Examining Major League Baseball Statcast Data to Identify Undervalued Players via Cluster Analysis and Principal Components

Jordan Rivera
University of California, Los Angeles

I. Introduction

Ever since the early 2000's, when the rise of sabermetrics and the Moneyball Era changed the game forever, Major League Baseball has become increasingly data driven. It started when general manager Billy Beane constructed the 2002 Oakland A's team with a strategy solely based on using analytics and data to evaluate players. This new, radical approach resulted in Oakland winning 103 games that season, the most in Major League Baseball, despite having the third lowest payroll of any team. In the twenty years since then, nearly every MLB team has adopted a similar strategy, hiring data analysts, statisticians, and analytically minded general managers, and examining new, advanced metrics in their efforts to evaluate players.

Thanks to new, cutting-edge technology such as Statcast, a tracking technology that was installed in every MLB stadium in 2015, teams have more data available to them than ever. Now, general managers have access to an overwhelmingly massive amount of baseball data that they can use to quantify the skills of the players they are evaluating. However, with this huge wealth of technology driven data becoming available, teams are only scratching the surface in terms of using these new metrics and analytics to their advantage. There is a lot of room for growth and exploration in this field.

The goal of this project is to tackle the same challenge that MLB general managers are facing: using these advanced statistics to evaluate players and make optimized decisions in order to build a winning team. The obvious goal is to build a team with as many high-value players as possible, but this is harder said than done. It's easy to identify the best players in the league, but the problem is that such players are coveted by every MLB team. These superstar players are expensive and could cost hundreds of millions of dollars in free agency. Acquiring these players via trade could be even more costly, as teams would likely have to trade away their own valuable players or high-ceiling prospects. The aim is therefore to identify players of value who are not as obviously skilled, players who are undervalued by the league. On the surface, these players are not considered great, and they might not have the same career success as star players, but their advanced statistics suggest that they are much better than how they are perceived. Such players most likely would not require as high a salary or as significant a trade return, so teams would be able to acquire them at low cost and low risk, making them ideal targets.

Therefore, the research question that we aim to answer is: Can we identify undervalued, inexpensive players who have similar statistical profiles as star players? Using cluster analysis,

we can split every players from the 2019 season into a set of clusters based on their advanced statistics. The goal is to find players who are in the same cluster as star players, who have comparable metrics yet are not as well-known and not as expensive. These players are ideal targets for acquisition, as they could be on the verge of becoming a star player. Furthermore, we can extend this research into developing a more general strategy for identifying undervalued players as a group – perhaps there are some statistics or metrics that are especially effective at finding these underrated players or predicting future stars.

Using principal components analysis, we aim to perform dimensional reduction and decrease the number of column variables in our data set. Our goal is to see how many principal components can explain 90% of the variability in the data, and whether we can interpret any latent variables represented. If we can detect any latent variables, these can be useful for analyzing differences between players and potentially identifying undervalued players.

Furthermore, using principal component regression, we aim to build a regression model that uses principal components to predict WAR (Wins Above Replacement) values for players. WAR is a metric produced from a statistical formula that measures a player's overall performance on the field. For example, a player with a WAR statistic of 5 has helped his team win approximately 5 more games than they would have with a replacement level player. Therefore, it is a good dependent variable for our research purposes. Building a model that could predict WAR output could be very useful for identifying valuable players. The final segment of this project will involve maximum likelihood estimation for WAR. We will fit a model to this dependent variable and calculate the MLE estimators of the model's parameters.

II. The Data Set

The data set that we are using for this analysis comes from Baseball Savant, an online database of MLB statistics. The database is massive and features thousands of different metrics and analytics used to measure professional baseball, including technology powered Statcast metrics. The data set we are using for this project focuses on player statistics, specifically batting statistics, from the 2019 MLB season. It was customizable, as we could choose which variables we wanted to include as well as a filter that could limit the number of observations. We chose to only include players who had a minimum of 100 plate appearances, as players with smaller

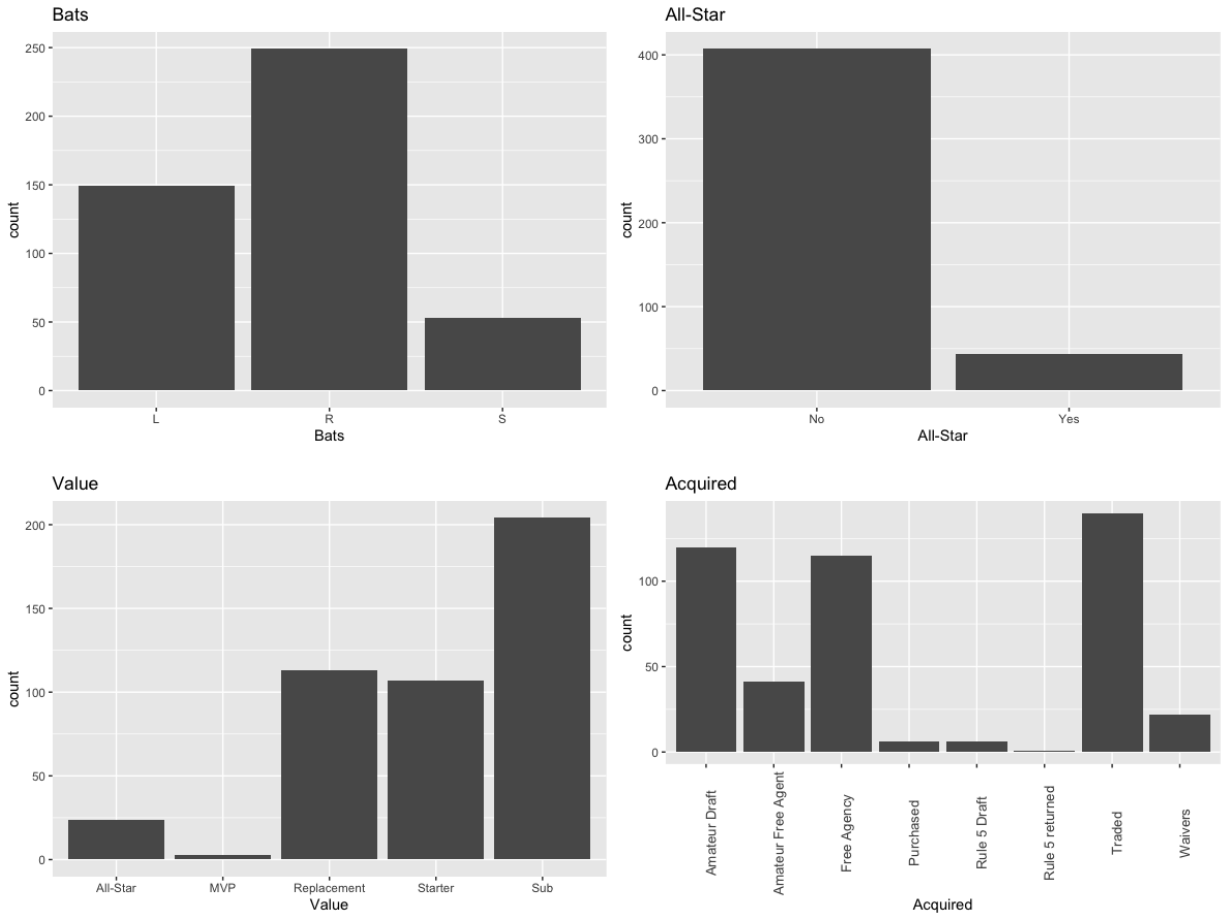
sample sizes of playing time could distort the results. There were 451 players who met this threshold, so there are 451 observations in our data set.

There are 52 total variables. Of these variables, 9 are categorical variables and 43 are quantitative variables. All 43 quantitative variables come from the original data set from Baseball Savant, although only 2 categorical variables (first name and last name) were included in that data set. We manually added 7 more categorical variables that we thought could be useful in examining the data. These variables were created using Baseball Savant and Baseball-Reference as the source. Table (1) provides descriptions and properties of the variables.

Table 1: Categorical Variables

Variable	**Description**
Last_name	Player's last name
First_name	Player's first name
Team	Player's team. If multiple, says number of teams (e.g., 2TM)
League	Player's league. Either National League or American League. If player played in both leagues, the designation is MLB
Bats	Side of the plate that the batter hits from
Position	Player's primary position
All-Star	Denotes whether the player was on the 2019 All-Star team
Acquired	How the player was acquired on his current team
Value	Estimates player's value based on their WAR statistic. There are 5 levels: MVP (8+), All-Star (5+), Starter (2+), Sub (0-2), Replacement (<0). This is a common statistic to measure player's production for the season

Figure 1: Barplots for Categorical Variables



Examining four of these variables closer, we see that there are more right-handed batters than left-handed batters and switch-hitters (players who can hit from both sides). The majority of players were not All-Stars in 2019, as only 44 players received that honor. This matches with the distribution across of the five values for the variable Value, where we see that there are few players playing at All-Star or MVP level. Finally, looking at the method of how these players were acquired, we see that most players were traded, drafted, and signed in free agency. Figure (1) displays bar plots for these variables. Table (2) provides descriptions and properties of the quantitative variables.

