # Runs better or worse than average the player was for all fielding from peak season
- Exceed_Prev_WAR: Did the player exceed previous season's WAR?
- Peak_WAR: Player's Wins Above Replacement from peak season
- Peak_OPS+: On-base plus slugging percentages adjusted for ballpark from peak season
- Peak_FG_BABIP: Batting average on balls in play from peak season

Figure 11

Permutation Importance: Top 5 Features – Random Forest Classification – Batters



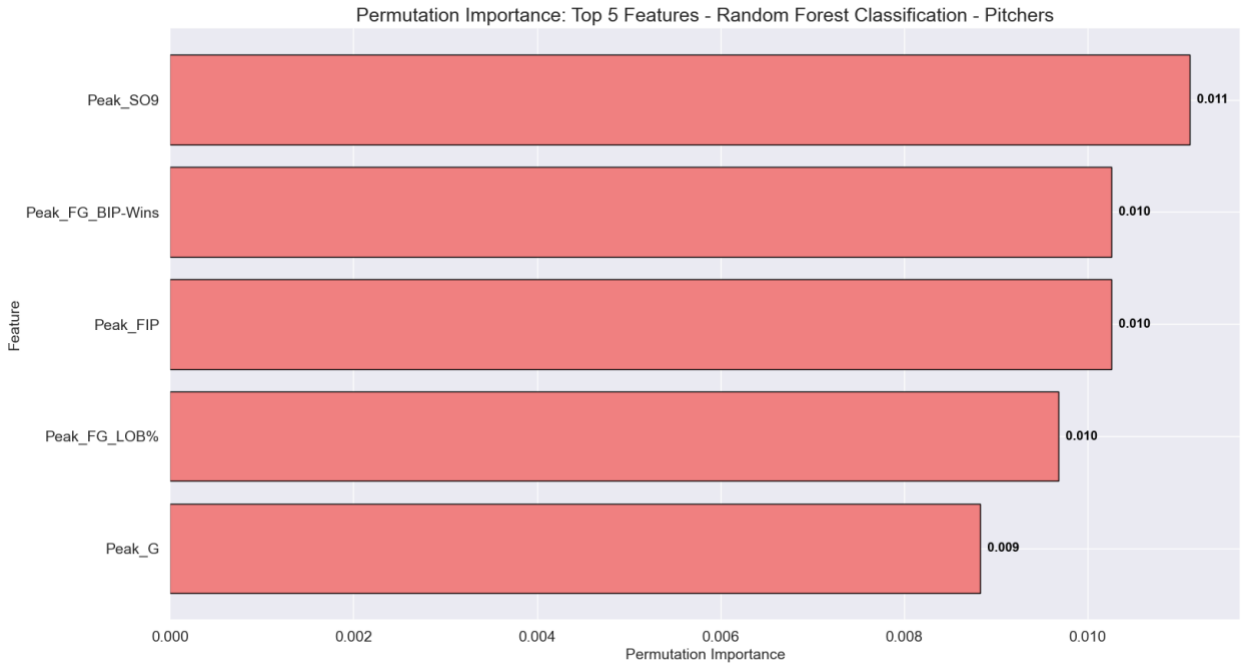
Note. This horizontal bar chart shows the top features and their permutation importance numbers.

For the starting pitcher data, the following features were found to have the greatest impact (see Figure 12):

- Peak_SO9: Strikeouts times 9 divided by innings pitched from peak season
- Peak_FG_BIP-Wins: Batting average allowed on balls in play Wins - # Wins a pitcher has added from peak season
- Peak_FIP: Fielding independent pitching from peak season
- Peak_FG_LOB%: Left on-base percentage from peak season
- Peak_G: # Games played from peak season

Figure 12

Permutation Importance: Top 5 Features – Random Forest Classification – Pitchers



Note. This horizontal bar chart shows the top features and their permutation importance numbers.

Evaluating Regression Models to Predict 2023 WAR

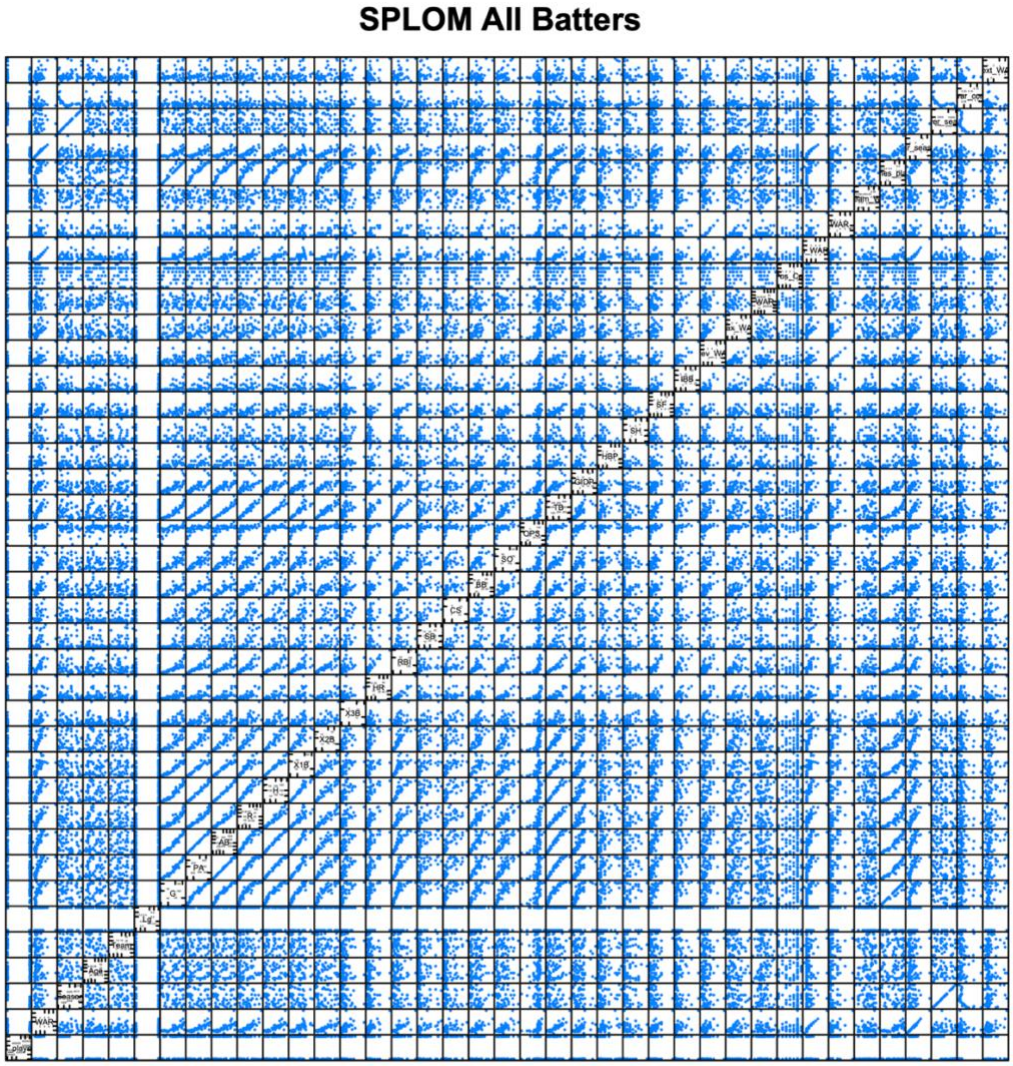
With the question answered of the players who are and are not predicted to exceed their 2022 WAR values in 2023, a natural next step would be to predict their 2023 WAR values. Because WAR is a floating numeric value, a regression seemed necessary in order to predict WAR. In this portion of the project, two regression techniques, multiple linear and lasso

regression, were compared to determine which would be more appropriate for predicting WAR. A subset of 37 of the dataset's features was used for each model (see Appendix B, Table B5) with 14,768 individual player seasons for training.

