HMW 1- Data 621

OMER OZEREN

Table of Contents

## Introduction

I have been given a dataset with 2276 records summarizing a major league baseball team’s season. All statistics have been adjusted to match the performance of a 162 game season. The objective is to build a linear regression model to predict the number of wins for a team. This report covers an attempt to build a model to predict number of wins of a baseball team in a season based on several offensive and deffensive statistics. Resulting model explained about 36% of variability in the target variable and included most of the provided explanatory variables. Some potentially variables were not included in the data set due to missing values.I used KNN for variable missing values imputtion.

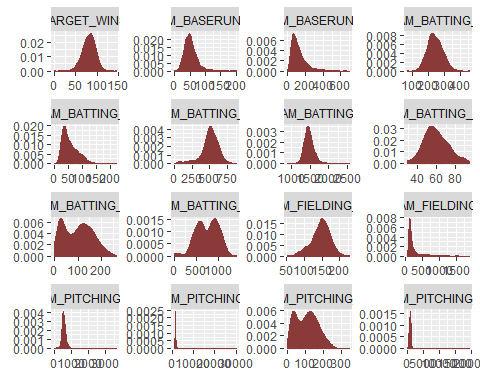
## DATA EXPLORATION

Each record in the data set represents the performance of the team for the given year adjusted to the current length of the season - 162 games. The data set includes 16 variables and the training set includes 2,276 records.

sumtable = data.frame(Variable = character(),  
 Min = integer(),  
 Median = integer(),  
 Mean = double(),  
 SD = double(),  
 Max = integer(),  
 Num\_NaN = integer())  
for (i in 2:17) {  
 sumtable <- rbind(sumtable, data.frame(Variable = colnames(training)[i],  
 Min = min(training[,i], na.rm=TRUE),  
 Median = median(training[,i], na.rm=TRUE),  
 Mean = mean(training[,i], na.rm=TRUE),  
 SD = sd(training[,i], na.rm=TRUE),  
 Max = max(training[,i], na.rm=TRUE),  
 Num\_NaN = sum(is.na(training[,i])))  
 )  
}  
colnames(sumtable) <- c("", "Min", "Median", "Mean", "SD", "Max", "Num of NaN")  
sumtable

Min Median Mean SD Max Num of NaN  
1 TARGET\_WINS 0 82.0 80.79086 15.75215 146 0  
2 TEAM\_BATTING\_H 891 1454.0 1469.26977 144.59120 2554 0  
3 TEAM\_BATTING\_2B 69 238.0 241.24692 46.80141 458 0  
4 TEAM\_BATTING\_3B 0 47.0 55.25000 27.93856 223 0  
5 TEAM\_BATTING\_HR 0 102.0 99.61204 60.54687 264 0  
6 TEAM\_BATTING\_BB 0 512.0 501.55888 122.67086 878 0  
7 TEAM\_BATTING\_SO 0 750.0 735.60534 248.52642 1399 102  
8 TEAM\_BASERUN\_SB 0 101.0 124.76177 87.79117 697 131  
9 TEAM\_BASERUN\_CS 0 49.0 52.80386 22.95634 201 772  
10 TEAM\_BATTING\_HBP 29 58.0 59.35602 12.96712 95 2085  
11 TEAM\_PITCHING\_H 1137 1518.0 1779.21046 1406.84293 30132 0  
12 TEAM\_PITCHING\_HR 0 107.0 105.69859 61.29875 343 0  
13 TEAM\_PITCHING\_BB 0 536.5 553.00791 166.35736 3645 0  
14 TEAM\_PITCHING\_SO 0 813.5 817.73045 553.08503 19278 102  
15 TEAM\_FIELDING\_E 65 159.0 246.48067 227.77097 1898 0  
16 TEAM\_FIELDING\_DP 52 149.0 146.38794 26.22639 228 286

training %>%  
 gather(variable, value, TARGET\_WINS:TEAM\_FIELDING\_DP) %>%  
 ggplot(., aes(value)) +   
 geom\_density(fill = "indianred4", color="indianred4") +   
 facet\_wrap(~variable, scales ="free", ncol = 4) +  
 labs(x = element\_blank(), y = element\_blank())



quick\_summary <- function(df){  
 df %>%  
 summary() %>%  
 kable() %>%  
 kable\_styling()  
}  
quick\_summary(training)

INDEX </th>

TARGET\_WINS

TEAM\_BATTING\_H

TEAM\_BATTING\_2B

TEAM\_BATTING\_3B

TEAM\_BATTING\_HR

TEAM\_BATTING\_BB

TEAM\_BATTING\_SO

TEAM\_BASERUN\_SB

TEAM\_BASERUN\_CS

TEAM\_BATTING\_HBP

TEAM\_PITCHING\_H

TEAM\_PITCHING\_HR

TEAM\_PITCHING\_BB

TEAM\_PITCHING\_SO

TEAM\_FIELDING\_E

TEAM\_FIELDING\_DP

Min. : 1.0

Min. : 0.00

Min. : 891

Min. : 69.0

Min. : 0.00

Min. : 0.00

Min. : 0.0

Min. : 0.0

Min. : 0.0

Min. : 0.0

Min. :29.00

Min. : 1137

Min. : 0.0

Min. : 0.0

Min. : 0.0

Min. : 65.0

Min. : 52.0

1st Qu.: 630.8

1st Qu.: 71.00

1st Qu.:1383

1st Qu.:208.0

1st Qu.: 34.00

1st Qu.: 42.00

1st Qu.:451.0

1st Qu.: 548.0

1st Qu.: 66.0

1st Qu.: 38.0

1st Qu.:50.50

1st Qu.: 1419

1st Qu.: 50.0

1st Qu.: 476.0

1st Qu.: 615.0

1st Qu.: 127.0

1st Qu.:131.0

Median :1270.5

Median : 82.00

Median :1454

Median :238.0

Median : 47.00

Median :102.00

Median :512.0

Median : 750.0

Median :101.0

Median : 49.0

Median :58.00

Median : 1518

Median :107.0

Median : 536.5

Median : 813.5

Median : 159.0

Median :149.0

Mean :1268.5

Mean : 80.79

Mean :1469

Mean :241.2

Mean : 55.25

Mean : 99.61

Mean :501.6

Mean : 735.6

Mean :124.8

Mean : 52.8

Mean :59.36

Mean : 1779

Mean :105.7

Mean : 553.0

Mean : 817.7

Mean : 246.5

Mean :146.4

3rd Qu.:1915.5

3rd Qu.: 92.00

3rd Qu.:1537

3rd Qu.:273.0

3rd Qu.: 72.00

3rd Qu.:147.00

3rd Qu.:580.0

3rd Qu.: 930.0

3rd Qu.:156.0

3rd Qu.: 62.0

3rd Qu.:67.00

3rd Qu.: 1682

3rd Qu.:150.0

3rd Qu.: 611.0

3rd Qu.: 968.0

3rd Qu.: 249.2

3rd Qu.:164.0

Max. :2535.0

Max. :146.00

Max. :2554

Max. :458.0

Max. :223.00

Max. :264.00

Max. :878.0

Max. :1399.0

Max. :697.0

Max. :201.0

Max. :95.00

Max. :30132

Max. :343.0

Max. :3645.0

Max. :19278.0

Max. :1898.0

Max. :228.0

NA

NA

NA

NA

NA

NA

NA

NA’s :102

NA’s :131

NA’s :772

NA’s :2085

NA

NA

NA

NA’s :102

NA

NA’s :286

Some initial observations:

* The response variable (TARGET\_WINS) looks to be normally distributed. This supports the working theory that there are good teams and bad teams. There are also a lot of average teams.
* There are also quite a few variables with missing values. We may need to deal with these in order to have the largest data set possible for modeling.
* A couple variables are bimodal (TEAM\_BATTING\_HR, TEAM\_BATTING\_SO TEAM\_PITCHING\_HR). This may be a challenge as some of them are missing values and that may be a challenge in filling in missing values.
* Some variables are right skewed (TEAM\_BASERUN\_CS, TEAM\_BASERUN\_SB, etc.). This might support the good team theory. It may also introduce non-normally distributed residuals in the model. We shall see.

#### Zero Values

There are also variables that have verly low values. Let’s see how big of a problem this is:

training %>%   
 gather(variable, value) %>%  
 filter(value == 0) %>%  
 group\_by(variable) %>%  
 tally() %>%  
 mutate(percent = n / nrow(training) \* 100) %>%  
 mutate(percent = paste0(round(percent, ifelse(percent < 10, 1, 0)), "%")) %>%  
 arrange(desc(n)) %>%  
 rename(`Variable With Zeros` = variable,  
 `Number of Records` = n,  
 `Share of Total` = percent) %>%  
 kable() %>%  
 kable\_styling()

Variable With Zeros

Number of Records

Share of Total

TEAM\_BATTING\_SO

20

0.9%

TEAM\_PITCHING\_SO

20

0.9%

TEAM\_BATTING\_HR

15

0.7%

TEAM\_PITCHING\_HR

15

0.7%

TEAM\_BASERUN\_SB

2

0.1%

TEAM\_BATTING\_3B

2

0.1%

TARGET\_WINS

1

0%

TEAM\_BASERUN\_CS

1

0%

TEAM\_BATTING\_BB

1

0%

TEAM\_PITCHING\_BB

1

0%

The report shows that missing values are nearly low.

#### Missing Values (NaN)

During our first look at the data it was noted that there were variables that are missing data. Here’s a look at what variables are missing data and how big of a problem it is:

training %>%   
 gather(variable, value) %>%  
 filter(is.na(value)) %>%  
 group\_by(variable) %>%  
 tally() %>%  
 mutate(percent = n / nrow(training) \* 100) %>%  
 mutate(percent = paste0(round(percent, ifelse(percent < 10, 1, 0)), "%")) %>%  
 arrange(desc(n)) %>%  
 rename(`Variable Missing Data` = variable,  
 `Number of Records` = n,  
 `Share of Total` = percent) %>%  
 kable() %>%  
 kable\_styling()

Variable Missing Data

Number of Records

Share of Total

TEAM\_BATTING\_HBP

2085

92%

TEAM\_BASERUN\_CS

772

34%

TEAM\_FIELDING\_DP

286

13%

TEAM\_BASERUN\_SB

131

5.8%

TEAM\_BATTING\_SO

102

4.5%

TEAM\_PITCHING\_SO

102

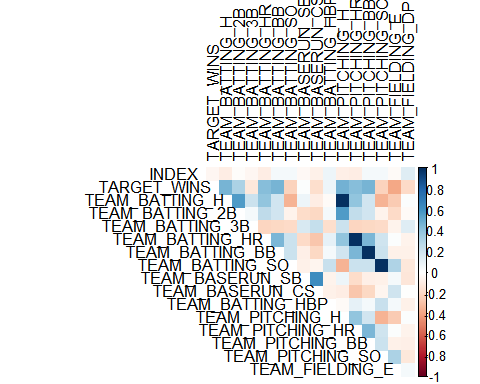
4.5%

The “TEAM\_BATTING\_HBP”" varriable has “NaN” nearly 92%. We will exclude this variable from consideration in our model.

#### Correlations Matrix

Let’s take a look at the correlations. The following is the correlations from the complete cases only:

training %>%   
 cor(., use = "complete.obs") %>%  
 corrplot(., method = "color", type = "upper", tl.col = "black", diag = FALSE)

 Anything over 0.5 or under -0.5 is highlighted in blue. The matrix was created using complete pairwise observations.

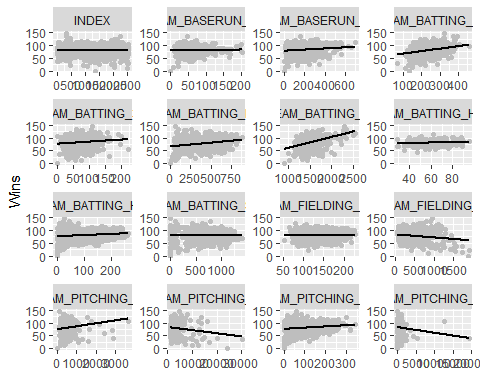
A few conclusions:

* Not surprisingly there is a very strong correlation between home runs batted in and home runs given up by pitching.
* There is a negative correlation between number of triples and home runs. A less powerful team may not have enough power to hit home runs, but they get a lot of triples.
* THere is a strong positive correlation between number of strikeouts and home runs. More swings of the bat results in more home runs.

#### Correlations: Endogenous and Exogenous Variables

Let’s take a look at how the Exogenous(Model Inputs) are correlated with the response variable(Endogenous):

training %>%  
 gather(variable, value, -TARGET\_WINS) %>%  
 ggplot(., aes(value, TARGET\_WINS)) +   
 geom\_point(fill = "indianred4", color="grey") +   
 geom\_smooth(method = "lm", se = FALSE, color = "black") +   
 facet\_wrap(~variable, scales ="free", ncol = 4) +  
 labs(x = element\_blank(), y = "Wins")



### Variable chacteristics

Each variable is presented below with corresponding basic statistics (minimum, median and maximum values, mean and standard deviation, number of records with missing values), boxplot, density plot with highlighted mean value, and scatterplot against outcome variable (TARGET\_WINS) with best fit line. This information is used to check general validity of data and adjust as necessary.

