CARLETON UNIVERSITY

SCHOOL OF
MATHEMATICS AND STATISTICS

HONOURS PROJECT

TITLE: Analysis of Major League Baseball Data

AUTHOR: Peter L'Oiseau

SUPERVISOR: Dr. Sanjoy K. Sinha

DATE: December 14, 2018

Abstract:

With the 2015 public release of Statcast data by Major League Baseball (MLB) Advanced Media the ever expanding field of baseball analytics had a world of new possibilities opened. These data given the right applications of statistical techniques, can offer new, invaluable insights into the game of baseball. In this project, we will use almost 4 MLB seasons worth of Statcast data to create a model that will predict the expected value of a batter's weighted on base average on balls in play. The new data on launch angle and launch speed teaches us about the relationships between these controllable factors and the success of a batter. We also determine which players are making good contact of the ball but not getting the expected results and which players are getting better results than their batted ball contact would suggest.

To do this we apply knowledge of linear regression and least squares coefficients to create predictions for MLB batter's weighted on base average on balls in play. Creating the "best" robust model also requires statistical rigour and this process is done by stepwise regression, where the ideal combination of variables is chosen from a group of nine possible predictors. Ensuring we have chosen an appropriate model for projecting a player's weighted on base average on balls in play, we will perform a leave one out cross-validation. Cross-validation will put two models to the test to see which one reduces test error more effectively.

The conclusion of this project includes a discussion of the projections of the final model, how it compares to existing models of this kind and where the research in this area can go in the future. This is an exciting time for the availability of data in the baseball world and this project hopes to be an effective use of statistical techniques and procedures applied to these data.

Table of Contents

1 Introduction

2 Data Access and Collection

3 Methodology

4 Analysis

5 Conclusion

7 Appendix

List of Tables

List of Figures

1 Introduction

1.1 Baseball Analytics History

Baseball since its formal inception into a professional league (the National League) in 1876 has been a game of numbers. Record keepers accounted for what they saw on the field in box scores of every game, recording statistics such as Home runs, Batting Average and Runs Batted In for individual batters and Wins, Earned Run Average and Strike Outs for individual pitchers. As a sport, baseball is uniquely suitable for statistical analysis as it is filled with a series of discrete events, each of which lasting at maximum ten seconds. As a consequence of this, cause and effect is easily attributable, which helps to understand an individual player's value in a given play. This is not the case in sports where the play flows for a long period of time like hockey, soccer and basketball and where network effects like teamwork need to be accounted for. But as one can imagine all the statistics initially being generated were not optimal for discerning the value of an individual player. The statistical side was always present to an observer of baseball, but in 1977 when Bill James, an American writer and statistician, published his *Baseball Abstract,* the growing world of "analytical baseball statistics" became part of the public discussion of baseball.

James in his book combed over the aforementioned box scores that were published for every game and analyzed the statistics not merely as season totals, but he looked at how different game states changed the numbers. He continued to publish these abstracts every year, each with a growing level of sophistication, turning his attention to creating new statistics like Runs-Created and Defensive Efficiency with much more complex formulas than previously seen in the baseball world. James brought a statistical knowledge to the field that had not been seen on the public side of the sport at all and is considered as the pioneer of what is now known as Sabermetrics coming from the acronym SABR (Society for American Baseball Research).

Since then academic statisticians began flowing into the game to push baseball statistical research forward. Publicly and privately for teams, statisticians began taking an empirical approach to answering strategic questions like: “What order should a manager bat players in?”, “Should a player bunt the ball, to give up an out to advance a runner?” and “When should a manager replace his starting pitcher with a reliever?” Concepts like expected value, conditional probability and regression to the mean changed the way the game was played over the period of many decades. Not only did in-game strategy change but the construction of a baseball team changed and continues to change because of the new ways to gather and analyze data about the game.

Famously in the book titled *Moneyball* by Michael Lewis, the General Manager of the Oakland Athletics Billy Beane and Assistant General Manager Paul DePodesta went all in on constructing a team and playing the game based on the insights of statistical research in 2002. The Athletics who were a team without much money to spend on acquiring players relative to their competitors, the Boston Red Sox or New York Yankees, needed to find a market inefficiency in order to compete for a championship. Beane made the call to listen to statisticians inside his organization which led to very impressive results for a team with such a small budget. This began what is commonly referred to as the “Statistical Revolution” inside baseball front offices. Now in 2018 every team in Major League Baseball has a large statistics and research department with proprietary data working to gain the slightest advantage in increasing their team’s probability of winning the championship.

Prediction and modeling of team and player performance is another huge innovation of the early 2000’s Statistical Revolution. In 2003, Nate Silver an American Statistician released his revolutionary projection system PECOTA (Player Empirical Comparison and Optimization Test Algorithm). PECOTA uses a kth-nearest neighbor approach to create player comparisons and project the upcoming season’s worth of performance from an individual player. This spawned a new market for statisticians to create the best model for a projection system of individual players, as well as teams. Thus with every advance in the world of statistics, machine learning and data science, it is applied to creating the optimal baseball team and product for the fan.

Most recently for the public side of baseball statistics, there was a big step forward in the world of technology used to gather in-game data. The year 2015 brought the public release of Statcast, which is a tracking technology of high resolution cameras and lasers to produce highly specific and accurate spatial data for the ball and the players on the field. Previously, there were measurement of outcomes like where the ball was hit and what base the player was able to get to. PITCH f/x which was used from 2006-2014 was able to track the velocity of the pitch and how it moved on the x and y axis, but not as precisely as Statcast. Statcast has also allowed the measurement of a batter's swing plane, how hard the batter hits the ball, how fast every player moves, and much more. These raw data resulted in many metrics that have been added to the baseball fans repertoire of statistics such as outs above average and expected weighted on base average. These new interesting numbers are what we will be looking at in this project to try to gain a more granular understanding of which batters are over and underperforming their projected outcomes and what a batter can do to increase performance.

1.2 Metric Explication:

In this project there are new metrics derived specifically from Statcast data as well as a reliable performance metric for evaluating a player's performance. A model will be built with the new metrics as predictor variables and the reliable established one as a response. This section is built for the reader's understanding of all the metrics' formulations and their value for understanding a player's performance. The response variable is weighted on-base average (woba).

$$WOBA = \frac{0.69 \times uBB + 0.72 \times HBP + 0.89 \times 1B + 1.27 \times 2B + 1.62 \times 3B + 2.10\; HR}{AB + BB - IBB + SF + HBP},$$

where uBB is an unintentional walk and IBB is an intentional walk and SF is a sacrifice fly. HBP is a hit by pitch, 1B is a single, 2B is a double, 3B is a triple and HR is a home run. WOBA is seen as more sophisticated metric than on base percentage and batting average, as weights for each outcome are calculated to reflect the value they provide towards scoring a run.

In this study, predictor variables include: Launch Speed, Launch Angle, Spin Rate, Velocity, Release Extension, Effective Speed, Pitch Location, Pitch Type and Batted Ball Direction. Launch Speed is how fast the ball is hit by the batter in miles per hour (MPH). Launch Angle is how high the ball is hit by the batter in degrees. Spin Rate is how much spin the pitcher puts on the balls in revolutions per minute (RPM). Velocity is how fast the pitch is thrown in MPH. Release Extension is how far from the mound (pitching position) the pitcher releases the ball in feet. Effective Speed is an adjustment to the Velocity statistic based on the Release Extension of a pitcher. Pitch Location is a categorical variable where the categories are low, mid and high based on where the pitch crosses the plate strictly on the y axis. Pitch Type is another categorical variable where the pitch is sorted into fastball, breaking ball and off speed. These types of pitches are created by the pitcher gripping and throwing the ball in different ways, all of which is tracked by Statcast. Lastly, Batted Ball Direction is a categorical variable sorted based on where the ball first lands on the ground, into the pull, push and middle thirds of the field. A balled hit to the pulled third of the field is when a batter hits a ball and it first lands on the side of the field from which they do not bat (i.e. a right handed batter hits the ball to left field or vice versa), and the push side is batting the ball to the same side of the field (i.e. a right handed batter hits the ball to the right side and vice versa).

These performance metrics and predictor variables are what will be used in order to conduct a multiple linear regression analysis and build a model for the expected performance on a batter in woba on balls in play.

2 Data Access and Collection

2.1 Baseball Savant

As previously noted Statcast, a product of Major League Baseball Advanced Media (MLBAM), started releasing their data to the public at the beginning of the 2015 MLB season. The data for this project was mined on September 12th, 2018 and thus has slightly less than four full season worth of league wide data. As Statcast continues to release data, a more complete data set will be available to conduct the analysis found in this project.

Statcast data are made available to the public by MLBAM through a website called Baseball Savant run by Darren Willman and Mike Petriello. Baseball Savant through their website allows one to query the Statcast data through thirty five different filters as well as a group by, sort by and sort order function. When a query is run, all the requested data appear on the webpage and can be downloaded as a comma separated values file for analysis.

For this project data were taken for batters from the beginning of the 2015 season until September 12th 2018 for both regular and post-season play. All queries are grouped by batter who must have a minimum of twenty-five batted ball results under the specified query in order for the rate metric like woba to stabilize. The three categorical variables as previously mentioned (Pitch Location, Pitch Type and Batted Ball Location) configured in all twenty seven possible ways require separate queries. Some of these queries however don't give any batters who qualify with twenty five results, which are:

Pitch Location	Pitch Type	Batted Ball Direction	Observations
High	Breaking Ball	Push	NULL
High	Off Speed	Pull	NULL
High	Off Speed	Push	NULL
High	Off Speed	Straight	NULL
Low	Break	Straight	NULL

Table 2.1: Summary of missing data

The remaining twenty-two queries do have players that qualify with the minimum number of balls hit satisfying all three criteria listed in the table and the number of batters who do for each query is listed in the Observations column and the average number of results for each batter is listed in the Average No. Results.