The full dataset was used initially on the multiple linear regression model with the understanding that multicollinearity would be present (see Figure 13).

Figure 13

Scatterplot Matrix of Regression Data



Note. This scatterplot matrix shows a random sampling of player records for 37 features.

Seventeen of the 37 features at were statistically significant at 0.1 or less and the initial Adjusted R^2 value was 0.509. To handle the multicollinearity, a stepwise method was employed to eliminate feature relationships with high correlations and reduce the features from 37 to 10. This model had an Adjusted R^2 value of 0.5056 and used the following features:

- Age
- Lg
- CS
- SO
- SH
- Max_WAR
- Max_WAR_Age
- Pos_Cat
- Prev_WAR_Class
- war_season

Before settling on the model, an all-subsets regression was completed. This method reduced the number of features down to the following nine:

- Age
- X1B
- SB
- BB
- Max_WAR
- Max_WAR_Age
- Pos_Cat

- Prev_WAR_Class
- war_season

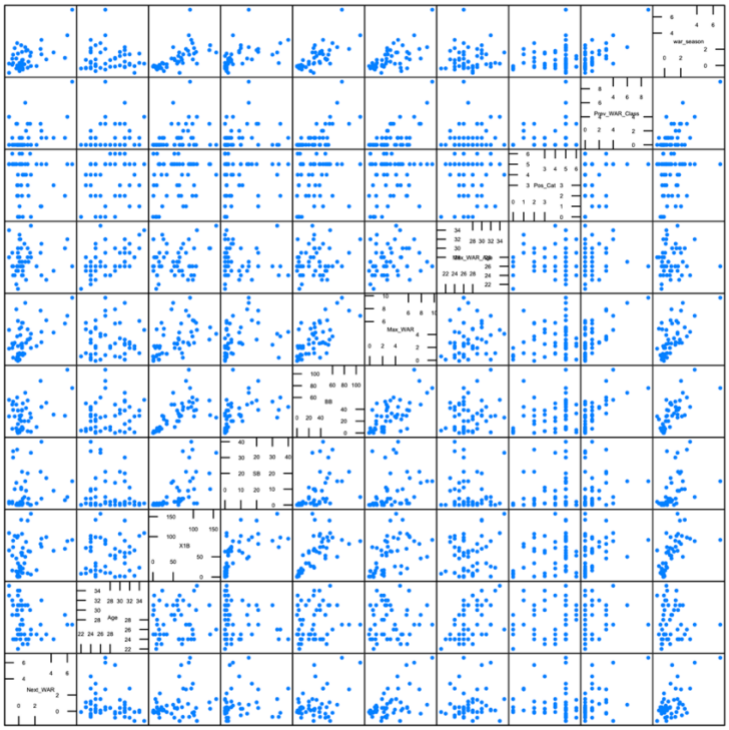
All features were statistically significant at 0.001, producing the following model:

Y = 1.3183109 - 0.1378358(Age) - 0.0031971(X1B) + 0.0085434(SB) + 0.0035137(BB) + 0.3448804(Max_WAR) + 0.0836675(Max_WAR_Age) - 0.0258308(Pos_Cat) + 0.1121132(Prev_WAR_Class) + 0.3154692(war_season) + ϵ

In addition, the Adjusted R^2 value increased to 0.5067, which meant that this model was able to explain 50.67% of the variation of the response values of the training data. From the revised scatterplot matrix, all multicollinearity had been removed from the model (see Figure 14).

Figure 14

Scatterplot Matrix with Features from Final Linear Regression Model

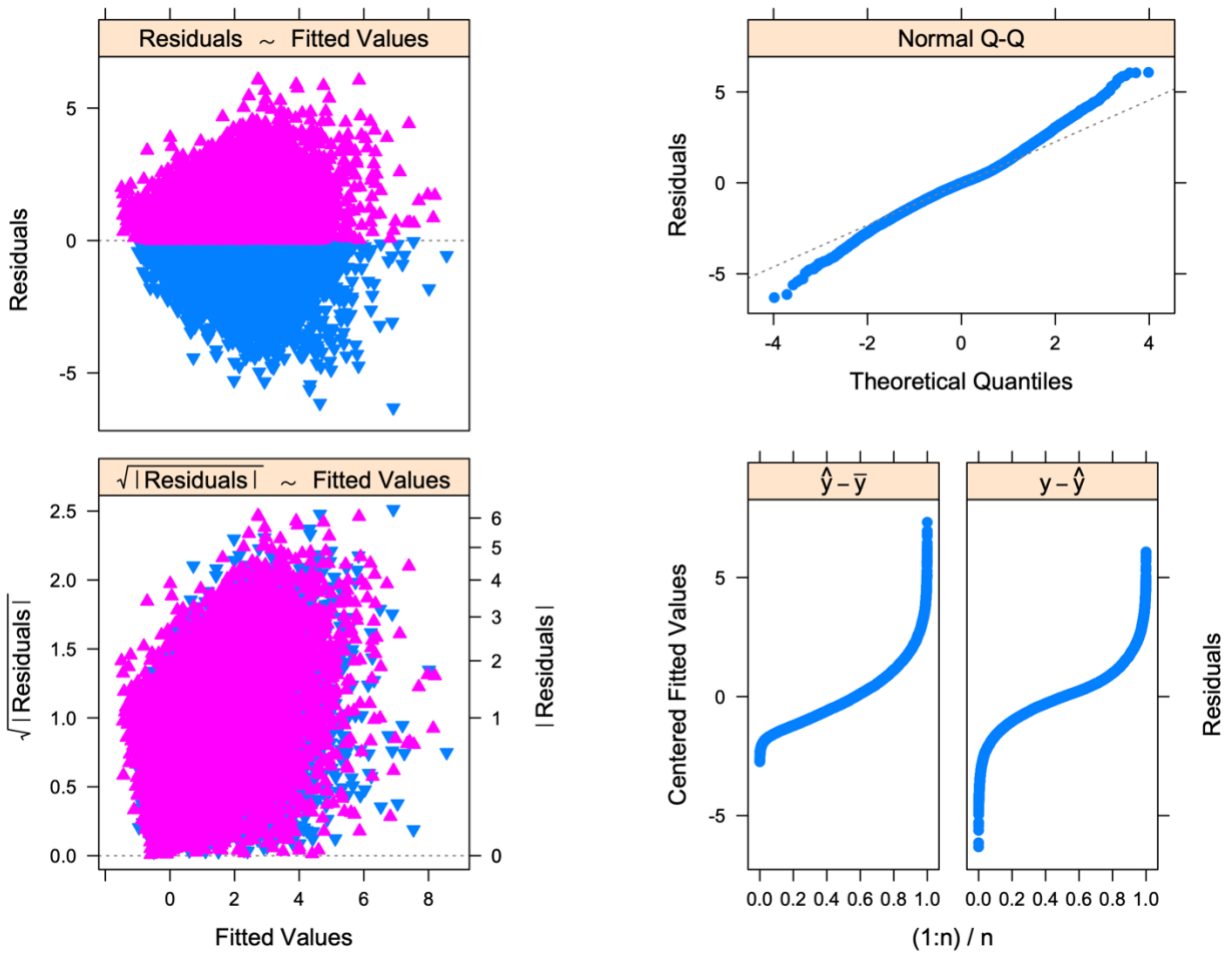


Note. This scatterplot matrix shows a random sampling of the player records for the 9 features that were selected by the all-subsets regression.

The residual plot of the fitted values showed that the residuals were randomly spaced around the horizontal axis. The Normal QQ plot seemed to show normality toward the center and then diverge as the tails approached (see Figure 15).

Figure 15

Residual and Normal Plots



Note. These residuals and normal plots are from the multiple linear regression model using 9 features.