#### TEAM\_BATTING\_H:

This variable represents number of team base hits:

Min

Median

Mean

SD

Max

Num of NaN

891

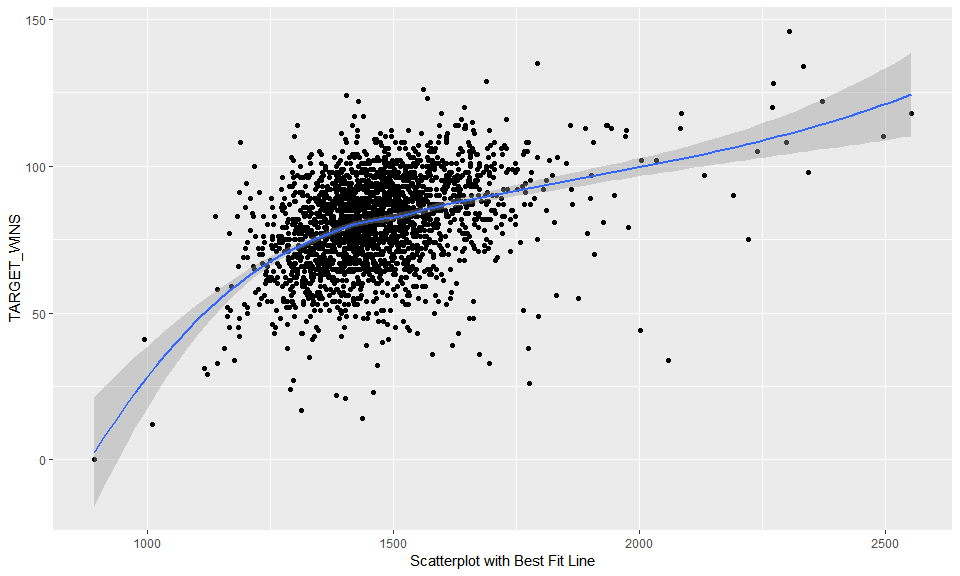
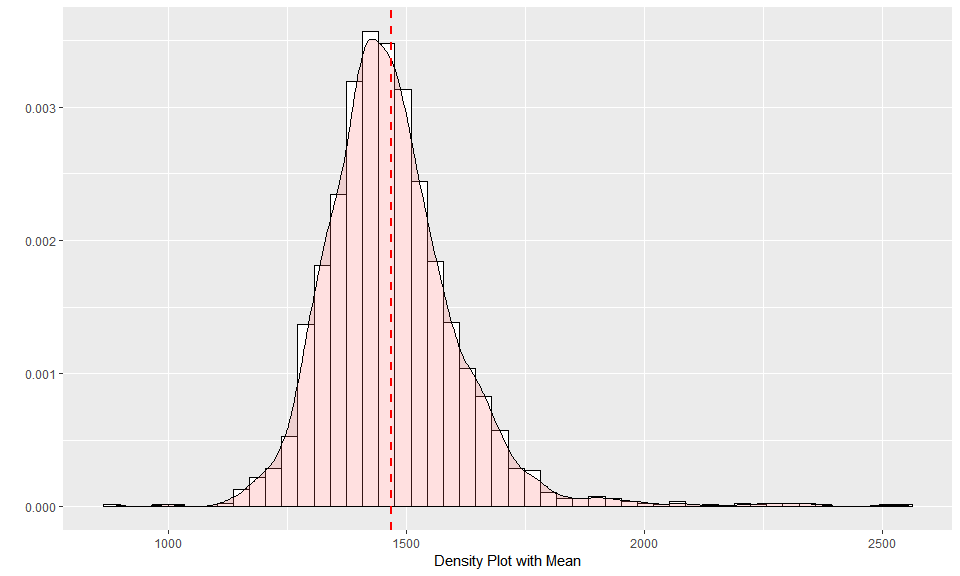
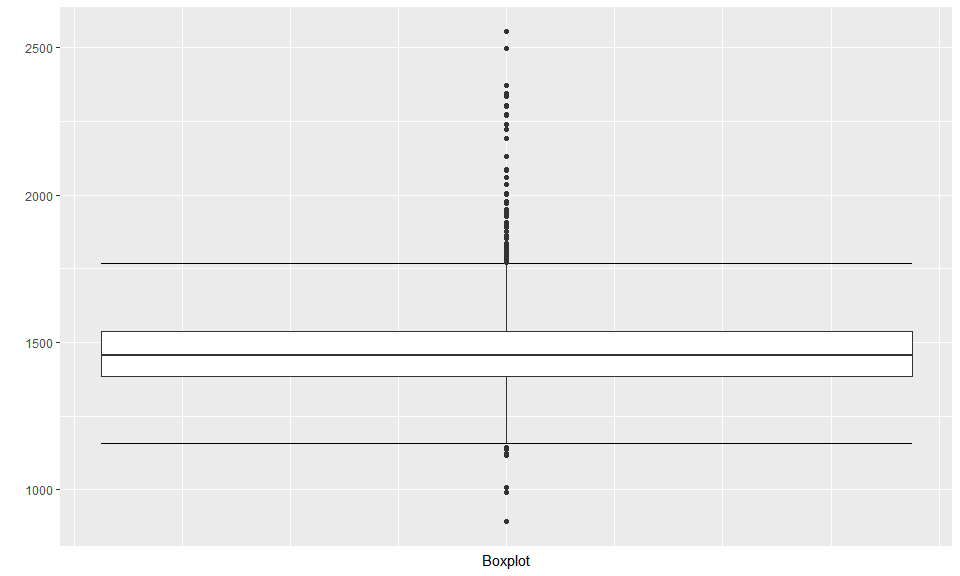
1454

1469.27

144.5912

2554

0



**Data Overview:** There are no missing values. The range and distribution are reasonable.

#### TEAM\_BATTING\_2B:

This variable represents number of team doubles:

Min

Median

Mean

SD

Max

Num of NaN

69

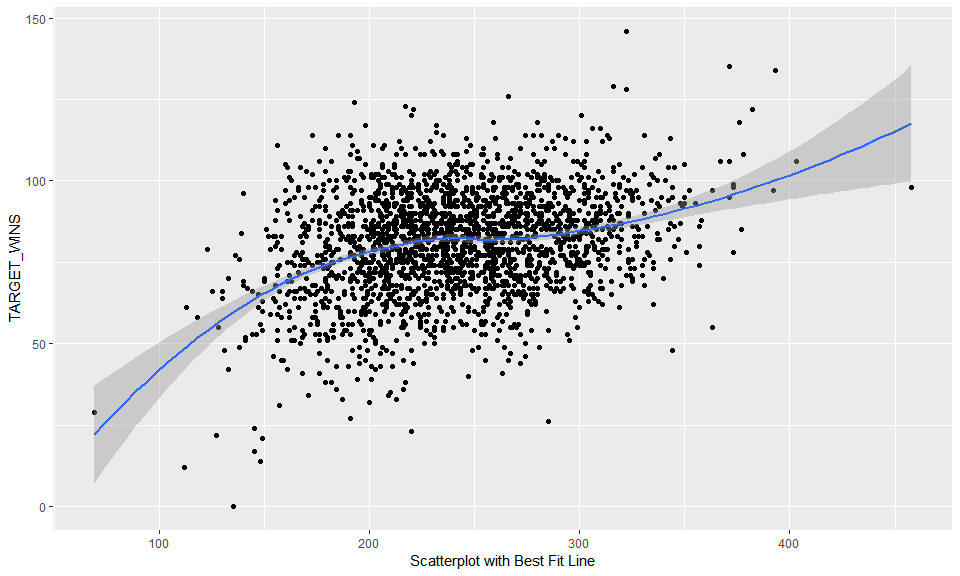
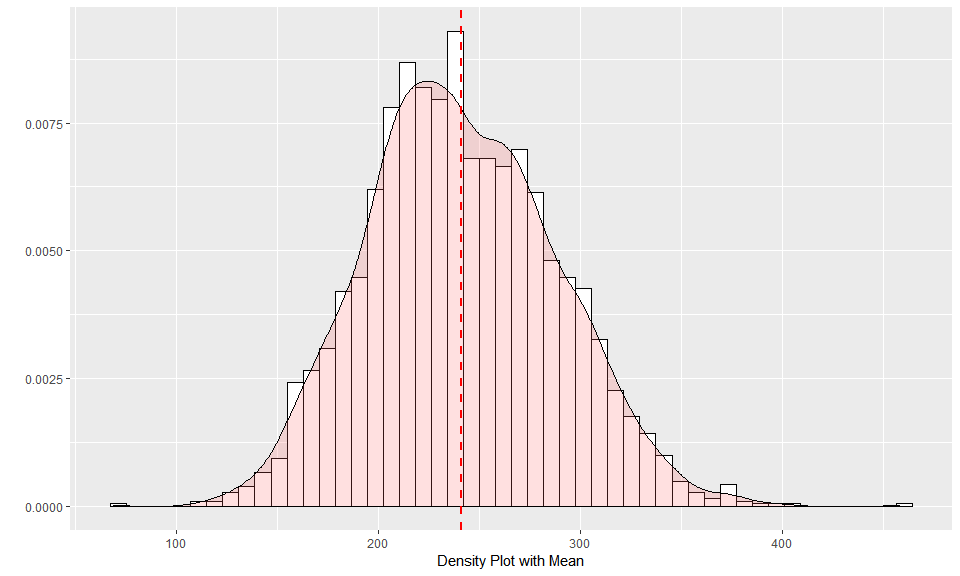
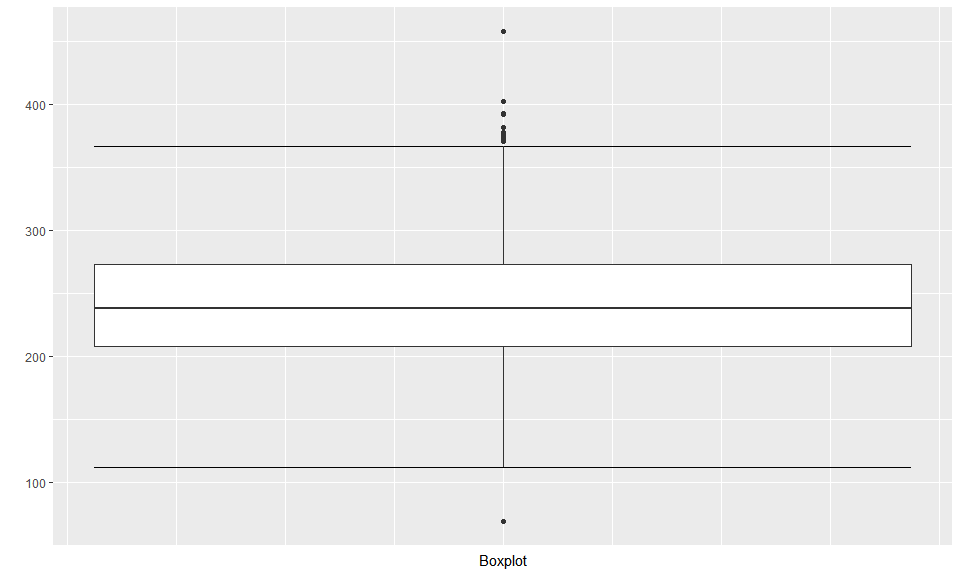
238

241.2469

46.80141

458

0



**Data Overview:** There are no missing values. The range and distribution are reasonable.

#### TEAM\_BATTING\_3B:

This variable represents number of team triples:

Min

Median

Mean

SD

Max

Num of NaN

0

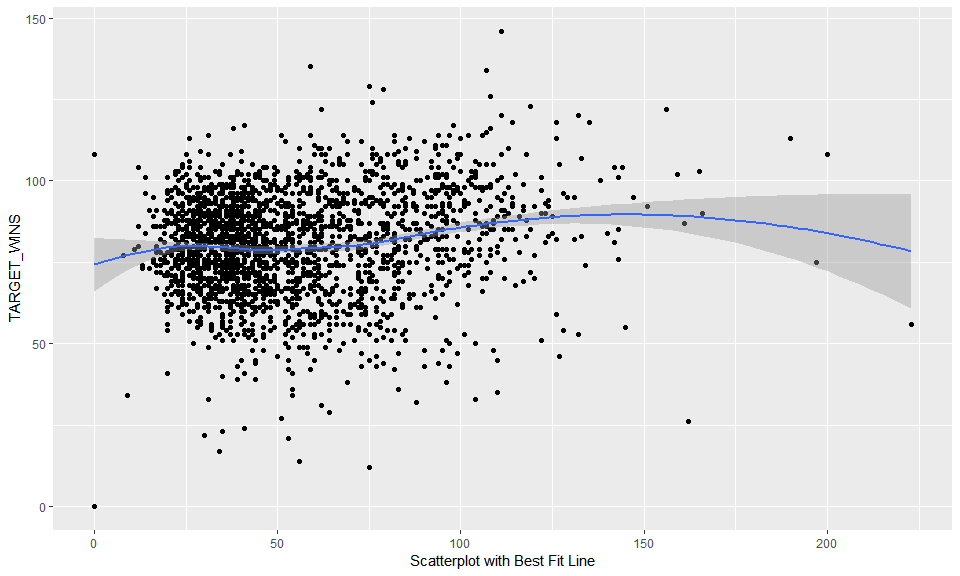
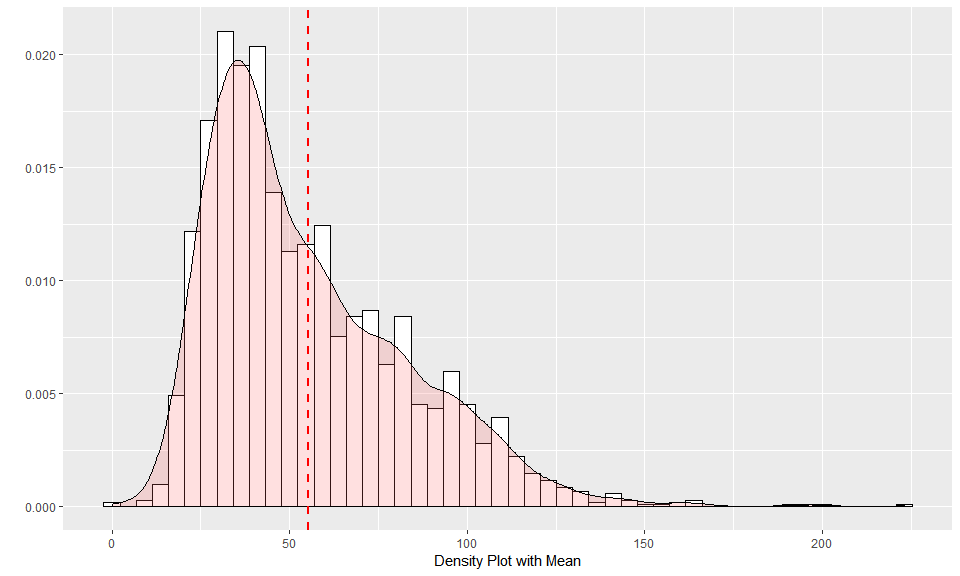
47

55.25

27.93856

223

0



**Data Overview:**The range and distribution are reasonable. There are 2 records with zero values which is unrealistic for a team in a season. One record (index 1347) has 12 variables with missing values, including the outcome variable. This record will be deleted from the data set. Second record (index 1494) has 7 missing variables, but it does have some recorded values in all categories - batting, pitching and fielding. Zero value for TEAM\_BATTING\_3B can be replaced with the median (because the distribution is right-skewed, median value will provide more realistic estimate).

