Base Bucks Playoff Baseball Predictor: Understanding How to Build a Playoff Baseball Team

Christopher Demirjian cjd2186@columbia.edu
Kaja Huruk koh2107@columbia.edu
Nicholas Isaza ni2253@columbia.edu

12/13/2023

STATGU4001 Introduction to Probability and Statistics

1. Abstract

This paper seeks to explore the underlying team statistics that best can predict a team's likelihood of reaching the MLB playoffs. The approach is two-fold, providing an "offensive" regression that regresses wins on offensive statistics and a "defensive" regression that regresses wins on defensive statistics. Data was collected from the 2008 season until the most recent 2023 season; however, 2020 was omitted from the data set due to the coronavirus pandemic leading to a shortened season and unique playoff format. These regressions helped show the significance of specific statistics influencing wins and the magnitude and direction of these statistics. As we are interested in creating a model that creates a playoff-caliber team, we then run a Monte-Carlo simulation for an imagined team with specific stats and show the percentage of that team making the playoffs by reaching a threshold amount of wins.

2. Introduction

The sport of baseball has quickly become a statistically driven and very analytical sport following the 2002 Oakland Athletics' impressive "Moneyball" playoff run. By utilizing a novel sabermetrics approach, General Manager Billy Beane was able to construct an incredible team on a limited budget that ended up beating teams with big-name superstar players as well as very deep pockets. This went on to revolutionize the sport and create the turbo-charged teams we see in the modern era that are able to nitpick every part of the game so as to maximize their success. In the spirit of this statistical takeover, we sought to look at data that tracked teams' average statistics starting from the 2008 season to the most recent 2023 season and see which statistics were the most important in influencing the record of a team over this period of time. Utilizing a linear regression model, we were able to figure out exactly this and see how influential some of these highly publicized statistics really are.

Focusing on making the playoffs was what mattered the most in terms of this report, as we regard it as the gateway to future successes for the franchise. If a team can specifically tailor its roster to augment its chances of making the playoffs, this will pay dividends down the road. In addition to various financial incentives for the teams in terms of ballpark sales and playoff-specific merchandising, as well as cash bonuses for players, the playoffs signal that the team is ready to compete. Consistently making the playoffs attracts star players who want to win games, which in turn creates a winning culture and solidifies the team's place as a competitive and important team in the league. For small market teams (teams not located in the biggest metropolitan places in the United States), making the playoffs also brings much-needed cash and media attention to the team but also encourages management to actively build off success instead of maintaining a degree of mediocrity extensively prevalent in these small market teams.

The largest challenges when attempting to come to these results was making sure that we could understand what these statistics were even measuring, as well as making sure it was actually relevant to predicting wins. In terms of understanding statistics, many of the advanced statistics are actually formulas using 4 or 5 different statistics to try and create an advanced measure that captures more than a simple statistic could capture. Take slugging percentage, for example, which is equal to (1B + 2Bx2 + 3Bx3 + HRx4)/AB, yielding a number that is very hard to contextualize in certain situations. Thus, when interpreting results and building the actual regression, it was important to carefully make sure the variables included in regressions were not almost identical to each other. Furthermore, the data set we downloaded had just about anything that could be measured about a roster, such as the average age of the pitchers to how many times a team's batters were hit by pitches. While these types of statistics may, of course, influence wins, they are not necessarily descriptive compared to others. In addition to that, interpreting the coefficient results for these would not be helpful in trying to construct a playoff-caliber team. If, for example, we saw that younger teams yielded higher wins, the logical instruction would be to tell teams to get younger if they wanted to win. However, younger players then influence much else in the real world that the model is not able to capture.

As previously mentioned, the main instrument used to understand how statistics influence wins was the linear regression model. After understanding the coefficients of the selected variables, we created a hypothetical team built with their metrics at a high enough level to hypothetically make the playoffs. By running a Monte Carlo simulation, we then introduce the variance of these statistics over the course of the

season to see the probability of the team we put together at the beginning of the season reaching the winning threshold at the end of the season. Overall, as a team, we were able to learn many skills, such as how to consolidate large data sets and discern what is important for our specific goals. Furthermore, we fine-tuned our knowledge of linear regressions and learned how to apply that to our Monte Carlo simulations.

3. Problem Statement and Data Sources

All the data used in this project is sourced and collected from Sportsreference's baseball statistics website "www.baseball-reference.com" This website contains organized tables of data for every possible statistic tracked in baseball. These tables can be easily extracted and downloaded from the website as Excel files. For the purpose of this project, the section "Season Team Stats" was utilized to extract the average statistics for all 30 MLB teams in the last 15 seasons (2008-2023). 2020 was omitted from our datasets, as the season was truncated from 162 to 50 games and featured an abnormally large yield of playoff teams as a result of a disrupted and shortened season in light of the COVID-19 pandemic and government lockdowns. The average offensive and defensive season statistics for each of the 30 teams in the past 15 years were then combined to create a 450-row master spreadsheet, "Master_Stats.csv" (See Appendix A), that features every recorded offensive team stat, defensive team stat, and win total for each team in each Major League Baseball season. Additionally, the league average of all 30 teams for each statistic is calculated in "Master_Stats_Average.csv" (See Appendix A).

The objective of this project is to find the optimal statistics that can be used in hitting and pitching (offense and defense) in order to reach the playoffs. Baseball is unlike any other sport in that the outcome of every play in each game is recorded and can be predicted using statistics. With baseball having such a statistically driven and measured set of strategies, we can use statistics to optimize the strengths and weaknesses of a baseball team to allow a team to find where they must improve on their roster to be on the path to the playoffs and possibly a championship.

4. Methodology

In order to determine how a team can construct their roster to make the playoffs, we had to determine which variables were significant to a baseball team winning games, how many games a baseball team needs to win to qualify for a playoff position, and what range these variables must lie within to reach this win threshold. To accomplish these sub-objectives, we used the following statistical tools:

- Linear Regression Model in R: determine which offensive and defensive statistics significantly influence team wins (see section 4a)
- Statistical Mean: determine how many wins the least winning playoff team needs (see section 4b)
- Variable Distribution: determine what value range is necessary for each statistic variable to reach the win threshold
- Monte Carlo Simulation: evaluate values used to test variables over numerous trials to extract win amounts

4a. Linear Regression Model

In baseball, there are a multitude of metrics used to measure an individual player's performance. A team is an assembly of players, therefore the statistics used to measure these player's performances can be averaged to calculate statistics to measure a team's performance. However, every aspect of a team's play on the field, offensive or defensive, is tracked using a statistical metric, making it difficult to understand which variables should be optimized to maximize a team's performance on the season itself. Baseball reference tracks 28 offensive and 35 defensive statistics to measure hitting and pitching performance, respectively. However, using all of these statistics in a model is unrealistic, as it is difficult to properly gauge which metric has the most significant influence on season win total. Therefore, we must figure out the most important variables to use in our model.

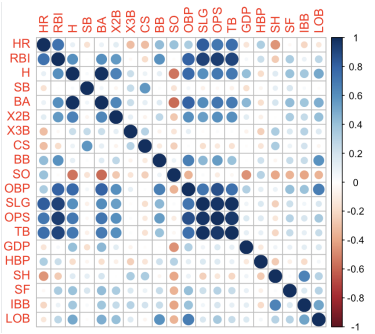
In order to find the most important statistics to use in the linear regression model, we first referenced the ten main team statistics ESPN displays on their MLB team stats homepage (ESPN, MLB

team stat leaders, 2023 regular season). This allowed us to narrow down the offense and defensive stats to the five categories each.

Using the five main batting statistics (BA, HR, RBI, H, SB), an offense baseline model (see Appendix B Model 1) was created; however, this model yielded a low R squared value of .375, and only three of five variables were seen as significant, so more variables had to be added to the model.

```
Call:
lm(formula = W.1 ~ BA + HR + RBI + H + SB, data = realdata)

Residuals:
     Min       1Q   Median       3Q      Max
-24.1411  -6.6796   0.5602   6.0668  29.2113

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  -8.98107   15.16602  -0.592 0.554029
BA          717.17788  220.73773   3.249 0.001246 **
HR           -0.02674    0.02281  -1.172 0.241652
RBI           0.11402    0.01457   7.825 3.76e-14 ***
H            -0.11899    0.03144  -3.784 0.000175 ***
SB            0.01063    0.01580   0.673 0.501438
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 9.714 on 444 degrees of freedom
Multiple R-squared:  0.3752,    Adjusted R-squared:  0.3682
F-statistic: 53.34 on 5 and 444 DF,  p-value: < 2.2e-16
```

(Left) Baseline 5 offensive statistic model



(Right) Highly Correlated Matrix of all relevant offensive statistics

From this baseline model, we were able to add statistical categories that measure different aspects of a team's performance until every relevant offensive statistic was included in the model. This expanded model has a higher R squared value of .4825, but only four of the twenty categories were statistically significant (see Appendix B Model 2).

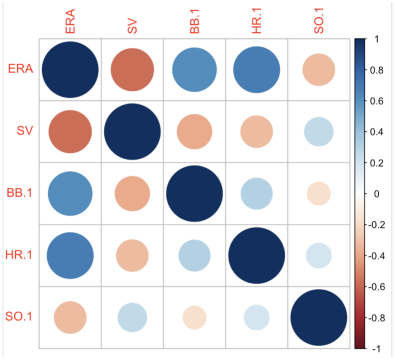
To increase the statistical significance of the variables in our model, variables that were highly correlated (such as OBP, SLG, OPS, TB, SO, LOB) were taken out from the model, thus yielding a more accurate model with a higher R squared value of .43 (see Appendix B Model 3); however, there were only ten statistically significant variables of the sixteen modeled.