Pitch Location	Pitch Type	Batted Ball Direction	Observations	Average No. Results
High	Breaking Ball	Pull	12	32.67
High	Breaking Ball	Straight	10	29
High	Fastball	Pull	251	52.61
High	Fastball	Push	309	54.96
High	Fastball	Straight	329	57.85
Mid	Breaking Ball	Pull	339	56.14
Mid	Breaking Ball	Push	209	38.53
Mid	Breaking Ball	Straight	333	55.26
Mid	Fastball	Pull	531	100.37
Mid	Fastball	Push	525	92.38
Mid*	Fastball	Straight	430	138.07
Mid	Off Speed	Pull	224	39.19
Mid	Off Speed	Push	56	31.29
Mid	Off Speed	Straight	195	39.79
Low	Breaking Ball	Pull	332	57.25
Low	Breaking Ball	Push	74	34.77
Low	Fastball	Pull	364	58.2
Low	Fastball	Push	258	44.3
Low*	Fastball	Straight	238	83.13
Low	Off Speed	Pull	216	41.73
Low	Off Speed	Push	9	27.56
Low	Off Speed	Straight	124	36.74

Table 2.2: Summary statistics for MLB data

The minimum number of results for the batter was raised to fifty in order for Baseball Savant to be able process the query.

Each query returned results for each observation in the thirty-one variables as listed below:

pitches	player_id	player_name	total_pitches	pitch_percent	ba	iso
babip	slg	woba	xwoba	xba	hits	abs
launch_speed	launch_angle	spin_rate	velocity	effective_speed	whiffs	swings
takes	eff_min_vel	release_extension	pos3_int_start_distance	pos4_int_start_distance	pos5_int_start_distance	pos6_int_start_distance
pos7_int_start_distance	pos8_int_start_distance	pos9_int_start_distance				

Table 2.3: Variables in MLB study

By combining the twenty-two data frames in R and adding three columns to correspond to the three categorical variables we queried upon, we have a data frame of 5366 observations on thirty four variables. Many of these variables are either identification variables or are highly correlated and thus are redundant for this project.

3 Methodology

3.1 Multiple Linear Regression

A multiple linear regression is an attempt to explain some response variable Y with a set of p>1 independent predictor variables $X_1,\dots, X_p$**.** Given n observations of the response variable Y and the p predictor variables, the multiple linear regression model has the form:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} \dots + \beta_p X_{ip} + \varepsilon_i \ \forall i = 1, \dots, n$$

Here ε is the error term assumed in the model due to the noise any real life data will have surrounding the underlying process. It is assumed $E(\varepsilon)=0$ and $V(\varepsilon)=\mathbf{I}\sigma^2$ where **I** is the identity matrix of size n, which means the error vector is random with mean zero and the elements of the error vector are uncorrelated. This implies: $E(Y) = X\beta$ where Y is a n x 1 vector, β is a (p+1) x 1 vector and X is an n x (p+1) matrix (the first column in X is a row of ones due to the intercept term). The β_j for j=0,…, p are parameters and in multiple linear regression they are estimated by least squares estimates $\widehat{\beta_J}$.

Least squares estimation for β_j is done by minimizing the sum of squares of errors:

$$S = \sum_{i=1}^{n} \varepsilon_i{}^2 = \sum_{i=1}^{n} (Y_i - \beta_0 - \beta_1 X_{i1} - \cdots - \beta_p X_{ip})^2$$

for the variable β_j (i.e. $\frac{\partial S}{\partial \beta_j} = 0$ and solve for β_j.) This process will result in having p+1 equations for the p+1 unknown β's and solving this system of equations gives the least squares estimates $\hat{\beta}$. These estimates will be displayed in the summary output of a linear model in R later. In matrix form, $\varepsilon'\varepsilon = (Y - \beta X)'(Y - \beta X)$ is the error sum of squares and the (p+1) x 1 vector $\hat{\beta}$ are the values that minimize that equation. The minimization process results in the normal equations $(X'X)\hat{\beta} = X'Y$, where in the case of the p+1 equations that don't depend on each other the (X'X) matrix is nonsingular, we have the least squares estimator $\hat{\beta} = (X'X)^{-1}X'Y$. The estimator $\hat{\beta}$ has the following properties:

1. It minimizes the error sum of squares $\varepsilon'\varepsilon$ regardless of the distributional properties of the errors.
2. It's elements are linear functions of the observations Y_1,…, Y_n and provide unbiased estimates for the elements of β, which have minimum variance regardless of the distributional properties of the errors.
3. If the errors are independent and $\varepsilon \sim N(0, \boldsymbol{I}\sigma^2)$, then $\hat{\beta}$ is the maximum likelihood estimate of β.

These are the properties that make least squares estimates an appropriate choice for the parameter vector in multiple linear regression and provide the foundation for a robust area of predictive analytics.

A common measure to assess to the quality of a linear regression model's fit is the R^2 statistic which is the percentage of variation explained by the model. The R^2 statistic (also called the coefficient of determination) is defined by,

$$R^2 = \frac{\sum(\widehat{Y_i} - \bar{Y})^2}{\sum(Y_i - \bar{Y})^2}$$

This will be one of the tools we will use to assess our models performance in upcoming sections.

3.2 Variable Selection Procedure

Our goal now is to create the “best” linear model given the n observations of a set of predictor variables $X_1,…, X_p$ for the response variable Y. Let $Z_1,…, Z_r$ all be functions of these predictors $X_1,…, X_p$; in the linear regression case these functions can be squares, inverses and powers, etc. In order to select the “best” model for the response variable one must choose a number t of these $Z_1,…, Z_r$. The decision of which t functions to use is a case of the bias-variance tradeoff which dominants much of the discussion of error when modelling. If one selects a large number t, bias error will decrease and result in more reliable fitted values. However, the average variance of $\hat{Y}_i$, the estimator for the ith observation of the response variable such that i=1,…,n, is equal to $t\sigma^2/n$. Therefore, selecting a smaller t will reduce the variance. In the pursuit of balancing the variance and bias of the error, a procedure must be outlined that weighs these factors. Stepwise regression is the variable selection procedure of choice here, but there is no unique procedure producing the “best” model in all cases. It is chosen because it considers the significance of the variables in the data set and is feasible for the relatively low number of predictors in our data. Stepwise regression is composed of two different procedures, forward selection and backward elimination, both of which will be examined in depth. In both cases, this procedure draws its power for the comparison of partial F-values. The partial F-value of Z_i, i=1,…, r, is calculated by inserting Z_i into the previously established regression equation, then calculating the regression coefficient $\hat{\beta}_i$ dividing it by its standard error and squaring it; $F_i = \left(\frac{\hat{\beta}_i}{s.e(\hat{\beta}_i)}\right)^2 \sim F(1, n-\upsilon, 1-\alpha)$ where υ is the number of predictors in the previous model and α is the significance level. Note in this project the t-value is used by R which is equivalent to the square of the F-value and produces the same results.

3.2.1 Forward Selection

Beginning with forward selection, the model begins simply as $Y = \beta_0 + \varepsilon$ where β_0 is the intercept and ε is the vector of errors. Next, the Z_j which is most highly correlated is added to the linear regression equation; making the new linear regression equation $Y = \beta_0 + \beta_j Z_j + \varepsilon$. A threshold for variable significance must be selected and if the variable Z_j clears this threshold the procedure continues and if it does not, the empty model, which is equivalent to $Y = \bar{Y}$, is chosen as the "best" model.

Given Z_j is significant, the next Z is chosen based on an analysis of the partial F-values of all predictors not in the regression equation. The Z_i with the largest partial F-value is added to the model and then all predictors in the model are checked for significance, the improvement or lack thereof in R^2 is noted and the process of analyzing partial F-values is restarted. Important to note, all predictors must be checked for significance when a new one is entered, because the effect of a new predictor can make predictors deemed in earlier iterations to be significant no longer so. When a predictor is entered into the model and that or any other predictor becomes insignificant the newly insignificant predictor is removed, the R^2 is noted and the partial F-values for predictors not in the model are calculated once again. Once the highest partial F-value of a predictor not in the model does not meet the threshold, the procedure is terminated and the "best" model by forward selection is obtained.

3.2.2 Backward Elimination

Stepwise regression's second component is backward elimination which does not always return the same model as forward selection but when it does, it strengthens the case for a linear model's claim as the "best". Backward elimination begins by fitting a linear regression equation with all predictors $Z_1,\ldots, Z_r$: $Y = \beta_0 + \beta_1 Z_1 + \cdots + \beta_r Z_r + \varepsilon$. Again, a threshold of significance must be selected and then all the partial F-values for the r predictors must be calculated. If all the partial F-values meet the threshold of significance the full model is obtained as the “best” model by backward elimination. If not, the predictor with the lowest partial F-value is removed from the model and then the model is refit and the R^2 value is recorded. Once the model is comprised of all predictors which are deemed to be significant, the procedure is finished and the “best” linear model is obtained by backward elimination.

3.3 Leave One Out Cross-Validation

Model evaluation is crucial to understanding the predictive power of a selected model and a common way to do this is cross-validation. Cross-validation is an approach where the n observations of the data are divided into training and test sets (in some forms of cross-validation there is also a validation set). These sets are designed to curb the problem known in modelling as overfitting. Overfitting is when the model follows the noise in the data too closely which means it will predict the training data well but have little predictive power on data it has not seen. To determine if a model is indeed following the noise too closely, the model is trained on one set and tested to see if it is overfitting on another.