#### TEAM\_BATTING\_HR:

This variable represents number of team triples:

Min

Median

Mean

SD

Max

Num of NaN

0

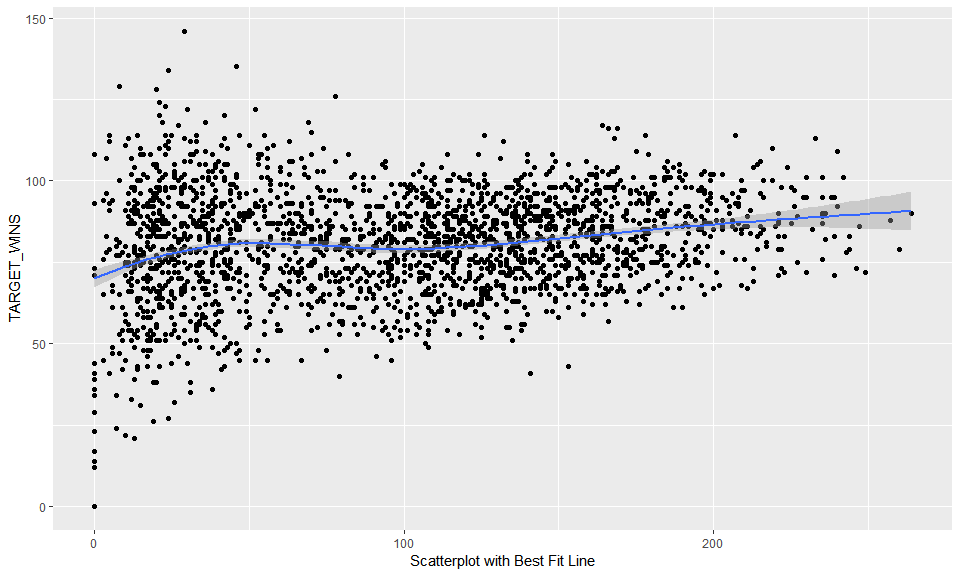
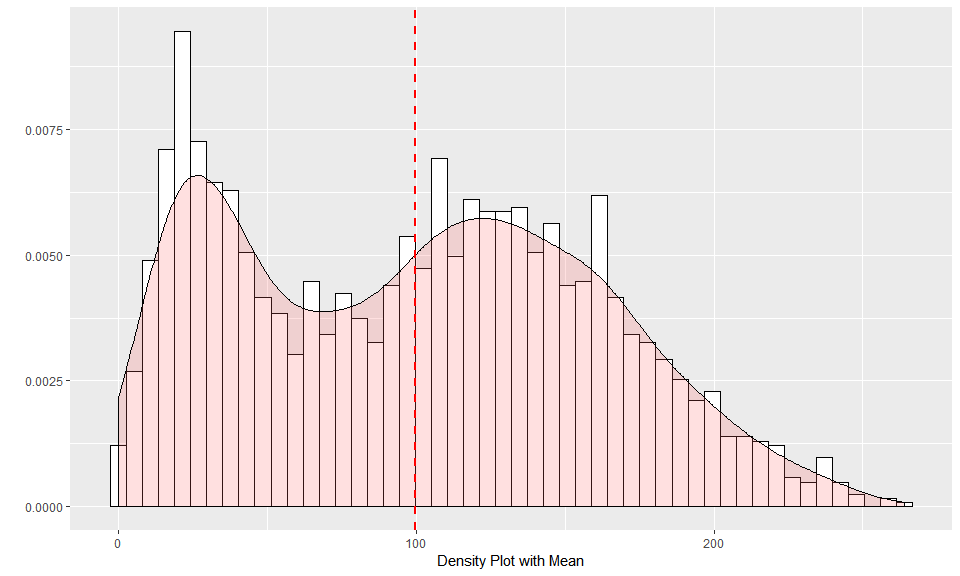
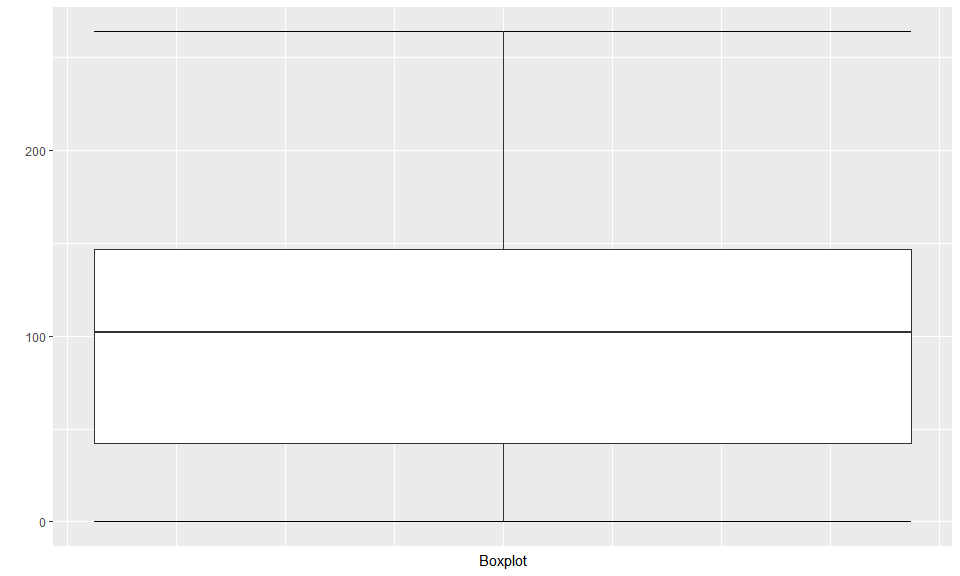
102

99.61204

60.54687

264

0



**Analysis:**There are some low values in the data. So zero doesn’t seem too unusual here either.

#### TEAM\_BATTING\_BB:

This variable represents Number of team walks

Min

Median

Mean

SD

Max

Num of NaN

0

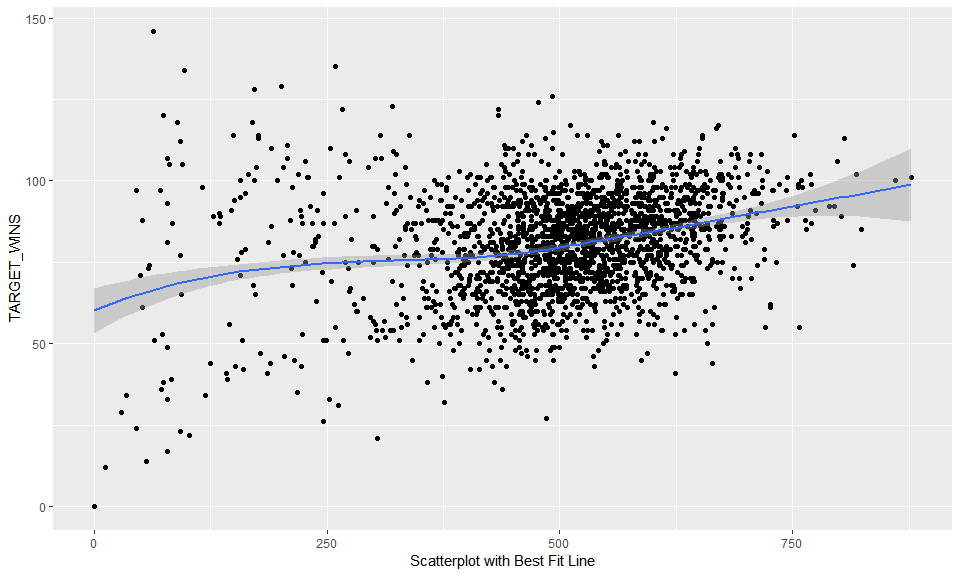
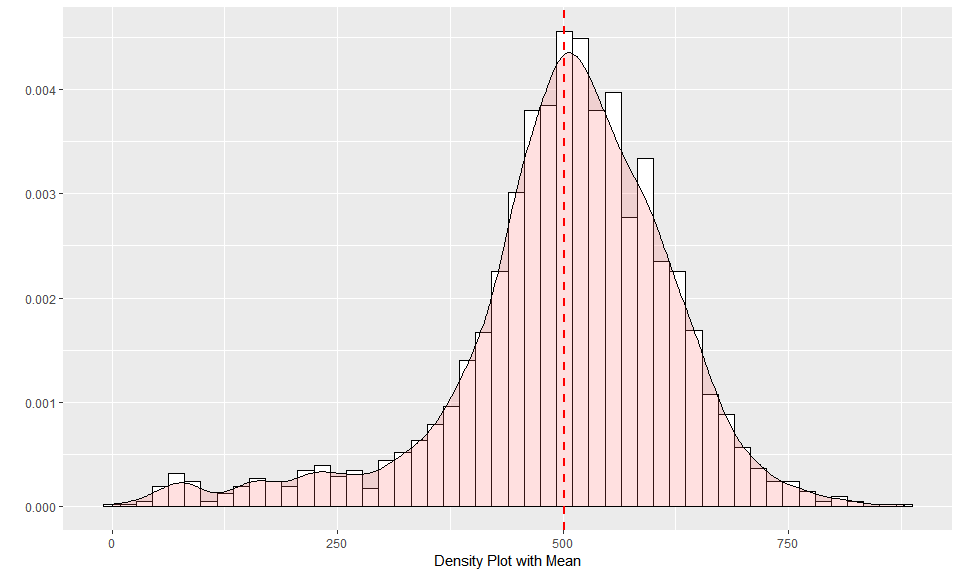
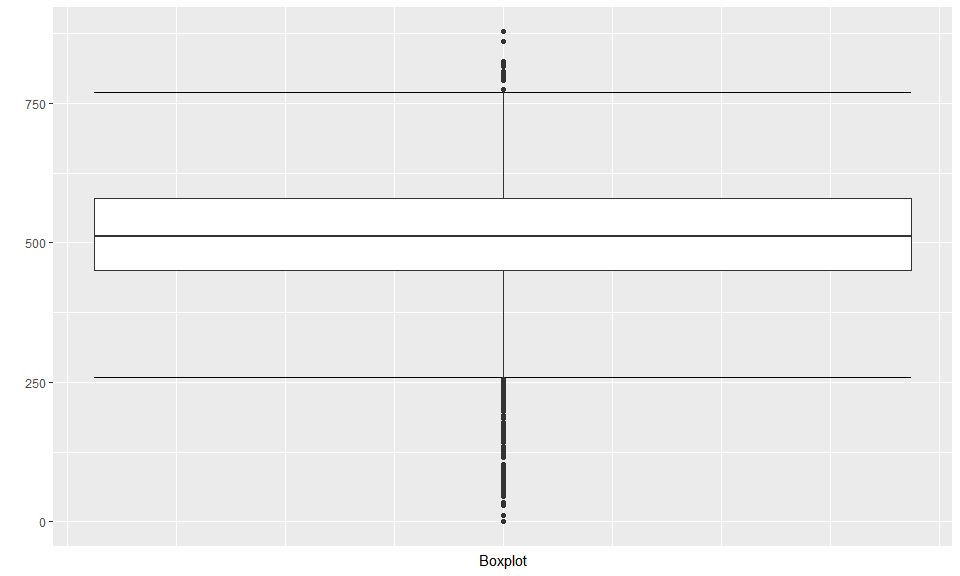
512