Table 2: Quantitative Variables

Variable	**Description**	**Mean**	**SD**	**Median**	**IQR**
Salary	Player's salary for the 2019 season	4217701	6143497	717500	4541500
WAR	Wins Above Replacement. Estimates the amount of wins the team gained by playing that player	1.366519	1.975868	0.9	2.45

	instead of a replacement-level player.				
Year	The year corresponding to the statistics. 2019 for all observations	2019	0	2019	0
Age	The player's age in years	28.416851	3.690117	28	5
AB	The player's number of at-bats	343.2772	156.3986	329	274.5
PA	The player's number of plate appearances	384.2772	176.2329	369	295.5
Hits	The player's number of hits	88.74945	46.91436	84	76
Doubles	The player's number of doubles	18.07761	10.51774	17	16
Triples	The player's number of triples	1.658537	1.928393	1	2
Home_runs	The player's number of home runs	14.58758	10.85862	12	15
Strikeouts	The player's number of strikeouts	84.58093	39.36563	82	58
Walks	The player's number of walks	33.53880	21.97888	29	28
K_percent	The percent of the player's plate appearances that end in a strikeout	22.974723	6.337741	22.8	8.2
BB_percent	The percent of the player's plate appearances that end in a walk	8.515078	3.135148	8.3	4.4
BA	The player's batting average, the percentage of at-bats that end in a hit	0.25029933	0.03688169	0.253	0.047
SLG	The player's slugging percentage, the average number of bases the player hits for per at-bat	0.42925277	0.08545524	0.425	0.114
OBP	The player's on-base percentage, the percent of plate appearances that end in the player reaching base	0.32040798	0.03985749	0.322	0.0485
OPS	The player's on-base plus slugging, the sum of SLG and OBP	0.7496497	0.1175807	0.744	0.149
ISO	The player's isolated power, a statistic that focuses on the player's tendency to hit for power. Calculated by subtracting hits from total bases and then dividing by at-bats	0.17891131	0.06355439	0.176	0.0875
RBI	Runs batted in, the number of runs driven in by the player while batting	47.73171	28.41301	43	43
CS	Caught stealing, the number of times the player was called out while attempting to steal a base	1.764967	2.084379	1	3
SB	The number of successful stolen bases	4.822616	6.876984	2	6
Games	The number of games played by the player	103.87140	36.90711	106	61
Runs	The number of runs scored by the player	49.64523	28.55806	45	42
xBA	The player's expected batting average based on the launch angle	0.24184479	0.02993397	0.244	0.042

	and exit velocity of each of their batted ball events				
xSLG	The player's expected slugging percentage based on the launch angle and exit velocity of each of their batted ball events	0.41337251	0.07870766	0.415	0.1065
wOBA	The player's weighted on-base average, a statistic similar to OBP but also factors in how he got on base	0.31729047	0.04419273	0.318	0.055
xwOBA	The player's expected weighted on-base average based on the launch angle and exit velocity of each of their batted ball events	0.31338581	0.04067658	0.314	0.055
xOBP	The player's expected on-base percentage based on the launch angle and exit velocity of each of their batted ball events	0.31481596	0.03647635	0.315	0.0455
xISO	The player's expected isolated power based on the launch angle and exit velocity of each of their batted ball events	0.17148559	0.06066324	0.170	0.082
Exit_velocity	The average exit velocity in miles per hour of the player's batted ball events	88.594235	2.202123	88.8	2.8
Launch_angle	The average launch angle of the player's batted ball events	12.860754	4.511219	13.0	6.3
Sweet_spot_percent	The percent of batted ball events that have a launch angle between 8 and 32 degrees, which is known as the launch angle sweet spot zone	33.413525	4.444124	33.5	6.0
Barrel_percent	The percent of batted ball events that are barrels, an event whose comparable hit types has led to a minimum .500 batting average and 1.500 slugging percentage	7.298004	4.135034	6.8	5.8
Hard_hit_percent	The percentage of batted ball events hit at least 95 mph	36.54745	7.87155	37.0	10.8
Zone_swing_percent	The percentage of pitches in the strike zone that the player swung at	67.633038	6.048856	67.8	8.05
OZ_swing_percent	The percentage of pitches outside of the strike zone that the player swung at	28.960310	6.509607	28.5	8.8
Whiff_percent	The percentage of pitches that the player swung and missed at	25.425499	6.181995	25.4	7.2
Swing_percent	The percentage of pitches that the player swung at	47.413525	5.238946	47.2	6.75

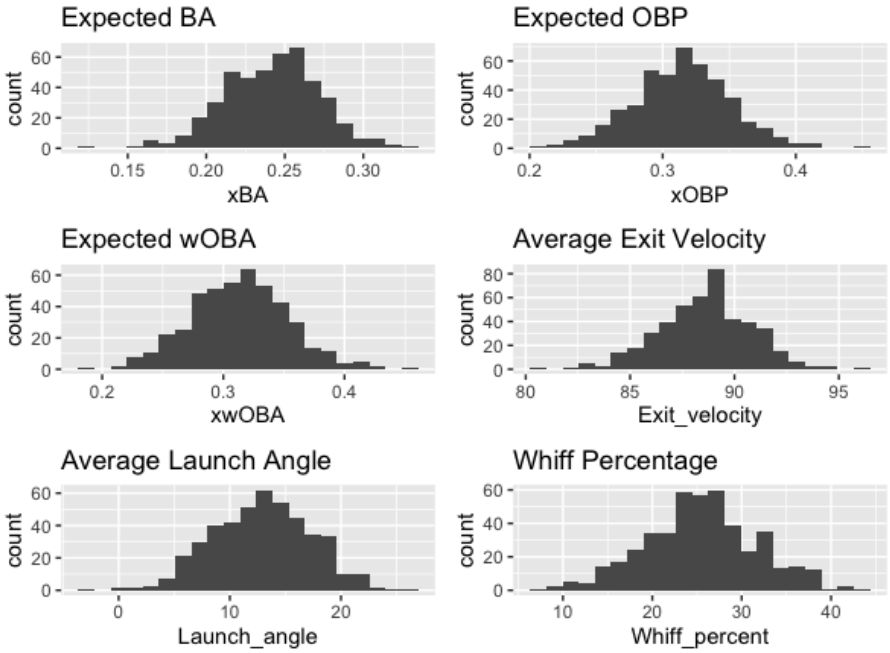
GB_percent	The percentage of batted balls that resulted in a ground ball	43.604878	6.980084	42.9	9.55
FB_percent	The percentage of batted balls that resulted in a fly ball	23.933038	5.142571	23.7	7.4
LD_percent	The percentage of batted balls that resulted in a line drive	24.995565	3.754515	25.1	4.6
Sprint_speed	The player's average speed when sprinting	27.038137	1.516578	27.1	2.0

There are zero missing values in this data set. Every observation has a value for each variable. Examining the variables, we see that many of the counting statistics are right skewed, as seen in Figure (2). It makes logical sense for these counting statistics to right skewed, as there are many average and below average players while there are fewer superstar players.

Figure 2: Right Skewed Variables

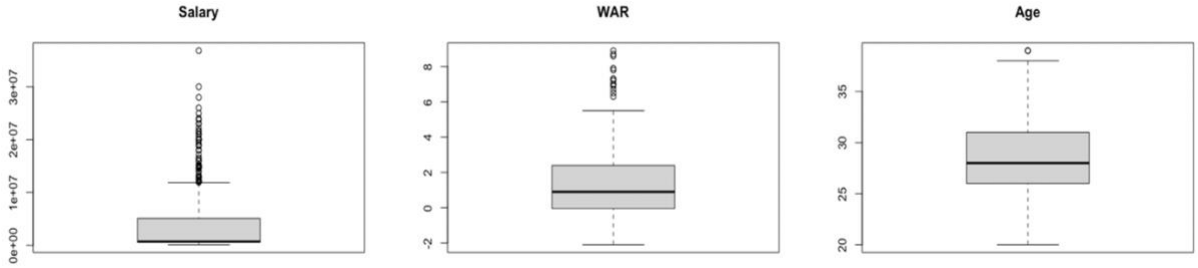
In comparison, other statistics, such as Exit Velocity, Launch Angle, and metrics that are measured in percentages are more normally distributed. In Figure (3), we display the histograms for some of these statistics.

Figure 3: Normally Distributed Variables



Analyzing the boxplots in Figure (4), we get a better understanding of the distributions and outliers for three of our numerical variables: Salary, WAR, and Age. We see from the Salary boxplot that there are many outliers at the top of the graph. These represent the highest paid players who are paid an annual salary in the tens of millions of dollars, which is far above the median salary of $717,500. Looking at the WAR boxplot, we see some outliers at the top, though not as many as for Salary. These outliers represent the superstar players of the league. As you can see, the distributions for Salary and WAR do not match up, suggesting that there are many overpaid players in the league who are not performing up to their expected standard, as well as many underpaid players who are performing above their pay rate. Finally, looking at the Age boxplot, we see that there is only one outlier, as the age distribution in the league is more normally distributed.

Figure 4: Boxplots for Salary, WAR, and Age



III. Cluster Analysis

In deciding which variables to use for cluster analysis, we must remember the research question we are trying to answer. The goal is to find undervalued players, meaning players whose advanced statistical profile suggests they are playing better than what their basic statistics show. This means they likely will not rack up the counting statistics such as home runs, hits, or WAR. These are common metrics used to evaluate players, and if they scored highly in these statistics, they would be seen as star players. We are looking for players who do not put up these kinds of numbers yet score similarly in more advanced metrics that are not as widely used or commonly known.

As such, we eliminated many of the counting variables, along with other variables deemed not as relevant before performing cluster analysis. Of the 43 numerical variables, we decided to use the following 19 variables: BA, SLG, OBP, OPS, ISO, xBA, xSLG, wOBA, xwOBA, xOBP, xISO, Exit_velocity, Launch_angle, Sweet_spot_percent, Barrel_percent, Hard_hit_percent, GB_percent, FB_perccent, and LD_percent.