In leave one out cross-validation, the test set is of size 1 and the training set is of size n-1. With this, the model is trained on data very similar to the whole data set, but the test is only on one data point so the error between the predicted value of the response variable $\hat{Y}$ in the training set and the actual value Y is relatively high. To curb this problem the process is repeated to ensure all n data points are used as the test set compared against the other n-1 points in the training set. Therefore the $MSE_i = (y_i - \hat{y_i})^2$ of a given i=1,…,n may be large but the $MSE = \frac{1}{n}\sum_{i=1}^{n} MSE_i$ is a strong estimate for the test error of the model.

Advantages of using this form of cross-validation over a validation set approach, where the data are split into random test and training sets of roughly equal size are as follows. First, this approach has far less bias because the model is trained on n-1 observations n times rather than training the model on roughly n/2 observations. Second, there is no randomness or problems with reproducibility in the selection of the training sets which implies the leave one out cross-validation estimate for test error is the same every time for a given data set. As such the leave one out cross-validation approach tends not to overestimate the error as much as other forms of cross-validation. This form of cross-validation is very computationally expensive, because the model is fit n times; when n is very large, this method is usually infeasible. Leave one out cross-validation is quite general and can be used for any kind of predictive modelling.

4 Analysis

4.1 Model Selection

	woba			
Predictors	*Estimates*	*std. Error*	*T-value*	*p*
(Intercept)	-0.6352	0.1914	-3.3188	**0.001**
launch angle	0.0008	0.0002	3.9483	<**0.001**
launch speed	0.0144	0.0004	40.5423	<**0.001**
velocity	-0.0393	0.0083	-4.7615	<**0.001**
effective speed	0.0395	0.0081	4.8526	<**0.001**
release extension	-0.0535	0.0207	-2.5841	**0.010**
spin rate	0.0001	0.0000	2.2205	**0.026**
batted ball direction 0	-0.1166	0.0032	-36.0896	<**0.001**
batted ball direction 1	-0.0709	0.0054	-13.1691	<**0.001**
pitch_location0	-0.0304	0.0053	-5.7517	<**0.001**
pitch location 2	0.0048	0.0043	1.1231	0.261
pitch type 2	-0.0047	0.0182	-0.2571	0.797
pitch type 0	0.0356	0.0174	2.0481	**0.041**
Observations	5366			

R^2 / adjusted R^2	0.428 / 0.426

The training set for the models used in the model selection process will be the 5366 observations of woba and the nine predictor variables outlined in Section 2.1. First, an F-value threshold must be chosen to remove or add a predictor to the model; here we choose F = 5 which depending on the degrees of freedom roughly corresponds to a p value of .01. We will begin with backward elimination and fit all nine predictors first order terms to woba; the summary output in R of this model is featured in Table 4.1.

Note, the three categorical variables (pitch location, pitch type and batted ball location) have multiple outputs in the table, this is because R treats them as factors and not as real numbers. In the case of these categorical variables, a baseline outcome is chosen by R and the remaining outcomes are compared relative to the baseline. Also note the t-value is merely the square root of the F-value, so we can eliminate the smallest absolute value for a t-value and in this case it's the second outcome (off speed) of pitch type which falls below the F = 5 threshold outlined earlier. Next, we fit the same model but without the predictor pitch type. Finally, note the R^2 statistic is 0.428 which is to say this model explains 42.8% of the variation in woba. We will note how this statistic changes throughout this process.

Table 4.1: Regression estimates for nine predictors

	woba			
Predictors	*Estimates*	*std. Error*	*T-value*	*p*
(Intercept)	-0.4022	0.1045	-3.8498	**<0.001**
launch angle	0.0008	0.0002	4.1531	**<0.001**
launch speed	0.0144	0.0004	40.5763	**<0.001**
Velocity	-0.0413	0.0080	-5.1397	**<0.001**
effective speed	0.0399	0.0081	4.9094	**<0.001**
release extension	-0.0497	0.0206	-2.4141	**0.016**
spin rate	0.0000	0.0000	0.3099	0.757

batted ball direction 0	-0.1160	0.0032	-36.4626	**<0.001**
batted ball direction 1	-0.0703	0.0053	-13.1674	**<0.001**
pitch_location0	-0.0317	0.0052	-6.0468	**<0.001**
pitch location 2	0.0036	0.0043	0.8433	0.399
Observations	5366			
R^2 / adjusted R^2	0.427 / 0.426			

Table 4.2: Regression estimates for eight predictors

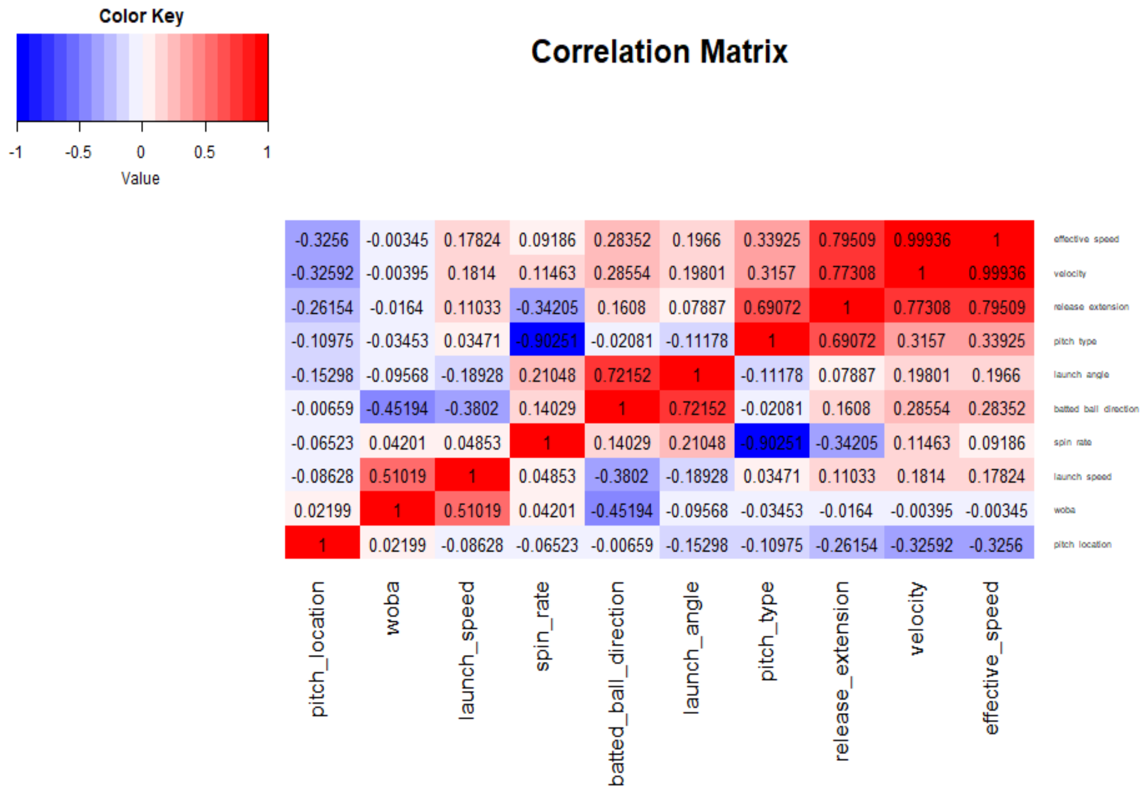
Here spin rate is the least significant predictor and it falls below the F = 5 threshold and thus is removed from the model. Here we see the change in the R^2 statistic from one model to the next is quite small, which implies that by removing pitch type from the model we have lost very little explanatory power. The process continues iteratively until all predictors are deemed significant which yields the following model:

	woba			
Predictors	*Estimates*	*std. Error*	*T-value*	*p*
(Intercept)	-0.2377	0.0890	-2.6699	**0.008**
launch angle	0.0016	0.0002	9.7544	<**0.001**
launch speed	0.0135	0.0003	39.5710	<**0.001**
velocity	-0.0500	0.0081	-6.1970	<**0.001**
effective speed	0.0495	0.0082	6.0667	<**0.001**
release extension	-0.0766	0.0173	-4.4245	<**0.001**
batted ball direction 0	-0.1145	0.0032	-35.8571	<**0.001**
batted ball direction 1	-0.0837	0.0048	-17.2861	<**0.001**
Observations	5366			
R^2 / adjusted R^2	0.414 / 0.414			

Table 4.3: Regression estimates for six predictors

Here all predictors are above the F-value threshold, so the process is concluded and the model containing the predictors launch angle, launch speed, velocity, effective speed, release extension and batted ball direction as $X_1,\ldots, X_6$ is deemed the “best” model. Little information has been lost compared to the full model, as this model explains 41.4% of the variation in woba.