This led us to create our final model, where all extra bases hit categories (X2B, X3B) were removed in favor of SLG, which better accounts for how hard the ball is hit (if the ball is hit harder, it will go farther, and you will have a better chance to score). Furthermore, generalized offensive statistics (RBI, HBP) and statistics that focused on poor offensive performance (SO, HBP, LOB, SH, SF) without much offensive incentive were removed as well. Statistics such as SB and CS (caught stealing) remained in the model, as stealing bases is an integral part of baseball, and the number of bases stolen is not significant unless we also take into account the number of failed base stealing attempts (see Appendix B Model 4).

In addition to this offensive statistical model, a defensive statistical model was created. This defensive statistic model was based on the five main pitching statistics (ERA, SV, BB, HR, SO). Although this model yielded a high R squared value of .59, all of these values were highly correlated, leading us to create a new model with more defensive statistics (see Appendix B Model 5).

```
lm(formula = W.1 ~ ERA + SV + BB.1 + HR.1 + SO.1, data = realdata)

Residuals:
    Min      1Q  Median      3Q     Max
-35.214  -5.197   0.084   5.525  21.314

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 103.284782   7.594853  13.599  < 2e-16 ***
ERA         -13.078494   1.600414  -8.172 3.20e-15 ***
SV            0.498886   0.062859   7.937 1.71e-14 ***
BB.1         -0.013208   0.008213  -1.608  0.10850
HR.1          0.056344   0.020877   2.699  0.00722 **
SO.1          0.006402   0.003321   1.928  0.05452 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.862 on 444 degrees of freedom
Multiple R-squared:  0.5907,    Adjusted R-squared:  0.5861
F-statistic: 128.2 on 5 and 444 DF,  p-value: < 2.2e-16
```

(Left) Baseline 5 Statistic Defensive Model



(Right) Highly Correlated Baseline Defensive statistics

Of these five statistics, walks were not statistically significant, informing us that this model did not account for a pitcher's control of the ball when facing a batter. For this reason, we added statistics that capture a pitcher's control of the ball, such as IBB, HBP, WP, IP[1] (see Appendix B Model 6).

In the final defensive statistic model, the number of pitchers was added[2] and the number of innings pitched by a pitcher was removed (IP is highly correlated to ERA, which makes sense, as ERA is a function of IP and Outs). With an R squared value of .608, and seven statistically significant variables out of ten, this model served as the final defensive model (see Appendix B Model 7).

Finally, we combined the significant offensive and defensive statistics into one linear regression model, but found that this model was not the best to use, as it overfit our data with a large R squared value of .835:

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   9.503018   9.686753   0.981 0.327126
BA          544.690881 152.210162   3.579 0.000385 ***
HR            0.092244   0.030110   3.064 0.002324 **
H            -0.023143   0.017474  -1.324 0.186049
SB            0.010725   0.010702   1.002 0.316845
CS           -0.071900   0.033861  -2.123 0.034290 *
BB            0.032818   0.004848   6.770 4.23e-11 ***
GDP          -0.078705   0.018552  -4.242 2.71e-05 ***
IBB           0.002529   0.025555   0.099 0.921223
SLG         -36.997889  57.904819  -0.639 0.523199
ERA         -11.833999   1.135482 -10.422  < 2e-16 ***
SV            0.346109   0.042095   8.222 2.36e-15 ***
BB.1         -0.008049   0.005790  -1.390 0.165191
HR.1         -0.004422   0.015075  -0.293 0.769416
SO.1          0.000592   0.002517   0.235 0.814175
IBB.1        -0.076395   0.022754  -3.357 0.000856 ***
WP           -0.002607   0.019967  -0.131 0.896168
X.P           0.009577   0.061215   0.156 0.875750
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.061 on 432 degrees of freedom
Multiple R-squared:  0.835,     Adjusted R-squared:  0.8285
F-statistic: 128.6 on 17 and 432 DF,  p-value: < 2.2e-16
```

(Above) linear regression model combining final offensive and defensive statistics

4b. Win Threshold Calculation

The win threshold is a simple but important calculation that we utilized as a measure for playoff berths. Because making the playoffs involves different situations every year, by looking at the win totals over the years, we can understand, on average, what is needed to guarantee a spot in the playoffs. Using the win data from the 2012 season onwards, due to a playoff format change that year, we isolate the five teams that made the playoffs in a given year and then take the number of wins for the team with the lowest record. This is done because we do not necessarily care how the team makes the playoffs; it is just that it does make it. By then taking the mean of these lowest winning teams in the American League, National League, and league in general, we were able to have a win value to use as a benchmark for entering the playoffs.

4c. Variable Distribution

For each of the chosen variables, which we divided into offensive and defensive stats tables so that they would be easier to understand by people who are not as familiar with baseball, we calculated the arithmetic mean and standard deviation. Then, we also computed the coefficient of variation. We did this because the variables have very different values, and we wanted to make it easier to compare the spread of distribution between variables. Finally, we created graphs in R to show the variable distribution. In each graph, we plotted the probability density function, which required scaling the y-axis to density. Each graph also shows the mean as ±1 standard deviation, indicated by a dashed line. We created the

[1] If a pitcher has good control, he will not need to walk batters intentionally and he will pitch more innings;

[2] Unlike batters, a team can use as many pitchers as they need in a game/season; if more pitchers are used, the quality and performance of these pitchers is inherently lower

histograms in order to gather more information, which was useful in conducting the Monte Carlo Simulation. Specifically, we wanted to see what kind of distributions the variables followed.

4d. Monte Carlo Simulation

The Monte Carlo simulation adds an extra dimension to this project by using the previous linear regression and predicting how many wins a team would have at the end of the season, given expected opening-day statistics and league-wide variance in the specific statistic. The rationale here being that teams are (mainly) constructed throughout the offseason and thus have a roster with average, or expected statistics for the upcoming season based on their performance the past season. If the team, therefore, trades for or acquires players that will move a given stat in the desired direction, the team enters the season with a baseline expectation for their performance (number of wins). However, because sports have intrinsic variability in them, no player will perform at the exact same level in one season as another. To account for this, we introduce the variance of that specific stat into the model so that the statistics are able to fluctuate accordingly, and these new season stats are therefore used to predict the number of wins in the season..

5. Analysis and Results

5a. Linear Regression Results

In our final offensive model, there are nine variables, all of which are statistically significant, with an R squared value of .419. Since all of these variables are statistically significant, they all strongly affect the number of wins a team can earn in a season, therefore they must be considered when we find out how to build a playoff team. The highest coefficient we see is that of the batting average, which is logical, as a player must make contact with the ball to score runs. Additionally, we see that Grounded into a Double play is detrimental to the number of wins a team can get, as getting two out of three possible batters out in one play ruins all momentum in an inning (of which there are only nine in a game). The most interesting coefficient, however, is the negative coefficient for Slugging. A baseball fanatic's intuition would lead one to believe that hitting the ball a lot and as hard as possible is optimal to get more wins, however having a power-heavy team leads to shortcomings in speed and contact.

```
lm(formula = W.1 ~ BA + HR + H + SB + CS + BB + GDP + IBB + SLG,
    data = realdata)

Residuals:
     Min       1Q   Median       3Q      Max
-26.0064  -6.0207  -0.3367   5.8521  28.8453

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -5.969e+01  1.409e+01  -4.235 2.79e-05 ***
BA           1.618e+03  2.719e+02   5.951 5.45e-09 ***
HR           2.750e-01  5.392e-02   5.100 5.05e-07 ***
H           -1.279e-01  3.157e-02  -4.052 6.00e-05 ***
SB           4.608e-02  1.900e-02   2.426 0.015678 *
CS          -1.959e-01  5.895e-02  -3.323 0.000964 ***
BB           5.273e-02  8.706e-03   6.057 2.98e-09 ***
GDP         -1.183e-01  3.402e-02  -3.477 0.000558 ***
IBB          1.457e-01  4.028e-02   3.618 0.000332 ***
SLG         -3.757e+02  1.041e+02  -3.608 0.000344 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 9.409 on 440 degrees of freedom
Multiple R-squared:  0.4191,    Adjusted R-squared:  0.4072
F-statistic: 35.27 on 9 and 440 DF,  p-value: < 2.2e-16
```

(Above) Final Linear Regression Model for Offensive Statistics

Therefore, from this linear regression model, it is optimal to build a team that is able to draw walks, make contact with the ball frequently and steal bases. This team's build warrants a roster of quick players that have a great eye for the strike zone and a keen ability to hit the ball into play.

In our final defensive model, there are nine variables, seven of which are statistically significant, with an R squared value of .608, meaning 61% of the variability in the wins can be explained by the defensive variables in the model. The seven variables that are statistically significant all strongly affect the number of wins a team can earn in a season, therefore they must be considered when we find out how to build a playoff team. The highest coefficient we see is that of the ERA, which is logical, as the ERA of a team determines how effective they are at allowing the opposing team to score. Therefore, a great team

will have a low ERA, as shown with the large negative coefficient. Furthermore, we see that the largest coefficients are for intentional walks and number of pitchers. The negative coefficient of intentional walks supports the intuition, as when a pitcher intentionally walks a player, it is an indication that the pitcher does not have control of the ball, and cannot properly get the batter out. Additionally, the number of pitchers has a large magnitude in the negative direction, indicating that a team wants to use as few pitchers as possible during the season. Quality pitchers will be able to play deeper into games, and pitch many games during the season as well. For this reason, having too many pitchers play on a team is an indication of a weak pitching staff, and a bad defensive team. One oddity in this regression's output is the extremely low coefficient of strikeouts, which is a miniscule .0079, implying that strikeouts are not as important for defense to win games.

Therefore, this linear regression model informs us that we must have a pitching staff that is able to prioritize control (minimizing walks and not hitting batters). If a pitching staff is focusing on control, they are sacrificing power, thus are unable to get as many strikeouts. The model therefore warrants that a pitching staff is built for control and longevity, minimizing the number of pitchers needed during the season, by having pitchers throw slower, putting less stress on their arm and maximizing the number of outs they obtain[3], hence winning more games.