The qualitative variable we decided to set aside for labeling the cluster was Value, which has five levels: MVP, All-Star, Starter, Sub, Replacement. These categories estimate the overall value the player contributed to his team to improve their chances at winning. The reason we chose this variable for this purpose connects back to the research question. Our goal here is not to perfectly classify our clusters with these five levels. Instead, we want to find players who have performed at Starter, Sub, and Replacement level yet are in the same cluster as players of All-Star and MVP caliber. These players possess the undervalued quality we are seeking to find. In our analysis, we also looked at other qualitative variables that could potentially be used to label clusters, such as League, Bats, Position, All-Star, and Acquired, but none of these labeled the clusters as well as the Value variable.

First, we apply the non-probabilistic k-means algorithm. We choose to divide the data into four clusters. My reason for choosing this was because while there are five levels in the Value variable, only 3 observations fall under the MVP level. See Figure (1) for the bar plot. There are only four levels with a significant number of observations, and because of this, we choose four clusters. Here are the results for this analysis:

Table 3: Cluster Sizes for K-Means with 4 Clusters

Cluster 1	Cluster 2	Cluster 3	Cluster 4
136	76	141	98

Table 4: Cluster Means, Medians, Standard Deviations

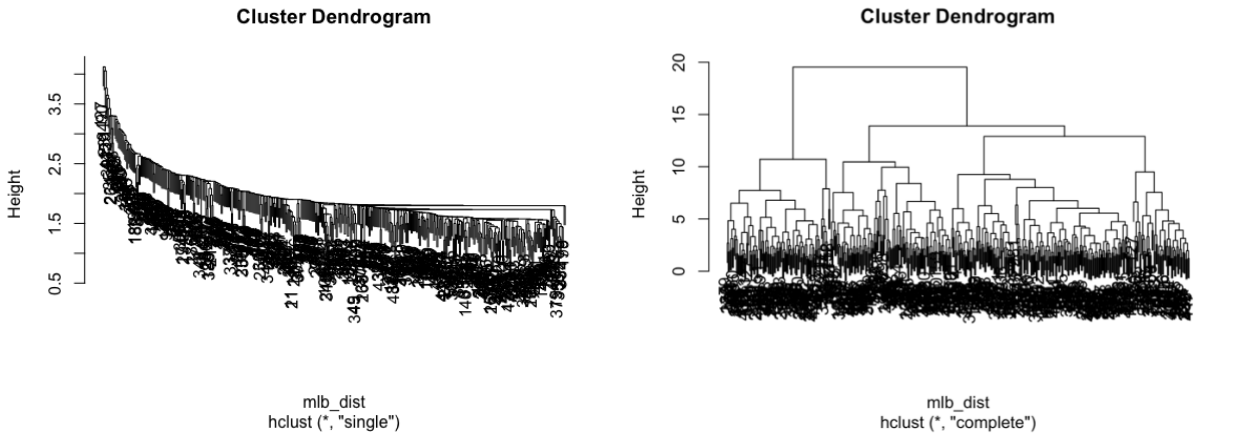
Variable\Cluster	Cluster 1			Cluster 2			Cluster 3			Cluster 4		
	Mean	Median	SD	Mean	Median	SD	Mean	Median	SD	Mean	Median	SD
BA	**0.2494338**	0.254	0.035	**0.2387237**	0.238	0.035	**0.2626879**	0.263	0.034	**0.2426531**	0.244	0.040
SLG	**0.4758971**	0.475	0.076	**0.3535526**	0.359	0.060	**0.4581348**	0.454	0.072	**0.3816735**	0.384	0.068
OBP	**0.3310221**	0.330	0.037	**0.2978421**	0.299	0.037	**0.3321348**	0.328	0.035	**0.3063061**	0.312	0.041
OPS	**0.8068971**	0.805	0.103	**0.6514474**	0.656	0.088	**0.7902411**	0.780	0.101	**0.6879592**	0.693	0.101
ISO	**0.2264779**	0.223	0.053	**0.1148684**	0.115	0.040	**0.1953333**	0.191	0.049	**0.1389388**	0.138	0.042
xBA	**0.2434338**	0.243	0.028	**0.2248421**	0.223	0.027	**0.2561631**	0.257	0.026	**0.2322245**	0.231	0.030
xSLG	**0.4617574**	0.459	0.064	**0.3335789**	0.329	0.045	**0.4504043**	0.444	0.060	**0.3548265**	0.359	0.051
wOBA	**0.3374191**	0.337	0.038	**0.2816579**	0.283	0.035	**0.3323901**	0.329	0.038	**0.2952653**	0.297	0.040
xwOBA	**0.3351176**	0.334	0.034	**0.2736316**	0.274	0.028	**0.3323830**	0.330	0.032	**0.2867245**	0.291	0.030
xOBP	**0.3273750**	0.326	0.035	**0.2876184**	0.288	0.031	**0.3280567**	0.324	0.031	**0.2994286**	0.301	0.032
xISO	**0.2183309**	0.210	0.048	**0.1087368**	0.109	0.032	**0.1941418**	0.189	0.043	**0.1225408**	0.124	0.032
Exit_velocity	**89.50515**	89.40	1.570	**87.03421**	87.30	1.488	**90.09929**	90.00	1.549	**86.37449**	86.55	1.597
Launch_angle	**17.216176**	16.90	2.701	**7.589474**	8.10	3.052	**10.629078**	10.90	2.764	**14.115306**	13.85	2.972
Sweet_spot_percent	**35.79118**	36.05	3.766	**28.71842**	28.65	3.660	**33.45532**	33.50	3.481	**33.69490**	33.70	4.332
Barrel_percent	**10.194118**	9.650	3.617	**3.742105**	3.65	2.183	**8.827660**	8.80	3.342	**3.835714**	3.75	2.031
Hard_hit_percent	**40.27059**	40.05	5.006	**30.42632**	31.25	5.267	**42.25248**	41.80	4.552	**27.91939**	29.25	5.469
GB_percent	**36.34265**	36.95	3.743	**53.43158**	52.85	4.153	**46.11844**	46.00	3.619	**42.44592**	42.15	3.762
FB_percent	**29.47132**	29.05	3.498	**18.06974**	18.15	3.023	**22.49645**	22.80	3.066	**22.86122**	22.45	3.306
LD_percent	**25.46618**	25.70	3.718	**22.25658**	22.60	3.412	**25.34539**	25.50	3.137	**25.96327**	26.00	3.954

Table (4) displays the cluster means, medians, and standard deviations for each of the 19 variables used in this analysis. Following the k-means algorithm, I also performed the nearest neighbor and furthest neighbor techniques. I centered and scaled the 19 variables, then performed the analysis. However, the results for nearest neighbor were very different from what I found using the k-means algorithm. Cluster 1 had 448 observations while the other three clusters only had 1 observation. Examining the dendrogram in Figure (5), it appears that some outliers were the cause of this. In the future, I would need to remove these outliers before running the algorithm. Using the furthest neighbor technique achieved better results, as the clusters were more evenly split. See Table (5) for the sample sizes for each cluster and Figure (5) for the dendrograms.

Table 5: Cluster Sizes for Nearest and Furthest Neighbor

	Cluster 1	Cluster 2	Cluster 3	Cluster 4
Nearest Neighbor	448	1	1	1
Furthest Neighbor	101	177	55	118

Figure 5: Dendrograms for Nearest and Furthest Neighbor



Returning to the results from our k-means algorithm, we examine the distribution of our Value variable across the four clusters in Table (6).

Table 6: Breakdown of Value variable across four clusters

Cluster\Value	MVP	All-Star	Starter	Sub	Replacement
1	3	10	49	58	16
2	0	0	6	34	36
3	0	13	42	57	29
4	0	1	10	55	32

As a reminder, the columns of Table (6) are arranged in descending order of value, with MVP designating highest value and Replacement meaning lowest value. As expected, the five levels of the Value variable do not classify the clusters perfectly, but we do see some trends. The 3 MVP-level players fall into Cluster 1, which also has 10 All-Star-level players and the fewest number of Replacement-level players, suggesting that it includes the top tier of players. Cluster 3 appears to group the next tier of players, as it includes 13 All-Star level players. Cluster 4 looks

slightly better than Cluster 2, but they have similar compositions and mostly include players who have been assigned to the bottom three value tiers.

With this in mind, we can examine the statistical profiles of the players in Cluster 1 and Cluster 3, as they contain not only the star players but also many other players who could be the undervalued players we are looking for. To better understand these results, we examine the scatterplots in Figures (6) and (7).