501.5589

122.6709

878

0



#### TEAM\_BATTING\_HBP:

This variable represents Number of team batters hit by pitch

Min

Median

Mean

SD

Max

Num of NaN

29

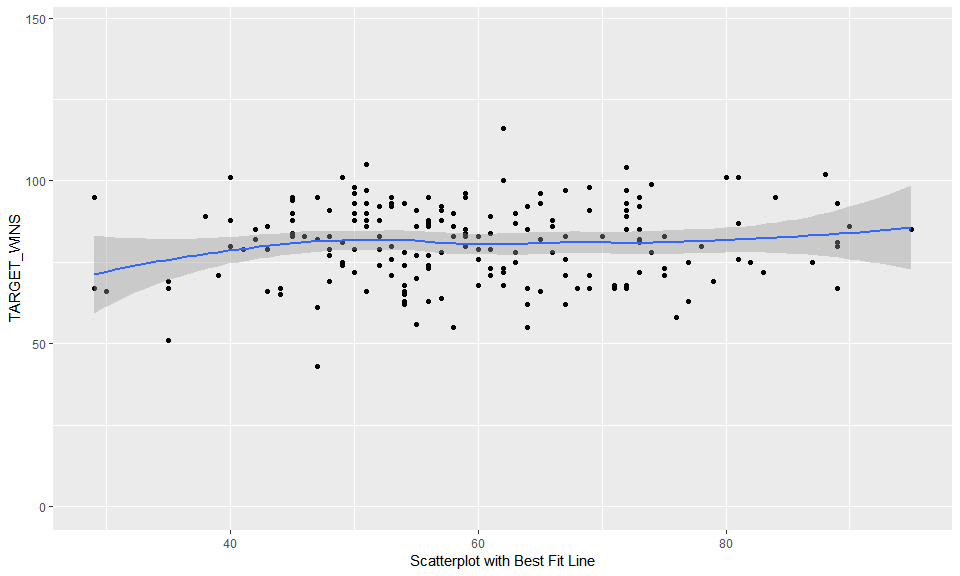
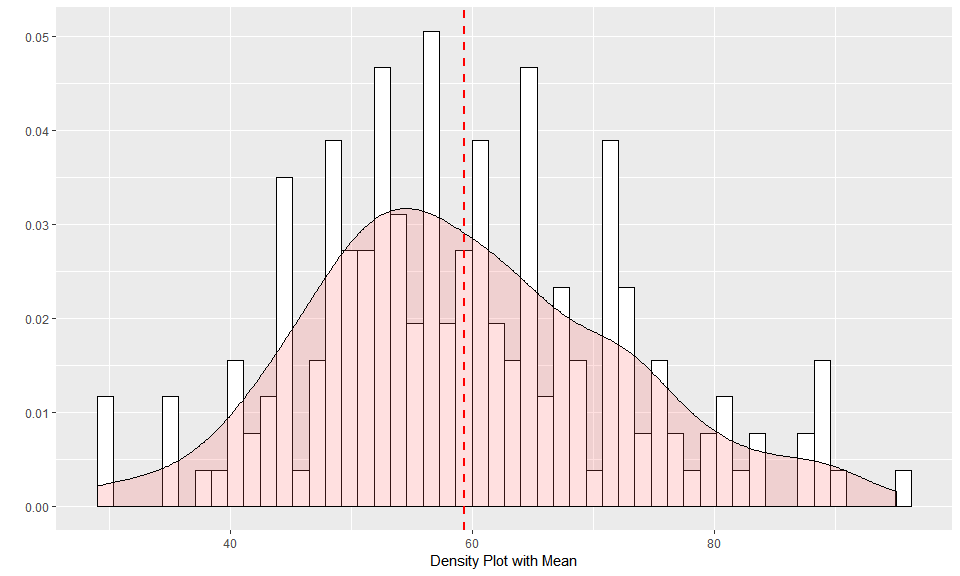
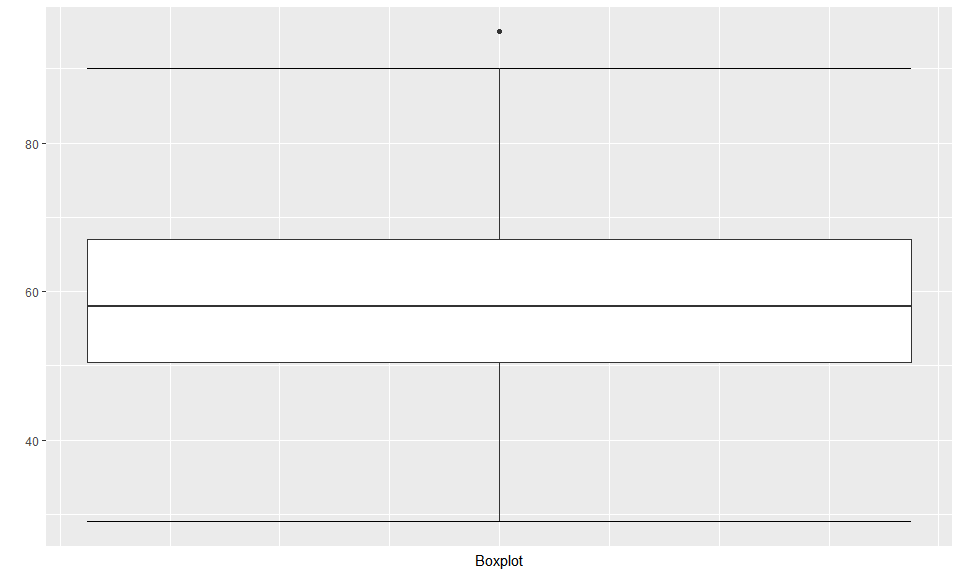
58

59.35602

12.96712

95

2085



**Analysis:** There are 2,085 records - 91.6% of data set - that are missing value. Because this variable is missing for majority of records, I wont consider this variable as input for regression model.

#### TEAM\_BATTING\_SO:

This variable represents Number of team strikeouts by batters

Min

Median

Mean

SD

Max

Num of NaN

0

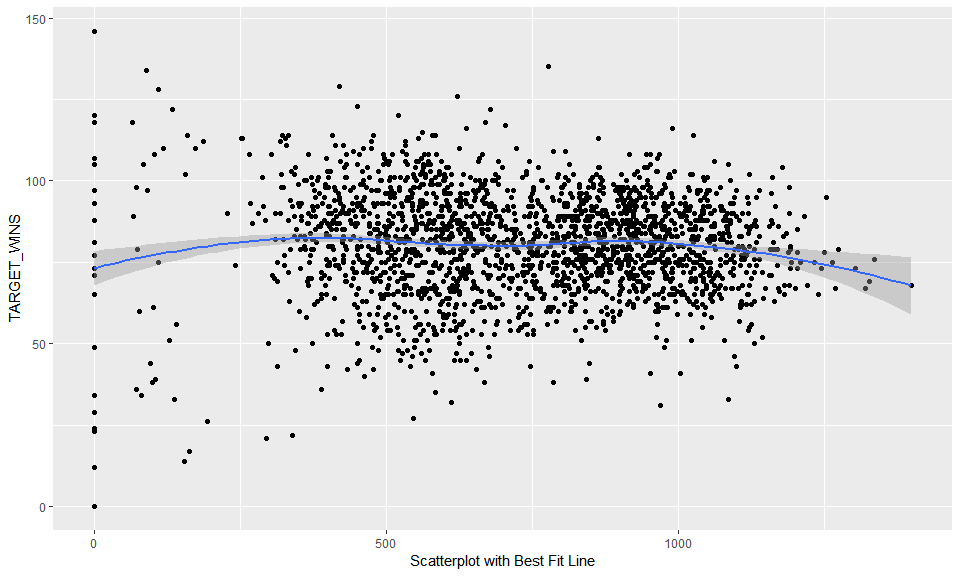
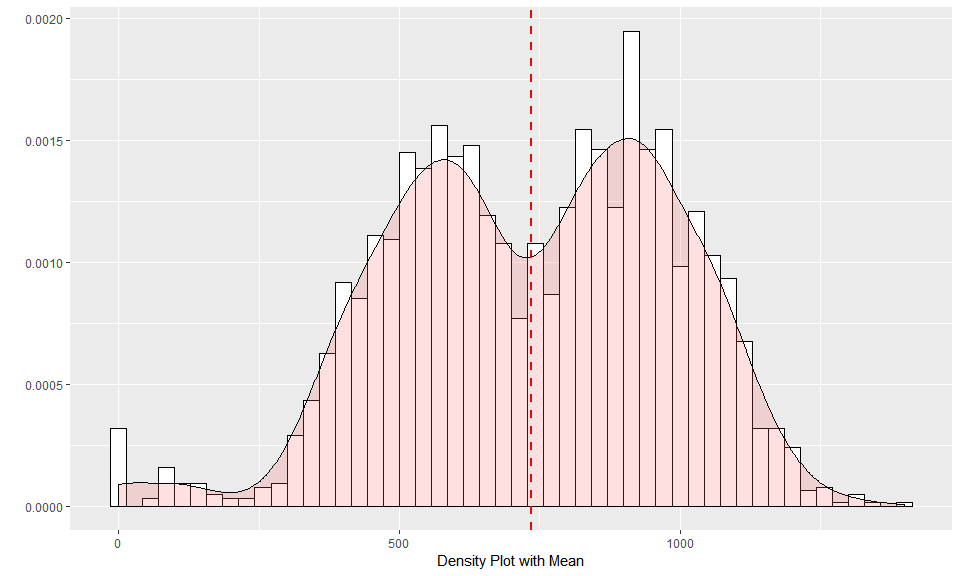
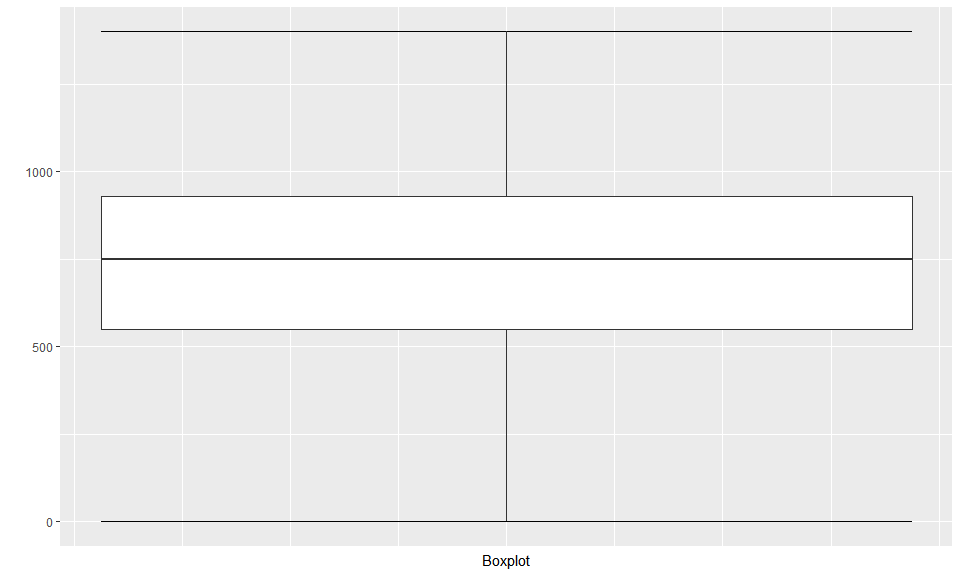
750

735.6053

248.5264

1399

102



**Analysis:** There are 122 records with missing or zero value (as wtih other variables a zero value is unrealistic). These values can be imputed. Similarly to homeruns, the distribution is multimodal, which is interesting enough for additional analysis. Another area of concern is a noticeable left tail. It is highly unlikely to have games without any strikeouts, so anything lower than 162 (average of 1 strikeout per game) is definitely suspect.