Next, we shall search for the “best” model given by forward selection which begins by examining the correlations of the predictors with the response.



pitch_location	woba	launch_speed	spin_rate	batted_ball_direction	launch_angle	pitch_type	release_extension	velocity	effective_speed	
-0.3256	-0.00345	0.17824	0.09186	0.28352	0.1966	0.33925	0.79509	0.99936	1	effective speed
-0.32592	-0.00395	0.1814	0.11463	0.28554	0.19801	0.3157	0.77308	1	0.99936	velocity
-0.26154	-0.0164	0.11033	-0.34205	0.1608	0.07887	0.69072	1	0.77308	0.79509	release extension
-0.10975	-0.03453	0.03471	-0.90251	-0.02081	-0.11178	1	0.69072	0.3157	0.33925	pitch type
-0.15298	-0.09568	-0.18928	0.21048	0.72152	1	-0.11178	0.07887	0.19801	0.1966	launch angle
-0.00659	-0.45194	-0.3802	0.14029	1	0.72152	-0.02081	0.1608	0.28554	0.28352	batted ball direction
-0.06523	0.04201	0.04853	1	0.14029	0.21048	-0.90251	-0.34205	0.11463	0.09186	spin rate
-0.08628	0.51019	1	0.04853	-0.3802	-0.18928	0.03471	0.11033	0.1814	0.17824	launch speed
0.02199	1	0.51019	0.04201	-0.45194	-0.09568	-0.03453	-0.0164	-0.00395	-0.00345	woba
1	0.02199	-0.08628	-0.06523	-0.00659	-0.15298	-0.10975	-0.26154	-0.32592	-0.3256	pitch location

Figure 4.1: Correlations among response and predictor variables

Figure 4.1 presents the correlation matrix as a heat map where the most strongly correlated predictor variable with woba is launch speed at .51019. Note, there are there categorical variables which make the data heterogeneous, so polyserial correlations are taken between the categorical variables and the numeric ones while polychoric correlations are taken between the categorical variables. We will begin our modeling with $woba = \beta_0 + \beta_1 launch\_angle$. However, in what will become important later in the analysis, the top right corner of the heat map shows velocity, effective speed, pitch type and to a slightly lesser degree release extension are very highly correlated. The reason this is important is that all the predictors in a multiple linear regression model are assumed to be independent and when predictors are highly correlated, they are creating instability in the model (i.e. the same information is being captured multiple times).

	woba			
Predictors	*Estimates*	*std. Error*	*T-value*	*p*
(Intercept)	-0.9147	0.0298	-30.7375	<**0.001**
launch speed	0.0147	0.0003	43.4461	<**0.001**
Observations	5366			
R^2 / adjusted R^2	0.260 / 0.260			

Table 4.4: Regression estimates with predictor launch speed

Launch speed easily clears the F = 5 threshold and the process continues this time by analyzing the partial F values of the other 8 predictors. The R^2 value of this model is only .26, but as more predictors are added we are likely to see an increase in this number.

	woba			
Predictors	*Estimates*	*std. Error*	*T-value*	*p*
(Intercept)	-0.7494	0.0293	-25.6161	<**0.001**
launch speed	0.0134	0.0003	40.8363	<**0.001**
batted ball direction 0	-0.1084	0.0031	-34.7900	<**0.001**
batted ball direction 1	-0.0564	0.0035	-16.3067	<**0.001**
Observations	5366			
R^2 / adjusted R^2	0.397 / 0.397			

Table 4.5: Regression estimates for two predictors

After analyzing all 8 partial F-values, batted ball direction has the highest partial F-Value and the other resulting partial F-values meet the threshold so batted ball direction is added to the model, and we note the R^2 statistic has increased to .397. This process continues as outlines in Section 3.2 and ends when we add the pitch location variable to the model and receive the output in Table 4.6:

	woba			
Predictors	*Estimates*	*std. Error*	*T-value*	*p*
(Intercept)	-0.3854	0.0892	-4.3186	<**0.001**
launch angle	0.0008	0.0002	4.1886	<**0.001**
launch speed	0.0144	0.0004	40.6119	<**0.001**
velocity	-0.0413	0.0080	-5.1448	<**0.001**
effective speed	0.0401	0.0081	4.9348	<**0.001**
release extension	-0.0531	0.0173	-3.0733	**0.002**
batted ball direction 0	-0.1161	0.0032	-36.5966	<**0.001**
batted ball direction 1	-0.0703	0.0053	-13.1715	<**0.001**
pitch_location0	-0.0318	0.0052	-6.0713	<**0.001**
pitch location 2	0.0035	0.0043	0.8315	0.406
Observations	5366			
R^2 / adjusted R^2	0.427 / 0.426			

Table 4.6: Regression estimates for seven predictors

By adding pitch location to the model we have one very significant predictor and one insignificant predictor, so the variable is removed and thus, the process ends. Notice the forward selection and backward elimination have both selected the same model as the “best”.

Next, we shall analyze the residuals to see if there are systematic trends causing a problem for the models predictive power. A systematic trend in this analysis would take the form of a non-random error plot or a non-normal distribution of residuals. First, the histogram of the residuals (actual values minus fitted values) of the model is chosen in Figure 4.2 where we see an approximately normal distribution.

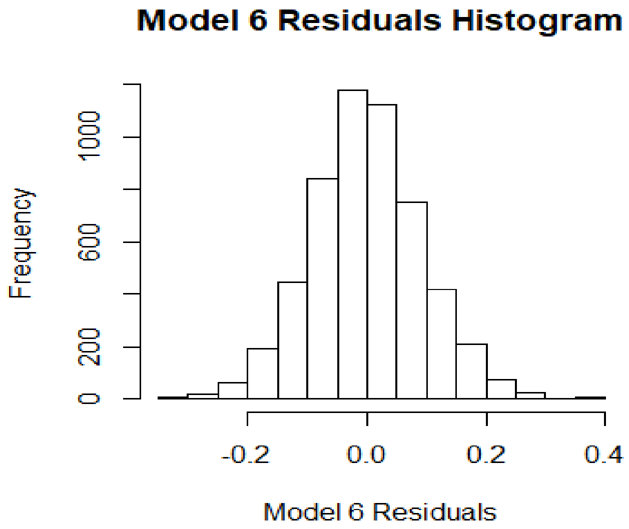


Figure 4.2: Histogram of residuals

The residuals look to be approximately normal around zero, but for further investigation of the normality of the residuals, one can observe the normal quantile-quantile plot, which maps the standardized residuals against the theoretical quantiles given a normal distribution. The plotted line beneath the data is what a perfect normal distribution of the residuals would look like, as shown in Figure 4.3

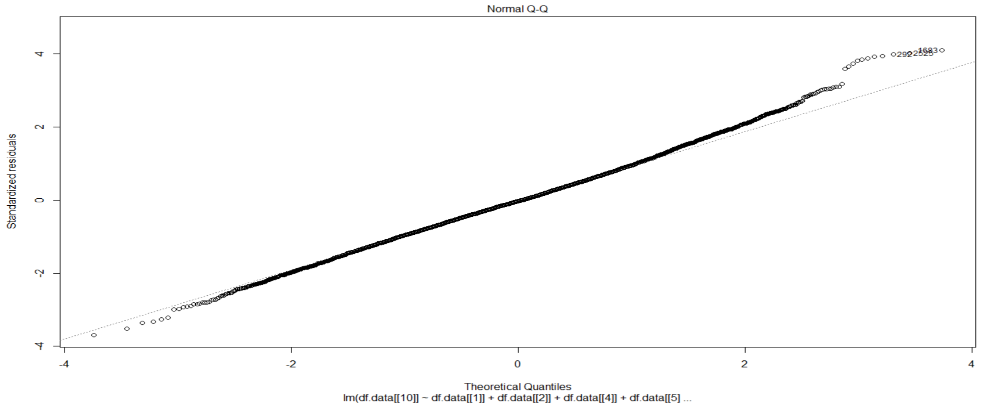


Figure 4.3: Quantile-Quantile plot of residuals

Within in two quantiles in either direction of the mean, the residuals appear to be perfectly normally distributed but there is more uncertainty moving toward the tails of the distribution of residuals in this case. The next piece of analysis is to observe the residuals plotted vs. the fitted values by the model, as shown in Figure 4.4.

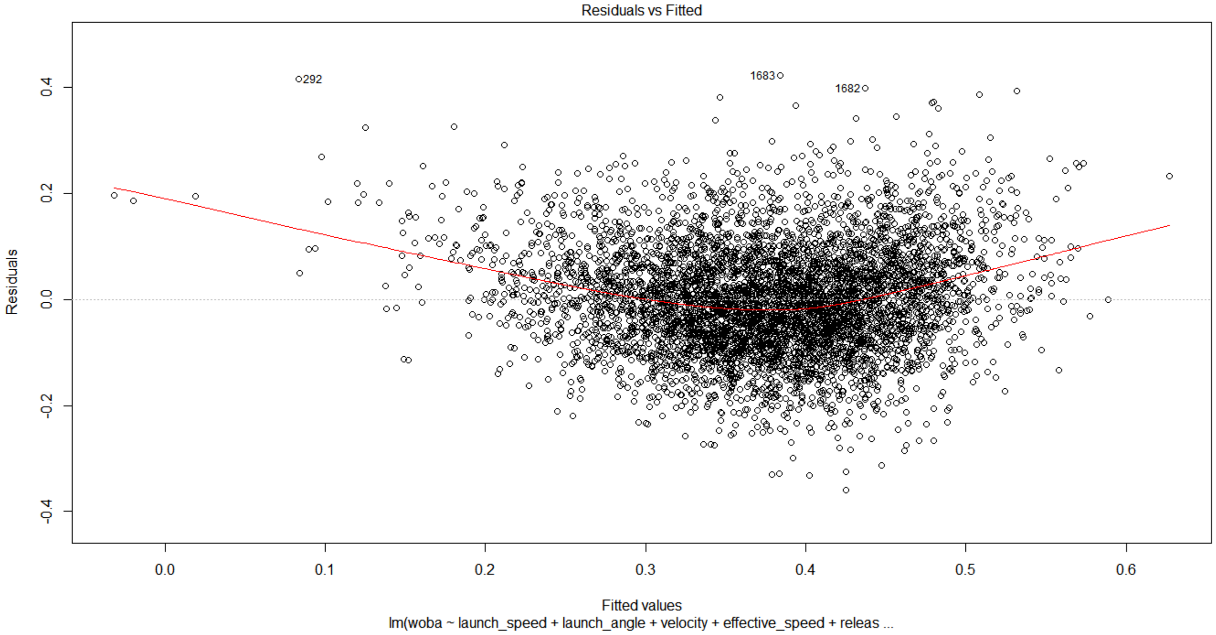


Figure 4.4: Plot of residuals against fitted values

To the naked eye, the plot does not reveal any obvious patterns and looks almost random around zero. However there are more large absolute values in the positive direction than in the negative. The smooth line R plots onto this graph maps the data with a slight quadratic trend. Further, we can look for trends by plotting the residuals vs. all the predictor variables in the model.