Reviewing the coefficient estimates provided the following observations about Next_WAR:

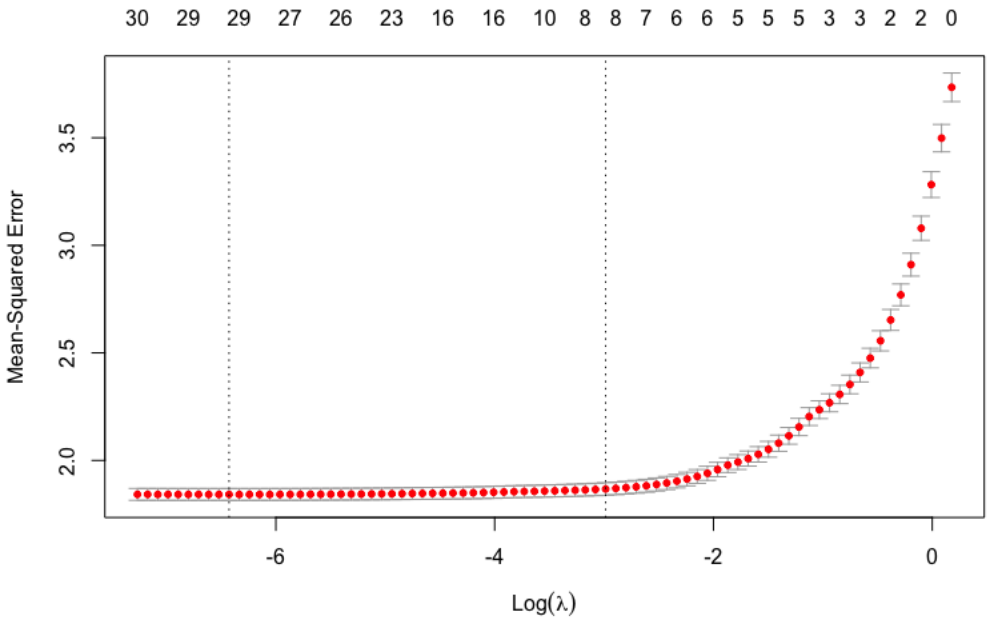
- Decreased by 0.1378358 when Age increased by 1
- Decreased by 0.0031971when X1B increased by 1
- Increased by 0.0085434 when SB increased by 1
- Increased by 0.0035137 when BB increased by 1
- Increased by 0.3448804 when Max_WAR increased by 1
- Increased by 0.0836675 when Max_WAR_Age increased by 1
- Decreased by 0.0258308 when Pos_Cat increased by 1 (C = 0, 1B = 1, 2B = 2, 3B = 3, SS = 4, OF = 5, DH = 6)
- Increased by 0.1121132 when Prev_WAR_Class increased by 1
- Increased by 0.3154692 when war_season increased by 1

Looking at the largest coefficient estimates, their impact on Next_WAR makes sense. As players age, their predicted Next_WAR value should decrease. Players with higher war_season (player's season WAR divided by the average season WAR) and Max_WAR (a player's highest WAR value in his career) values should positively impact predicted Next_WAR values.

With a lasso regression, the L1 penalty associated with this regression forces some of the coefficient estimates to zero. In effect, this is a method of variable selection producing a sparse model in which only a subset of the variables is used (James, Witten, Hastie, & Tibshirani, 2021, p. 241). As such, the full dataset was used with the understanding that multicollinearity would be present and that the lasso model would perform variable selection. Using, Next_WAR as the response variable, a matrix of all predictor variables also was created. A k-fold cross validation was completed to find the optimal lambda value (0.00194649) that minimized Mean Squared Error (MSE) (see Figure 16).

Figure 16

MSE by Lamba Value



Note. This plot shows the optimal lambda value (0.00194649) that minimized MSE.

Using the best lambda value, 28 coefficients were selected out of 37 (see Table 2). The rest were reduced to 0.

Table 2

Lasso Regression Coefficients and their Estimates

Coefficient/Estimate		Coefficient/Estimate	
Age	-0.1374474	SH	-0.009243828
Team	0.001023863	SF	-0.002436568
Lg	-0.08.026940	IBB	0.003014843
G	-0.001930639	Prev_WAR	0.01840499
R	0.002866738	Max_WAR	0.3418047
X1B	-0.005278239	Max_WAR_Age	0.08140411
X3B	0.01565901	Pos_Cat	-0.02978787
SB	0.007536027	Season_WAR_Class	0.09786154e
CS	0.003990475	Prev_WAR_Class	0.08629849
BB	0.003094779	Team_WL	-0.003169164
SO	-0.002307260	Games_played	0.5201730
OPS+	-0.0000005750008	war_season	0.1838946
GIDP	0.006894613	player_season	0.0000007089201
HBP	-0.003169534	war_corr	0.3037931

Note. This table shows the coefficients selected by the lasso regression.

The best lambda and best model were used to predict the WAR values on the training set. Using these predictions, an R^2 value was determined to be 0.5099058, explaining 50.99058% of the variation of the response values of the training data.

Comparing the two regression models, the lasso regression did all of the work of selecting the appropriate features – selecting 28 features. After an all-subsets regression, the multiple linear regression model used 9 features. Looking closer at the features, there were similarities among those with the largest estimates. Age, Max_WAR, and war_season were among the features in both models that had the largest coefficient estimates. Looking at the R^2 values between the two regression models, lasso was slightly better at 0.5099058 compared to 0.5067 for the multiple linear regression model. Given the large number of predictors that would

seem appropriate to predict WAR, as well as the slight improvement in the R^2 value, it would seem that the lasso regression would be the appropriate regression to continue to build upon in order to predict WAR for upcoming seasons.

Threats to the Validity of this Study

The main weakness of this project is that it never develops an adequate model for predicting performance. WAR was selected as the measure of performance, and it proved to be difficult to develop a model that would either predict it within a category or as a specific value. Would it have been more appropriate to narrow the terms of performance – i.e., focus on a specific aspect like offense only or even hits within the offensive category? With the exception of individual play data that can be attributed to a player, all publicly available player data was used in this project. Was this data enough? Was there missing data that could have assisted with the prediction of WAR? Some features were engineered; however, are there additional features that needed to be developed in order to predict WAR? It is assumed that teams are predicting player performance, but what are the best practices? There is not much research available demonstrating success – either by taking a similar approach or a different one.

In terms of assumptions, the approach taken to predict whether a player will exceed their 2022 WAR and to evaluate models to predict WAR used individual player seasons for data. It was assumed based on research that this was the appropriate approach. However, an aggregation of individual player data might have been another approach worth taking.

Future Work

This project ends short of developing a model that accurately predicts WAR for upcoming seasons. There is a question if WAR is the appropriate metric to predict player performance. Is it too variable? Should individual statistics focused on specific parts of the game be used instead? For example, should On-base Plus Slugging Percentages (OPS) be used instead to predict offensive performance specifically. To continue this work, additional research should be done on the appropriate response variables, as well as possible predictors. This project attempted to develop some engineered features; however, the models were never accurate enough to indicate that they were successful. Additional models should also be researched. Several were examined for the WAR prediction –Naïve Bayes, Logistic Regression, neural networks, Random Forest, multiple linear regression, and lasso regression. An appropriate next step might be to try to interview analytics teams from sites like Baseball-Reference or Fangraphs or even an MLB team.

Reflections

I learned several things through the completion of this project. With all of the data cleaning and feature engineering, I think I became a better Python and R programmer. I learned more about baseball statistics than I knew previously – specifically about Wins Above Replacement. It has been several years since I have done any academic research and writing, so I do feel like I was able to regain some of those skills as well. I was fairly comfortable with the Random Forest machine learning algorithm and multiple linear regression from some of my previous coursework; however, I enjoyed the opportunity to put them into practice. In addition, I enjoyed learning about lasso regression, which I had not encountered previously. One of the selling points of the data science program at Loyola was the capstone experience. It has been rewarding to put many of the lessons learned into practice and it has been a satisfying end to the program. I think the scaffolding of the proposal course and this final course has provided milestones to make this project a reality. For anyone doing a future project, I would recommend following those milestones as closely as possible – they kept me on track while balancing other courses, my work schedule, and my personal life.