```
lm(formula = W.1 ~ ERA + SV + BB.1 + HR.1 + SO.1 + IBB.1 + HBP.1 +
    WP + X.P, data = realdata)

Residuals:
    Min      1Q  Median      3Q     Max
-33.861  -4.768   0.041   5.092  23.264

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 106.880693   7.713767  13.856  < 2e-16 ***
ERA         -12.329008   1.619567  -7.613 1.65e-13 ***
SV            0.479416   0.062015   7.731 7.34e-14 ***
BB.1         -0.002017   0.009018  -0.224 0.823122
HR.1          0.043754   0.021333   2.051 0.040857 *
SO.1          0.007891   0.003655   2.159 0.031420 *
IBB.1        -0.115681   0.030460  -3.798 0.000166 ***
HBP.1         0.011848   0.036028   0.329 0.742418
WP           -0.035269   0.029428  -1.198 0.231375
X.P          -0.231896   0.090738  -2.556 0.010934 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.727 on 440 degrees of freedom
Multiple R-squared:  0.6083,    Adjusted R-squared:  0.6003
F-statistic: 75.92 on 9 and 440 DF,  p-value: < 2.2e-16
```

(Above) Final Linear Regression Model for Defensive Statistics

5b. Win Threshold Results

League	Mean_Wins
AL	89.09091
NL	88.00000
MLB	87.36364

By taking the arithmetic mean of the least winning playoff team from 2012 to 2023, we reach these AL, NL, and MLB results. As we can see, they are all very close to each other, as expected. To make the playoffs in the AL, one would need 90 wins, the NL 89, and generally speaking, would need 88 wins to make the playoffs. Thus, for future predictions, since we are dealing with a general team, we will establish 88 as the necessary win threshold.

[3] To maximize the number of outs obtained by a pitcher is minimize ERA, as ERA is inversely proportional to the number of innings pitched (low ERA means lots of innings pitched and few runs allowed)

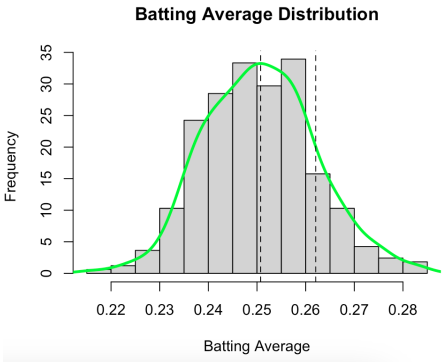
5c. Variable Distribution Results

We divided our variables into offensive and defensive stats and calculated the league average, standard deviation, and the coefficient of variation, measured as a percentage, for each of them.

Offensive Stats	BA	HR	H	SB	CS	BB	GDP	IBB	SLG
League Average	0.251	181.273	1379.812	88.636	31.364	504.182	119.091	28	0.409
Standard Deviation	0.012	39.599	78.267	29.547	8.960	63.592	16.239	12.262	0.027
Coefficient of Variation (%)	4.516	21.855	5.671	33.379	28.555	12.611	13.626	43.769	6.540

Out of the offensive statistics, IBB and SB have the highest dispersions around the mean, with the coefficient of variation equal to 43.769% and 33.379% respectively. This signifies a high variation in the number of IBB and SB. As shown in our linear regression, IBB is statistically significant (***) so its high variation points towards the fact that there are teams which could improve their performance in this area and thus their overall results. On the other hand, H and BA have the lowest coefficients of variation, 5.671% and 4.516% respectively, showing that they are concentrated around the mean. There is little variation in the number of Hits and the Batting Average.

For each of the variables, we created a histogram, as seen in this example for Batting Average:



From the graphs, we were able to confirm that all the variables follow a normal distribution. This gave us a better understanding of the variables, as well as provided us with information about the ranges of each variable, which became useful in the Monte Carlo simulation later on.

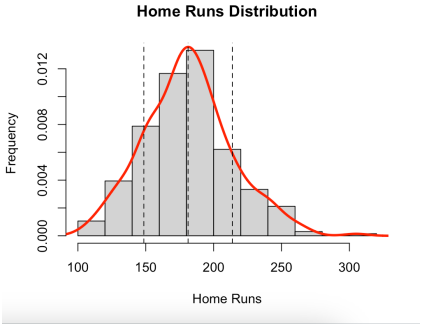
We completed the same steps for defensive stats:

Defensive Stats	ERA	SV	BB.1	HR.1	SO.1	IBB.1	HBP.1	WP	#P (X.P)
League Average	4.115	41.182	504.182	181.273	1321	28	60.091	57.909	25.909
Standard Deviation	0.548	7.377	58.675	32.601	132.809	12.777	12.967	13.267	5.376
Coefficient of Variation (%)	13.310	17.867	11.636	17.993	10.053	45.608	21.531	22.961	18.680

In a similar manner to the offensive stats, the IBB is the least concentrated around the mean with 45.608% of coefficient of variation, significantly higher than any other variable. The Strikeout and Base

on Balls are the two variables which are the most concentrated around the mean, their coefficient of variation equal 10.053% and 11.636% respectively. This means that there is little variation in the number of SO and BB between different team performances.

We plotted the histograms for each of the defensive variables:

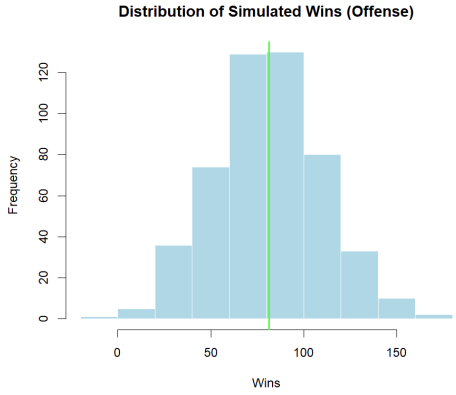


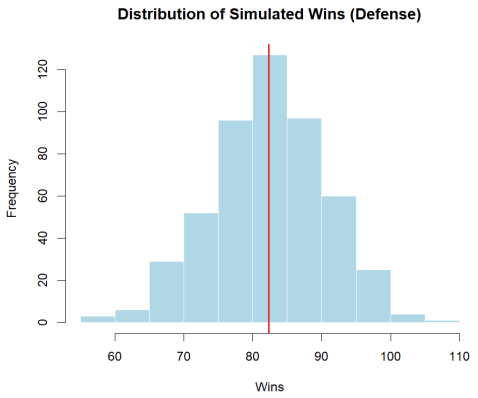
Once again, all the variables followed the normal distribution, which gave us more information on how to proceed with the Monte Carlo Simulation. The rest of the histograms for other variables can be found in Appendix D.

5d. Monte Carlo Simulation Results

The results of the Monte Carlo simulation are very helpful in helping us predict a team's future wins, given statistical expectations for the team at the beginning of the year. We treat offense and defense as two separate units, as we have for the rest of this project, and as is normal in most baseball analyses. All of these simulations that follow were run for 500 iterations.

This first simulation is run using the offensive and defensive models with league averages and league standard deviations as inputs for every relevant statistic as found in the linear regression models. We do this to ensure that our model is able to create a distribution centered close to the league average in wins, 80.63, in both cases.

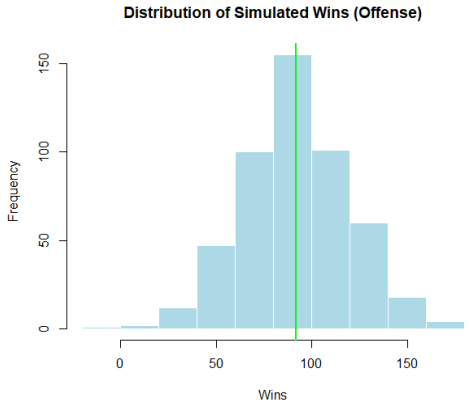




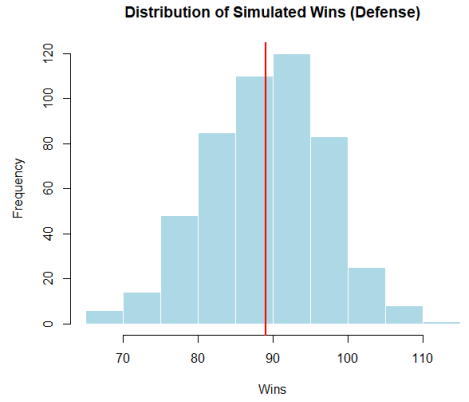
The histogram on the left shows the distribution of the 500 offensive iterations with a mean of 81.20 wins, very close to the actual league average of 80.63 wins, validating this approach as a way to predict wins. The same result stands for the histogram on the right, showing the distribution of the 500 defensive iterations with a mean of 82.3 wins. This number is still very close to the mean of 80.63, and we, therefore, now proceed to create a team that can make the playoffs by reaching a mean value of 88 wins.

We now move on to constructing a team that can reach the 88-win threshold as described earlier. Theoretically speaking, there are infinite ways to approach this problem; one could move variables in

every direction and influence the mean number of wins every single time. However, for the sake of brevity, we will provide one Monte Carlo simulation using each model that reaches the winning threshold.