#### TEAM\_BASERUN\_SB:

This variable represents Number of team stolen bases

Min

Median

Mean

SD

Max

Num of NaN

0

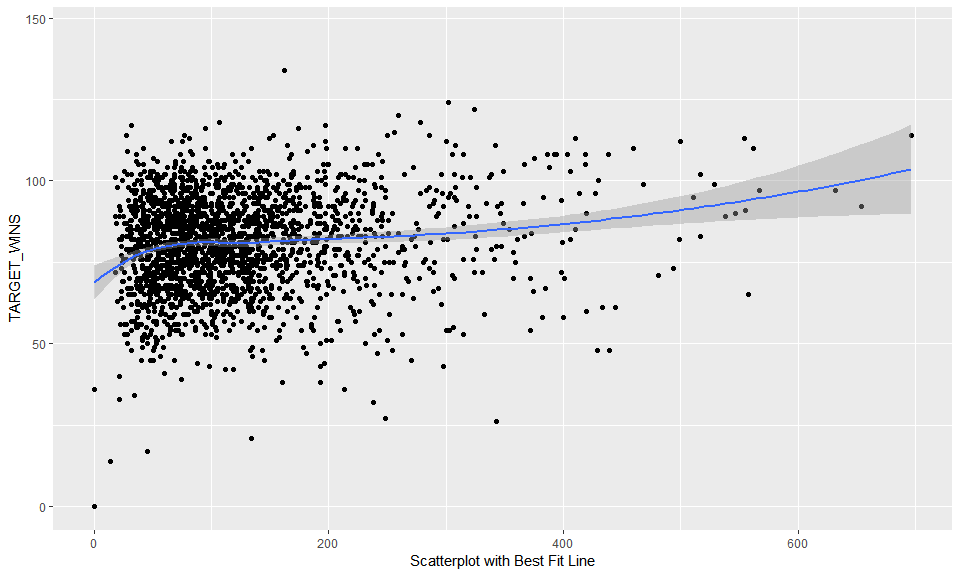
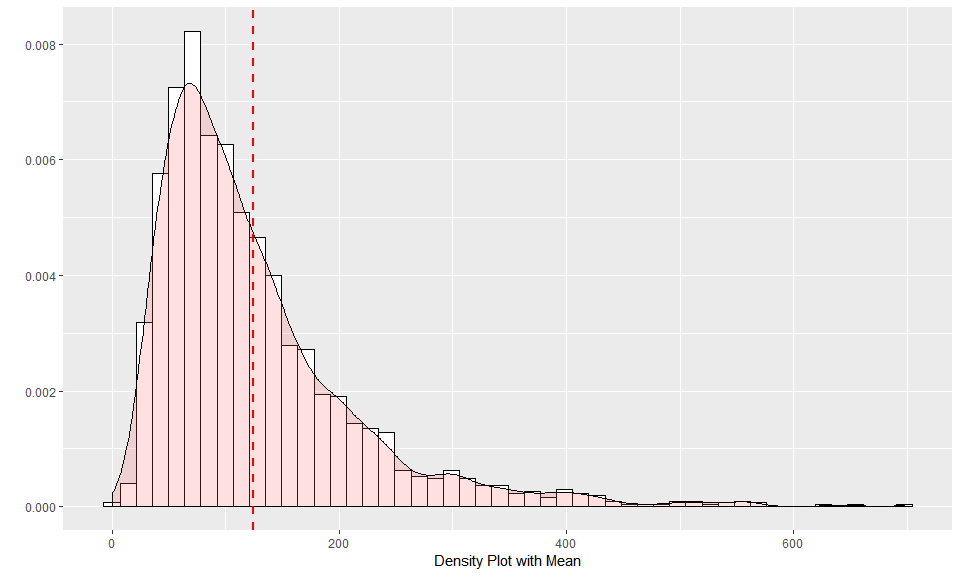
101

124.7618

87.79117

697

131



#### TEAM\_BASERUN\_CS:

This variable represents Number of team runners caught stealing

Min

Median

Mean

SD

Max

Num of NaN

0

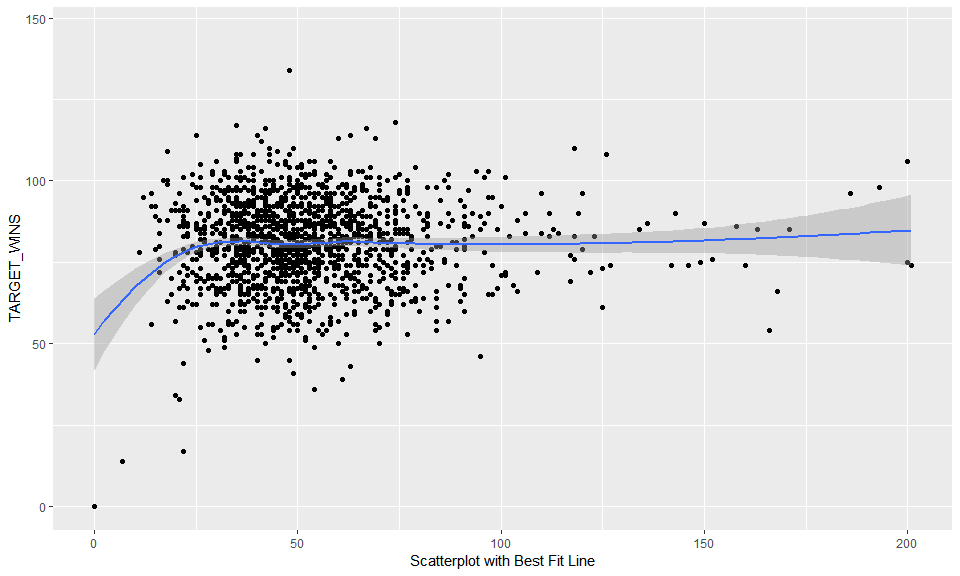
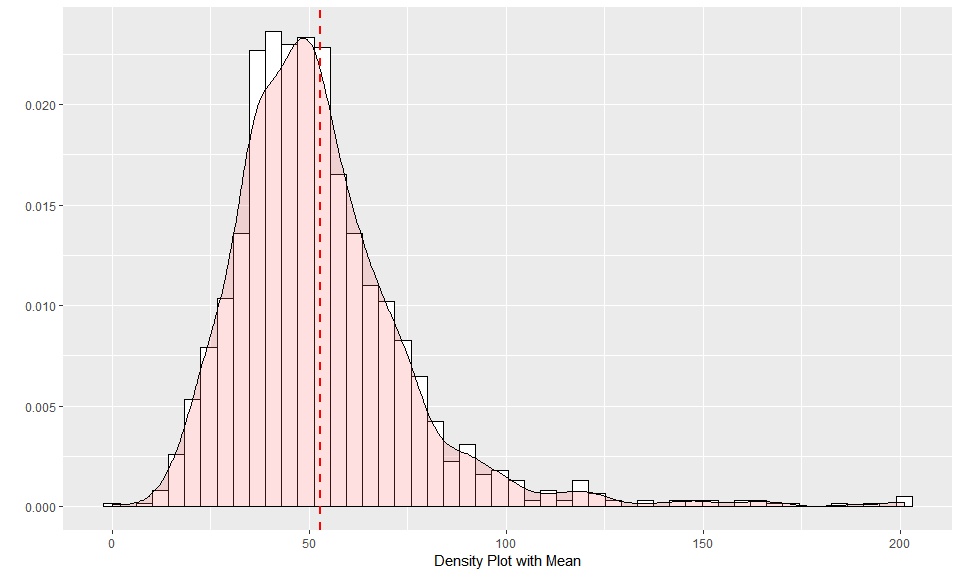
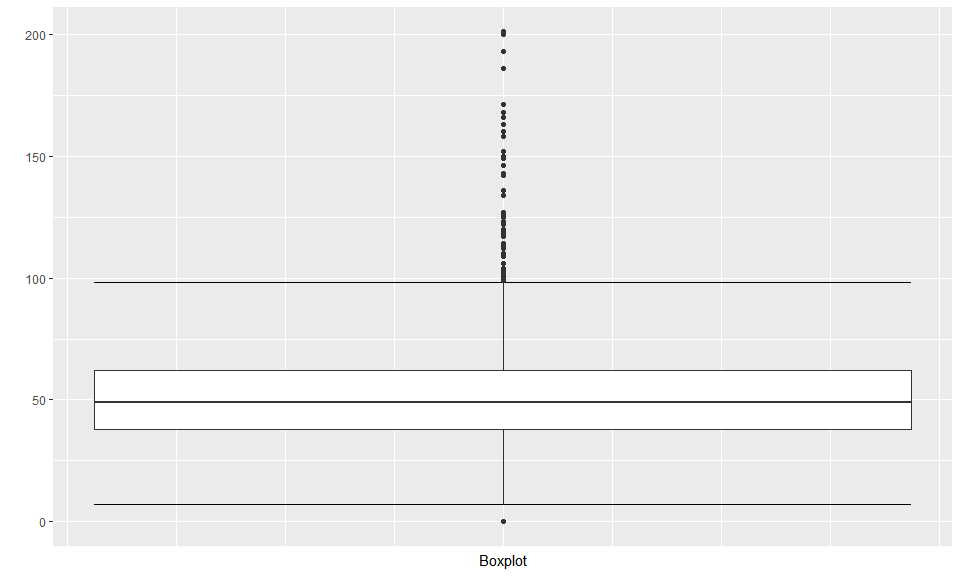
49

52.80386

22.95634

201

772



#### TEAM\_FIELDING\_E:

This variable represents Number of team fielding errors

Min

Median

Mean

SD

Max

Num of NaN

65

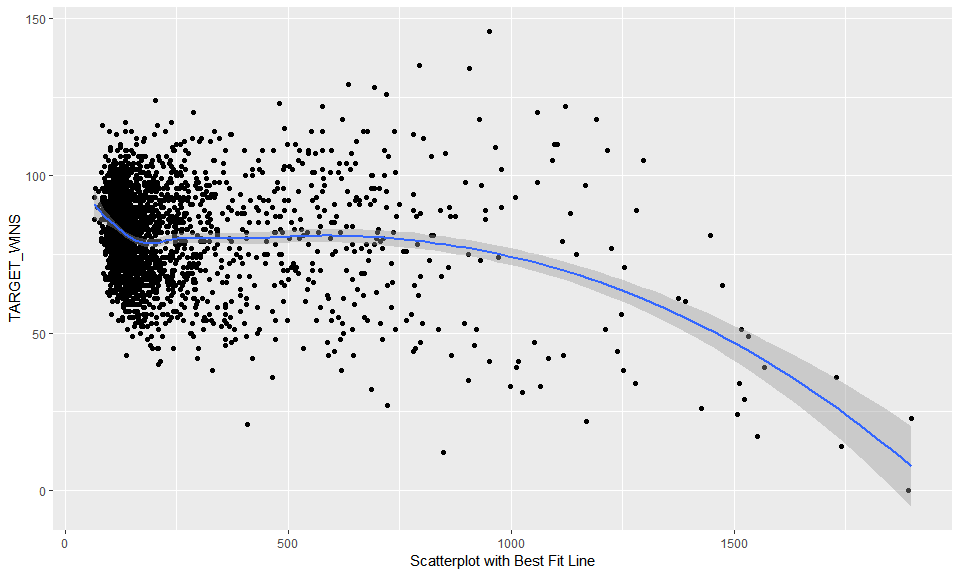
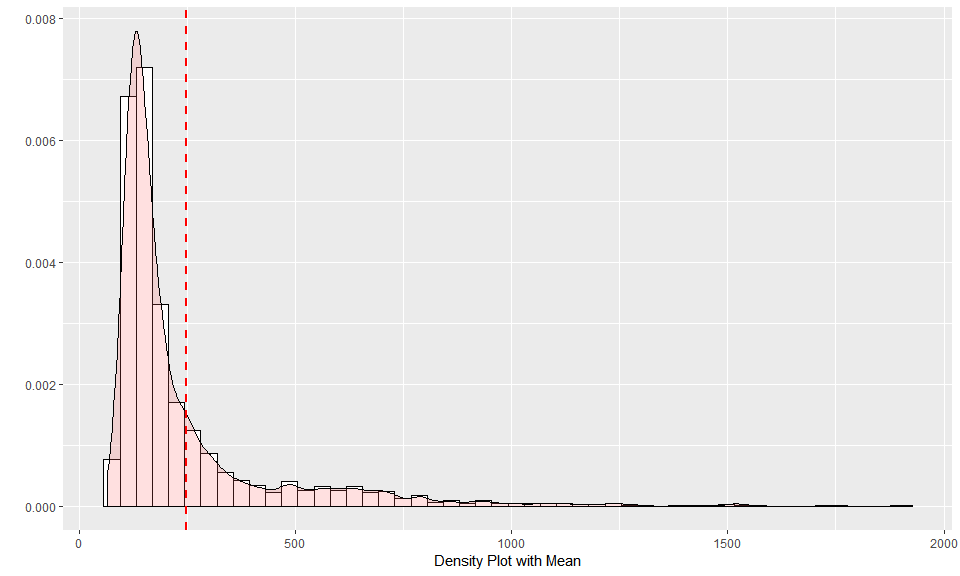
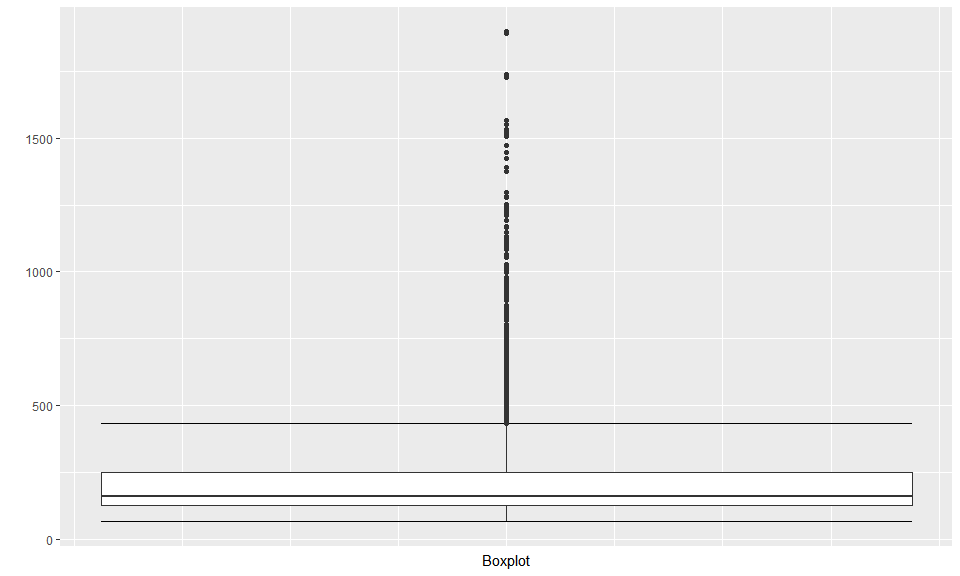
159