References

Baseball-Reference. (2022). *Data Coverage.* https://www.baseball-reference.com/about/coverage.shtml

Baseball-Reference. (2022). *WAR comparison chart*. https://www.baseball-reference.com/about/war_explained_comparison.shtml

Baumer, B.S., Jensen, S.T., & Matthews, G.J. (2015). openWAR: An open source system for evaluating overall player performance in major league baseball. *Journal of Quantitative Analysis in Sports, 11*(2), 69-84. https://www.degruyter.com/document/doi/10.1515/jqas-2014-0098/html

Baumer, B. & Zimbalist, A. (2014). *The sabermetric revolution*. University of Pennsylvania Press.

Bradbury, J.C. (2009). *Peak athletic performance and ageing: Evidence from baseball*. *Journal of Sports Sciences, 27*(6), 599-610. https://doi.org/10.1080/02640410802691348

Brown, M. (2022, Apr 7). *How Major League Baseball could crack $11 billion in revenues in 2022*. Forbes. https://www.forbes.com/sites/maurybrown/2022/04/07/how-major-league-baseball-could-crack-11-billion-in-revenues-in-2022/?sh=68098b77f63a

Carig, M. (2018, Sep 27). *The Yankees have paired the best bullpen ever with cutting-edge tools to optimize its usage. Will it be enough in the playoffs?* The Athletic. https://theathletic.com/552336/2018/09/27/the-yankees-have-paired-the-best-bullpen-ever-with-cutting-edge-tools-to-optimize-its-usage-will-it-be-enough-in-the-playoffs/?redirected=1

Castrovince, A. (2019, Feb 3). *The influence of WAR on modern front offices*. MLB.com. https://www.mlb.com/news/war-embraced-by-mlb-front-offices-c303484670

Castrovince, A. (2020). *A fan's guide to baseball analytics*. Sports Publishing.

Common, D. (2014, Aug 18). *How the defensive shift and big data are changing the game*. CBC. https://www.cbc.ca/news/world/how-the-defensive-shift-and-big-data-are-changing-baseball-1.2739619

Davidoff, K. (2019, Nov 19). *Marcell Ozuna among players smartly embracing analytics in MLB free agency.* New York Post. https://nypost.com/2019/11/19/marcell-ozuna-among-players-smartly-embracing-analytics-in-mlb-free-agency/

Hakes, J.K., & Turner, C. (2011). *Pay, productivity and aging in Major League Baseball*. *Journal of Productivity Analysis, 35*(1), 61-74. https://www.jstor.org/stable/23883797

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). *An Introduction to Statistical Learning with Applications in R*. Springer.

MLB. (2022). *Expected Weighted On-base Average (xwOBA)*. https://www.mlb.com/glossary/statcast/expected-woba

Marchi, M., Albert, J., & Baumer, B.S. (2019). *Analyzing baseball data with R*. CRC Press.

Ng, K. (2017). Analyzing major league baseball player's performance based on age and experience. *Journal of Sports Economics & Management, 7*(2). 78-100. http://sportsem.uv.es/j_sports_and_em/index.php/JSEM/article/view/66

Ozanian, M. & Teitelbaum, J. (2022, Mar 24). *Baseball's most valuable teams 2022: Yankees hit $6 billion as new CBA creates new revenue streams*. Forbes. https://www.forbes.com/sites/mikeozanian/2022/03/24/baseballs-most-valuable-teams-2022-yankees-hit-6-billion-as-new-cba-creates-new-revenue-streams/?sh=244684d600a2

Raschka, S. & Mirjalili, V. (2019). *Python Machine Learning*. Packt.

Schrage, M. (2019, July 15). *What baseball can teach you about using data to improve yourself*. Harvard Business Review. https://hbr.org/2019/07/what-baseball-can-teach-you-about-using-data-to-improve-yourself

Schulz, R., Musa, D., Staszewski, J., & Seigler, R.S. (1994). The relationship between age and major league baseball performance: Implications for development. *Psychology and Aging, 9*(2). 274-286. https://doi.org/10.1037/0882-7974.9.2.274

Spotrac. (2022). *MLB team payroll tracker*. https://www.spotrac.com/mlb/payroll/2022/?ref=trending-pages

Sun, HC., Lin, TY., & Tsai, YL. (2023). Performance prediction in major league baseball by long short-term memory networks. *International Journal of Data Science and Analytics, 15*, 93-104. https://arxiv.org/pdf/2206.09654.pdf

Weinberg, N. (2014, Sept 19). *Calculating position player WAR, A complete example*. FanGraphs. https://library.fangraphs.com/calculating-position-player-war-a-complete-example/

Weinberg, N. (2017, Apr 17). *Calculating pitcher WAR, A complete example*. FanGraphs. https://library.fangraphs.com/calculating-pitcher-war-a-complete-example/

Appendix A

Relevant links to code and output on Github:

- Peak Ages by Position – Batters and Pitchers
- Exceed 2022 WAR – Batters – Data Clean and Feature Engineering
- Exceed 2022 WAR – Batters – Correlation and Results
- Exceed 2022 WAR – Pitchers – Data Clean and Feature Engineering
- Exceed 2022 WAR – Batters – Correlation and Results
- Stepwise Linear Regression – Batters
- Lasso Regression - Batters

Appendix B

Table B1

Batter features for binary classification

Feature	Description	Type	Source
Rk	Player's season rank	Integer	Stathead Baseball
Player	Player name	String	Stathead Baseball
WAR	Player's Wins Above Replacement	Float	Stathead Baseball
Season	Year played	Integer	Stathead Baseball
Age	Age of player	Integer	Stathead Baseball
Team	Team of player	String	Stathead Baseball
Lg	League of player	String	Stathead Baseball
G	# Games played	Integer	Stathead Baseball
PA	# Plate appearances	Integer	Stathead Baseball
AB	# At-bats	Integer	Stathead Baseball
R	# Runs	Integer	Stathead Baseball
H	# Hits	Integer	Stathead Baseball
1B	# Singles	Integer	Stathead Baseball
2B	# Doubles	Integer	Stathead Baseball
3B	# Triples	Integer	Stathead Baseball
HR	# Home runs	Integer	Stathead Baseball
RBI	# Runs batted in	Integer	Stathead Baseball
SB	# Stolen bases	Integer	Stathead Baseball
CS	# Caught stealing	Integer	Stathead Baseball
BB	# Walks	Integer	Stathead Baseball
SO	# Strikeouts	Integer	Stathead Baseball
OPS+	On-base plus slugging percentages adjusted for ballpark	Float	Stathead Baseball
TB	# Total bases	Integer	Stathead Baseball
GIDP	# Grounded into double play	Integer	Stathead Baseball
HBP	# Hit by pitch	Integer	Stathead Baseball
SH	# Sacrifice hits	Integer	Stathead Baseball
SF	# Sacrifice flies	Integer	Stathead Baseball
IBB	# Intentional walks	Integer	Stathead Baseball
WAA	Wins added by the player above average	Float	Stathead Baseball
oWAR	Offensive Wins Above Replacement	Float	Stathead Baseball
dWAR	Defensive Wins Above Replacement	Float	Stathead Baseball
Rbat	# Runs better or worse than average the player was as a hitter	Integer	Stathead Baseball
Rdp	# Runs better or worse than average the player was at avoiding grounding into double play	Integer	Stathead Baseball