This distribution is formulated using a hypothetical offense that is average in all statistics; however, its batting average is now .256, and it has 191 home runs. The .256 BA comes from the fact that it is very hard to raise the batting average of a team over the course of one off-season. Thus, a one standard deviation shift would be hard to rationalize. However, a .005 increase, which is about 40% of one standard deviation, is much more manageable. Furthermore, increasing home runs to 191 is still less than one standard deviation from the mean and represents hitting ten more home runs than the league average as a team. This feat is very doable by adding players known for their home run abilities. After running this Monte Carlo simulation, we reached a mean of 91.7 wins on the year, signaling that we can expect this team to make the playoffs.



This distribution comes about for a hypothetical defense that is average everywhere except in ERA, SV, BB.1, and WP. The reason this requires much more manipulation than the offensive simulation arises from the fact that the defensive stats are much tighter in distribution (see 5c or Appendix D). ERA was lowered to 3.85, a fairly decent decrease from the mean but not yet one standard deviation from the mean. Lowering this ERA could be done by acquiring a very good starting pitcher with a low ERA, relievers who do not let up many earned runs, or fielders who consistently make good defensive plays. Thus, any combination of these three will lead the team in the right direction of lowering their ERA. Saves are increased to 46, almost one standard deviation above the mean, which requires that relief pitchers and, specifically, closers are able to hold onto leads. This increase can be attained by signing a very good closer that will close the game out with the lead intact. BB.1 or walks is decreased to 485, nearly a third of a standard deviation from the mean, and is a very reasonable decrease to make if the team is already focusing on acquiring pitchers with control based on the other factors. Finally, wild pitches are decreased to 50, around half a standard deviation from the mean, and by the same logic as walks, wild pitches will naturally decrease by improving the control qualities of the pitching staff. Using these statistics results in an average number of 89.02 wins, above the 88-win threshold, and into the playoffs.

6. Conclusions

The aim of our project was to analyze baseball offensive and defensive statistics in order to make recommendations for the teams to optimize their roster in the offseason to make the postseason. Based on the significance of various variables, we wanted to be able to advise teams on how to maximize their chance of reaching the playoffs. Using statistical tools such as linear regressions and Monte Carlo simulation, we found that when it comes to offensive statistics, Batting Average (BA) and Ground into Double Play (GDP) have statistically the highest influence on the number of wins by a team. Out of the variables related to defense, we showed that Earned Run Average (ERA), Intentional Base on Balls (IBB), and the Number of Pitchers (X.P) are the most important to a team's performance. Surprisingly, Slugging (SLG) and Strikeouts (SO) turned out not to be as important as we initially expected them to be.

These results could be used as a guide for baseball teams on what to focus on in order to increase their chances of advancing to the playoffs. The offensive results yielded from this experiment inform us that a team can bolster their roster with quick, contact-focused players to increase their batting average and number of stolen bases and decrease the number of double plays invoked. On the defensive side, our experiment showed that in addition to increasing the team's ERA, it is important to improve control when pitching, as wild pitches, intentional walks, and other metrics of unruly pitching dampen a team's ability to get more wins.

For future work, we can run more linear regression models that instead consider the relationship between fielding statistics and wins. However, current fielding metrics are not as accurate and consistent as hitting and pitching statistics. In addition, given more time and more statisticians, this experiment can be run on a larger set of data, as baseball statistics have been tracked for almost a hundred years. We can also create a more fine-grained approach to this analysis by considering the statistics for each position on offense and defense, such as having a designated table for shortstops, closing pitchers, and starting pitchers. Finally, additional information could be gained by conducting a more in-depth analysis of the performance of each team in comparison with the others. In our project, we focused on the averages for the whole league, so calculating them for each team across the years could provide additional insight. It would make it easier to compare our findings with the actual performance of each team over the years.

7. Appendices

Appendix A: CSV File Guides

Guide to Master_Stats.csv:
Green: Offensive Statistics
Red: Defensive Statistics
NOTE: in R, to reference a defensive statistic that have same column name as an offensive statistic use "stat.1"

Year	Tm	#Bat	…	IBB	LOB	#P	PAge	…	LOB.1	W.1
2008	Arizona Diamondbacks	41	…	49	1142	20	29.4	…	1109	82
2008	Atlanta Braves	49	…	56	1274	28	28.6	…	1144	72
…	…	…	…	…	…	…	…	…	…	…
2023	Washington Nationals	48	…	6	1082	28	28	…	1167	71

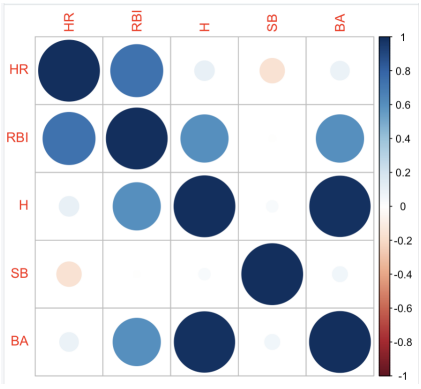
Guide to Master_Average_Stats.csv:
Green: Offensive Stats
Red/Orange: Defensive Stats
Yellow: League Average Wins
Pink: Relevant Seasons averaged in last row (only want seasons with current playoff format of 2 wild card teams)

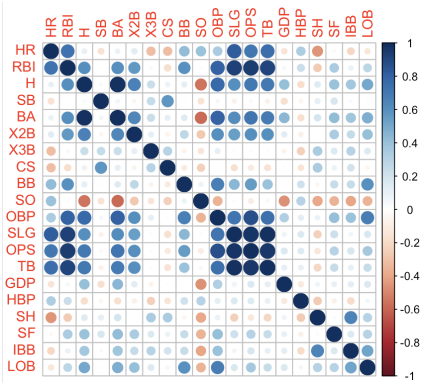
Year	Tm	#Bat	…	IBB	LOB	#P	PAge	…	SO/W	LOB	W
2008	League Average	43	…	44	1166	22	28.6	…	2.01	1166	80
2009	League Average	42	…	39	1161	22	28.4	…	2.02	1161	81
2010	League Average	42	…	41	1144	21	28.4	…	2.17	1144	81
2011	League Average	43	…	41	1128	22	28.3	…	2.3	1128	80
2012	League Average	43	…	35	1103	22	28.4	…	2.48	1103	81
…	…	…	…	…	…	…	…	…	…	…	…
2023	League Average	49	…	16	1099	29	28.9	…	2.65	1099	81
2012-2023	**Average of stats (2012-2023)**	**46.090909091**	**…**	**28**	**1099.09091**	**25.909090909**	**28.5181818**	**…**	**2.62**	**1099.09091**	**80.6363636**

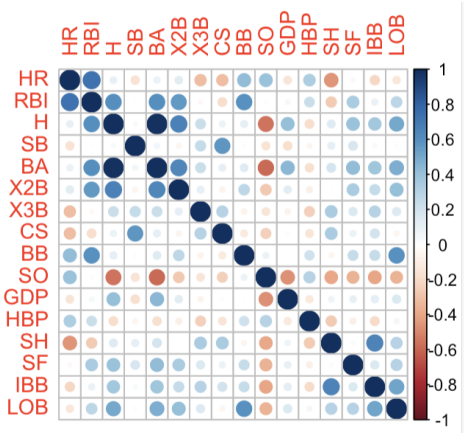
Appendix B: Linear Regression Models and Correlation Plots

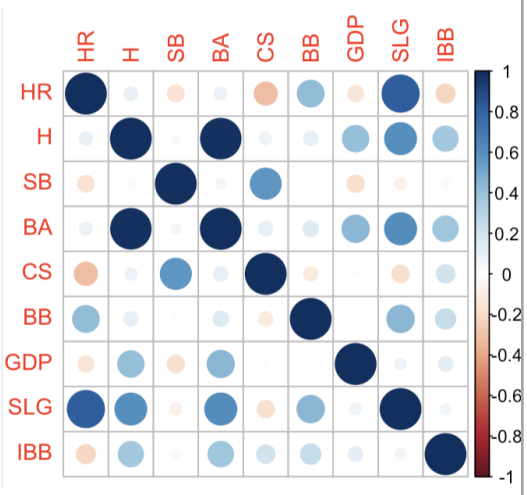
Offensive Statistic Models:

Model Number	Linear Regression Model (dependent on season wins)	Correlation Between Variables

Model 1 (Baseline)	*Base 5 ESPN Statistics: BA, HR, RBI, H, SB*
	`Call:`
	`lm(formula = W.1 ~ BA + HR + RBI + H + SB, data = realdata)`
	
	`Residuals:`
	`Min 1Q Median 3Q Max`
	`-24.1411 -6.6796 0.5602 6.0668 29.2113`
	
	`Coefficients:`
	`Estimate Std. Error t value Pr(>|t|)`
	`(Intercept) -8.98107 15.16602 -0.592 0.554029`
	`BA 717.17788 220.73773 3.249 0.001246 **`
	`HR -0.02674 0.02281 -1.172 0.241652`
	`RBI 0.11402 0.01457 7.825 3.76e-14 ***`
	`H -0.11899 0.03144 -3.784 0.000175 ***`