Figure 6: Launch Angle vs Barrel Percentage

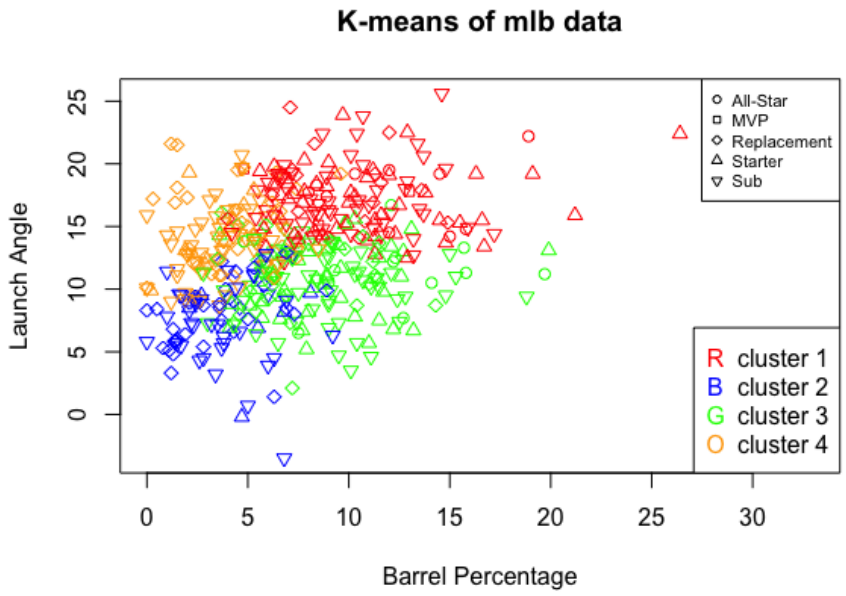


Figure (6) plots Launch Angle against Barrel Percentage, two advanced statistics that have only recently become available in the last few years thanks to new technology. We see that the four clusters are somewhat defined. Cluster 1, which we denoted as having the best players, takes up the top-right portion of the scatterplot. This suggests that higher launch angles and higher barrel percentages lead to more productive hitting. Cluster 3, which we identified as including the second-best tier of players, has a similar barrel percentage as Cluster 1, but a lower launch angle. Cluster 4 has a higher launch angle but lower barrel percentage, suggesting that these players hit the ball in the air but do not make solid contact. Cluster 2 has the lowest launch angle and barrel percentage of the four clusters. These players do not hit the ball in the air as often and they do not make solid contact as often, a combination that is unlikely to result in success.

Figure 7: Exit Velocity vs Ground Ball Percentage

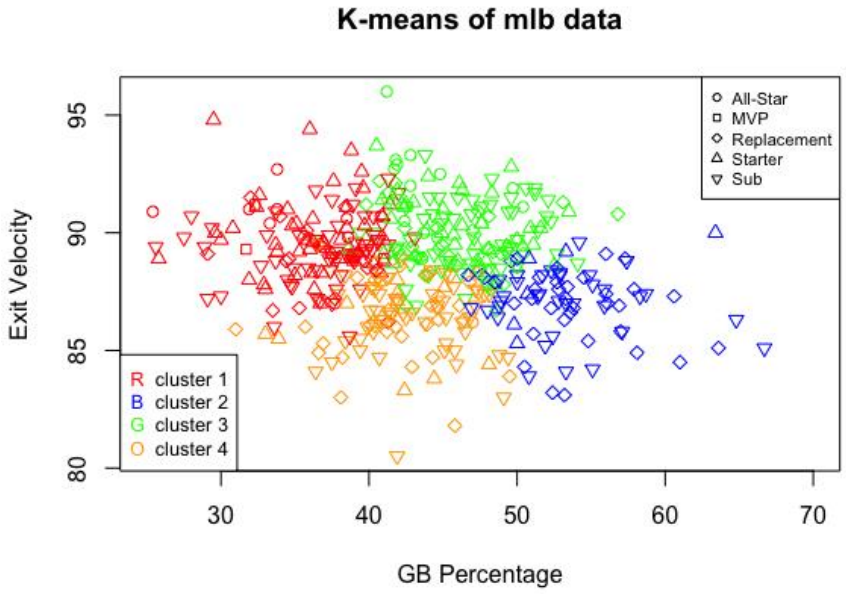


Figure (7) displays Exit Velocity against Ground Ball Percentage. Again, the four clusters are well defined. We identify Cluster 1 on the top-left side, suggesting that these players have higher average exit velocities and hit the ball on the ground the least often. Cluster 3, our second-best tier of players, has a higher ground ball percentage but a similar exit velocity as Cluster 1, meaning they hit the ball just as hard but on the ground more often. This is consistent with the results from Figure (6). Cluster 4 and Cluster 2 take up the bottom of the graph, indicating that these players have the lowest average exit velocities, with Cluster 2 hitting the ball on the ground the most often.

Circling back to our research question, can we identify undervalued players who have similar statistical profiles as star players? Using cluster analysis, we were able to group players based on 19 different advanced metrics. We learned that players in Cluster 1 were most likely to achieve success on the field, as they had the highest number of MVP-level and All-Star-level players. If I were the general manager of an MLB team tasked with finding undervalued players, I would pursue inexpensive, lesser-known players within this cluster. More generally, this analysis suggests that players who hit the ball in the air more often and who make solid contact are the most valuable players. There are many Sub-level and Replacement-level players who are hitting the ball very similarly to star players yet are not achieving the same success. Perhaps it's because of bad luck or maybe there is a mechanical adjustment that they need to make to unlock

their potential. Either way, these players have the ability to be stars yet will not cost a lot to acquire, making them ideal, cost-effective targets.

IV. Principal Components for Dimension Reduction and Latent Variable Seeking

While we only used 19 of the 43 numerical variables for cluster analysis, we used 41 numerical variables for principal components analysis. We removed two variables. The first was Year, which was the same for all observations, as the data set includes statistics for the 2019 season. Therefore, this variable was irrelevant. The second variable we removed was WAR. The reason we removed this variable was because it will serve as the dependent variable when we perform principal components regression in Section V. As a result, we cannot use it when constructing the principal components. All of the other 41 numerical variables were selected, as all of them are relevant statistics that we wish to incorporate.

The first step of principal components is to check whether it is even necessary to do the analysis. We do this by checking the correlations between the centered and scaled independent variables. From a logical standpoint, it is clear that many of the variables will be highly correlated with each other, as many of them depend on one another or are inherently related in some way. For example, AB (at-bats) and PA (plate appearances) have a correlation of 0.995, which makes sense because an at-bat is a type of plate appearance. Similarly, Hard-Hit Percentage and Exit Velocity have a correlation of 0.885, which also makes sense because hard-hit balls are classified as balls hit with an exit velocity of 95mph, which is a high exit velocity. Table (7) displays the correlations between the first ten variables. Only the first ten are displayed for presentation purposes.

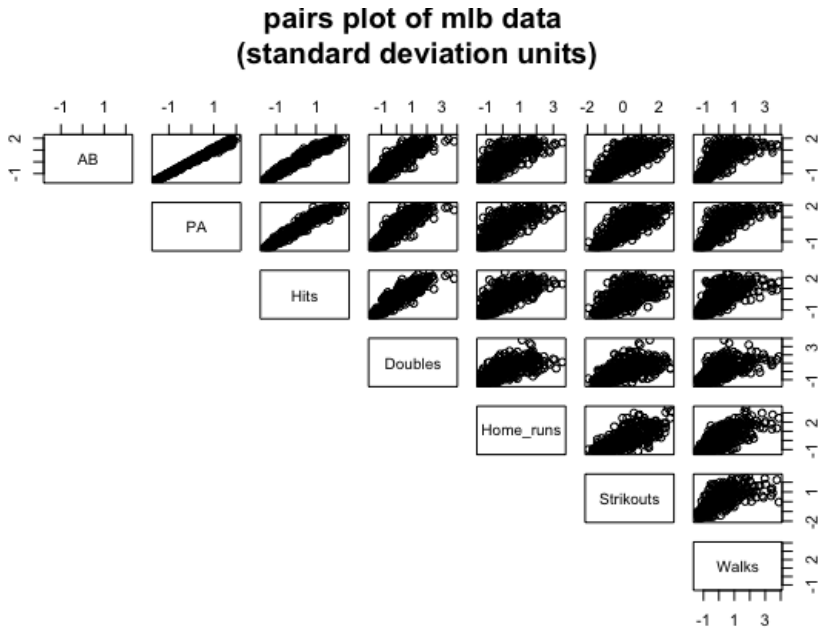
Table 7: Correlations Between the First Ten Variables

	Salary	Age	AB	PA	Hits	Doubles	Triples	Home_runs	Strikeouts	Walks
Salary	1.000	0.584	0.306	0.296	0.292	0.452	-0.141	0.393	0.236	-0.066
Age	0.584	1.000	-0.505	-0.520	-0.514	-0.262	-0.383	-0.358	-0.395	-0.598
AB	0.306	-0.505	1.000	0.995	0.981	0.859	0.412	0.746	0.588	0.604
PA	0.296	-0.520	0.995	1.000	0.969	0.852	0.422	0.771	0.611	0.673
Hits	0.292	-0.514	0.981	0.969	1.000	0.844	0.447	0.691	0.474	0.528
Doubles	0.452	-0.262	0.859	0.852	0.844	1.000	0.636	0.535	0.347	0.475
Triples	-0.141	-0.383	0.412	0.422	0.447	0.636	1.000	0.034	-0.132	0.366
Home_runs	0.393	-0.358	0.746	0.771	0.691	0.535	0.034	1.000	0.852	0.691
Strikeouts	0.236	-0.395	0.588	0.611	0.474	0.347	-0.132	0.852	1.000	0.588

Walks	-0.066	-0.598	0.604	0.673	0.528	0.475	0.366	0.691	0.588	1.000

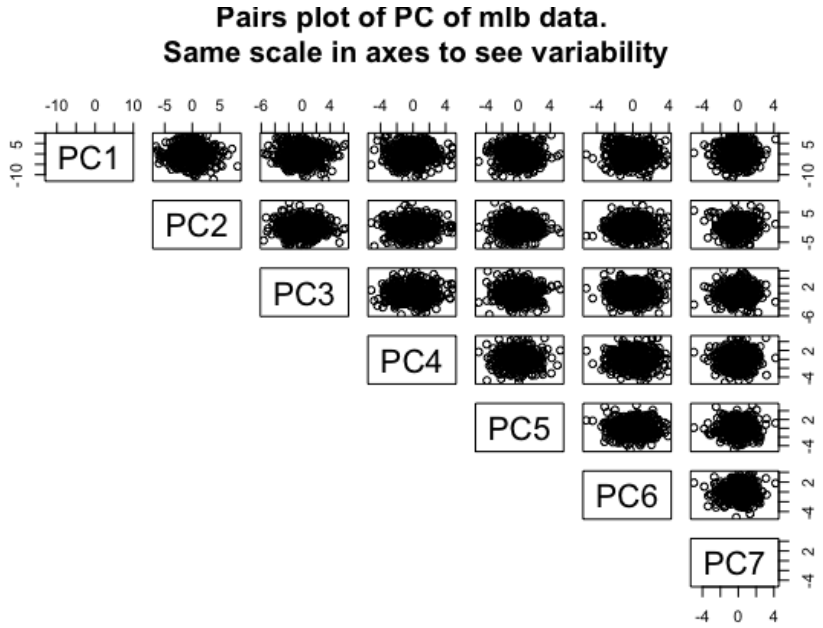
Looking at the correlations between variables, we confirm that many of the variables are indeed highly correlated with one another. Figure (8) is a pairs plot for 7 of the independent variables. From this plot, we see visually that the variables are strongly correlated.