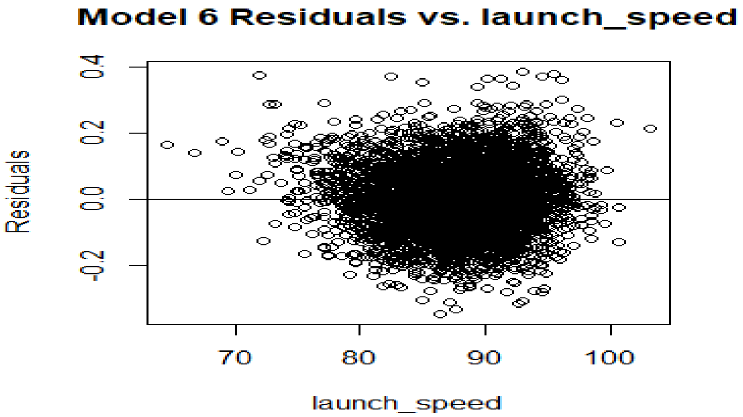


Figure 4.5: Plot of residuals vs. predictor launch speed

It is clear from Figure 4.5 that most of the residuals cluster around zero when the launch speed is between 70 and 100, but again there appear to be more high positive values than low negative ones.

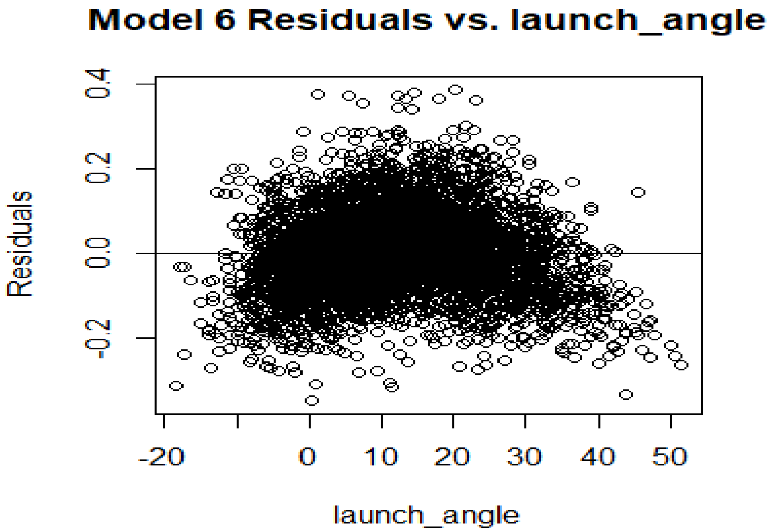


Figure 4.6: Plot of residuals vs. predictor launch angle

With launch angle there appears to be a clear negative quadratic pattern when mapped against residuals. This is an important fact to note and it aligns with what is seen when mapping launch angle against woba in Figure 4.7.

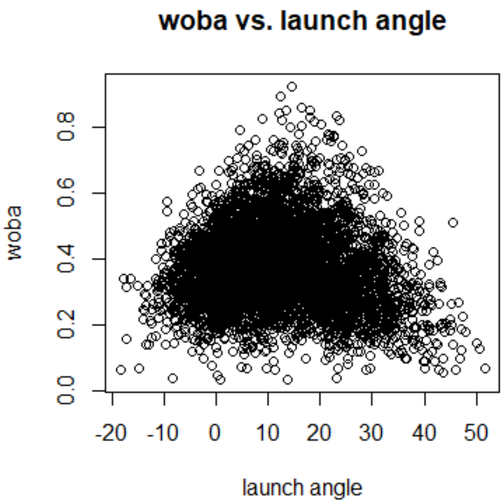


Figure 4.7: Plot of woba vs. predictor launch angle

This signals launch angle has a quadratic relationship which is negative and significant. We can test this prediction by adding a quadratic launch angle term to the multiple linear regression model, making it a polynomial linear regression.

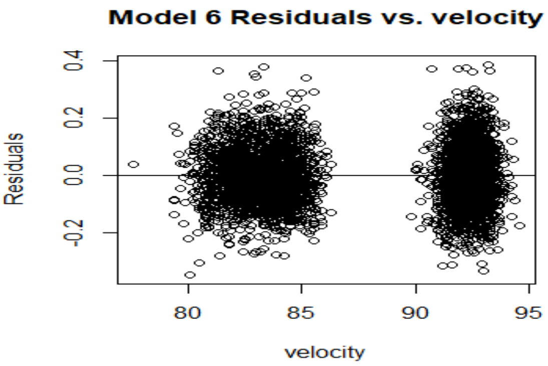


Figure 4.8 Plot of residuals vs. predictor velocity

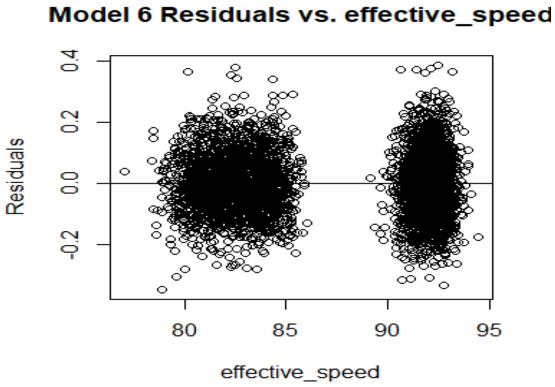


Figure 4.9: Plot of residuals vs. predictor effective speed

Figures 4.8 and 4.9 present plots of residuals against the predictors velocity and effective speed, respectively. The plots do not indicate any clear patterns in the residuals against the predictors.

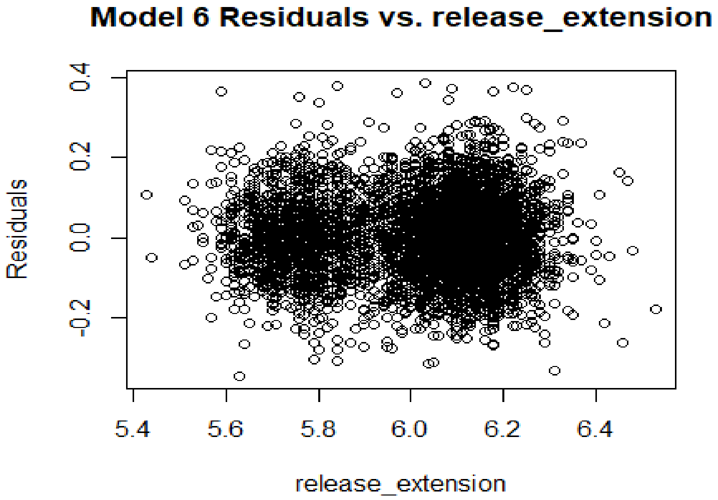


Figure 4.10: Plot of residuals vs. predictor release extension

In Figure 4.10, release extension has two large ellipsis centered around zero. This is because there are less data points with release extension between 5.8 and 6 both above and below zero.

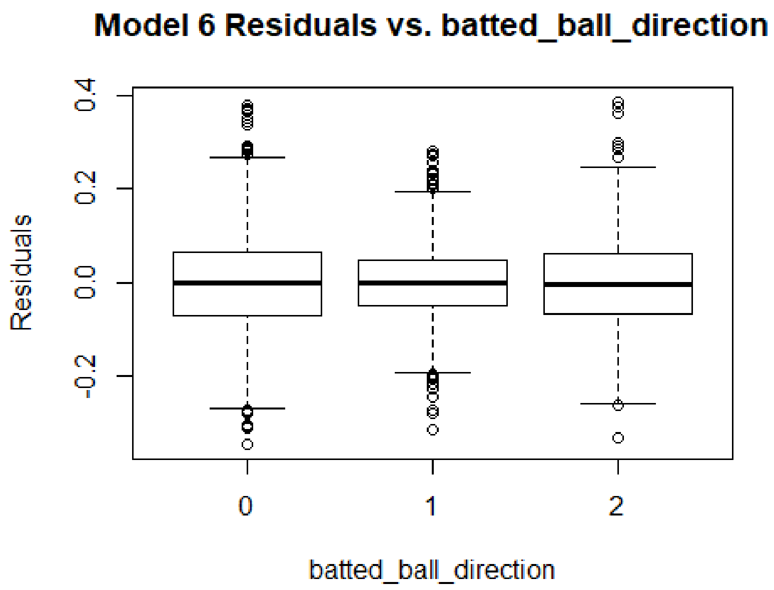


Figure 4.11: Boxplots of residuals vs. predictor batted ball direction

The boxplots of residuals in Figure 4.11 do not show any systematic pattern with respect to the different batted ball directions.