Rbaser	# Runs better or worse than average the player was for all fielding	Integer	Stathead Baseball
Rbaser + Rdp_x	# Runs better or worse than average the player was for all fielding plus # Runs better or worse than average the player was at avoiding grounding into double play	Integer	Stathead Baseball
Rfield	# Runs better or worse than average the player was for all fielding	Integer	Stathead Baseball
FG_AVG	Batting average	Float	Fangraphs
FG_BABIP	Batting average on balls in play	Float	Fangraphs
FG_Bat	Batting runs above average	Float	Fangraphs
FG_BB/K	Walks per strikeout ratio	Float	Fangraphs
FG_BB%	Walk percentage	Float	Fangraphs
FG_BsR	Base Running turns stolen bases, caught stealing, and other base running plays into runs above and below average	Float	Fangraphs
FG_ISO	Isolated power – extra bases per at bat	Float	Fangraphs
FG_K%	Strikeout percentage	Float	Fangraphs
FG_L-WAR	Legacy WAR calculation	Float	Fangraphs
FG_Def	Defense Runs Above Average	Float	Fangraphs
FG_Events	Calculated batted balls	Integer	Fangraphs
FG_playerid	Player ID	Integer	Fangraphs
FG_SLG	Slugging percentage	Float	Fangraphs
FG_Spd	Speed score	Float	Fangraphs
FG_WAR	Fangraphs' Wins Above Replacement	Float	Fangraphs
FG_wOBA	Weighted On Base Average	Float	Fangraphs
FG_wRAA	Weighted Runs Above Average	Float	Fangraphs
FG_wRC	Weighted Runs Created	Float	Fangraphs
FG_wSB	Weighted stolen bases	Float	Fangraphs
FG_OBP	On-base Percentage	Float	Fangraphs
FG_OPS	On-base Plus Slugging Percentage	Float	Fangraphs
Prev_WAR	Player's previous season WAR	Float	Engineered
Next_WAR	Player's next season WAR	Float	Engineered
Peak_Rk	Player's season rank from peak season		Stathead Baseball
Peak_WAR	Player's Wins Above Replacement from peak season	Float	Stathead Baseball
Peak_Season	Year played from peak season	Integer	Stathead Baseball
Peak_Age	Age of player from peak season	Integer	Stathead Baseball
Peak_Team	Team of player from peak season	String	Stathead Baseball
Peak_Lg	League of player from peak season	String	Stathead Baseball
Peak_G	# Games played from peak season	Integer	Stathead Baseball

Peak_PA	# Plate appearances from peak season	Integer	Stathead Baseball
Peak_AB	# At-bats from peak season	Integer	Stathead Baseball
Peak_R	# Runs from peak season	Integer	Stathead Baseball
Peak_H	# Hits from peak season	Integer	Stathead Baseball
Peak_1B	# Singles from peak season	Integer	Stathead Baseball
Peak_2B	# Doubles from peak season	Integer	Stathead Baseball
Peak_3B	# Triples from peak season	Integer	Stathead Baseball
Peak_HR	# Home runs from peak season	Integer	Stathead Baseball
Peak_RBI	# Runs batted in from peak season	Integer	Stathead Baseball
Peak_SB	# Stolen bases from peak season	Integer	Stathead Baseball
Peak_CS	# Caught stealing from peak season	Integer	Stathead Baseball
Peak_BB	# Walks from peak season	Integer	Stathead Baseball
Peak_SO	# Strikeouts from peak season	Integer	Stathead Baseball
Peak_OPS+	On-base plus slugging percentages adjusted for ballpark from peak season	Float	Stathead Baseball
Peak_TB	# Total bases from peak season	Integer	Stathead Baseball
Peak_GIDP	# Grounded into double play from peak season	Integer	Stathead Baseball
Peak_HBP	# Hit by pitch from peak season	Integer	Stathead Baseball
Peak_SH	# Sacrifice hits from peak season	Integer	Stathead Baseball
Peak_SF	# Sacrifice flies from peak season	Integer	Stathead Baseball
Peak_IBB	# Intentional walks from peak season from peak season	Integer	Stathead Baseball
Peak_WAA	Wins added by the player above average from peak season	Float	Stathead Baseball
Peak_oWAR	Offensive Wins Above Replacement from peak season	Float	Stathead Baseball
Peak_dWAR	Defensive Wins Above Replacement from peak season	Float	Stathead Baseball
Peak_Rbat	# Runs better or worse than average the player was as a hitter from peak season	Integer	Stathead Baseball
Peak_Rdp	# Runs better or worse than average the player was at avoiding grounding into double play from peak season	Integer	Stathead Baseball
Peak_Rbaser	# Runs better or worse than average the player was for all fielding from peak season	Integer	Stathead Baseball
Rbaser + Rdp_y	# Runs better or worse than average the player was for all fielding plus # Runs better or worse than average the player was at avoiding	Integer	Stathead Baseball

	grounding into double play from peak season		
Peak_Rfield	# Runs better or worse than average the player was for all fielding from peak season	Integer	Stathead Baseball
Peak_FG_AVG	Batting average from peak season	Float	Stathead Baseball
Peak_FG_BABIP	Batting average on balls in play from peak season	Float	Fangraphs
Peak_FG_Bat	Batting runs above average from peak season	Float	Fangraphs
Peak_FG_BB/K	Walks per strikeout ratio from peak season	Float	Fangraphs
Peak_FG_BB%	Walk percentage from peak season	Float	Fangraphs
Peak_FG_BsR	Base Running turns stolen bases, caught stealing, and other base running plays into runs above and below average from peak season	Float	Fangraphs
Peak_FG_ISO	Isolated power – extra bases per at bat from peak season	Float	Fangraphs
Peak_FG_K%	Strikeout percentage from peak season	Float	Fangraphs
Peak_FG_L-WAR	Legacy WAR calculation from peak season	Float	Fangraphs
Peak_FG_Def	Defense Runs Above Average from peak season	Float	Fangraphs
Peak_FG_Events	Calculated batted balls from peak season	Integer	Fangraphs
Peak_FG_SLG	Slugging percentage from peak season	Float	Fangraphs
Peak_FG_Spd	Speed score from peak season	Float	Fangraphs
Peak_FG_WAR	Fangraphs' Wins Above Replacement from peak season	Float	Fangraphs
Peak_FG_wOBA	Weighted On Base Average from peak season	Float	Fangraphs
Peak_FG_wRAA	Weighted Runs Above Average from peak season	Float	Fangraphs
Peak_FG_wRC	Weighted Runs Created from peak season	Float	Fangraphs
Peak_FG_wSB	Weighted stolen bases from peak season	Float	Fangraphs
Peak_FG_OBP	On-base Percentage from peak season	Float	Fangraphs
Peak_FG_OPS	On-base Plus Slugging Percentage from peak season	Float	Fangraphs
Peak_Prev_WAR	Player's previous season WAR from peak season	Float	Engineered

Peak_Next_WAR	Player's next season WAR from peak season	Float	Engineered
Peak_Pos_Cat	Player's position from peak season; C = 0, 1B = 1, 2B = 2, 3B = 3, SS = 4, OF = 5, DH = 6	Integer	Engineered
Exceed_Prev_WAR	Did the player exceed previous season's WAR? 0-No, 1-Yes	0/1	Engineered
Team_WL	Player's team's win-loss percentage for the season	Float	Engineered
Games_played	Percentage of games the player played in the season	Float	Engineered

Table B2

Batter features for binary classification post correlation analysis

Feature	Description	Type	Source
WAR	Player's Wins Above Replacement	Float	Stathead Baseball
Season	Year played	Integer	Stathead Baseball
Age	Age of player	Integer	Stathead Baseball
Team	Team of player	String	Stathead Baseball
Lg	League of player	String	Stathead Baseball
3B	# Triples	Integer	Stathead Baseball
HR	# Home runs	Integer	Stathead Baseball
SB	# Stolen bases	Integer	Stathead Baseball
CS	# Caught stealing	Integer	Stathead Baseball
BB	# Walks	Integer	Stathead Baseball
OPS+	On-base plus slugging percentages adjusted for ballpark	Float	Stathead Baseball
GIDP	# Grounded into double play	Integer	Stathead Baseball
HBP	# Hit by pitch	Integer	Stathead Baseball
SH	# Sacrifice hits	Integer	Stathead Baseball
SF	# Sacrifice flies	Integer	Stathead Baseball
IBB	# Intentional walks	Integer	Stathead Baseball
Rdp	# Runs better or worse than average the player was at avoiding grounding into double play	Integer	Stathead Baseball
Rbaser	# Runs better or worse than average the player was for all fielding	Integer	Stathead Baseball
Rfield	# Runs better or worse than average the player was for all fielding	Integer	Stathead Baseball
FG_BABIP	Batting average on balls in play	Float	Fangraphs
FG_BB/K	Walks per strikeout ratio	Float	Fangraphs
FG_BB%	Walk percentage	Float	Fangraphs
FG_BsR	Base Running turns stolen bases, caught stealing, and other base running plays into runs above and below average	Float	Fangraphs
FG_ISO	Isolated power – extra bases per at bat	Float	Fangraphs
FG_K%	Strikeout percentage	Float	Fangraphs
FG_Def	Defense Runs Above Average	Float	Fangraphs
FG_Events	Calculated batted balls	Integer	Fangraphs
FG_playerid	Player ID	Integer	Fangraphs
FG_Spd	Speed score	Float	Fangraphs
FG_wSB	Weighted stolen bases	Float	Fangraphs
Prev_WAR	Player's previous season WAR	Float	Engineered