	`SB 0.01063 0.01580 0.673 0.501438`
	`---`
	`Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1`
	
	`Residual standard error: 9.714 on 444 degrees of freedom`
	`Multiple R-squared: 0.3752, Adjusted R-squared: 0.3682`
	`F-statistic: 53.34 on 5 and 444 DF, p-value: < 2.2e-16`
		

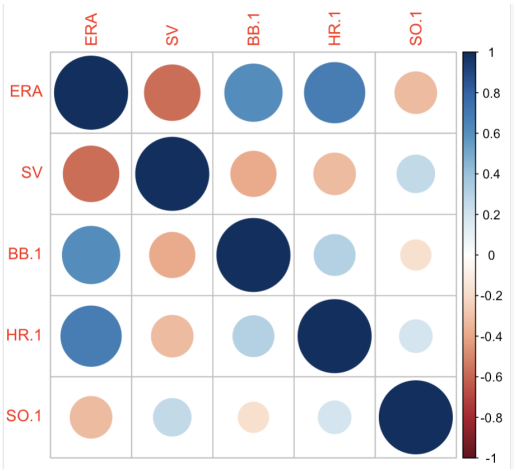
Model 2	*Add all variables*
	`Coefficients: (1 not defined because of singularities)`
	`Estimate Std. Error t value Pr(>|t|)`
	`(Intercept) 2.771e+01 4.119e+01 0.673 0.501445`
	`BA 1.624e+03 1.257e+03 1.292 0.196971`
	`HR 3.990e-01 4.546e-01 0.878 0.380514`
	`RBI 2.996e-02 3.389e-02 0.884 0.377158`
	`H -1.092e-01 8.684e-02 -1.257 0.209430`
	`SB 2.570e-02 1.886e-02 1.363 0.173697`
	`X2B 7.195e-02 1.516e-01 0.475 0.635266`
	`X3B 5.100e-02 3.121e-01 0.163 0.870267`
	`CS -1.943e-01 7.339e-02 -2.648 0.008400 **`
	`BB 5.466e-02 1.188e-01 0.460 0.645581`
	`SO -1.140e-02 4.659e-03 -2.448 0.014778 *`
	`OBP 2.451e+02 1.229e+03 0.199 0.842047`
	`SLG -7.112e+02 1.171e+03 -0.607 0.543953`
	`OPS -5.027e+01 8.937e+02 -0.056 0.955166`
	`TB NA NA NA NA`
	`GDP -1.578e-01 4.718e-02 -3.345 0.000893 ***`
	`HBP 8.134e-02 1.220e-01 0.667 0.505336`
	`SH -6.666e-02 3.387e-02 -1.968 0.049723 *`
	`SF 1.025e-01 8.500e-02 1.206 0.228405`
	`IBB 2.571e-01 4.837e-02 5.316 1.71e-07 ***`
	`LOB -6.608e-02 3.309e-02 -1.997 0.046485 *`
	`---`
	`Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1`
	
	`Residual standard error: 8.983 on 430 degrees of freedom`
	`Multiple R-squared: 0.4825, Adjusted R-squared: 0.4597`
	`F-statistic: 21.1 on 19 and 430 DF, p-value: < 2.2e-16`
		

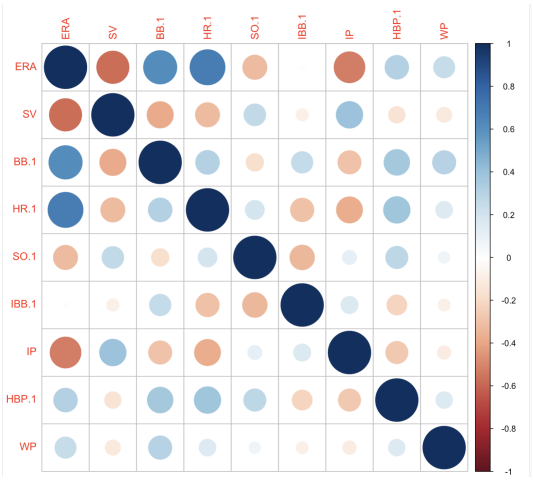
Model 3	*Remove Highly Correlated Variables (OBP, SLG, OPS, TB)*
	`Coefficients:`
	`Estimate Std. Error t value Pr(>|t|)`
	`(Intercept) 31.336293 23.559292 1.330 0.184184`
	`BA 653.189418 242.374275 2.695 0.007314 **`
	`HR -0.014816 0.027392 -0.541 0.588856`
	`RBI 0.031823 0.033684 0.945 0.345308`
	`H -0.039950 0.046726 -0.855 0.393037`
	`SB 0.024759 0.018768 1.319 0.187811`
	`X2B -0.063668 0.023701 -2.686 0.007502 **`
	`X3B -0.228566 0.058861 -3.883 0.000119 ***`
	`CS -0.186886 0.072513 -2.577 0.010288 *`
	`BB 0.073512 0.031856 2.308 0.021489 *`
	`SO -0.011283 0.004643 -2.430 0.015492 *`
	`GDP -0.156476 0.046970 -3.331 0.000938 ***`
	`HBP 0.097757 0.045693 2.139 0.032961 *`
	`SH -0.067455 0.033539 -2.011 0.044917 *`
	`SF 0.092927 0.065511 1.418 0.156764`
	`IBB 0.255629 0.048143 5.310 1.76e-07 ***`
	`LOB -0.063563 0.032868 -1.934 0.053777 .`
	`---`
	`Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1`
	
	`Residual standard error: 8.962 on 433 degrees of freedom`
	`Multiple R-squared: 0.4814, Adjusted R-squared: 0.4622`
	`F-statistic: 25.12 on 16 and 433 DF, p-value: < 2.2e-16`
		

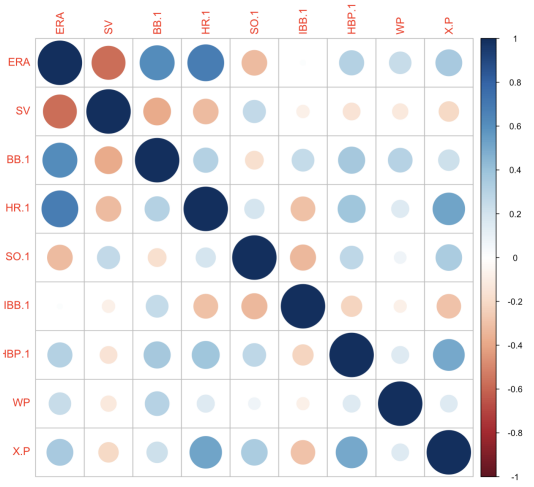
Model 4 (FINAL)	*Remove Extra Base Hits (X2B, X3B, RBI),*
	Remove SH, SF, LOB, SO, HBP
	Add SLG
	`lm(formula = W.1 ~ BA + HR + H + SB + CS + BB + GDP + IBB + SLG, data = realdata)`
	`Residuals:`
	`Min 1Q Median 3Q Max`
	`-26.0064 -6.0207 -0.3367 5.8521 28.8453`
	`Coefficients:`
	`Estimate Std. Error t value Pr(>|t|)`
	`(Intercept) -5.969e+01 1.409e+01 -4.235 2.79e-05 ***`
	`BA 1.618e+03 2.719e+02 5.951 5.45e-09 ***`
	`HR 2.750e-01 5.392e-02 5.100 5.05e-07 ***`
	`H -1.279e-01 3.157e-02 -4.052 6.00e-05 ***`
	`SB 4.608e-02 1.900e-02 2.426 0.015678 *`
	`CS -1.959e-01 5.895e-02 -3.323 0.000964 ***`
	`BB 5.273e-02 8.706e-03 6.057 2.98e-09 ***`
	`GDP -1.183e-01 3.402e-02 -3.477 0.000558 ***`
	`IBB 1.457e-01 4.028e-02 3.618 0.000332 ***`
	`SLG -3.757e+02 1.041e+02 -3.608 0.000344 ***`
	`---`
	`Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1`
	`Residual standard error: 9.409 on 440 degrees of freedom`
	`Multiple R-squared: 0.4191, Adjusted R-squared: 0.4072`
	`F-statistic: 35.27 on 9 and 440 DF, p-value: < 2.2e-16`
		

Defensive Statistic Models:

Model Number	Linear Regression Model (dependent on season wins)	Correlation Between Variables

Model 5 (Baseline)	*Base 5 ESPN Statistics: ERA, SV, BB, HR, SO*
	`lm(formula = W.1 ~ ERA + SV + BB.1 + HR.1 + SO.1, data = realdata)`
	`Residuals:`
	`Min 1Q Median 3Q Max`
	`-35.214 -5.197 0.084 5.525 21.314`
	`Coefficients:`
	`Estimate Std. Error t value Pr(>|t|)`
	`(Intercept) 103.284782 7.594853 13.599 < 2e-16 ***`
	`ERA -13.078494 1.600414 -8.172 3.20e-15 ***`
	`SV 0.498886 0.062859 7.937 1.71e-14 ***`
	`BB.1 -0.013208 0.008213 -1.608 0.10850`
	`HR.1 0.056344 0.020877 2.699 0.00722 **`
	`SO.1 0.006402 0.003321 1.928 0.05452 .`
	`---`
	`Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1`
	`Residual standard error: 7.862 on 444 degrees of freedom`
	`Multiple R-squared: 0.5907, Adjusted R-squared: 0.5861`
	`F-statistic: 128.2 on 5 and 444 DF, p-value: < 2.2e-16`
		

Model 6	*Add statistics for ball control (IP, HBP.1, WP)*
	`lm(formula = W.1 ~ ERA + SV + BB.1 + HR.1 + SO.1 + IBB.1 + IP + HBP.1 + WP, data = realdata)`
	
	`Residuals:`
	`Min 1Q Median 3Q Max`
	`-32.686 -4.671 -0.025 5.378 21.626`
	
	`Coefficients:`
	`Estimate Std. Error t value Pr(>|t|)`
	`(Intercept) -39.274158 44.013053 -0.892 0.372704`
	`ERA -11.393291 1.656289 -6.879 2.08e-11 ***`
	`SV 0.461630 0.062122 7.431 5.66e-13 ***`
	`BB.1 -0.001838 0.008967 -0.205 0.837666`
	`HR.1 0.032015 0.021153 1.513 0.130881`
	`SO.1 0.005681 0.003494 1.626 0.104635`
	`IBB.1 -0.127228 0.030680 -4.147 4.04e-05 ***`
	`IP 0.099053 0.029187 3.394 0.000752 ***`
	`HBP.1 -0.002682 0.034607 -0.078 0.938252`
	`WP -0.041348 0.029335 -1.409 0.159400`
	`---`
	`Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1`
	
	`Residual standard error: 7.684 on 440 degrees of freedom`
	`Multiple R-squared: 0.6126, Adjusted R-squared: 0.6047`
	`F-statistic: 77.31 on 9 and 440 DF, p-value: < 2.2e-16`
		

Model 7 (FINAL)	*Remove Highly Correlated Variables (IP), Add X.P*
	`lm(formula = W.1 ~ ERA + SV + BB.1 + HR.1 + SO.1 + IBB.1 + HBP.1 + WP + X.P, data = realdata)`
	
	`Residuals:`
	`Min 1Q Median 3Q Max`
	`-33.861 -4.768 0.041 5.092 23.264`
	
	`Coefficients:`
	`Estimate Std. Error t value Pr(>|t|)`
	`(Intercept) 106.880693 7.713767 13.856 < 2e-16 ***`
	`ERA -12.329008 1.619567 -7.613 1.65e-13 ***`
	`SV 0.479416 0.062015 7.731 7.34e-14 ***`
	`BB.1 -0.002017 0.009018 -0.224 0.823122`
	`HR.1 0.043754 0.021333 2.051 0.040857 *`
	`SO.1 0.007891 0.003655 2.159 0.031420 *`
	`IBB.1 -0.115681 0.030460 -3.798 0.000166 ***`
	`HBP.1 0.011848 0.036028 0.329 0.742418`
	`WP -0.035269 0.029428 -1.198 0.231375`
	`X.P -0.231896 0.090738 -2.556 0.010934 *`
	`---`
	`Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1`
	
	`Residual standard error: 7.727 on 440 degrees of freedom`
	`Multiple R-squared: 0.6083, Adjusted R-squared: 0.6003`
	`F-statistic: 75.92 on 9 and 440 DF, p-value: < 2.2e-16`
		

Appendix C: R Scripts

Linear Regression Model:

```
rm(list=ls())
library(dplyr)
library(corrplot)
data <- read.csv("Master_Stats.csv")

#omit irrelevant stats, such as batter age, plate appears
realdata <- select(data, -X.Bat, -G,PAge, -L, -W.L., -G.1, -GS, -GF, -CG,
                   -tSho, -cSho, -BK, -BatAge, -R,
                    -AB, -PA, -PAge,)

colnames(realdata)

#offense: start at 5 stats and iterate to see different models
base  <- lm(W.1~ BA + HR + RBI + H + SB, data=realdata)
b_2B  <- lm(W.1~ BA + HR + RBI + H + SB + X2B, data=realdata)
b_3B  <- lm(W.1~ BA + HR + RBI + H + SB + X2B + X3B, data=realdata)
b_CS  <- lm(W.1~ BA + HR + RBI + H + SB + X2B + X3B + CS, data=realdata)
b_BB  <- lm(W.1~ BA + HR + RBI + H + SB + X2B + X3B + CS + BB, data=realdata)
b_SO  <- lm(W.1~ BA + HR + RBI + H + SB + X2B + X3B + CS + BB + SO, data=realdata)
b_OBP <- lm(W.1~ BA + HR + RBI + H + SB + X2B + X3B + CS + BB + SO + OBP, data=realdata)
b_SLG <- lm(W.1~ BA + HR + RBI + H + SB + X2B + X3B + CS + BB + SO + OBP + SLG, data=realdata)
b_OPS <- lm(W.1~ BA + HR + RBI + H + SB + X2B + X3B + CS + BB + SO + OBP + SLG + OPS, data=realdata)
b_TB  <- lm(W.1~ BA + HR + RBI + H + SB + X2B + X3B + CS + BB + SO + OBP + SLG + OPS + TB, data=realdata)
b_GDP <- lm(W.1~ BA + HR + RBI + H + SB + X2B + X3B + CS + BB + SO + OBP + SLG + OPS + TB + GDP, data=realdata)
b_HBP <- lm(W.1~ BA + HR + RBI + H + SB + X2B + X3B + CS + BB + SO + OBP + SLG + OPS + TB + GDP + HBP, data=realdata)
b_SH  <- lm(W.1~ BA + HR + RBI + H + SB + X2B + X3B + CS + BB + SO + OBP + SLG + OPS + TB + GDP + HBP + SH, data=realdata)
b_SF  <- lm(W.1~ BA + HR + RBI + H + SB + X2B + X3B + CS + BB + SO + OBP + SLG + OPS + TB + GDP + HBP + SH + SF, data=realdata)
b_IBB <- lm(W.1~ BA + HR + RBI + H + SB + X2B + X3B + CS + BB + SO + OBP + SLG + OPS + TB + GDP + HBP + SH + SF + IBB, data=realdata)
```

```r
b_LOB <- lm(W.1~ BA + HR + RBI + H + SB + X2B + X3B + CS + BB + SO + OBP + SLG + OPS + TB + GDP + HBP + SH + SF + IBB + LOB, data=realdata)
b_remove <- lm(W.1~ BA + HR + RBI + H + SB + X2B + X3B + CS + BB + SO + GDP + HBP + SH + SF + IBB + LOB, data=realdata)

base_d <- lm(W.1 ~ ERA + SV + BB.1 + HR.1 + SO.1, data= realdata)

#load most important Offensive stats into a model
offense <- lm(W.1 ~ BA + HR + H + SB + CS + BB + GDP + IBB + SLG, data = realdata)

#load most important Defensive stats into a model
defense <- lm(W.1 ~ ERA + SV + BB.1 + HR.1 + SO.1 + IBB.1 + IP + HBP.1+ WP, data=realdata)

#try combining both stats (led to overfit data)
combined  <- lm(W.1 ~ BA + HR + H + SB + CS + BB + GDP + IBB + SLG + ERA + SV + BB.1 + HR.1 + SO.1 + IBB.1 + WP + X.P, data=realdata)
summary(offense)
summary(defense)
summary(combined)

#check correlation between offensive variables
stats_o<-c("HR", "H", "SB", "BA", "CS", "BB", "GDP", "SLG", "IBB")

#check correlation between defensive variables
stats_d<- c("ERA", "SV", "BB.1", "HR.1", "SO.1", "IBB.1", "IP", "HBP.1",  "WP")

correlation_matrix <- cor(realdata[, stats_o])
corrplot(cor(realdata[, stats_o]))
print(correlation_matrix)

correlation_matrix <- cor(realdata[, stats_d])
corrplot(cor(realdata[, stats_d]))
print(correlation_matrix)
```

Win Thresholds:

```r
# Win thresholds
AL <- data.frame(wins = c(87, 86, 92, 96, 91, 85, 89, 86, 88, 92, 88))
mean_ALwins <- mean(AL$wins)
NL <- data.frame(wins = c(84, 87, 88, 89, 90, 87, 87, 90, 88, 90, 88))
mean_NLwins <- mean(NL$wins)
MLB <- data.frame(wins = c(84, 86, 88, 89, 90, 85, 87, 86, 88, 90, 88))
mean_MLBwins <- mean(MLB$wins)

# Creating a table
wins_table <- data.frame(
  League = c("AL", "NL", "MLB"),
  Mean_Wins = c(mean_ALwins, mean_NLwins, mean_MLBwins)
)

# View the table
print(wins_table)
```

Probability Distribution Histograms:

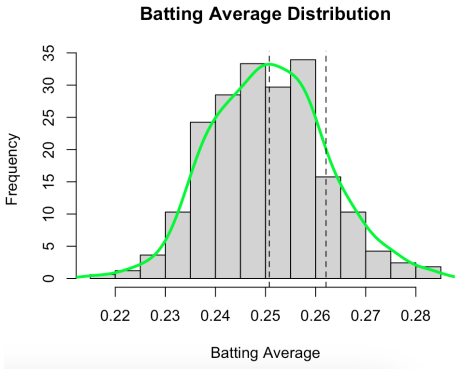
```r
#histograms
hist(data$HR, probability = TRUE, xlab = "Home Runs", ylab = "Frequency", main = "Home Run Distribution")
lines(density(data$HR), col = 'green', lwd = 3)
abline(v = mean(data$HR), lty = 2)
abline(v = mean(data$HR)+sd(data$HR), lty =2)
abline(v = mean(data$HR)-sd(data$HR), lty =2)
```

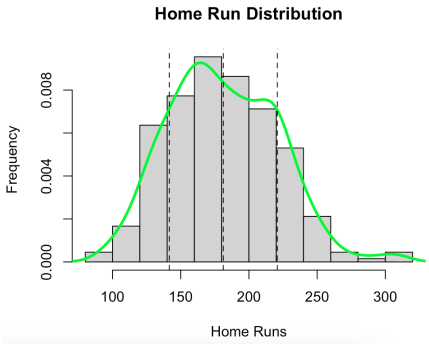
Monte Carlo Simulation:

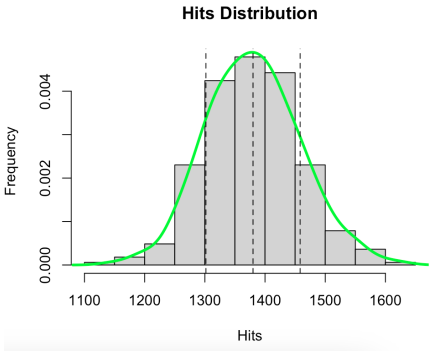
```r
rm(list=ls())
library(dplyr)
library(corrplot)
data <- read.csv("Master_Stats.csv")

#load most important Offensive stats into a model
offense <- lm(W.1 ~ BA + HR + H + SB + CS + BB + GDP + IBB + SLG, data = data)

#load most important Defensive stats into a model
defense <- lm(W.1 ~ ERA + SV + BB.1 + HR.1 + SO.1 + IBB.1 + WP + X.P, data=data)

#summaries for understanding
summary(offense)
summary(defense)

#creating Monte carlo offense
monte_carlo_regression_off <- function(num_simulations, offense, specific_values, variances) {
  # Extract coefficients from the original model
  original_coefficients <- coef(offense)

  # Initialize a matrix to store simulation results
  simulation_results <- matrix(NA, nrow = num_simulations, ncol = length(original_coefficients))