Figure 8: Pairs Plot for 7 Variables



Knowing that there are high correlations between variables, we have justification for performing the principal components analysis. To perform this analysis, we construct the sample variance-covariance matrix of the centered and scaled data, then obtain the eigenvalues and eigenvectors of the matrix. The principal components matrix is constructed by multiplying the centered and scaled data by the matrix of eigenvectors. A total of 41 principal components are obtained. We confirm that the eigenvectors are orthonormal by checking the cross product of the variance-covariance matrix with itself, which results in a diagonal matrix. Similarly, we confirm that the principal components are orthogonal by checking the cross product of the PC matrix with itself. Both properties are met. In Figure (9), we show the pairs plot for the first 7 principal components. Only 7 are shown for presentation purposes. As seen in the pairs plot, none of the principal components are correlated with each other.

Figure 9: Pairs plot of the First 7 Principal Components



Next, we interpret the principal components. We examine the eigenvalues of the variance-covariance matrix once again. The cumulative sums of the eigenvalues, scaled to 100, are displayed in Table (8).

Table 8: Cumulative Sums of the Eigenvalues

1	**1:2**	**1:3**	**1:4**	**1:5**	**1:6**	**1:7**	**1:8**	**1:9**	**1:10**
39.81825	53.52605	62.87159	70.42889	76.26328	81.59771	85.12186	87.53346	89.50952	91.15081
1:11	**1:12**	**1:13**	**1:14**	**1:15**	**1:16**	**1:17**	**1:18**	**1:19**	**1:20**
92.35449	93.51805	94.47824	95.32360	96.00780	96.66211	97.25081	97.67970	98.05463	98.37025
1:21	**1:22**	**1:23**	**1:24**	**1:25**	**1:26**	**1:27**	**1:28**	**1:29**	**1:30**
98.63268	98.87148	99.08429	99.24820	99.40071	99.53686	99.64981	99.72651	99.79598	99.85895
1:31	**1:32**	**1:33**	**1:34**	**1:35**	**1:36**	**1:37**	**1:38**	**1:39**	**1:40**
99.91117	99.95429	99.98005	99.98819	99.99585	99.99869	99.99964	99.99988	99.99993	99.99998
1:41									
100.00000									

The cumulative sums of the eigenvalues are equal to the amount of variability explained by the corresponding principal components. For example, the first principal component explains 39.818% of the variance in the data set. The first and second principal components explain 53.526% of the variance. Our threshold was that our selected principal components must explain 90% of the variance. Therefore, we choose the first 10 principal components, as we see in Table

(8) that together they explain 91.151% of the variance. We have successfully reduced our dimensions from 41 variables to 10 principal components. Next, we examine the correlations of the principal components with the original variables in Table (9).

Table 9: Correlations Between the Principal Components and Original Variables

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10
Salary	-0.352	-0.082	0.211	-0.171	0.507	-0.278	0.026	-0.456	-0.124	0.028
Age	-0.015	0.075	0.332	-0.079	0.561	-0.233	0.082	-0.549	-0.067	0.231
AB	-0.792	-0.472	-0.096	-0.281	0.170	-0.066	0.035	0.112	0.037	0.025
PA	-0.816	-0.427	-0.059	-0.311	0.155	-0.065	0.035	0.114	0.007	0.008
Hits	-0.828	-0.502	-0.045	-0.120	0.113	-0.043	0.020	0.078	0.027	0.046
Doubles	-0.797	-0.422	-0.068	-0.098	0.117	0.027	0.025	0.078	0.054	0.037
Triples	-0.343	-0.494	-0.134	-0.186	-0.327	0.219	-0.018	-0.161	0.199	0.124
Home_runs	-0.895	0.048	-0.201	-0.124	0.111	-0.095	-0.129	0.123	-0.053	0.050
Strikeouts	-0.700	-0.083	-0.370	-0.454	-0.010	-0.092	0.249	0.113	-0.064	0.120
Walks	-0.793	0.003	0.211	-0.450	0.014	-0.067	0.044	0.116	-0.183	-0.115
K_percent	0.223	0.561	-0.526	-0.287	-0.269	-0.038	0.335	-0.036	-0.108	0.209
BB_percent	-0.370	0.508	0.461	-0.402	-0.226	-0.062	0.069	0.014	-0.291	-0.184
BA	-0.646	-0.368	0.168	0.534	-0.151	0.073	-0.058	-0.042	-0.103	0.159
SLG	-0.855	0.134	-0.112	0.349	-0.116	0.062	-0.212	-0.030	-0.066	0.174
OBP	-0.761	0.010	0.421	0.206	-0.278	0.035	-0.020	-0.034	-0.280	0.019
OPS	-0.880	0.101	0.061	0.323	-0.178	0.057	-0.161	-0.033	-0.143	0.133
ISO	-0.775	0.394	-0.249	0.160	-0.068	0.040	-0.253	-0.016	-0.029	0.142
RBI	-0.892	-0.194	-0.128	-0.126	0.181	-0.096	-0.062	0.129	-0.020	0.039
CS	-0.261	-0.537	-0.233	-0.288	-0.327	0.155	-0.002	-0.307	0.097	-0.120
SB	-0.265	-0.508	-0.146	-0.285	-0.410	0.117	-0.016	-0.424	0.075	-0.145
Games	-0.724	-0.435	-0.061	-0.342	0.192	-0.051	0.071	0.136	-0.018	0.034
Runs	-0.889	-0.326	-0.045	-0.213	0.020	-0.020	-0.045	0.048	-0.028	0.000
xBA	-0.723	-0.276	0.246	0.467	0.044	-0.015	0.099	-0.026	0.209	-0.141
xSLG	-0.878	0.275	-0.099	0.260	0.029	-0.077	0.009	-0.046	0.148	-0.081
wOBA	-0.870	0.091	0.132	0.303	-0.212	0.058	-0.138	-0.038	-0.174	0.123
xwOBA	-0.896	0.238	0.179	0.193	-0.050	-0.065	0.055	-0.042	0.069	-0.149
xOBP	-0.772	0.138	0.497	0.080	-0.132	-0.035	0.104	-0.021	-0.061	-0.228
xISO	-0.783	0.493	-0.250	0.108	0.016	-0.092	-0.038	-0.047	0.088	-0.035
Exit_velocity	-0.627	0.426	-0.168	0.089	-0.076	-0.384	0.074	-0.043	0.295	-0.090
Launch_angle	-0.242	0.433	-0.168	-0.250	0.315	0.635	-0.276	-0.064	0.015	-0.062
Sweet_spot_percent	-0.479	0.172	0.119	0.194	0.096	0.517	0.544	-0.020	-0.014	0.083
Barrel_percent	-0.632	0.603	-0.333	0.039	-0.084	-0.173	0.056	-0.055	0.052	-0.012
Hard_hit_percent	-0.626	0.459	-0.187	0.105	-0.041	-0.368	0.150	-0.038	0.286	-0.074
Zone_swing_percent	-0.072	-0.245	-0.642	0.331	0.202	0.009	0.063	-0.076	-0.331	-0.417
OZ_swing_percent	0.161	-0.406	-0.600	0.454	0.271	0.000	0.028	0.023	-0.012	0.119
Whiff_percent	0.067	0.507	-0.662	-0.142	-0.197	-0.170	0.299	-0.081	-0.213	0.061
Swing_percent	0.127	-0.409	-0.688	0.449	0.270	0.029	0.047	-0.042	-0.160	-0.142
GB_percent	0.406	-0.450	0.110	0.153	-0.290	-0.680	0.070	0.043	-0.076	0.051
FB_percent	-0.405	0.577	-0.204	-0.247	0.212	0.398	-0.251	-0.047	0.059	-0.052
LD_percent	-0.304	-0.104	0.268	0.243	0.045	0.459	0.685	0.032	0.060	0.029
Sprint_speed	-0.016	-0.319	-0.304	-0.062	-0.674	0.188	-0.076	-0.132	-0.024	0.016

Analyzing Table (9), we look for high correlations between principal components and original variables. These correlations tell us how strongly each principal component loads onto each variable. For example, Principal Component 1 has correlations of -0.75 or less for Age, PA, Hits, Doubles, Home Runs, Walks, SLG, OBP, OPS, ISO, RBI, Runs, xSLG, wOBA, xwOBA, xOBP, and xISO. Notice that all of the strongest correlations for PC1 are negative. The variables that PC1 is especially correlated with are xwOBA (-0.896), Home Runs (-0.895), and RBI (-0.892). It is difficult to assign a general latent variable from these different variables, but an argument could be made that Principal Component 1 measures a mixture of power and production by a hitter. Since PC1 is negatively correlated to most of these statistics, hitters with a very negative score for Principal Component 1 will generally be more powerful and very productive. Players who do not have as much power or have not played or produced as much will have a score closer to 0.