Peak_Rk	Player's season rank from peak season		Stathead Baseball
Peak_WAR	Player's Wins Above Replacement from peak season	Float	Stathead Baseball
Peak_Age	Age of player from peak season	Integer	Stathead Baseball
Peak_Team	Team of player from peak season	String	Stathead Baseball
Peak_Lg	League of player from peak season	String	Stathead Baseball
Peak_3B	# Triples from peak season	Integer	Stathead Baseball
Peak_HR	# Home runs from peak season	Integer	Stathead Baseball
Peak_SB	# Stolen bases from peak season	Integer	Stathead Baseball
Peak_CS	# Caught stealing from peak season	Integer	Stathead Baseball
Peak_BB	# Walks from peak season	Integer	Stathead Baseball
Peak_OPS+	On-base plus slugging percentages adjusted for ballpark from peak season	Float	Stathead Baseball
Peak_GIDP	# Grounded into double play from peak season	Integer	Stathead Baseball
Peak_HBP	# Hit by pitch from peak season	Integer	Stathead Baseball
Peak_SH	# Sacrifice hits from peak season	Integer	Stathead Baseball
Peak_SF	# Sacrifice flies from peak season	Integer	Stathead Baseball
Peak_IBB	# Intentional walks from peak season from peak season	Integer	Stathead Baseball
Peak_Rdp	# Runs better or worse than average the player was at avoiding grounding into double play from peak season	Integer	Stathead Baseball
Peak_Rfield	# Runs better or worse than average the player was for all fielding from peak season	Integer	Stathead Baseball
Peak_FG_BABIP	Batting average on balls in play from peak season	Float	Fangraphs
Peak_FG_Bat	Batting runs above average from peak season	Float	Fangraphs
Peak_FG_BB/K	Walks per strikeout ratio from peak season	Float	Fangraphs
Peak_FG_BB%	Walk percentage from peak season	Float	Fangraphs
Peak_FG_BsR	Base Running turns stolen bases, caught stealing, and other base running plays into runs above and below average from peak season	Float	Fangraphs
Peak_FG_K%	Strikeout percentage from peak season	Float	Fangraphs
Peak_FG_Def	Defense Runs Above Average from peak season	Float	Fangraphs
Peak_FG_Events	Calculated batted balls from peak season	Integer	Fangraphs

Peak_FG_Spd	Speed score from peak season	Float	Fangraphs
Peak_FG_wRAA	Weighted Runs Above Average from peak season	Float	Fangraphs
Peak_FG_wSB	Weighted stolen bases from peak season	Float	Fangraphs
Peak_Prev_WAR	Player's previous season WAR from peak season	Float	Engineered
Peak_Pos_Cat	Player's position from peak season	Integer	Engineered
Exceed_Prev_WAR	Did the player exceed previous season's WAR? 0-No, 1-Yes	0/1	Engineered
Team_WL	Player's team's win-loss percentage for the season	Float	Engineered
Games_played	Percentage of games the player played in the season	Float	Engineered

Table B3

Pitcher features for binary classification

Feature	Description	Type	Source
Rk	Player's season rank	Integer	Stathead Baseball
Player	Player name	String	Stathead Baseball
WAR	Wins Above Replacement	Float	Stathead Baseball
GS	Games Started	Integer	Stathead Baseball
Season	Season year	Integer	Stathead Baseball
Age	Player's age in a specific season	Integer	Stathead Baseball
Team	Player's team in a specific season	Integer	Stathead Baseball
Lg	League the player played in	Integer	Stathead Baseball
W	# Wins	Integer	Stathead Baseball
L	# Losses	Integer	Stathead Baseball
W-L%	Win-Loss Percentage	Float	Stathead Baseball
Dec	# Decisions	Integer	Stathead Baseball
ERA	Earned Run Average	Integer	Stathead Baseball
G	# Games	Integer	Stathead Baseball
CG	# Complete Games	Integer	Stathead Baseball
SHO	# Shutouts	Integer	Stathead Baseball
SV	# Saves	Integer	Stathead Baseball
IP	# Innings pitched	Float	Stathead Baseball
H	# Hits allowed	Integer	Stathead Baseball
R	# Runs allowed	Integer	Stathead Baseball
ER	# Earned runs allowed	Integer	Stathead Baseball
HR	# Home runs allowed	Integer	Stathead Baseball
BB	# Walks allowed	Integer	Stathead Baseball
IBB	# Intentional walks allowed	Integer	Stathead Baseball
SO	# Strikeouts	Integer	Stathead Baseball
HBP	# Batters hit	Integer	Stathead Baseball
BK	# Balks	Integer	Stathead Baseball
WP	# Wild pitches	Integer	Stathead Baseball
BF	# Batters faced	Integer	Stathead Baseball
ERA+	Earned run average adjusted to ballparks	Integer	Stathead Baseball
FIP	Fielding independent pitching	Float	Stathead Baseball
WHIP	Walks plus hits divided by innings pitched	Float	Stathead Baseball
H9	Hits times 9 divided by innings pitched	Float	Stathead Baseball
HR9	Home runs times 9 divided by innings pitched	Float	Stathead Baseball
BB9	Walks times 9 divided by innings pitched	Float	Stathead Baseball

SO9	Strikeouts times 9 divided by innings pitched	Float	Stathead Baseball
SO/BB	Strikeouts per walks	Float	Stathead Baseball
WAA	Wins above the average player	Float	Stathead Baseball
FG_BABIP	Batting average allowed on balls in play	Float	Fangraphs
FG_LOB%	Left on-base percentage	Float	Fangraphs
FG_WAR	Wins Above Replacement	Float	Fangraphs
FG_RA9-WAR	Runs allowed based Wins Above Replacement	Float	Fangraphs
FG_BIP-Wins	Batting average allowed on balls in play Wins - # Wins a pitcher has added	Float	Fangraphs
FG_LOB-Wins	Left on-base Wins added as a result of stranding runners	Float	Fangraphs
FG_FDP-Wins	Fielding dependent wins; sum of BIP-Wins and LOB-Wins	Float	Fangraphs
FG_K-BB%	Strikeouts less walks percentage	Float	Fangraphs
FG_kwERA	ERA estimator based on strikeouts and walks	Float	Fangraphs
FG_TTO%	Walks, strikeouts, homeruns percentage	Float	Fangraphs
FG_AVG+	Batting average allowed (ballpark effects)	Integer	Fangraphs
FG_K%+	Strikeout rate (ballpark effects)	Integer	Fangraphs
FG_BB%+	Walk rate (ballpark effects)	Integer	Fangraphs
FG_Events	Calculated batted balls	Integer	Fangraphs
FG_WPA	Win probability added	Float	Fangraphs
FG_-WPA	Negative win probability added	Float	Fangraphs
FG_+WPA	Positive win probability added	Float	Fangraphs
FG_RE24	Run expectancy 24 base out state	Float	Fangraphs
FG_REW	Run expectancy wins	Float	Fangraphs
FG_pLI	Average leverage index	Float	Fangraphs
FG_inLI	Inning leverage index	Float	Fangraphs
FG_gmLI	Game leverage index	Float	Fangraphs
FG_Pulls	# Times pitcher has been removed from a game	Integer	Fangraphs
FG_WPA/LI	Situational wins	Float	Fangraphs
FG_Clutch	Clutch score	Float	Fangraphs
FG_SD	Shutdowns (relief)	Integer	Fangraphs
FG_MD	Meltdowns (relief)	Integer	Fangraphs
FG_ERA-	ERA (ballpark effects and league average)	Integer	Fangraphs
FG_FIP-	Fielding independent pitching minus (ballpark effects and league average)	Integer	Fangraphs