246.4807

227.771

1898

0



#### TEAM\_FIELDING\_DP:

This variable represents Number of team fielding double plays

Min

Median

Mean

SD

Max

Num of NaN

52

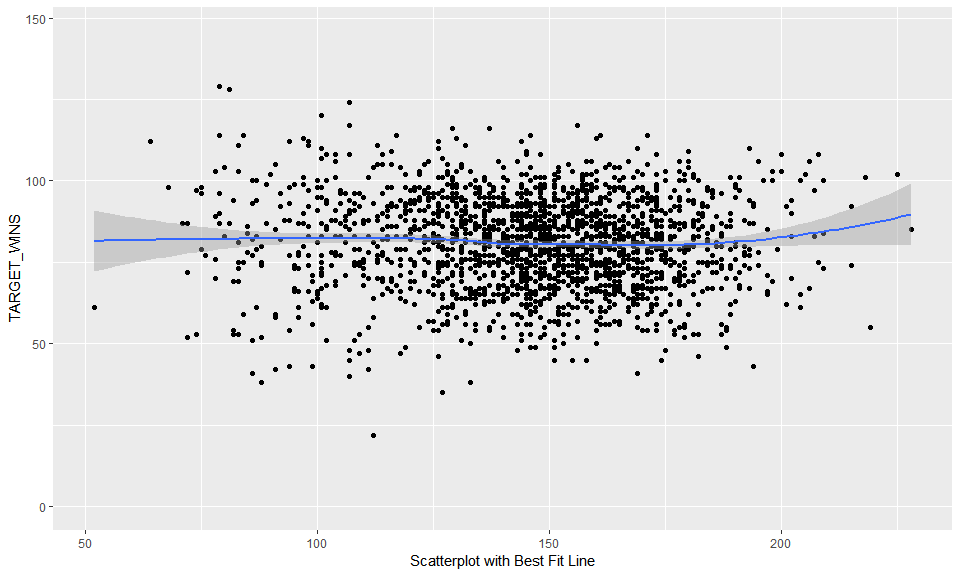
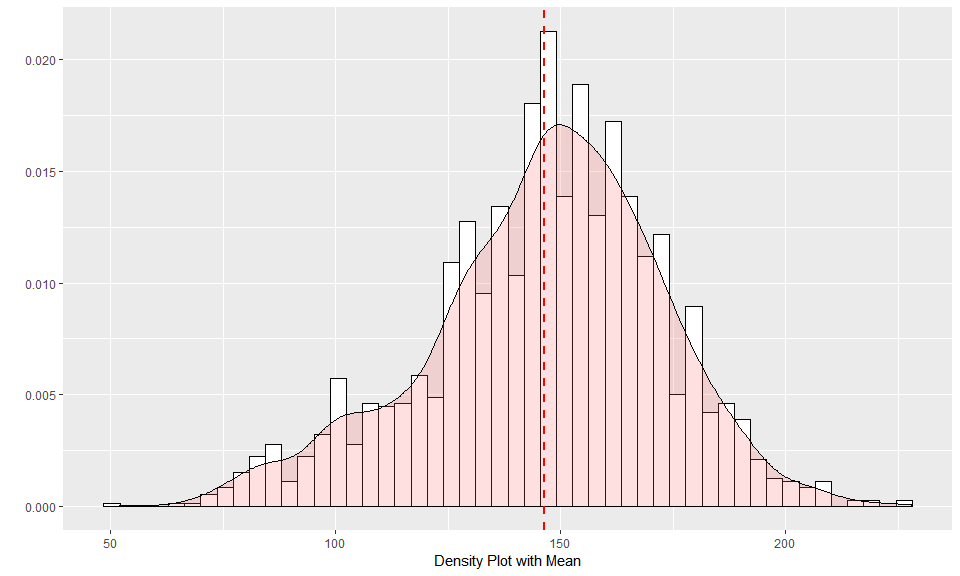
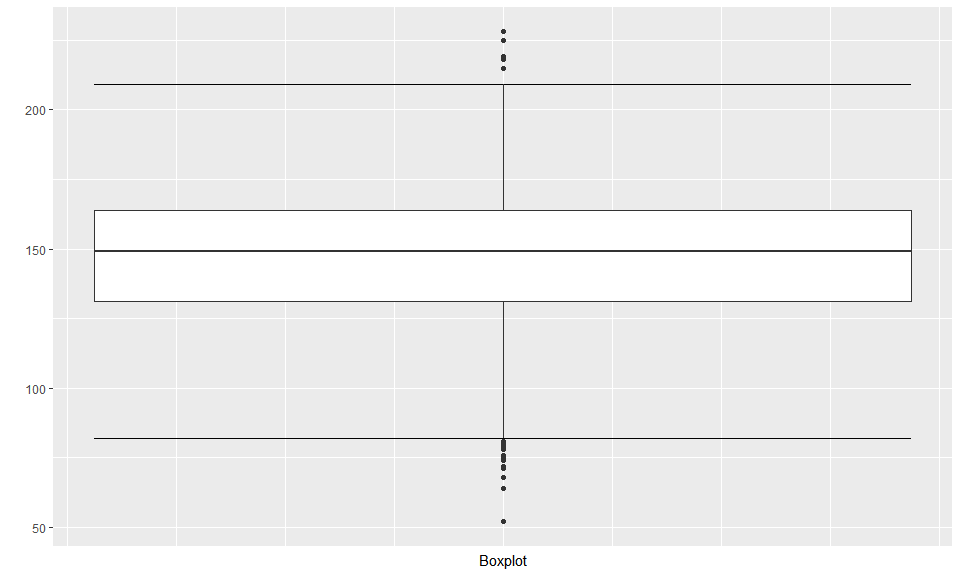
149

146.3879

26.22639

228

286



#### TEAM\_PITCHING\_BB:

This variable represents Number of walks given up by pitchers

Min

Median

Mean

SD

Max

Num of NaN

0

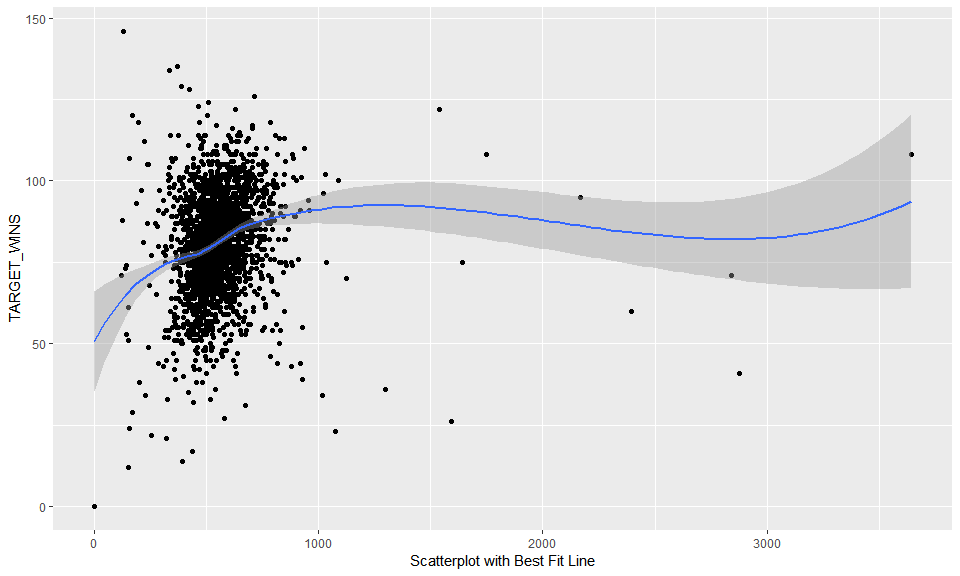
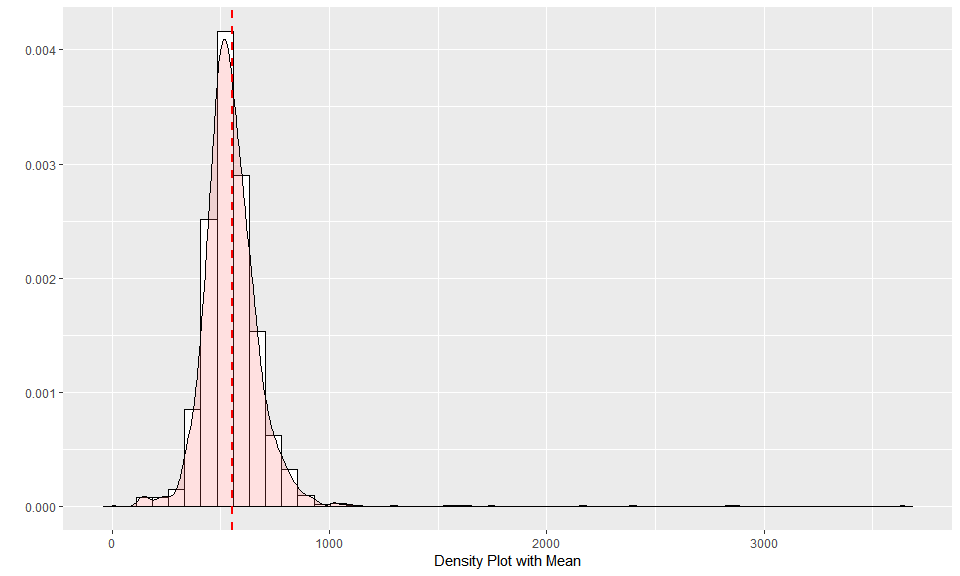
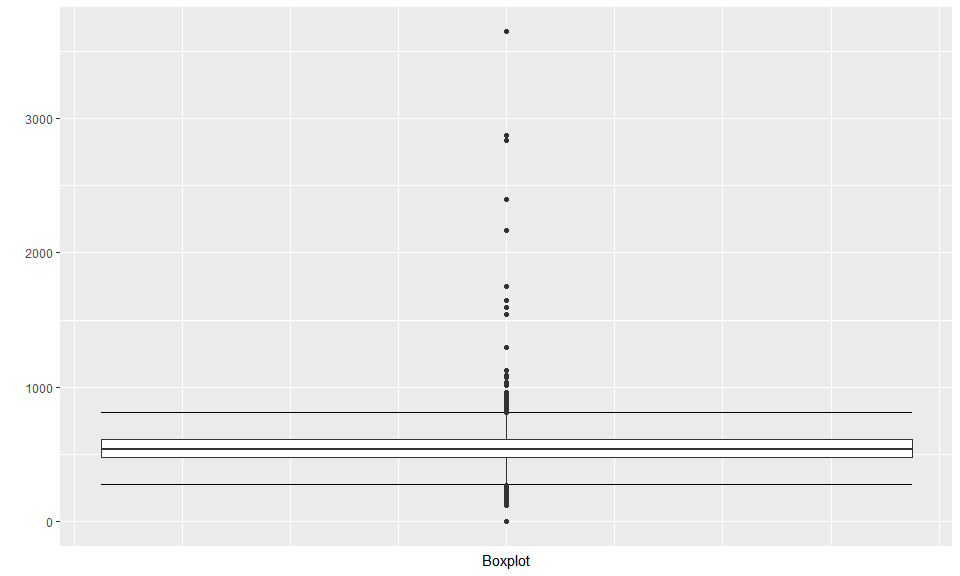
536.5

553.0079

166.3574

3645

0



**Analysis:** There are no missing values with the exception of record 1347 which will be deleted from model building. There are some unrealistic outliers.