The model given as the "best" by both forward and backward stepwise regression has a lot of what one would expect from the residuals of a good model. The residuals are approximately normally distributed as seen in Figure 4.2 and Figure 4.3. The residuals, however show a slight quadratic pattern when mapped against the fitted values of the model. This pattern, however, is slight and difficult to detect with the naked eye. Most importantly, for our analysis, launch angle shows a quadratic pattern when mapped against the residuals of the model. In light of this, we add the quadratic launch angle term to the already established model and the summary output is shown in Table 4.7.

	woba			
Predictors	*Estimates*	*std. Error*	*T-value*	*p*
(Intercept)	-0.2542	0.0835	-3.0434	**0.002**
launch angle	0.0072	0.0003	27.9025	**<0.001**

launch speed	0.0123	0.0003	37.8920	<**0.001**
velocity	-0.0471	0.0076	-6.2241	<**0.001**
effective speed	0.0482	0.0077	6.2936	<**0.001**
release extension	-0.0799	0.0162	-4.9173	<**0.001**
batted ball direction 0	-0.1268	0.0030	-41.8438	<**0.001**
batted ball direction 1	-0.0750	0.0046	-16.4703	<**0.001**
launch angle^2	-0.0002	0.0001	-26.9995	<**0.001**
Observations	5366			
R^2 / adjusted R^2	0.484 / 0.484			

Table 4.7: Regression estimates with a quadratic term

In Table 4.7, the launch angle^2 term meets the previously established partial F-value threshold and does not make another predictor fall below it. Most interestingly, the R^2 statistic has moved from .414 up to .484, so the launch angle^2 term is a very useful addition to our model.

Recall the heat map of the predictors and response correlations showed velocity and effective speed as very highly correlated. This, as mentioned earlier, may be a sign they are dependent processes and would violate the assumption of multiple linear regression that all predictors are independent. Next, let us examine the effect of removing one of these predictors (the one with the smaller partial F-value) from the model. The results are shown in Table 4.8.

	woba			
Predictors	*Estimates*	*std. Error*	*T-value*	*p*
(Intercept)	-0.6611	0.0522	-12.6623	<**0.001**
launch angle	0.0073	0.0003	28.3120	<**0.001**
launch speed	0.0122	0.0003	37.4885	<**0.001**
effective speed	0.0006	0.0005	1.3630	0.173

release extension	-0.0087	0.0116	-0.7483	0.454
batted ball direction 0	-0.1265	0.0030	-41.6167	<**0.001**
batted ball direction 1	-0.0767	0.0046	-16.8202	<**0.001**
launch angle^2	-0.0002	0.0001	-26.9948	<**0.001**
Observations	5366			
R^2 / adjusted R^2	0.481 / 0.480			

Table 4.8: Regression estimates with a quadratic term, with predictor velocity removed

It is clear from Table 4.8 that not much has been lost by way of the R^2 statistic, but both the effective speed and release extension, variables have fallen below the F-value threshold. Removing release extension which has the lower F value, yields estimates as shown in Table 4.9.

	woba			
Predictors	*Estimates*	*std. Error*	*T-value*	*p*
(Intercept)	-0.6923	0.0314	-22.0686	<**0.001**
launch angle	0.0073	0.0003	28.3346	<**0.001**
launch speed	0.0122	0.0003	37.7752	<**0.001**
effective speed	0.0003	0.0003	1.2513	0.211
batted ball direction 0	-0.1265	0.0030	-41.6129	<**0.001**
batted ball direction 1	-0.0765	0.0046	-16.8063	<**0.001**
launch angle^2	-0.0002	0.0001	-26.9857	<**0.001**
Observations	5366			
R^2 / adjusted R^2	0.481 / 0.480			

Table 4.9: Regression estimates with a quadratic term, with predictor release extension removed

The model suffers no noticeable change in R^2 and still shows effective speed below the F-value threshold. The backward elimination process continues by removing effective speed and we are left with estimates that are shown in Table 4.10

	woba			
Predictors	*Estimates*	*std. Error*	*T-value*	*p*
(Intercept)	-0.6729	0.0273	-24.6557	<**0.001**
launch angle	0.0072	0.0003	28.4322	<**0.001**
launch speed	0.0123	0.0003	40.1341	<**0.001**
batted ball direction 0	-0.1258	0.0030	-42.0450	<**0.001**
batted ball direction 1	-0.0752	0.0044	-16.9740	<**0.001**
launch angle^2	-0.0002	0.0001	-27.2031	<**0.001**
Observations	5366			
R^2 / adjusted R^2	0.481 / 0.480			

Table 4.10: Regression estimates with a quadratic term, with effective velocity removed

In this model, all predictors meet the F-value threshold and more of the variation in woba is being explained than in the “best” first degree model chosen by stepwise regression.

To ensure we have selected an appropriate model we will once again analyze the errors this new model produces. Figure 4.12 shows the histogram of the residuals from the model, as used in Table 4.10. Figure 4.13 plots the residuals against the fitted values.

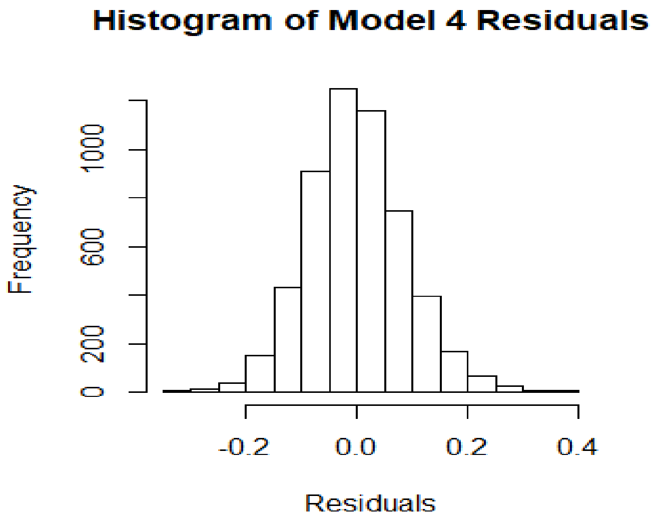


Figure 4.12: Histogram of residuals for the “best” model in Table 4.10

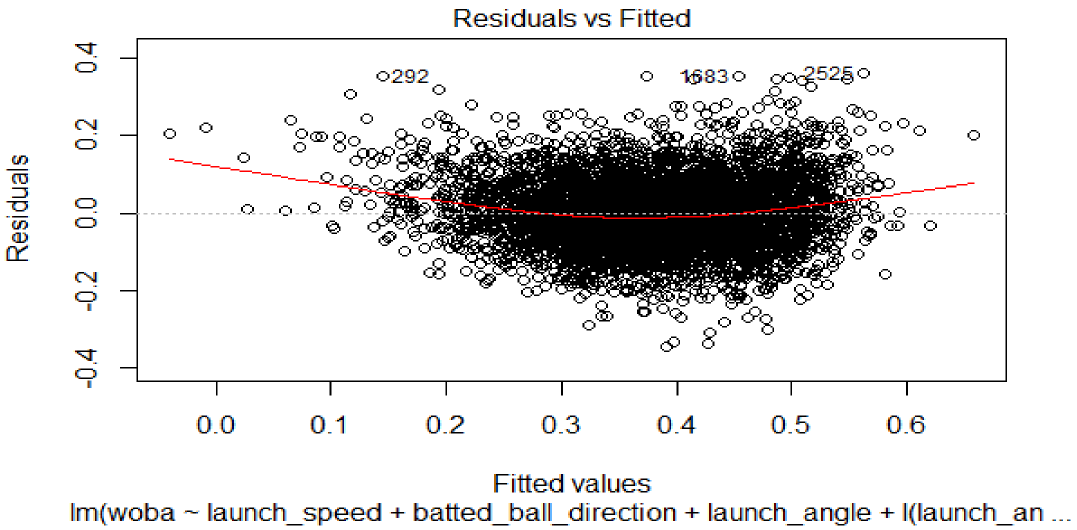


Figure 4.13: Plot of residuals vs. fitted values for the “best” model in Table 4.10

The distribution of residuals is approximately normal and the residuals mapped against the predicted values look similar to those in Figure 4.4 with a slightly less quadratic pattern to it. Most importantly, we see a large change in the residuals mapped against launch angle as Figure 4.14 shows no systematic pattern among the residuals.

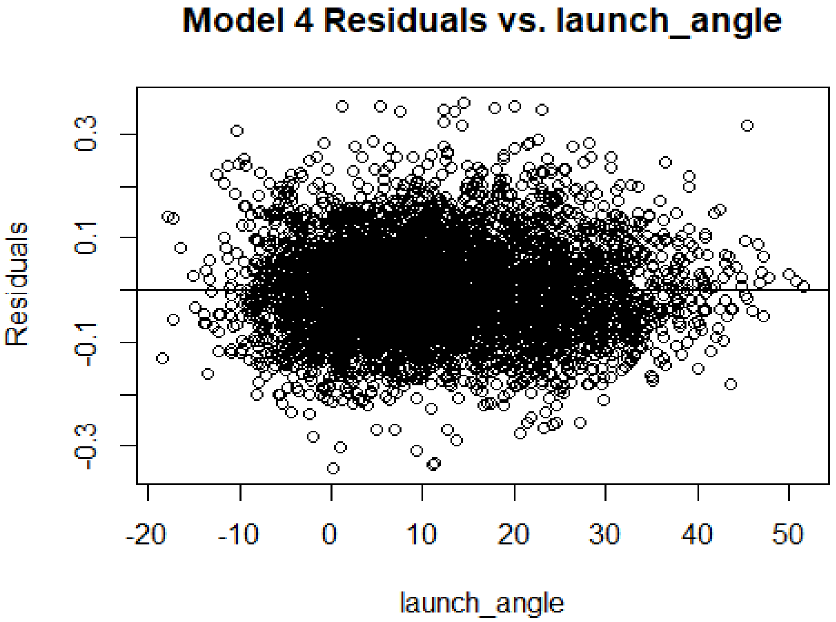


Figure 4.14: Plot of residuals vs. launch angle for the model in Table 4.10

The previous systematic trend in Figure 4.6 has been eliminated, which is another positive sign for the predictive power of the model. However, more can be done to test the stability of this model, most notably by cross-validation.