FG_K%	Strikeout percentage	Float	Fangraphs
FG_BB%	Walk percentage	Float	Fangraphs
FG_E-F	ERA and FIP differential	Float	Fangraphs
FG_playerid	Unique player ID	Integer	Fangraphs
Next_WAR	Player's next season WAR	Float	Engineered
Prev_WAR	Player's previous season WAR	Float	Engineered
Peak_Rk	Player's season rank from peak season	Integer	Stathead Baseball
Peak_WAR	Wins Above Replacement from peak season	Float	Stathead Baseball
Peak_GS	Games Started from peak season	Integer	Stathead Baseball
Peak_Season	Season year from peak season	Integer	Stathead Baseball
Peak_Age	Player's age in peak season	Integer	Stathead Baseball
Peak_Team	Player's team in a peak season	Integer	Stathead Baseball
Peak_Lg	League the player played in from peak season	Integer	Stathead Baseball
Peak_W	# Wins from peak season	Integer	Stathead Baseball
Peak_L	# Losses from peak season	Integer	Stathead Baseball
Peak_W-L%	Win-Loss Percentage from peak season	Float	Stathead Baseball
Peak_Dec	# Decisions from peak season	Integer	Stathead Baseball
Peak_ERA	Earned Run Average from peak season	Integer	Stathead Baseball
Peak_G	# Games from peak season	Integer	Stathead Baseball
Peak_CG	# Complete Games from peak season	Integer	Stathead Baseball
Peak_SHO	# Shutouts from peak season	Integer	Stathead Baseball
Peak_SV	# Saves from peak season	Integer	Stathead Baseball
Peak_IP	# Innings pitched from peak season	Float	Stathead Baseball
Peak_H	# Hits allowed from peak season	Integer	Stathead Baseball
Peak_R	# Runs allowed from peak season	Integer	Stathead Baseball
Peak_ER	# Earned runs allowed from peak season	Integer	Stathead Baseball
Peak_HR	# Home runs allowed from peak season	Integer	Stathead Baseball
Peak_BB	# Walks allowed from peak season	Integer	Stathead Baseball
Peak_IBB	# Intentional walks allowed from peak season	Integer	Stathead Baseball
Peak_SO	# Strikeouts from peak season	Integer	Stathead Baseball
Peak_HBP	# Batters hit from peak season	Integer	Stathead Baseball
Peak_BK	# Balks from peak season	Integer	Stathead Baseball
Peak_WP	# Wild pitches from peak season	Integer	Stathead Baseball
Peak_BF	# Batters faced from peak season	Integer	Stathead Baseball
Peak_ERA+	Earned run average adjusted to ballparks from peak season	Integer	Stathead Baseball

Peak_FIP	Fielding independent pitching from peak season	Float	Stathead Baseball
Peak_WHIP	Walks plus hits divided by innings pitched from peak season	Float	Stathead Baseball
Peak_H9	Hits times 9 divided by innings pitched from peak season	Float	Stathead Baseball
Peak_HR9	Home runs times 9 divided by innings pitched from peak season	Float	Stathead Baseball
Peak_BB9	Walks times 9 divided by innings pitched from peak season	Float	Stathead Baseball
Peak_SO9	Strikeouts times 9 divided by innings pitched from peak season	Float	Stathead Baseball
Peak_SO/BB	Strikeouts per walks from peak season	Float	Stathead Baseball
Peak_WAA	Wins above the average player from peak season	Float	Stathead Baseball
Peak_FG_BABIP	Batting average allowed on balls in play from peak season	Float	Fangraphs
Peak_FG_LOB%	Left on-base percentage from peak season	Float	Fangraphs
Peak_FG_WAR	Wins Above Replacement from peak season	Float	Fangraphs
Peak_FG_RA9-WAR	Runs allowed based Wins Above Replacement from peak season	Float	Fangraphs
Peak_FG_BIP-Wins	Batting average allowed on balls in play Wins - # Wins a pitcher has added from peak season	Float	Fangraphs
Peak_FG_LOB-Wins	Left on-base Wins added as a result of stranding runners from peak season	Float	Fangraphs
Peak_FG_FDP-Wins	Fielding dependent wins; sum of BIP-Wins and LOB-Wins from peak season	Float	Fangraphs
Peak_FG_K-BB%	Strikeouts less walks percentage from peak season	Float	Fangraphs
Peak_FG_kwERA	ERA estimator based on strikeouts and walks from peak season	Float	Fangraphs
Peak_FG_TTO%	Walks, strikeouts, homeruns percentage from peak season	Float	Fangraphs
Peak_FG_AVG+	Batting average allowed (ballpark effects) from peak season	Integer	Fangraphs
Peak_FG_K%+	Strikeout rate (ballpark effects) from peak season	Integer	Fangraphs
Peak_FG_BB%+	Walk rate (ballpark effects) from peak season	Integer	Fangraphs

Peak_FG_Events	Calculated batted balls from peak season	Integer	Fangraphs
Peak_FG_WPA	Win probability added from peak season	Float	Fangraphs
Peak_FG_-WPA	Negative win probability added from peak season	Float	Fangraphs
Peak_FG_+WPA	Positive win probability added from peak season	Float	Fangraphs
Peak_FG_RE24	Run expectancy 24 base out state from peak season	Float	Fangraphs
Peak_FG_REW	Run expectancy wins from peak season	Float	Fangraphs
Peak_FG_pLI	Average leverage index from peak season	Float	Fangraphs
Peak_FG_inLI	Inning leverage index from peak season	Float	Fangraphs
Peak_FG_gmLI	Game leverage index from peak season	Float	Fangraphs
Peak_FG_Pulls	# Times pitcher has been removed from a game from peak season	Integer	Fangraphs
Peak_FG_WPA/LI	Situational wins from peak season	Float	Fangraphs
Peak_FG_Clutch	Clutch score from peak season	Float	Fangraphs
Peak_FG_SD	Shutdowns (relief) from peak season	Integer	Fangraphs
Peak_FG_MD	Meltdowns (relief) from peak season	Integer	Fangraphs
Peak_FG_ERA-	ERA (ballpark effects and league average) from peak season	Integer	Fangraphs
Peak_FG_FIP-	Fielding independent pitching minus (ballpark effects and league average) from peak season	Integer	Fangraphs
Peak_FG_K%	Strikeout percentage from peak season	Float	Fangraphs
Peak_FG_BB%	Walk percentage from peak season	Float	Fangraphs
Peak_FG_E-F	ERA and FIP differential from peak season	Float	Fangraphs
Peak_Next_WAR	Player's next season WAR from peak season	Float	Engineered
Peak_Prev_WAR	Player's previous season WAR from peak season	Float	Engineered
Exceed_Prev_WAR	Did the player exceed previous season's WAR? 0-No, 1-Yes	0/1	Engineered
Games_played	Percentage of games the player played in the season	Float	Engineered
Team_WL	Player's team's win-loss percentage for the season	Float	Engineered