#### TEAM\_PITCHING\_H:

This variable represents Number of base hits given up by pitchers

Min

Median

Mean

SD

Max

Num of NaN

1137

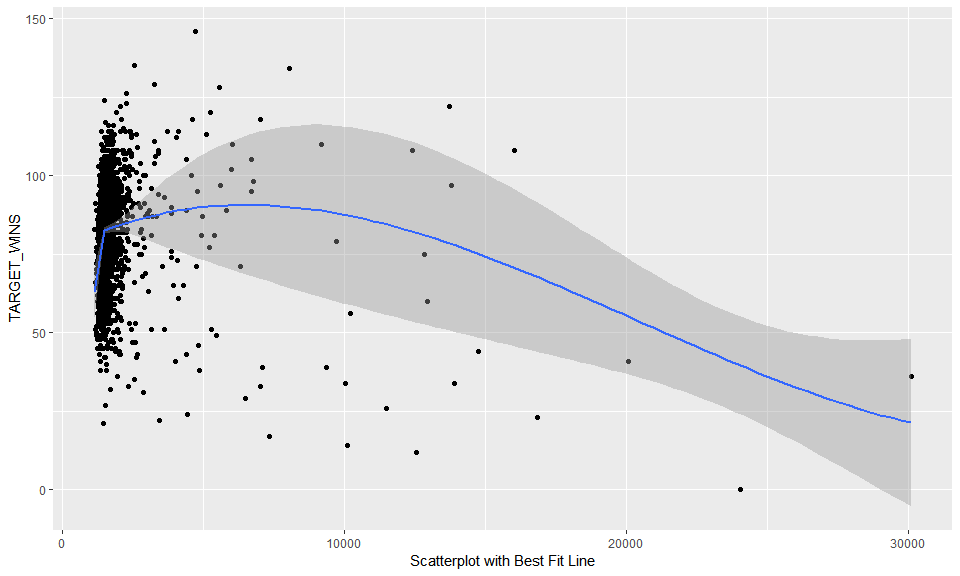
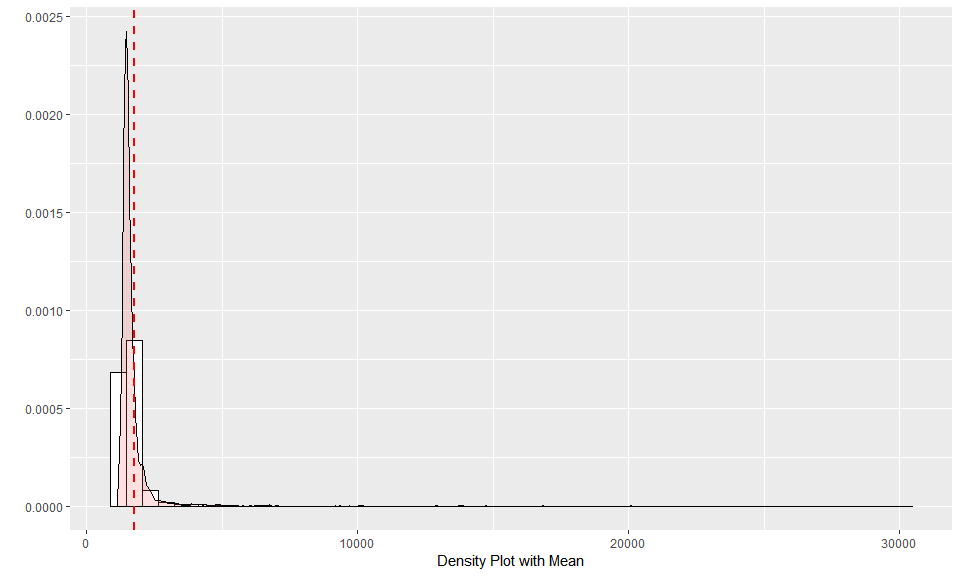
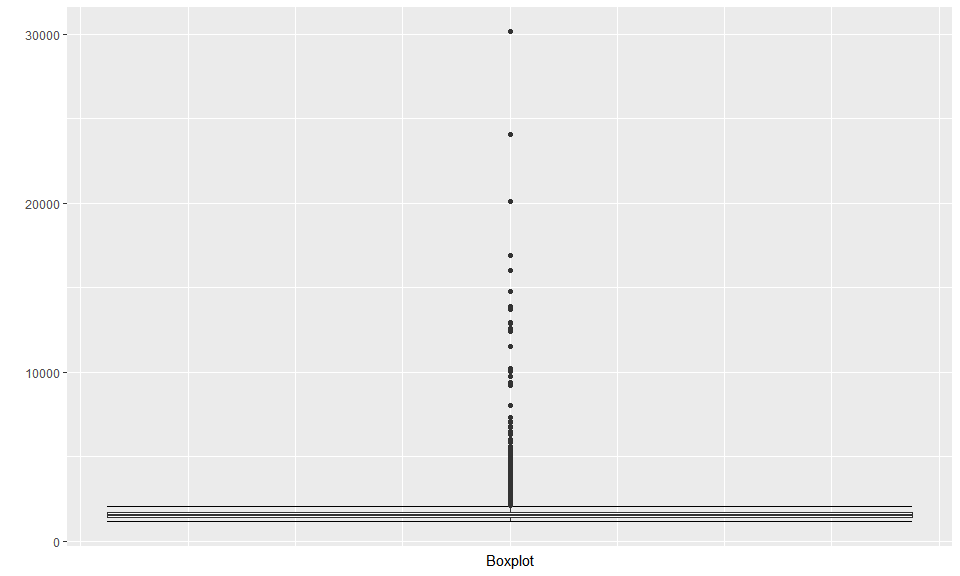
1518

1779.21

1406.843

30132

0



**Analysis:** Similar to TEAM\_PITCHING\_BB above, there are no missing value, but there issues with outliers. Based on visualizations, this variable will be capped at 13,000 and any value over this will be set to this cap.

#### TEAM\_PITCHING\_SO:

This variable represents Number of strikeouts by pitchers

Min

Median

Mean

SD

Max

Num of NaN

0

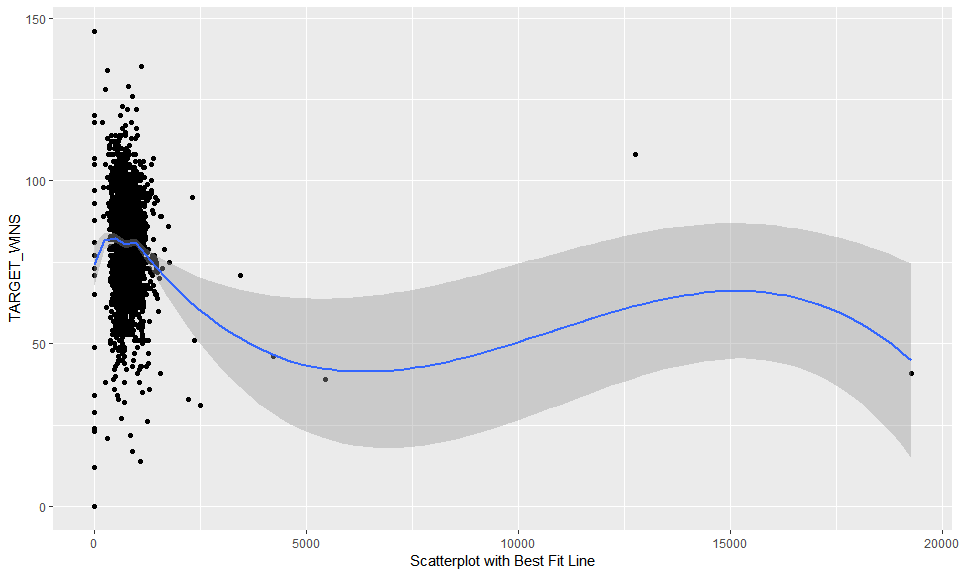
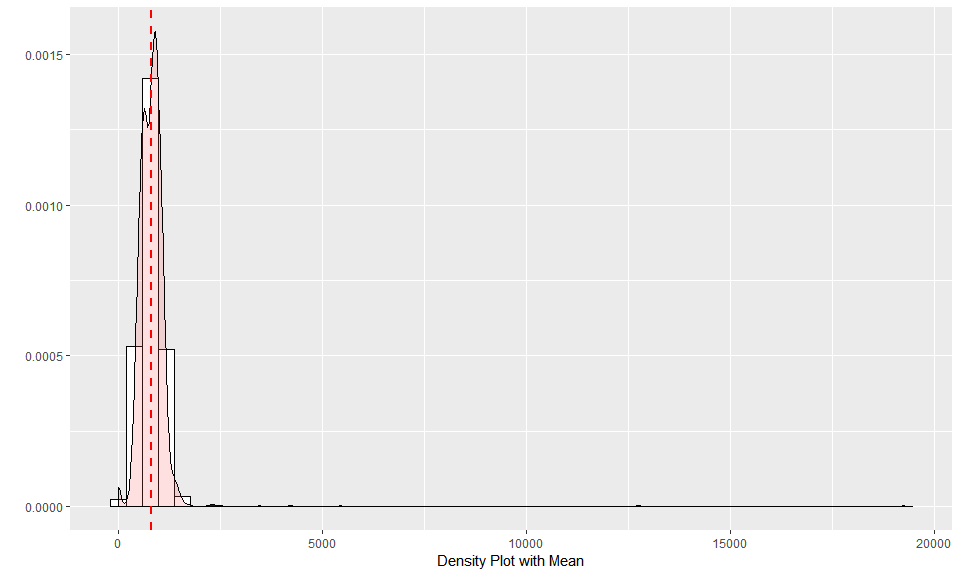
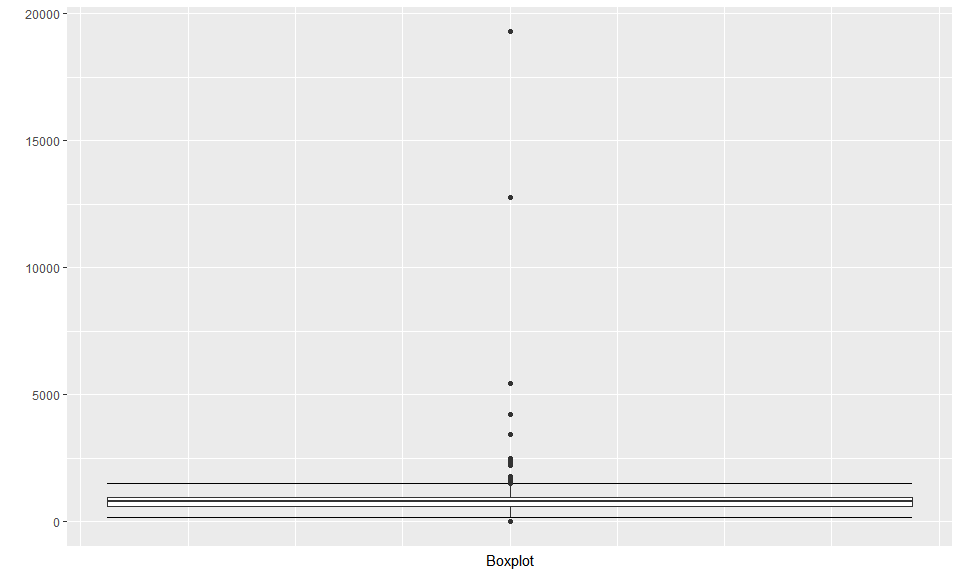
813.5

817.7305

553.085

19278

102



**Analysis:** This variable has 122 missing or zero values. They can be imputed as needed. There is also an outlier issue as graph shows.

#### TARGET\_WINS:

This variable represents Number of wins **(Outcome)**

Min

Median

Mean

SD

Max

Num of NaN

0

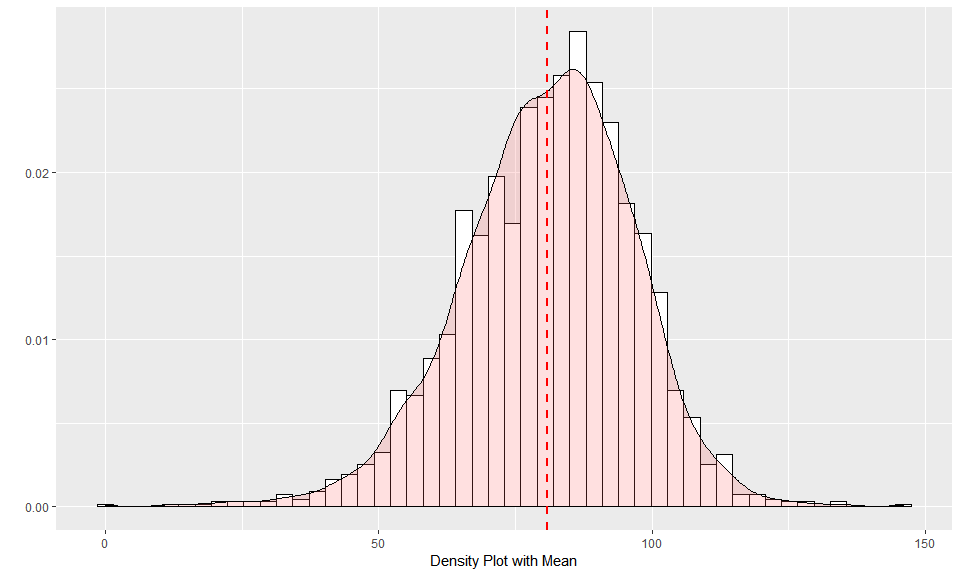
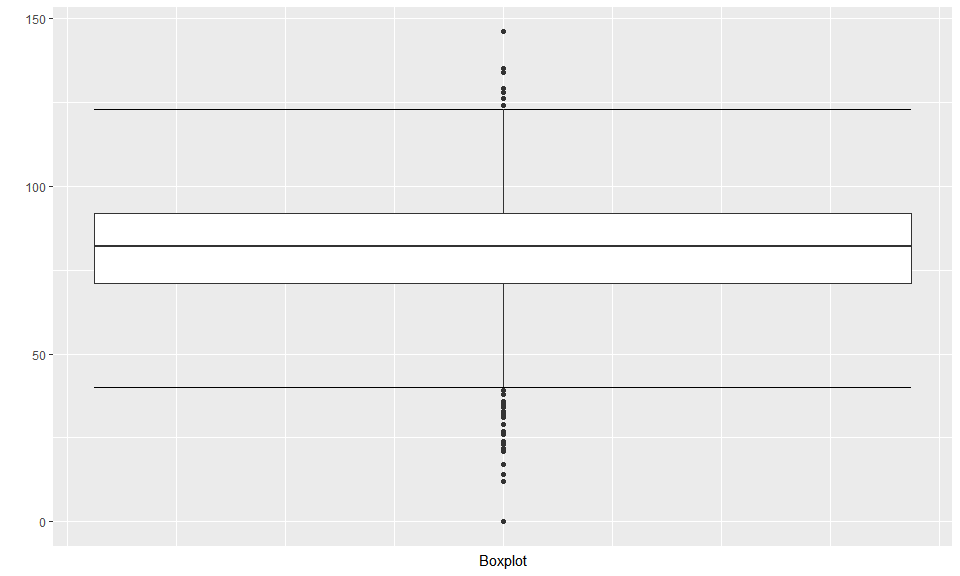
82

80.79086

15.75215

146

0

 **Analysis:** The range and distribution are reasonable. There are no missing values with the exception of record 1347.

## DATA PREPARATION

### Fixing Missing/Zero Values

First we will remove the invalid data and prep it for imputation. We will drop the hit by pitcher variable from the dataset.