4.2 Model Assessment

We have selected a model and feel confident in the stability of it based on the residual analysis, we can perform a cross-validation. We will proceed with a leave one out cross-validation as previously outlined. Doing this will output the test error rate as well as the distribution of the errors across all possible training sets of size n-1. The distribution of R^2 values across the same sets is also presented below. The histogram plots of the mean squared errors and R^2 values across all possible training sets are shown in Figures 4.15 and 4. 16, respectively.

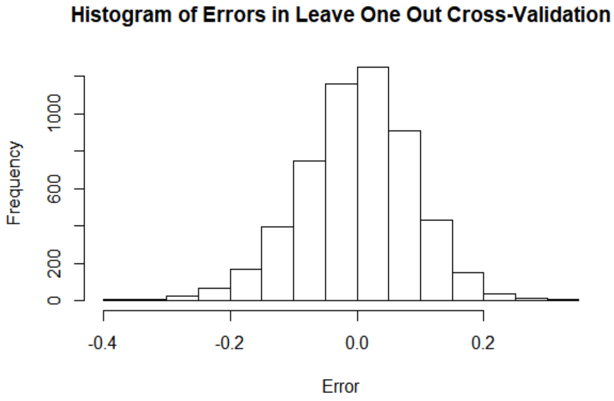


Figure 4.15: Histogram of errors in cross-validation

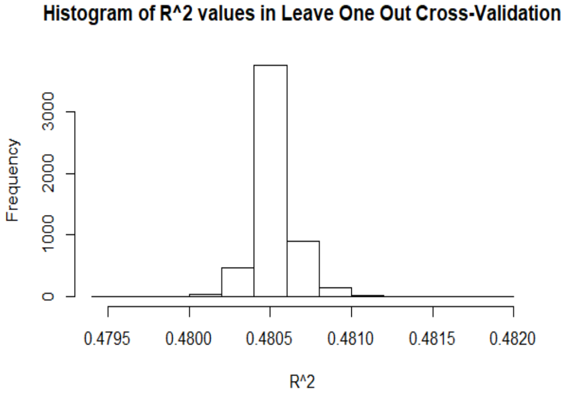


Figure 4.16: Histogram of R^2 values in cross-validation

The distributions of both are normal and relatively narrow, which should offer confidence in stability of the model. Also provided by the leave one out cross-validation is the mean squared error, which is 0.0079, and for reference the best linear model by stepwise regression has a mean squared error of 0.0089 by this method. This implies the addition of the launch angle^2 and the removal of the velocity, effective speed and release extension terms have improved the capability of the model on test data. We also see the R^2 is very consistently in the .480-.481 range for all the training sets of size n-1.

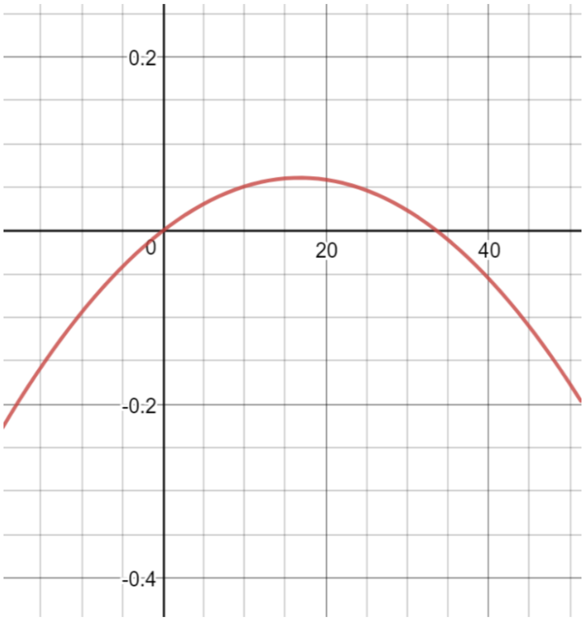


Figure 4.17: Effect of launch angle on woba

We can also interpret $\beta_0,\ldots,\beta_5$ the regression coefficients given to us, by this model. The intercept and coefficient for launch speed are straight forward to interpret in the linear regression context. The intercept at -0.673 is the predicted response variable (woba) output if all variables are zero, which however, does not mean much because if all variables are zero the ball has not been hit and thus a woba on balls in play is incalculable. Having the coefficient for launch speed at 0.012 implies while holding all x's constant for every mile per hour a batter adds in launch speed, they gain is what is called 12 points in woba. The batted ball coefficients work similarly, but understanding of categorical variables is important. In this case, a ball which is pulled is considered the baseline result (this is because most balls are pulled) and balls hit straightaway are coded as 1 and balls pushed are coded as 2, so the coefficients are relative to a pulled ball. This implies a batter loses 126 points in woba if they hit the ball straightaway compared to a pulled ball and lose 75 points of woba if they push the again relative to pulling it. This is in accordance to common baseball wisdom which says most players have better outcomes to the pull side. This gap is shrinking especially on groundballs due to the increased prevalence of the defensive shift. Lastly, to explain the polynomial coefficients for launch angle holding the other independent x's constant and removing the intercept, Figure 4.27 is a picture of launch angle's effect on woba. By taking the coefficient of the linear term and dividing it by two times the negative coefficient of the quadratic of this formula, we find the optimal launch angle is 17° which adds 61 points of woba, the range of launch angles which add points to woba are 0° to 33.5°; anything above 33.5° or below 0° results in a loss of woba in this model.

To test the model further we can gather another set of data and analyze the predictions. This set of data is aggregated to correspond to players who have had at least 50 batted balls pulled, pushed and hit straight away (150 in total) in the last 4 MLB seasons, where 555 players met this threshold. These data are new because the data is split strictly on batted ball direction which changes the values of all the input variables (9 categories of batted balls are combined for each batted ball direction). It allows us to see which players are significantly over performing and underperforming their woba. First, a list of 10 biggest under performers according to this model is shown in Table 4.11.

Player Name	woba	xwoba	nxwoba	nxwoba-woba
Tuffy Gosewisch	0.244	0.285	0.366	0.123

Erik Kratz	0.286	0.322	0.402	0.116
Eric Campbell	0.259	0.359	0.375	0.116
Jake Smolinski	0.288	0.325	0.391	0.103
Casey McGehee	0.256	0.318	0.356	0.101
Taylor Motter	0.276	0.327	0.376	0.099
Dustin Ackley	0.298	0.341	0.393	0.096
Aaron Hill	0.296	0.313	0.391	0.095
Rob Refsnyder	0.279	0.333	0.370	0.091
Albert Pujols	0.333	0.379	0.420	0.087

Table 4.11: List of 10 biggest underperformers

Included in this table is the statistic nxwoba, which is the predicted value of each player's woba by the previously outlined model and its difference to woba. Also, included is xwoba, which is similarly an estimator for a player's woba based on the player's batted ball quality of contact and eliminates the effect of defense. Next, a list of players over performing their woba is presented in Table 4.12.

Player Name	woba	xwoba	nxwoba	nxwoba-woba
Mallex Smith	0.361	0.287	0.245	-0.116
Ezequiel Carrera	0.374	0.323	0.274	-0.100
Dee Gordon	0.338	0.275	0.238	-0.100
Ryan Schimpf	0.461	0.432	0.363	-0.099
Jarrett Parker	0.460	0.437	0.365	-0.095
Keon Broxton	0.461	0.435	0.366	-0.095
Willy Adames	0.434	0.361	0.340	-0.094
Shohei Ohtani	0.517	0.496	0.431	-0.086
Trevor Story	0.496	0.447	0.411	-0.085
Aaron Judge	0.564	0.578	0.479	-0.085

Table 4.12: List of 10 biggest overperformers

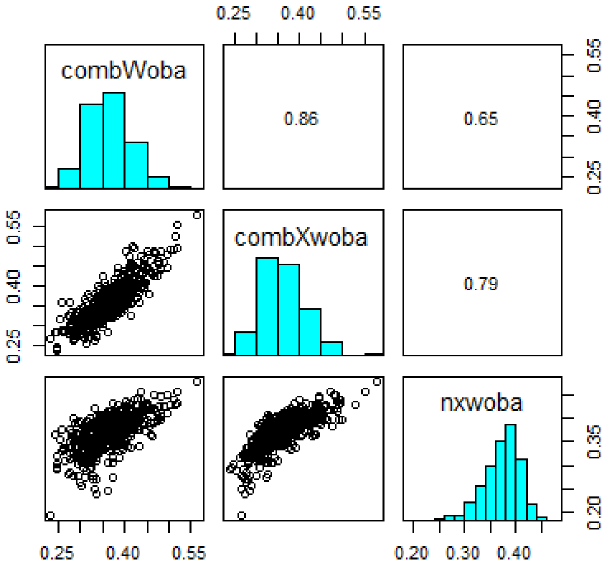


Figure 4.18: Comparison of woba and its estimators

Note, in 19 of the 20 most extreme errors the xwoba and nxwoba statistics agree on the direction of the player's regression. One thing to be mindful of when analyzing these tables is, these are statistics only on balls in play; if one would like a holistic assessment of the players expected woba for all contexts, a simple adjustment to factor in walks, strikeouts, hit by pitches and other non-contact plate appearances would need to be made. Finally, a comparison of the scatterplots, distributions and pairwise correlations of woba and its two estimators is shown in Figure 4.18.