Table B4

Pitcher features for binary classification post correlation analysis

Feature	Description	Type	Source
WAR	Wins Above Replacement	Float	Stathead Baseball
Season	Season year	Integer	Stathead Baseball
Age	Player's age in a specific season	Integer	Stathead Baseball
Team	Player's team in a specific season	Integer	Stathead Baseball
Lg	League the player played in	Integer	Stathead Baseball
W-L%	Win-Loss Percentage	Float	Stathead Baseball
G	# Games	Integer	Stathead Baseball
CG	# Complete Games	Integer	Stathead Baseball
SHO	# Shutouts	Integer	Stathead Baseball
SV	# Saves	Integer	Stathead Baseball
IBB	# Intentional walks allowed	Integer	Stathead Baseball
HBP	# Batters hit	Integer	Stathead Baseball
BK	# Balks	Integer	Stathead Baseball
WP	# Wild pitches	Integer	Stathead Baseball
ERA+	Earned run average adjusted to ballparks	Integer	Stathead Baseball
FIP	Fielding independent pitching	Float	Stathead Baseball
SO/BB	Strikeouts per walks	Float	Stathead Baseball
FG_LOB%	Left on-base percentage	Float	Fangraphs
FG_BIP-Wins	Batting average allowed on balls in play Wins - # Wins a pitcher has added	Float	Fangraphs
FG_LOB-Wins	Left on-base Wins added as a result of stranding runners	Float	Fangraphs
FG_FDP-Wins	Fielding dependent wins; sum of BIP-Wins and LOB-Wins	Float	Fangraphs
FG_kwERA	ERA estimator based on strikeouts and walks	Float	Fangraphs
FG_TTO%	Walks, strikeouts, homeruns percentage	Float	Fangraphs
FG_AVG+	Batting average allowed (ballpark effects)	Integer	Fangraphs
FG_K%+	Strikeout rate (ballpark effects)	Integer	Fangraphs
FG_Events	Calculated batted balls	Integer	Fangraphs
FG_pLI	Average leverage index	Float	Fangraphs
FG_inLI	Inning leverage index	Float	Fangraphs
FG_gmLI	Game leverage index	Float	Fangraphs
FG_Clutch	Clutch score	Float	Fangraphs
FG_FIP-	Fielding independent pitching minus (ballpark effects and league average)	Integer	Fangraphs

FG_BB%	Walk percentage	Float	Fangraphs
FG_E-F	ERA and FIP differential	Float	Fangraphs
FG_playerid	Unique player ID	Integer	Fangraphs
Prev_WAR	Player's previous season WAR	Float	Engineered
Peak_WAR	Wins Above Replacement from peak season	Float	Stathead Baseball
Peak_Team	Player's team in a peak season	Integer	Stathead Baseball
Peak_Lg	League the player played in from peak season	Integer	Stathead Baseball
Peak_W-L%	Win-Loss Percentage from peak season	Float	Stathead Baseball
Peak_G	# Games from peak season	Integer	Stathead Baseball
Peak_CG	# Complete Games from peak season	Integer	Stathead Baseball
Peak_SHO	# Shutouts from peak season	Integer	Stathead Baseball
Peak_SV	# Saves from peak season	Integer	Stathead Baseball
Peak_IBB	# Intentional walks allowed from peak season	Integer	Stathead Baseball
Peak_HBP	# Batters hit from peak season	Integer	Stathead Baseball
Peak_BK	# Balks from peak season	Integer	Stathead Baseball
Peak_WP	# Wild pitches from peak season	Integer	Stathead Baseball
Peak_ERA+	Earned run average adjusted to ballparks from peak season	Integer	Stathead Baseball
Peak_FIP	Fielding independent pitching from peak season	Float	Stathead Baseball
Peak_SO9	Strikeouts times 9 divided by innings pitched from peak season	Float	Stathead Baseball
Peak_SO/BB	Strikeouts per walks from peak season	Float	Stathead Baseball
Peak_FG_BABIP	Batting average allowed on balls in play from peak season	Float	Fangraphs
Peak_FG_LOB%	Left on-base percentage from peak season	Float	Fangraphs
Peak_FG_BIP-Wins	Batting average allowed on balls in play Wins - # Wins a pitcher has added from peak season	Float	Fangraphs
Peak_FG_LOB-Wins	Left on-base Wins added as a result of stranding runners from peak season	Float	Fangraphs
Peak_FG_FDP-Wins	Fielding dependent wins; sum of BIP-Wins and LOB-Wins from peak season	Float	Fangraphs
Peak_FG_kwERA	ERA estimator based on strikeouts and walks from peak season	Float	Fangraphs
Peak_FG_Events	Calculated batted balls from peak season	Integer	Fangraphs

Peak_FG_pLI	Average leverage index from peak season	Float	Fangraphs
Peak_FG_inLI	Inning leverage index from peak season	Float	Fangraphs
Peak_FG_gmLI	Game leverage index from peak season	Float	Fangraphs
Peak_FG_Pulls	# Times pitcher has been removed from a game from peak season	Integer	Fangraphs
Peak_FG_Clutch	Clutch score from peak season	Float	Fangraphs
Peak_FG_FIP-	Fielding independent pitching minus (ballpark effects and league average) from peak season	Integer	Fangraphs
Peak_FG_BB%	Walk percentage from peak season	Float	Fangraphs
Peak_Prev_WAR	Player's previous season WAR from peak season	Float	Engineered
Exceed_Prev_WAR	Did the player exceed previous season's WAR? 0-No, 1-Yes	0/1	Engineered
Games_played	Percentage of games the player played in the season	Float	Engineered
Team_WL	Player's team's win-loss percentage for the season	Float	Engineered

Table B5

Initial dataset for regression model comparison

Feature	Description	Type	Source
WAR	Player's Wins Above Replacement	Float	Stathead Baseball
Season	Year played	Integer	Stathead Baseball
Age	Age of player	Integer	Stathead Baseball
Team	Team of player	String	Stathead Baseball
Lg	League of player	String	Stathead Baseball
G	# Games played	Integer	Stathead Baseball
PA	# Plate appearances	Integer	Stathead Baseball
AB	# At-bats	Integer	Stathead Baseball
R	# Runs	Integer	Stathead Baseball
H	# Hits	Integer	Stathead Baseball
X1B	# Singles	Integer	Stathead Baseball
X2B	# Doubles	Integer	Stathead Baseball
X3B	# Triples	Integer	Stathead Baseball
HR	# Home runs	Integer	Stathead Baseball
RBI	# Runs batted in	Integer	Stathead Baseball
SB	# Stolen bases	Integer	Stathead Baseball
CS	# Caught stealing	Integer	Stathead Baseball
BB	# Walks	Integer	Stathead Baseball
SO	# Strikeouts	Integer	Stathead Baseball
OPS+	On-base plus slugging percentages adjusted for ballpark	Float	Stathead Baseball
TB	# Total bases	Integer	Stathead Baseball
GIDP	# Grounded into double play	Integer	Stathead Baseball
HBP	# Hit by pitch	Integer	Stathead Baseball
SH	# Sacrifice hits	Integer	Stathead Baseball
SF	# Sacrifice flies	Integer	Stathead Baseball
IBB	# Intentional walks	Integer	Stathead Baseball
Prev_WAR	Player's previous season WAR	Float	Engineered
Max_WAR	Player's maximum WAR value across career	Float	Engineered
Max_WAR_Age	Age at which the player had his max WAR value in his career	Float	Engineered
Pos_Cat	Player's position; C = 0, 1B = 1, 2B = 2, 3B = 3, SS = 4, OF = 5, DH = 6	Integer	Engineered
Season_WAR_Class	Places the player's WAR value in a category between 0 and 6	Integer	Engineered
Prev_WAR_Class	Places the player's previous WAR value in a category between 0 and 6	Integer	Engineered
Team_WL	Player's team's win-loss percentage for the season	Float	Engineered

Games_played	Percentage of games the player played in the season	Float	Engineered
war_season	WAR divided by the average WAR in a given season	Float	Engineered
player_season	Season number for a player's career (e.g., season 5 out of 10)	Integer	Engineered
war_corr	Correlation between the player's season number and his season WAR	Float	Engineered