  # Monte Carlo simulations
  for (i in 1:num_simulations) {
    #Introducing variation in the input variables
    simulated_data <- data.frame(
      BA = rnorm(1, mean = specific_values$BA, sd = variances$BA),
      HR = rnorm(1, mean = specific_values$HR, sd = variances$HR),
      H = rnorm(1, mean = specific_values$H, sd = variances$H),
      SB = rnorm(1, mean = specific_values$SB, sd = variances$SB),
      CS = rnorm(1, mean = specific_values$CS, sd = variances$CS),
      BB = rnorm(1, mean = specific_values$BB, sd = variances$BB),
      GDP = rnorm(1, mean = specific_values$GDP, sd = variances$GDP),
```

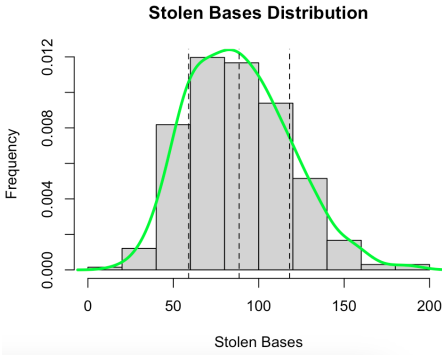
```
      IBB = rnorm(1, mean = specific_values$IBB, sd = variances$IBB),
      SLG = rnorm(1, mean = specific_values$SLG, sd = variances$SLG)
    )

    # Predict using the original coefficients and simulated data above
    simulated_offense <- predict(offense, newdata = simulated_data)

    # Store results
    simulation_results[i, ] <- coef(lm(simulated_offense ~ BA + HR + H + SB + CS
                                       + BB + GDP + IBB + SLG, data = simulated_data))
  }

  colnames(simulation_results) <- names(original_coefficients)
  return(simulation_results)
}

# Set specific values for each variable of interest
specific_values <- list(BA = 0.256, HR = 191, H = 1379, SB = 88, CS = 31, BB = 504,
                        GDP = 119, IBB = 28, SLG = 0.409)

# Set standard deviations (variances) for each variable
variances <- list(BA = 0.012, HR = 39.599, H = 78.267, SB = 29.547, CS = 8.96,
                  BB = 63.592, GDP = 16.239, IBB = 12.262, SLG = 0.027)

# Run Monte Carlo simulation with specific values and variations
num_simulations <- 500
results <- monte_carlo_regression_off(num_simulations, offense, specific_values, variances)
# Extract the 'Intercept' column from the 'results' matrix- Wins
intercept_column <- results[, "(Intercept)"]

# Calculate the mean of the 'Intercept' column- Mean Wins
mean_intercept <- mean(intercept_column)

# Plotting a histogram of simulated intercepts for offense
hist(intercept_column, main = "Distribution of Simulated Wins (Offense)", xlab = "Wins", ylab = "Frequency",
     col = "lightblue", border = "white")
abline(v = mean_intercept, col = "green", lwd = 2)  # Add a red vertical line for the mean to distinguish

#creating Monte carlo defense

monte_carlo_regression_def <- function(num_simulations, defense, specific_values, variances) {
  # Extract coefficients from the original model
  original_coefficientsd <- coef(defense)

  # Initialize a matrix to store simulation results
  simulation_resultsd <- matrix(NA, nrow = num_simulations, ncol = length(original_coefficientsd))

  #Monte Carlo simulations
  for (i in 1:num_simulations) {
    #Introducing variation in the input variables
    simulated_data <- data.frame(
      ERA = rnorm(1, mean = specific_values$ERA, sd = variances$ERA),
      SV = rnorm(1, mean = specific_values$SV, sd = variances$SV),
      BB.1 = rnorm(1, mean = specific_values$BB.1, sd = variances$BB.1),
      HR.1 = rnorm(1, mean = specific_values$HR.1, sd = variances$HR.1),
      SO.1 = rnorm(1, mean = specific_values$SO.1, sd = variances$SO.1),
      IBB.1 = rnorm(1, mean = specific_values$IBB.1, sd = variances$IBB.1),
      WP = rnorm(1, mean = specific_values$WP, sd = variances$WP),
      X.P = rnorm(1, mean = specific_values$X.P, sd = variances$X.P)
    )

    # Predict using the original coefficients and simulated data
    simulated_defense <- predict(defense, newdata = simulated_data)

    # Store results
    simulation_resultsd[i, ] <- coef(lm(simulated_defense ~ ERA + SV + BB.1 +
                                        HR.1 + SO.1 + IBB.1 + WP + X.P, data = simulated_data))
  }

  colnames(simulation_resultsd) <- names(original_coefficientsd)
  return(simulation_resultsd)
}

# Set specific values for each variable of interest
specific_values <- list(ERA = 3.85, SV = 46, BB.1 = 485, HR.1 = 181.273,
                        SO.1 = 1321, IBB.1 = 28, WP = 50, X.P = 25.909)

# Set standard deviations (variances) for each variable
variances <- list(ERA = 0.548, SV = 7.377, BB.1 = 58.675, HR.1 = 32.601, SO.1 = 132.809,
                  IBB.1 = 12.777, WP = 13.267, X.P = 5.376)

# Run Monte Carlo simulation with specific values and variations
num_simulations <- 500
results <- monte_carlo_regression_def(num_simulations, defense, specific_values, variances)

# Extract the 'Intercept' column from the 'results' matrix- Wins
intercept_column <- results[, "(Intercept)"]
# Calculate the mean of the 'Intercept' column- Mean wins
mean_intercept <- mean(intercept_column)