For Principal Component 2, we see that the variables that are most positively correlated (0.5 or higher) are K%, BB%, Barrel%, Whiff%, and FB%. The variables that are most negatively correlated (-0.5 or lower) are Hits, CS, and SB. Notice that the five variables that are positively correlated are all percentages. Players with high scores for PC2 will have high strikeout rates and whiff rates (meaning they swing and miss a lot), yet also have high walk rates, barrel rates, and fly ball rates (all of which are positive metrics). However, these players will get fewer hits, fewer times caught stealing, and fewer stolen bases. Therefore, I would say that Principal Component 2 measures a mixture of contact and speed. Players with positive scores are slower and do not make as much contact, yet they hit the ball harder, just not as often. Players with negative scores are faster and make more contact, yet do not hit the ball as hard.

For Principal Component 3, there are 4 main variables that are strongly correlated, all of them negatively correlated. These variables have a correlation with PC3 of -0.6 or less. They are Zone Swing%, OZ Swing%, Whiff%, and Swing%. This principal component is much simpler to interpret, as all four of these variables are very connected with each other, as they all have to do with how much a player swings while hitting. Therefore, a latent variable we can assign to this principal component is plate discipline. A player with a very negative score for PC3 has low plate discipline, meaning they swing a lot at pitches. A player with a score closer to 0 has high plate discipline, meaning they do not swing at as many pitches.

We could continue this practice for the rest of the principal components. For example, Principal Component 4 is most positively correlated with BA and xBA, making it a decent measure for batting average in general. Principal Component 5 is most positively correlated with Age and Salary, while most negatively correlated with sprint speed. The general latent variable we could assign to this is experience. Experienced players are obviously older and more expensive, yet they do not run as fast as the younger players. We could continue with the rest of the principal components, but it is worth noting that since the later principal components do not explain as much variability, they are not as strongly correlated with the variables, making it difficult to identify the latent variable represented by each. However, for the 5 principal components we did interpret, the latent variables are Power/Production, Contact/Speed, Plate Discipline, Batting Average, and Experience. Overall, we were able to successfully reduce the number of columns variables from 41 original variables to 10 principal components. We also constructed a biplot, but there were too many observations to the point that it was not presentable.

V. Principal Components Regression for Prediction

Next, we use the principal components to build a regression model that we will use to predict values for a dependent variable. As mentioned in Section IV, we set aside the variable WAR to use as the dependent variable. The reason for selecting this variable is that it is a good measure for the overall value a player has contributed to his team. Building a model that predicts this variable could be useful in evaluating players. This connects back to our overall goal of developing an analytical process for identifying undervalued MLB players. This model could be a useful tool for this, as we can analyze the predicted WAR values and identify high-value players. Of these players, some might be stars who already produce a WAR value similar to the predicted value, or some might be average players who have an actual WAR value much lower than the predicted value. These players could be identified as undervalued players, because while they might not have had the results of a star player, the model suggests that they are playing like one. These players could be on the cusp of a breakout season in which they generate a high WAR value.

To perform the regression, we first normalized the principal components. This was done by dividing each of the 41 principal components by its norm. We also centered and scaled the

dependent variable, WAR. Next, we created a new data frame in R comprised of the centered and scaled dependent variable and the 41 normalized principal components. This data frame would be used as our new data set for which we would build the regression model. We split this data set into two subsets: a training set and a testing set. This was done by randomly selecting 10 observations to be used for the testing set and assigning the remaining 441 observations to the training set. The training set was used to build the model, which would be used to predict WAR values for the testing set.

To select which principal components we would use for the regression model, we analyzed the correlation between the normalized principal components and the centered and scaled dependent variable WAR. Table (10) displays these correlations.

Table 10: Correlations Between Normalized PCs and WAR

PC1	**PC2**	**PC3**	**PC4**	**PC5**	**PC6**	**PC7**	**PC8**	**PC9**	**PC10**
-0.79731	-0.13931	0.05618	-0.00036	-0.17823	0.05657	-0.14201	0.00859	-0.08589	0.03764
PC11	**PC12**	**PC13**	**PC14**	**PC15**	**PC16**	**PC17**	**PC18**	**PC19**	**PC20**
0.04454	0.04066	0.02988	0.01275	0.04046	0.07076	-0.07232	0.06172	-0.04319	0.10311
PC21	**PC22**	**PC23**	**PC24**	**PC25**	**PC26**	**PC27**	**PC28**	**PC29**	**PC30**
0.12523	-0.00857	-0.01558	-0.02846	0.08642	0.10716	0.05412	-0.0053	0.0388	0.00964
PC31	**PC32**	**PC33**	**PC34**	**PC35**	**PC36**	**PC37**	**PC38**	**PC39**	**PC40**
-0.08713	-0.0015	-0.04247	0.07285	-0.00688	0.00394	0.03051	-0.03452	4e-05	-0.008
PC41									
0.0078									

We chose the principal components that were the most strongly correlated with the centered and scaled dependent variable WAR. The seven most strongly correlated principal components were, in descending order: PC1, PC5, PC7, PC2, PC21, PC26, PC20. We decided to use these seven principal components for our regression model. The reason we chose to use seven was mostly a trial-and-error process. We attempted to balance between achieving a high R^2 value for our training data and not overfitting our model. After trying out 5 to 10 independent variables, 7 seemed like it was giving us the best overall model. We fitted a multiple linear regression model using these seven principal components on our training data. Table (11) displays the summary table for the coefficients of the resulting regression model.

Table 11: Coefficients for Principal Components Regression Model

	Estimate	Std. Error	t value	Pr(>|t|)
PC1	-16.940006	0.518808	-32.652	< 2e-16
PC5	-3.783718	0.514920	-7.348	1.01e-12
PC7	-3.027748	0.519489	-5.828	1.09e-08
PC2	-3.026332	0.519657	-5.824	1.12e-08
PC21	2.592544	0.516255	5.022	7.49e-07
PC26	2.233192	0.517475	4.316	1.97e-05
PC20	2.218774	0.520365	4.264	2.47e-05

Looking at the results of the regression model, we see that all of the coefficients are statistically significant values. Our model has an R^2 value of 0.7428 and an adjusted R^2 of 0.7387. This means that approximately 74% of the variability in the dependent variable WAR is explained by this model. The root mean square error was 0.507394. We examine the histogram of residuals in Figure (10) and the residual plot in Figure (11).

Figure 10: Histogram of Residuals for Principal Components Regression Model

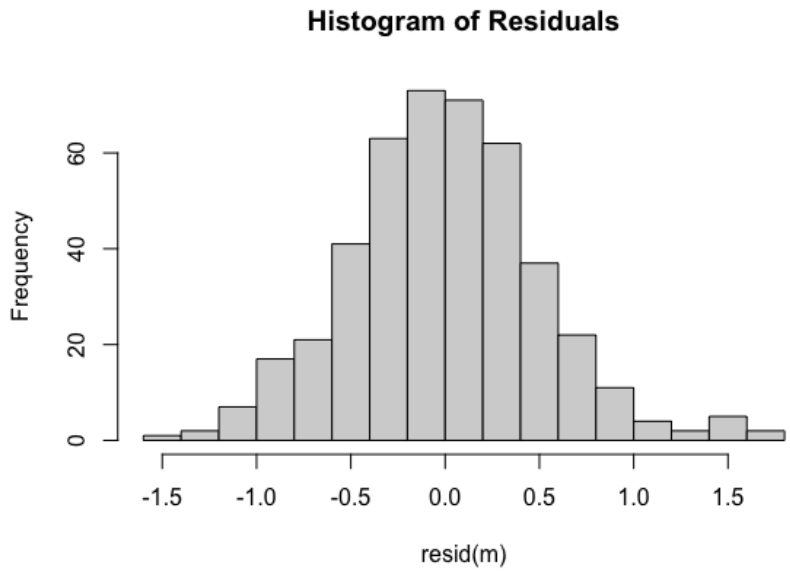
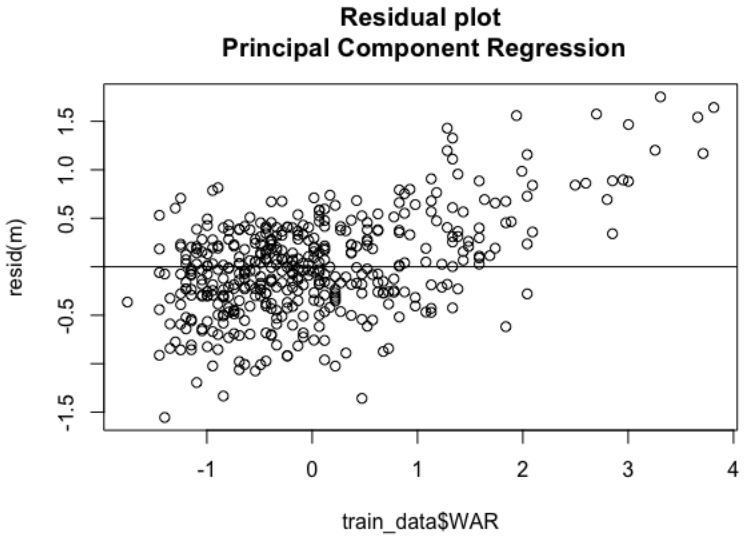


Figure 11: Residual Plot for Principal Components Regression Model



Looking at the histogram in Figure (10), we see that our residuals are normally distributed, which fulfills an assumption of multiple linear regression. However, if we examine the residual plot in Figure (11), we see that there still appears to be a pattern among the residuals: a positive correlation between the residuals and the dependent variable. Our model is strong, yet it does not explain all of the variability in the data. To build a stronger model, we could consider performing transformations on the independent and dependent variables. For the sake of this project, however, we will continue to use this model.