clean\_data <- function(df){  
 # Change 0's to NA so they too can be imputed  
 df <- df %>%  
 mutate(TEAM\_BATTING\_SO = ifelse(TEAM\_BATTING\_SO == 0, NA, TEAM\_BATTING\_SO))  
 # Remove the high pitching strikeout values  
 df[which(df$TEAM\_PITCHING\_SO > 5346),"TEAM\_PITCHING\_SO"] <- NA  
 # Drop the hit by pitcher variable  
 df %>%  
 select(-TEAM\_BATTING\_HBP)  
}  
training <- clean\_data(training)  
evaluation <- clean\_data(evaluation)

### KNN imputation

set.seed(42)  
knn <- training %>% knnImputation()  
apply\_func <- function(df, knn){  
 impute\_me <- is.na(df$TEAM\_BATTING\_SO)  
 df[impute\_me,"TEAM\_BATTING\_SO"] <- knn[impute\_me,"TEAM\_BATTING\_SO"]   
 impute\_me <- is.na(df$TEAM\_BASERUN\_SB)  
 df[impute\_me,"TEAM\_BASERUN\_SB"] <- knn[impute\_me,"TEAM\_BASERUN\_SB"]   
 impute\_me <- is.na(df$TEAM\_BASERUN\_CS)  
 df[impute\_me,"TEAM\_BASERUN\_CS"] <- knn[impute\_me,"TEAM\_BASERUN\_CS"]   
 impute\_me <- is.na(df$TEAM\_PITCHING\_SO)  
 df[impute\_me,"TEAM\_PITCHING\_SO"] <- knn[impute\_me,"TEAM\_PITCHING\_SO"]  
 impute\_me <- is.na(df$TEAM\_FIELDING\_DP)  
 df[impute\_me,"TEAM\_FIELDING\_DP"] <- knn[impute\_me,"TEAM\_FIELDING\_DP"]  
 return(df)  
}  
training <- apply\_func(training, knn)

### Feature Engineering

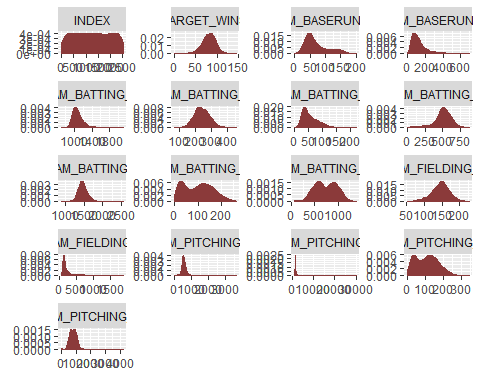
The batting singles is not included but we can back it out of the hits. We will do this.

add\_features <- function(df){  
 df %>%  
 mutate(TEAM\_BATTING\_1B = TEAM\_BATTING\_H - TEAM\_BATTING\_2B - TEAM\_BATTING\_3B - TEAM\_BATTING\_HR)  
}  
training <- add\_features(training)  
evaluation <- add\_features(evaluation)

### Model Data Look

Here’s what the data look like after imputation and correction:

training %>%  
 gather(variable, value) %>%  
 ggplot(., aes(value)) +   
 geom\_density(fill = "indianred4", color="indianred4") +   
 facet\_wrap(~variable, scales ="free", ncol = 4) +  
 labs(x = element\_blank(), y = element\_blank())



quick\_summary <- function(df){  
 df %>%  
 summary() %>%  
 kable() %>%  
 kable\_styling()  
}  
quick\_summary(training)

INDEX </th>

TARGET\_WINS

TEAM\_BATTING\_H

TEAM\_BATTING\_2B

TEAM\_BATTING\_3B

TEAM\_BATTING\_HR

TEAM\_BATTING\_BB

TEAM\_BATTING\_SO

TEAM\_BASERUN\_SB

TEAM\_BASERUN\_CS

TEAM\_PITCHING\_H

TEAM\_PITCHING\_HR

TEAM\_PITCHING\_BB

TEAM\_PITCHING\_SO

TEAM\_FIELDING\_E

TEAM\_FIELDING\_DP

TEAM\_BATTING\_1B

Min. : 1.0

Min. : 0.00

Min. : 891

Min. : 69.0

Min. : 0.00

Min. : 0.00

Min. : 0.0

Min. : 66

Min. : 0.0

Min. : 0.0

Min. : 1137

Min. : 0.0

Min. : 0.0

Min. : 0.0

Min. : 65.0

Min. : 52.0

Min. : 709.0

1st Qu.: 630.8

1st Qu.: 71.00

1st Qu.:1383

1st Qu.:208.0

1st Qu.: 34.00

1st Qu.: 42.00

1st Qu.:451.0

1st Qu.: 554

1st Qu.: 67.0

1st Qu.: 43.0

1st Qu.: 1419

1st Qu.: 50.0

1st Qu.: 476.0

1st Qu.: 618.5

1st Qu.: 127.0

1st Qu.:130.0

1st Qu.: 990.8

Median :1270.5

Median : 82.00

Median :1454

Median :238.0

Median : 47.00

Median :102.00

Median :512.0

Median : 733

Median :104.0

Median : 58.0

Median : 1518

Median :107.0

Median : 536.5

Median : 797.0

Median : 159.0

Median :147.0

Median :1050.0

Mean :1268.5

Mean : 80.79

Mean :1469

Mean :241.2

Mean : 55.25

Mean : 99.61

Mean :501.6

Mean : 735

Mean :124.7

Mean : 69.7

Mean : 1779

Mean :105.7

Mean : 553.0

Mean : 795.8

Mean : 246.5

Mean :145.4

Mean :1073.2

3rd Qu.:1915.5

3rd Qu.: 92.00

3rd Qu.:1537

3rd Qu.:273.0

3rd Qu.: 72.00

3rd Qu.:147.00

3rd Qu.:580.0

3rd Qu.: 925

3rd Qu.:153.2

3rd Qu.: 89.0

3rd Qu.: 1682

3rd Qu.:150.0

3rd Qu.: 611.0

3rd Qu.: 957.0

3rd Qu.: 249.2

3rd Qu.:162.0

3rd Qu.:1129.0

Max. :2535.0

Max. :146.00

Max. :2554

Max. :458.0

Max. :223.00

Max. :264.00

Max. :878.0

Max. :1399

Max. :697.0

Max. :201.0

Max. :30132

Max. :343.0

Max. :3645.0

Max. :4224.0

Max. :1898.0

Max. :228.0

Max. :2112.0

## BUILD MODELS

### Model 1

The first model includes several variables, selected manually, that have higher than average correlation to the target variable. They cover hitting, walking and fielding errors.

Call:  
lm(formula = TARGET\_WINS ~ TEAM\_BATTING\_H + TEAM\_BATTING\_BB +   
 TEAM\_FIELDING\_E, data = training)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-50.901 -9.079 0.080 9.089 51.837   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 3.662987 3.299955 1.110 0.267   
TEAM\_BATTING\_H 0.049554 0.002081 23.807 < 2e-16 \*\*\*  
TEAM\_BATTING\_BB 0.015938 0.003134 5.085 3.98e-07 \*\*\*  
TEAM\_FIELDING\_E -0.014908 0.001746 -8.538 < 2e-16 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 13.71 on 2272 degrees of freedom  
Multiple R-squared: 0.2437, Adjusted R-squared: 0.2427   
F-statistic: 244.1 on 3 and 2272 DF, p-value: < 2.2e-16

All variables are significant, but the value is relatively small at 0.2356.

### Model 2

The second model expand the base hit variable, TEAM\_BATTING\_H, into its components - singles, doubles, triples and home runs.

Call:  
lm(formula = TARGET\_WINS ~ TEAM\_BATTING\_1B + TEAM\_BATTING\_2B +   
 TEAM\_BATTING\_3B + TEAM\_BATTING\_HR + TEAM\_BATTING\_BB + TEAM\_FIELDING\_E,   
 data = training)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-52.307 -8.825 0.088 8.699 63.698   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 7.866838 3.433496 2.291 0.022043 \*   
TEAM\_BATTING\_1B 0.045773 0.003141 14.572 < 2e-16 \*\*\*  
TEAM\_BATTING\_2B 0.021963 0.007412 2.963 0.003075 \*\*   
TEAM\_BATTING\_3B 0.160496 0.014912 10.763 < 2e-16 \*\*\*  
TEAM\_BATTING\_HR 0.080311 0.007755 10.356 < 2e-16 \*\*\*  
TEAM\_BATTING\_BB 0.012342 0.003200 3.857 0.000118 \*\*\*  
TEAM\_FIELDING\_E -0.018474 0.001946 -9.495 < 2e-16 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 13.52 on 2269 degrees of freedom  
Multiple R-squared: 0.2648, Adjusted R-squared: 0.2628   
F-statistic: 136.2 on 6 and 2269 DF, p-value: < 2.2e-16

All variables are still significant and is slightly improved at 0.2574.

### Higher Order Stepwise Regression

For the third model we will use a stepwise regression method using a backwards elimination process. We also introduce some higher order polynomial variables.

full\_formula <- "TARGET\_WINS ~ TEAM\_BATTING\_2B + TEAM\_BATTING\_3B + TEAM\_BATTING\_HR + TEAM\_BATTING\_BB + TEAM\_BATTING\_SO + TEAM\_BASERUN\_SB + TEAM\_BASERUN\_CS + TEAM\_PITCHING\_H + TEAM\_PITCHING\_HR + TEAM\_PITCHING\_BB + TEAM\_PITCHING\_SO + TEAM\_FIELDING\_E + TEAM\_FIELDING\_DP + TEAM\_BATTING\_1B + I(TEAM\_BATTING\_2B^2) + I(TEAM\_BATTING\_3B^2) + I(TEAM\_BATTING\_HR^2) + I(TEAM\_BATTING\_BB^2) + I(TEAM\_BATTING\_SO^2) + I(TEAM\_BASERUN\_SB^2) + I(TEAM\_BASERUN\_CS^2) + I(TEAM\_PITCHING\_H^2) + I(TEAM\_PITCHING\_HR^2) + I(TEAM\_PITCHING\_BB^2) + I(TEAM\_PITCHING\_SO^2) + I(TEAM\_FIELDING\_E^2) + I(TEAM\_FIELDING\_DP^2) + I(TEAM\_BATTING\_1B^2) "  
full\_model <- lm(full\_formula, training)  
step\_back <- MASS::stepAIC(full\_model, direction="backward", trace = F)  
poly\_call <- summary(step\_back)$call  
step\_back <- lm(poly\_call[2], training)  
summary(step\_back)

Call:  
lm(formula = poly\_call[2], data = training)  
  
Residuals:  
 Min 1Q Median 3Q Max   
-42.356 -8.122 -0.046 7.607 62.986   
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 8.045e+01 1.448e+01 5.555 3.10e-08 \*\*\*  
TEAM\_BATTING\_2B 1.587e-01 4.641e-02 3.420 0.000636 \*\*\*  
TEAM\_BATTING\_3B 1.223e-01 1.639e-02 7.462 1.21e-13 \*\*\*  
TEAM\_BATTING\_BB -2.121e-01 1.842e-02 -11.513 < 2e-16 \*\*\*  
TEAM\_BATTING\_SO 4.911e-02 8.382e-03 5.859 5.34e-09 \*\*\*  
TEAM\_BASERUN\_SB 2.511e-02 5.260e-03 4.773 1.93e-06 \*\*\*  
TEAM\_PITCHING\_H -4.660e-03 1.181e-03 -3.946 8.17e-05 \*\*\*  
TEAM\_PITCHING\_HR 1.451e-01 2.173e-02 6.675 3.09e-11 \*\*\*  
TEAM\_PITCHING\_BB 2.400e-02 3.617e-03 6.636 4.03e-11 \*\*\*  
TEAM\_FIELDING\_E -5.477e-02 5.779e-03 -9.477 < 2e-16 \*\*\*  
TEAM\_FIELDING\_DP -9.404e-02 1.358e-02 -6.926 5.62e-12 \*\*\*  
TEAM\_BATTING\_1B -3.278e-02 1.900e-02 -1.725 0.084587 .   
I(TEAM\_BATTING\_2B^2) -2.176e-04 9.132e-05 -2.383 0.017273 \*   
I(TEAM\_BATTING\_HR^2) 3.910e-04 7.362e-05 5.311 1.19e-07 \*\*\*  
I(TEAM\_BATTING\_BB^2) 1.930e-04 1.586e-05 12.168 < 2e-16 \*\*\*  
I(TEAM\_BATTING\_SO^2) -3.677e-05 5.228e-06 -7.032 2.68e-12 \*\*\*  
I(TEAM\_BASERUN\_CS^2) 5.579e-04 8.535e-05 6.537 7.75e-11 \*\*\*  
I(TEAM\_PITCHING\_H^2) 9.878e-08 3.674e-08 2.689 0.007230 \*\*   
I(TEAM\_PITCHING\_HR^2) -4.589e-04 8.618e-05 -5.325 1.11e-07 \*\*\*  
I(TEAM\_FIELDING\_E^2) 1.141e-05 4.147e-06 2.751 0.005988 \*\*   
I(TEAM\_BATTING\_1B^2) 3.795e-05 7.795e-06 4.868 1.20e-06 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 12.32 on 2255 degrees of freedom  
Multiple R-squared: 0.3941, Adjusted R-squared: 0.3887   
F-statistic: 73.33 on 20 and 2255 DF, p-value: < 2.2e-16

This model has the highest adjusted R-squared value .Some variables p-values are not in 95 % siginificant level but they are in 90 % significant level which is acceptable.

## SELECT MODELS

In order to select which model is the “best” we will test it against a validation(evaluation) set. We will examine the difference between the predicted and actual values.