# Plotting a histogram of simulated intercepts for the 'defense' model
hist(intercept_column, main = "Distribution of Simulated Wins (Defense)", xlab =
       "Wins", ylab = "Frequency", col = "lightblue", border = "white")
abline(v = mean_intercept, col = "red", lwd = 2)  # Add a red vertical line for the mean
```

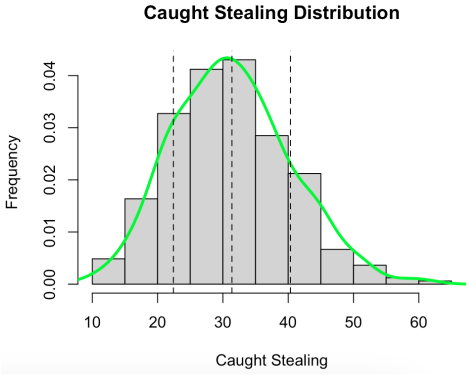
Appendix D: Histograms

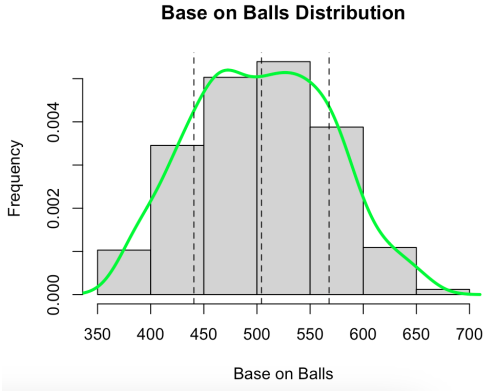


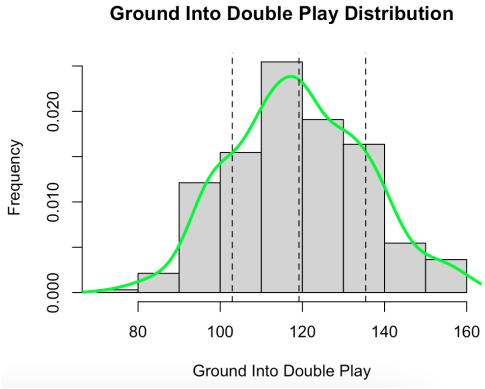


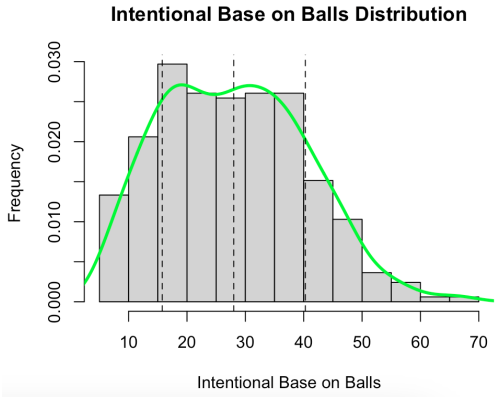


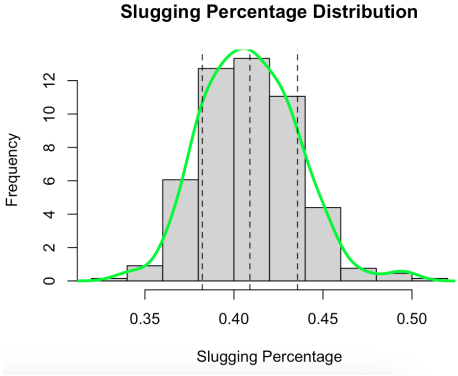


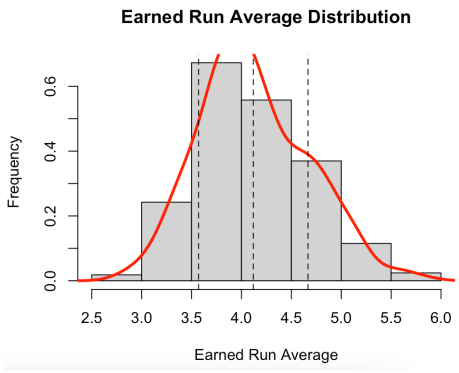




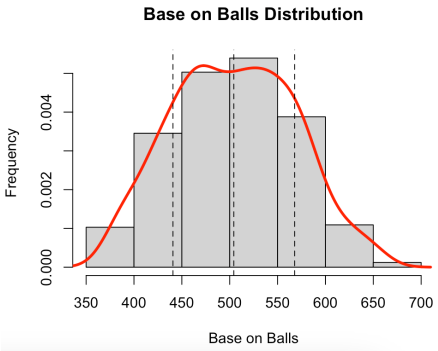


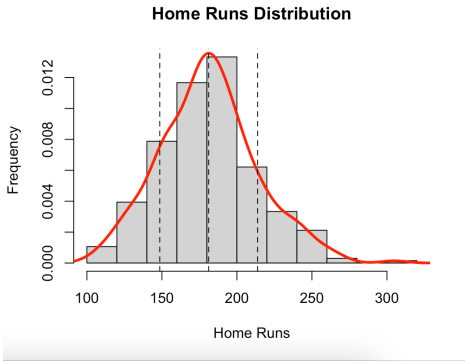


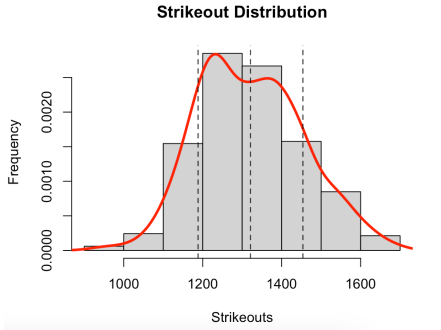


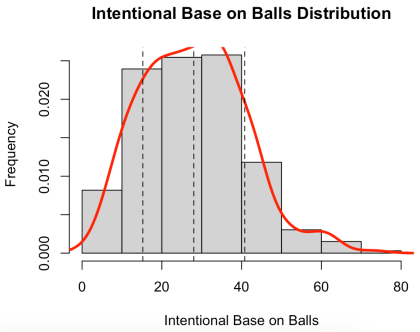

Earned Run Average Distribution
Frequency
0.0
0.2
0.4
0.6
2.5
3.0
3.5
4.0
4.5
5.0
5.5
6.0
Earned Run Average

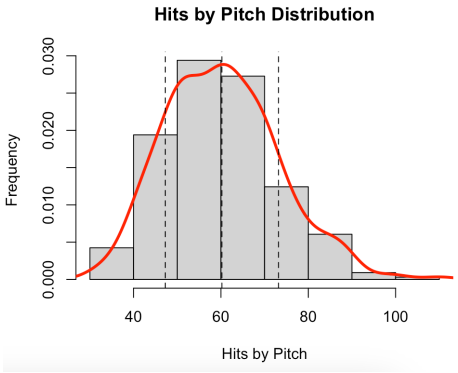

Saves Distribution
Frequency
0.00
0.02
0.04
20
30
40
50
60
Saves

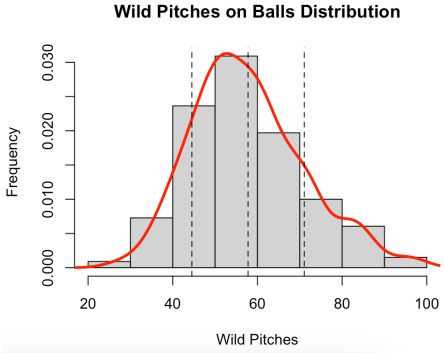

Base on Balls Distribution
Frequency
0.000
0.002
0.004
350
400
450
500
550
600
650
700
Base on Balls

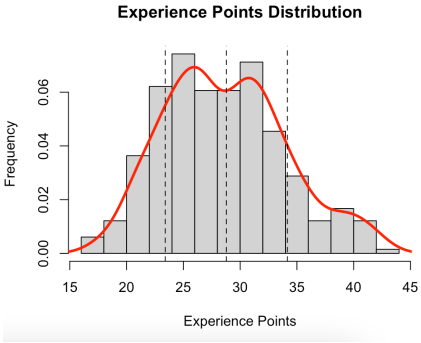

Home Runs Distribution
Frequency
0.000
0.004
0.008
0.012
100
150
200
250
300
Home Runs


Strikeout Distribution
Frequency
0.0000
0.0010
0.0020
1000
1200
1400
1600
Strikeouts


Intentional Base on Balls Distribution
Frequency
0.000
0.010
0.020
0
20
40
60
80
Intentional Base on Balls


Hits by Pitch Distribution
Frequency
0.000
0.010
0.020
0.030
40
60
80
100
Hits by Pitch


Wild Pitches on Balls Distribution
Frequency
0.000
0.010
0.020
0.030
20
40
60
80
100
Wild Pitches


Experience Points Distribution
Frequency
0.00
0.02
0.04
0.06
15
20
25
30
35
40
45
Experience Points

Glossary of Statistical Abbreviations:

Offensive Statistics:	**Defensive Statistics:**
#Bat -- Number of Players used in Games **BatAge -- Batters' average age** Weighted by AB + Games Played **R/G** -- Runs Scored Per Game **G** -- Games Played or Pitched **PA -- Plate Appearances** **AB** -- At Bats **R** -- Runs Scored/Allowed **H** -- Hits/Hits Allowed **HR** -- Home Runs Hit/Allowed **RBI** -- Runs Batted In **SB** -- Stolen Bases **CS** -- Caught Stealing **BB** -- Bases on Balls/Walks **SO** -- Strikeouts **BA -- Hits/At Bats** **OBP -- (H + BB + HBP)/(At Bats + BB + HBP + SF)** **SLG -- Total Bases/At Bats or** **(1B + 2*2B + 3*3B + 4*HR)/AB** **OPS -- On-Base + Slugging Percentages** **OPS+ -- OPS+ 100*[OBP/lg OBP + SLG/lg SLG - 1]** Adjusted to the player's ballpark(s) **TB -- Total Bases** Singles + 2 x Doubles + 3 x Triples + 4 x Home Runs. **GDP -- Double Plays Grounded Into** Only includes standard 6-4-3, 4-3, etc. double plays. **HBP** -- Times Hit by a Pitch. **SH** -- Sacrifice Hits (Sacrifice Bunts) **SF -- Sacrifice Flies** First tracked in 1954. **IBB -- Intentional Bases on Balls** First tracked in 1955. **LOB** -- Runners Left On Base	**#P -- Number of Pitchers used in Games** **PAge -- Pitchers' average age** Weighted by 3*GS + G + SV **RA/G** -- Runs Allowed Per Game **W** -- Wins **L** -- Losses **W-L% -- Win-Loss Percentage** W / (W + L) **ERA -- 9 * ER / IP** **G** -- Games Played or Pitched **GS** -- Games Started **GF** -- Games Finished **CG** -- Complete Game **tSho** -- Shutouts by a team No runs allowed in a game by one or more pitchers. **cSho** -- Shutouts No runs allowed and a complete game. **SV** -- Saves **IP** -- Innings Pitched **H** -- Hits/Hits Allowed **R** -- Runs Scored/Allowed **ER** -- Earned Runs Allowed **HR** -- Home Runs Hit/Allowed **BB** -- Bases on Balls/Walks **IBB -- Intentional Bases on Balls** **SO** -- Strikeouts **HBP** -- Times Hit by a Pitch. **BK** -- Balks **WP** -- Wild Pitches **BF** -- Batters Faced **ERA+ -- ERA+ 100*[lgERA/ERA]** Adjusted to the player's ballpark(s). **FIP -- Fielding Independent Pitching** this stat measures a pitcher's effectiveness at preventing HR, BB, HBP and causing SO (13*HR + 3*(BB+HBP) - 2*SO)/IP + $Constant_{lg}$ **WHIP -- (BB + H)/IP** **H9 -- 9 x H / IP** **HR9 -- 9 x HR / IP** **BB9 -- 9 x BB / IP** **SO9 -- 9 x SO / IP** **SO/W -- SO/W or SO/BB** **LOB** -- Runners Left On Base

8. Bibliography

"2008 Major League Baseball Team Statistics." *Baseball*, www.baseball-reference.com/leagues/majors/2008.shtml. Accessed 13 Dec. 2023.

"2009 Major League Baseball Team Statistics." *Baseball*, www.baseball-reference.com/leagues/majors/2009.shtml. Accessed 13 Dec. 2023.

"2010 Major League Baseball Team Statistics." *Baseball*, www.baseball-reference.com/leagues/majors/2010.shtml. Accessed 13 Dec. 2023.

"2011 Major League Baseball Team Statistics." *Baseball*, www.baseball-reference.com/leagues/majors/2011.shtml. Accessed 13 Dec. 2023.

"2012 Major League Baseball Team Statistics." *Baseball*, www.baseball-reference.com/leagues/majors/2012.shtml. Accessed 13 Dec. 2023.

"2013 Major League Baseball Team Statistics." *Baseball*, www.baseball-reference.com/leagues/majors/2013.shtml. Accessed 13 Dec. 2023.

"2014 Major League Baseball Team Statistics." *Baseball*, www.baseball-reference.com/leagues/majors/2014.shtml. Accessed 13 Dec. 2023.

"2015 Major League Baseball Team Statistics." *Baseball*, www.baseball-reference.com/leagues/majors/2015.shtml. Accessed 13 Dec. 2023.

"2016 Major League Baseball Team Statistics." *Baseball*, www.baseball-reference.com/leagues/majors/2016.shtml. Accessed 13 Dec. 2023.

"2017 Major League Baseball Team Statistics." *Baseball*, www.baseball-reference.com/leagues/majors/2017.shtml. Accessed 13 Dec. 2023.

"2018 Major League Baseball Team Statistics." *Baseball*, www.baseball-reference.com/leagues/majors/2018.shtml. Accessed 13 Dec. 2023.

"2019 Major League Baseball Team Statistics." *Baseball*, www.baseball-reference.com/leagues/majors/2019.shtml. Accessed 13 Dec. 2023.

"2021 Major League Baseball Team Statistics." *Baseball*, www.baseball-reference.com/leagues/majors/2021.shtml. Accessed 13 Dec. 2023.

"2022 Major League Baseball Team Statistics." *Baseball*, www.baseball-reference.com/leagues/majors/2022.shtml. Accessed 13 Dec. 2023.

"2023 Major League Baseball Team Statistics." *Baseball*, www.baseball-reference.com/leagues/majors/2023.shtml. Accessed 13 Dec. 2023.

"MLB Team Stat Leaders, 2023 Regular Season." *ESPN*, ESPN Internet Ventures, www.espn.com/mlb/stats/_/view/team. Accessed 13 Dec. 2023.