Next, we use this regression model to predict values for the centered and scaled dependent variable WAR in our testing data set. Table (12) displays the predicted values along with the actual centered and scaled values, along with the residuals.

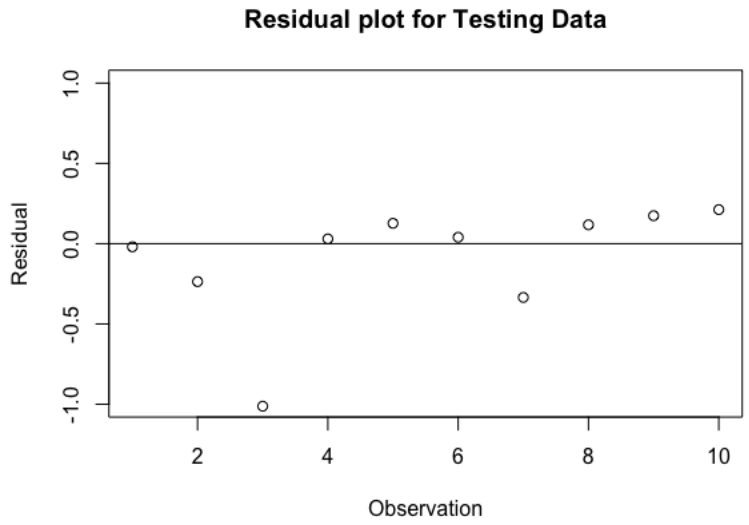
Table 12: Predicted and Actual Centered and Scaled WAR Values

Observation	Predicted Value	Actual Value	Residual
1	1.90925249	1.88953978	-0.01971272
2	-0.75894296	-0.99526834	-0.23632538
3	0.72567611	-0.28671898	-1.01239509
4	-0.56947011	-0.53977232	0.02969779
5	-0.16044123	-0.03366563	0.12677560
6	-0.68085795	-0.64099366	0.03986429
7	0.09865779	-0.23610831	-0.33476610

8	-0.45472409	-0.33732965	0.11739444
9	-0.61254760	-0.43855098	0.17399662
10	1.47506359	1.68709710	0.21203351

Examining the values in Table (12), we see that our model does a decent job at predicting the values for dependent variable. Figure (12) displays the residual plot for these ten observations in the testing data set.

Figure 12: Residual Plot for Testing Data

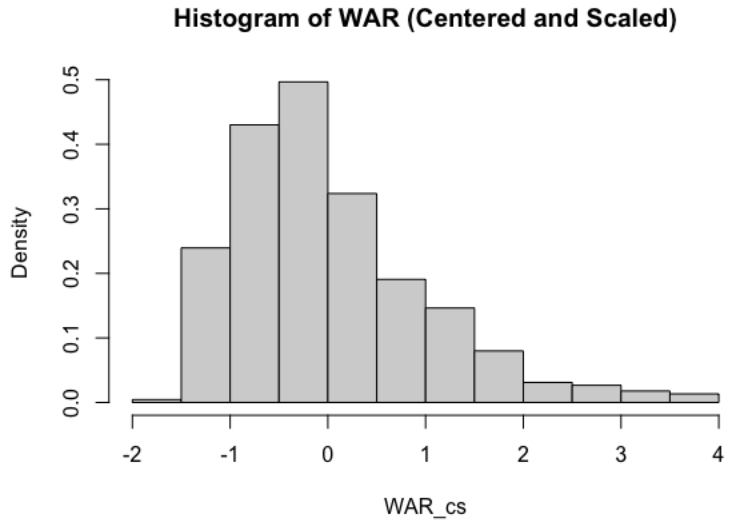


The root mean square error was 0.3606672 for these residuals. We see that most of the residuals are close to 0. The third residual, however, is an outlier. As we can see in Figure (12), it is much further from 0 than the other residuals, and we see in Table (12) that it has a value of -1.012. This means that our predicted value for WAR was much greater than the actual value for WAR. This connects back to our goal of identifying undervalued players. This player represented in Observation 3 could potentially be one of these undervalued players that we are looking for, as their predicted WAR is higher than their actual WAR. In other words, they might not have played very well during this season, but the model suggests that based on their statistics, their expected value was much higher than what it really was. Perhaps the disparity was due to bad luck, ballpark factors, or other small reasons, but from a general manager standpoint, this player would be one that I would target for acquisition, as I would expect him to perform closer to his expected value in future seasons.

VI. Maximum Likelihood Estimation

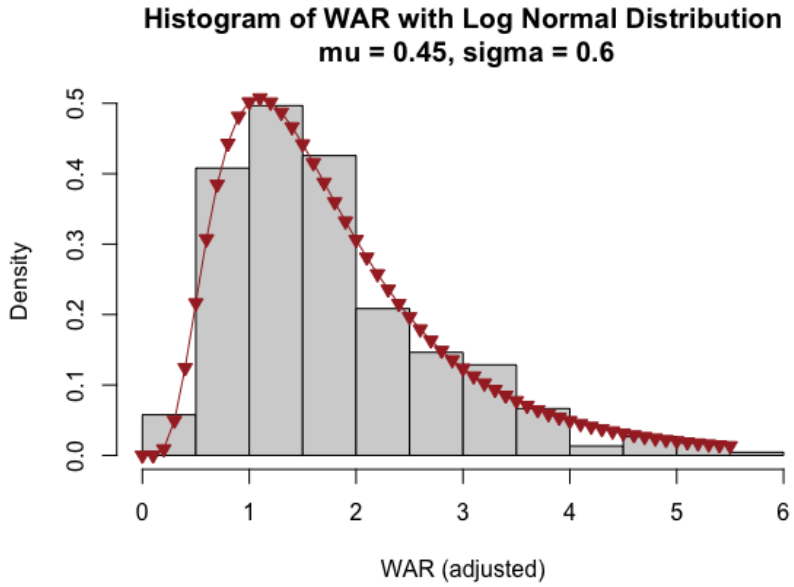
In this section of the project, we will further explore the dependent variable that we used in Section V. Our goal is to select a model that fits this data well, and then find the maximum likelihood estimators for the model's parameters. Figure (13) displays a histogram of the centered and scaled variable WAR.

Figure 13: Histogram for Centered and Scaled WAR



We see that the distribution is right skewed. Judging by the shape and skewness of the histogram, we hypothesize that the log normal model is a good fit for this variable. However, since the log normal model is only for positive values, we will first have to perform a transformation to make all values positive. We do this by finding the minimum value for this variable and then adding the magnitude of it to each observation. We also add 0.01 to each observation to ensure that every value is positive. This transformation successfully retains the shape of the distribution, simply shifting it to the right, specifically by a value of 1.764428. Next, we estimate starting values for the parameters of the log normal model, μ and σ. This is done by simulating log normal distributions over the histogram. Using a method of trial-and-error, we estimated starting values of $\mu = 0.45$ and $\sigma = 0.6$. Figure (14) displays the simulated distribution over the histogram. Note that the data has been shifted so that all values are positive.

Figure 14: Histogram for WAR with Simulated Log Normal Distribution



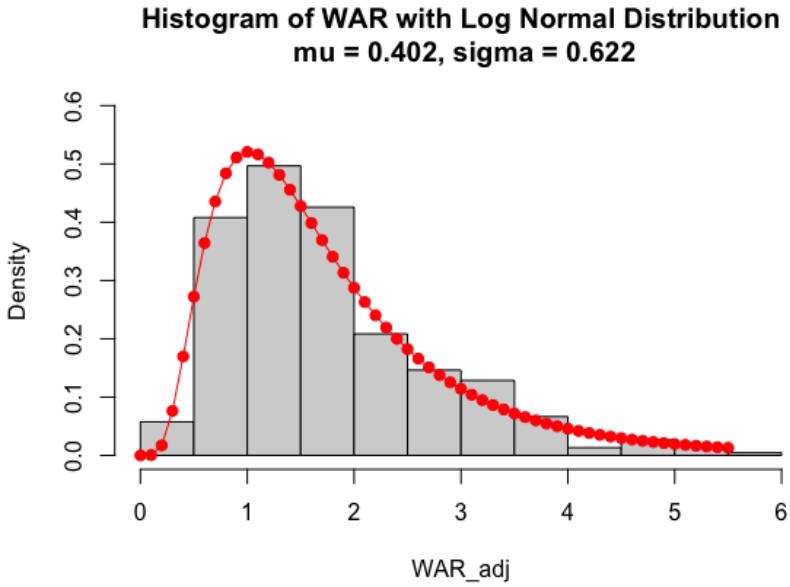
As seen in Figure (14), the log normal model with starting parameters ($\mu = 0.45$, $\sigma = 0.6$) fits the histogram well. Now that we know that the log normal model is potentially a good fit for this data, we calculate the MLE estimators for μ and σ. Using the log likelihood function for the log normal distribution and the nlm() function in R, we obtain the estimated values for the parameters for the WAR values in our data set. In Table (13), we display the MLEs along with the standard errors and 95% confidence intervals for each of the parameters.