The distribution of the predicted values on this new data set appears to be slightly left skewed and correlates with woba at 65 percent with a generally linear pattern in the scatterplot. Interestingly the nxwoba actually correlates better with the estimator xwoba and has a more clearly linear trend to the scatterplot. As nxwoba compares to xwoba as an estimator for woba, it is not strong according to this analysis. The stronger correlation with xwoba than woba is to be expected, however, because they are both defense neutral statistics which attempt to capture the true talent of the batter.

5 Conclusion

5.1 Summary of Findings

Regarding the top 10 under and over performers tables by the model derived from this project, there are clear insights to be gained. A caveat, many of the players that appear on these lists have close to the minimum number of batter balls, however, not all do and those with smaller sample sizes fit the patterns of those with larger sample sizes. From the underperforming table, half of the players are below average runners, Pujols, Gosewisch and Kratz especially. There is a relevant correlation between sprint speed and batting average in balls in play, which is made of the same component outcomes of woba on balls in play. For below average runners, we may expect their nxwoba to be consistently higher than their woba for this reason. The over performers list, however, is full of players, especially Ohtani, Story, Shimpf and Judge, who have extraordinarily high strikeout rates. These players have extraordinarily high woba on balls in play and for this reason alone our model would expect their true talent woba on balls in ball to be lower but still very high. Some players in this mold are able to withstand this all or nothing approach at the plate, like Ohtani and Judge, while others are marred by their strikeouts, like Broxton and Schimpf. We see the inverse effect of speed at top of the over performers leaderboard as Carrera, Smith and Gordon are some of the fastest runners in the game. They make a lot of weak contact but are able to get significantly more infield hits than the average player which would explain some of the gap in woba and nxwoba. Using the 554 of 555 players in the testing sample (Jacob deGrom excluded due to lack of sprint speed data), the difference between woba and nxwoba has a correlation of .396 with sprint speed. When sprint speed is used as a predictor added to the current model, with the 554 player test data used as training data, it is deemed significant and adds .0274 in the R^2 statistic. An additional set of data to test on and more precise sprint speed measurements, would be next step to building a more accurate and comprehensive model.

The mean squared error on the last set of data is .00154 which only lends more confidence in the models predictive power. By leveraging new advanced data from MLBAM and foundational statistical techniques like multiple linear regression, stepwise regression model selection and model assessment tools: cross-validation and residual analysis, we are able to build a highly functional model for woba on balls in play. This model is able to provide reliable predictions for woba and follows closely the well-established and respected statistic xwoba.

5.2 Opportunities for Future Research

Analyzing the extreme values of the errors for this model raises an interesting question. If sprint speed was factored into the model how big of a factor would it be for predicting woba on balls in play? Further it raises a methodological question about what we are trying to do with a model and statistic like the one derived in this paper. All the variables which ended up in the model are clear reflections of a player's offensive capability, launch angle, launch speed and batted ball direction. This metric appears to be capturing just how good a player is at hitting the ball, not how good of a runner they are, which is relevant to reaching base. If a player makes weak contact, they are surely a less impressive batter and they are systematically unappreciated by this metric, if they are extraordinarily fast. Including sprint speed in the model would change the statistic from an evaluation of player's ability to hit the ball well to a measure of their ability to generate value while they are batting and by extension running.

Additionally instead of using a multiple linear regression approach, a ridge regression approach could be taken to address the potential collinearity that exists between predictors in the data set. Another method of attempting understand a players batting value through Statcast metrics would be to train a model based on every individual batted ball available rather than sorting by player and requiring a minimum number of batted balls. woba scores of individual batted balls would not be stable but presumably with greatly increased size of the training data the model produced would be more predictive. There are logistically problems in trying to mine this data however and with the resources available for this project, this approach was not feasible.

6 Bibliography:

Drasgow, F. (1986) *Polychoric and polyserial correlations*. Pp. 68-74 in S. Kotz and N. Johnson, eds. The Encyclopedia of Statistics, Volume 7. Wiley.

Draper, N. R., & Smith, H. (1981). *Applied regression analysis* (3rd edition). New York: Wiley.

Guthery, F. S., & Bingham, R. L. (2007). A primer on interpreting regression models. *Journal of Wildlife* Management, 71(3), 684-692

James, B. *The Bill James Baseball Abstract: 1977,* privately printed, 1977.

James, B. *The Bill James Baseball Abstract.* (1983, September). The Atlantic, 59. Retrieved from http://link.galegroup.com.proxy.library.carleton.ca/apps/doc/A31392555/CPI?u=ocul_carleton&sid=CPI&xid=55c13e14

James, G., Witten, D., Hastie, T., Tibshirani, R., & SpringerLink (Online service). (2013). *An introduction to statistical learning: With applications in R.* New York, NY: Springer.

Lewis, M. *Moneyball: the Art of Winning an Unfair Game*. W.W. Norton, 2013.

Marchi, M., & Albert, J. (2018; 2013; 2017 ;). *Analyzing baseball data with R.* Philadelphia, PA: Chapman and Hall/CRC.

Padres and dodgers swept out of play-offs: SINCE 1876 NATIONAL LEAGUE OF PROFESSIONAL BASEBALL CLUBS. (1996). South China Morning Post (1946-Current)

Podhorzer, M. (2017, July 17). Diving into Statcast Sprint Speed. Retrieved from https://www.fangraphs.com/fantasy/diving-into-statcast-sprint-speed/

Willman, D., & Petriello, M. (2015). Statcast Leaderboard. Retrieved September, 2018, from https://baseballsavant.mlb.com/

7 Appendix A: Glossary

Batted Ball Direction: a categorical variable sorted based on where the ball first lands on the ground, into the pull, push and middle thirds of the field. A balled hit to the pulled third of the field is when a batter hits a ball and it first lands on the side of the field from which they do not bat (i.e. a right handed batter hits the ball to left field or vice versa), and the push side is batting the ball to the same side of the field (i.e. a right handed batter hits the ball to the right side and vice versa).

Effective Speed: an adjustment to the Velocity statistic based on the Release Extension of a pitcher.

Launch Angle: how high the ball is hit by the batter in degrees.

Launch Speed: how fast the ball is hit by the batter in miles per hour (MPH).

Pitch Location: a categorical variable where the categories are low, mid and high based on where the pitch crosses the plate strictly on the y axis.

Pitch Type: another categorical variable where the pitch is sorted into fastball, breaking ball and off speed. These types of pitches are created by the pitcher gripping and throwing the ball in different ways, all of which is tracked by Statcast.

Release Extension: how far from the mound (pitching position) the pitcher releases the ball in feet.

Spin Rate: how much spin the pitcher puts on the balls in revolutions per minute (RPM).

Velocity: how fast the pitch is thrown in MPH.

Weighted On-Base Average (woba): is a weighted average based on a plate appearances outcome that is used as a batter performance metric.

Appendix B: Code

All code for this project was written in R and the following functions were used to execute the necessary procedures for this project.

This function is input a data frame and returns the Pearson, polychoric or polyserial correlations in the form of a heat map correlation matrix.

```
library(polycor)
statcast_hitter_cor_mat <- function(input){

  if (!require("gplots")) {
    install.packages("gplots", dependencies = TRUE)
    library(gplots)
  }

  s<-hetcor(input)

  df_cor <- round(s$correlations,5)

  col<- colorRampPalette(c("blue", "white", "red"))(20)

  heatmap.2(x = df_cor,cellnote = df_cor, notecol="black", main = "Correlation Matrix",
        density.info="none",trace="none", col = col, dendrogram="none", margins =c(12,8),cexRow=0.5)
}
```

This function requires a data frame and a list of columns, it outputs a scatter plot matrix of all the variables included in the columns list. On the lower diagonal are all the pairwise scatterplots, the upper diagonal has the pairwise correlations and the diagonal itself has a picture of the distribution of the variable.

```
library(polycor)
statcast_scatter_plot <- function(df, columns){
  #install ggplots2
  if (!require("ggplot2")) {
    install.packages("ggplot2", dependencies = TRUE)
    library(ggplot2)
  }
  #clean data
  df<- df[ , columns]

  #histogram
  panel.hist <- function(x, ...)
  {
    usr <- par("usr"); on.exit(par(usr))
    par(usr = c(usr[1:2], 0, 1.5) )
    h <- hist(x, plot = FALSE)
    breaks <- h$breaks; nB <- length(breaks)
    y <- h$counts; y <- y/max(y)
    rect(breaks[-nB], 0, breaks[-1], y, col = "cyan", ...)
  }
  #correlation
  panel.cor <- function(x, y, ...)
  {
    par(usr = c(0, 1, 0, 1))
    h<- hetcor(x, y),
    txt <- as.character(format(h, digits=2))
    text(0.5, 0.5, txt)
  }

  #scatterplot matrix
  pairs(df, diag.panel = panel.hist, upper.panel = panel.cor)
}
```

This function requires a data frame, the name of the response variable and a model. It then performs a leave one out cross-validation and returns the histogram of the errors and the mean squared error of the model under leave one out cross-validation.

```
score = list()
library(ModelMetrics)

LOOCV_function = function(x, label, model){
  for(i in 1:nrow(x)){
```

```
  training = x[-i,]
   validation = x[i,]
  pred = predict(model, validation[,setdiff(names(validation),label)])
  score[[i]] = (pred-validation[[label]]) # score/error of ith fold
 }
 hist(unlist(score),main = "Histogram of Errors in Leave One Out Cross-validation", xlab = "Error")
 mse<-(1/nrow(x))*sum(unlist(score)^2)
 mse
}
```