Table 13: MLE Estimators for μ and σ

Parameter	MLE Estimates	Standard Error	95% Confidence Interval	
			Lower Bound	Upper Bound
μ	0.4018392	0.02928662	0.3444385	0.4592399
σ	0.6219530	0.02071707	0.5813483	0.6625577

As seen in Table (13), our MLE value for μ is 0.402 and our MLE value for σ is 0.622. These values are relatively close to our original estimated values of 0.45 and 0.6, and are what determine the shape of the log normal model. Now that we have obtained a maximum likelihood model for fitting the variable WAR, we can plot the model over the histogram, which we see in Figure (15).

Figure 15: Histogram of WAR with MLE Log Normal Model



The close fit between the histogram and the model supports our hypothesis that the variable WAR follows a log normal model. This model can therefore be useful for calculating probabilities. For instance, we can use this model to calculate the probabilities of obtaining values larger or smaller than those predicted in Part V. First, we perform the same adjustment to the predicted WAR values by adding 1.764428 to each value. Again, this does not change the shape of the distribution but simply ensures that all values are positive. Then, for each point, we calculate the area under the curve to the left of the point and to the right of the point. These values give the probabilities of obtaining values less than and greater than the obtained value. Table (14) displays these probabilities. Notice that for each observation, the sum of the probability of obtaining a value less than the predicted value and greater than the predicted value is 1, which makes sense as the total area under the curve is 1.

Table 14: Probabilities of Obtaining WAR Values Less Than or Greater Than Predicted Values

Predicted Value	Shifted Value	Probability < Value	Probability > Value
1.90718676	3.671615	0.9257869	0.07421306
-0.76096488	1.003463	0.2609129	0.73908706
0.72362723	2.488056	0.7937365	0.20626351
-0.57153782	1.192891	0.3584881	0.64151188
-0.16252757	1.601901	0.5443926	0.45560740
-0.68294556	1.081483	0.3014811	0.69851895
0.09661569	1.861044	0.6378045	0.36219548
-0.45677870	1.307650	0.4149539	0.58504608
-0.61451959	1.149909	0.3366930	0.66330698
1.47304512	3.237473	0.8930265	0.10697348

The probabilities of obtaining a value less than the predicted values can also be interpreted as percentiles. For example, for the first observation, the probability of obtaining a value less than the predicted value is 0.9257869. Therefore, the model is predicting that this player is in the 92nd percentile for WAR, or in other words, has a WAR value that is higher than 92% of the players in the league. This player is likely to be a star player, as they are within the top 10% of the league. Looking at the third observation, which we identified as an undervalued player in Section V, the model predicts that this player is in the 79th percentile for WAR, whereas his actual value for WAR would fall in the 49th percentile. This reinforces our decision to identify this hitter as an undervalued player that is ideal for acquisition.

VII. Conclusions

With the amount of Major League Baseball data and new statistics becoming available thanks to technology, there is near limitless room for exploration in baseball analytics. It is a growing movement within the sport to use data to optimize player and team performance. One specific field that we have discussed is the method of using data analysis to evaluate players. The goal of this paper was to analyze advanced Major League Baseball statistics to develop a strategy for evaluating players and identifying undervalued hitters. These players possess statistical

profiles that suggest they have the potential to play much better than how they have performed, making them less expensive, low-risk targets.

We obtained the data set, which measured hitting statistics from the 2019 season, from Baseball Savant and Baseball-Reference. We examined the different variables of the data set in Section II. In Section III, we performed k-means non-probabilistic cluster analysis on the data, splitting the observations into four different clusters. We attempted to label the clusters with several different categorical variables, but in the end, the categorical variable that we decided to use was Value, as it resulted in the best labeling of clusters and best served our purpose of comparing actual value with predicted value. We found that Cluster 1 included the most MVP-level and All-Star level players, suggesting that it represented the top tier of valuable hitters. Cluster 3 appeared to represent the second tier of hitters, followed by Cluster 4, and then lastly Cluster 2.

We therefore examined the mean vectors of the clusters, as the mean statistics for Cluster 1 or Cluster 3 could provide insight as to the kind of players we would want to target. For example, the players in Cluster 1 contain not only the top tier star players, but also other players who have similar metrics as these stars but are more underrated. Examining the results in Table (4) and the scatterplots in Figures (6) and (7), we see that the players in Cluster 1 have the highest average launch angle, barrel percentage, and exit velocity, while also displaying the lowest ground ball percentage. In other words, players who hit the ball in the air at high speeds are more likely to play like an All-Star or MVP-level player. This might seem intuitive, but examining the players in Cluster 1, we find a lot of players who share this statistical profile yet are not considered stars. I filtered the players in Cluster 1 to only include non-All-Star players making less than $1 million and came up with a group of players that could be cheap acquisitions with high upside. Some undervalued, low-paid players in this cluster included Alex Dickerson, Trent Grisham, Mitch Haniger, Teoscar Hernandez, Tyler O'Neill, Donovan Solano, and Luke Voit. None of these players had great seasons in 2019, as they all were labeled as "Sub" for the Value variable, yet their advanced statistics were similar to those of star players. At the end of the 2019 season, none of these players held great trade value or had high salaries, making them cheap targets for rival teams. Two years later, in 2021, we see that several of these players have blossomed and their values have drastically increased, indicating that our cluster analysis would have served as a good predictor for valuable players.

In Section IV, we performed Principal Components Analysis on the data set and reduced the number of dimensions from 41 numerical variables to 10 principal components. We attempted to identify latent variables within these principal components, and came up with the following assignments for the first five principal components: Power/Production, Contact/Speed, Plate Discipline, Batting Average, and Experience. This strategy was also useful in reducing multicollinearity, which is important for building a regression model, as seen in Section V.

For the regression model, we set aside the variable WAR, which is a good measure for a player's actual value, as the dependent variable. Therefore, by building a model that predicts this variable, we could have a useful tool for evaluating players. For instance, if a player's predicted WAR was much higher than their actual WAR, it could be the marker for an undervalued player, The model would suggest that this player is performing like a star even though the actual results might not be there, at least not yet. We constructed the regression model using the seven normalized principal components most correlated with WAR and used this model to predict WAR values in a small testing data set. The model was relatively accurate in predicting values and the results also gave us insight for identifying undervalued players. Specifically, the third observation in the testing data set, whom we identified as Yasiel Puig, had a predicted WAR value a full standard deviation above his actual WAR value. We can generalize this by examining all players from the 2019 season and seeing which ones have a predicted WAR value 1 standard deviation higher than their actual value. Some players who fall into this category who we would consider to be undervalued are David Dahl, Renato Nunez, Franmil Reyes, and Nick Senzel. Like the players listed earlier, all of these players made less than $1 million and were listed as Sub-level players for the Value variable. None of them performed very well in terms of WAR, yet our model suggests that their expected WAR based on their statistics was much higher. These players could therefore be identified as undervalued players who we could target.

In Section VI, we fit a log normal model to the distribution of the variable WAR and calculated the maximum likelihood estimators for the parameters. Using this log normal model, we were able to calculate probabilities of obtaining values less than a certain point by finding the area under the curve to the left of the point. We interpreted this probability as an estimated percentile. For example, using the third observation in our testing data, the predicted value for WAR was at a point on the distribution where 79% of the area under the curve was to the left. This player, who we identified as Yasiel Puig, therefore was predicted to score in the 79th

percentile for WAR. Using the same log normal model on his actual value, we see that he actually scored in the 49th percentile. We identified this disparity as another marker for undervalued players.

Baseball is a business, and the general manager of an MLB team has the responsibility of building a competitive roster while keeping the payroll as low as possible. Therefore, developing a strategy for identifying underrated and cost-effective players can be instrumental for a team's success, both on the field and financially. The analyses and models discussed in this paper could potentially be a very useful tool for Major League Baseball teams as they seek to make optimized decisions.

VIII. Acknowledgements

The author would like to acknowledge Dr. Juana Sanchez and the Statistics 102B course at the University of California, Los Angeles. The analysis strategies and techniques used in this paper were learned from the lecture notes and R programs provided by Dr. Sanchez. We would also like to acknowledge Baseball Savant and Baseball-Reference for the Major League Baseball data set, as well as for providing definitions for advanced baseball analytics. The analyses in this project were performed in RStudio.

IX. References

Sources used for constructing data set:

https://baseballsavant.mlb.com/leaderboard/custom?year=2019&type=batter&filter=&sort=0&sortDir=asc&min=100&selections=player_age,b_ab,b_total_pa,b_total_hits,b_double,b_triple,b_home_run,b_strikeout,b_walk,b_k_percent,b_bb_percent,batting_avg,slg_percent,on_base_percent,on_base_plus_slg,isolated_power,b_rbi,r_total_caught_stealing,r_total_stolen_base,b_game,r_run,xba,xslg,woba,xwoba,xobp,xiso,exit_velocity_avg,launch_angle_avg,sweet_spot_percent,barrel_batted_rate,hard_hit_percent,z_swing_percent,oz_swing_percent,whiff_percent,swing_percent,groundballs_percent,flyballs_percent,linedrives_percent,sprint_speed,&chart=false&x=xba&y=xba&r=no&chartType=beeswarm

https://www.baseball-reference.com/leagues/majors/2019-standard-batting.shtml

https://www.baseball-reference.com/leagues/majors/2019-value-batting.shtml