Moneyball

David Moste, Vanita Thompson, Sadia Perveen

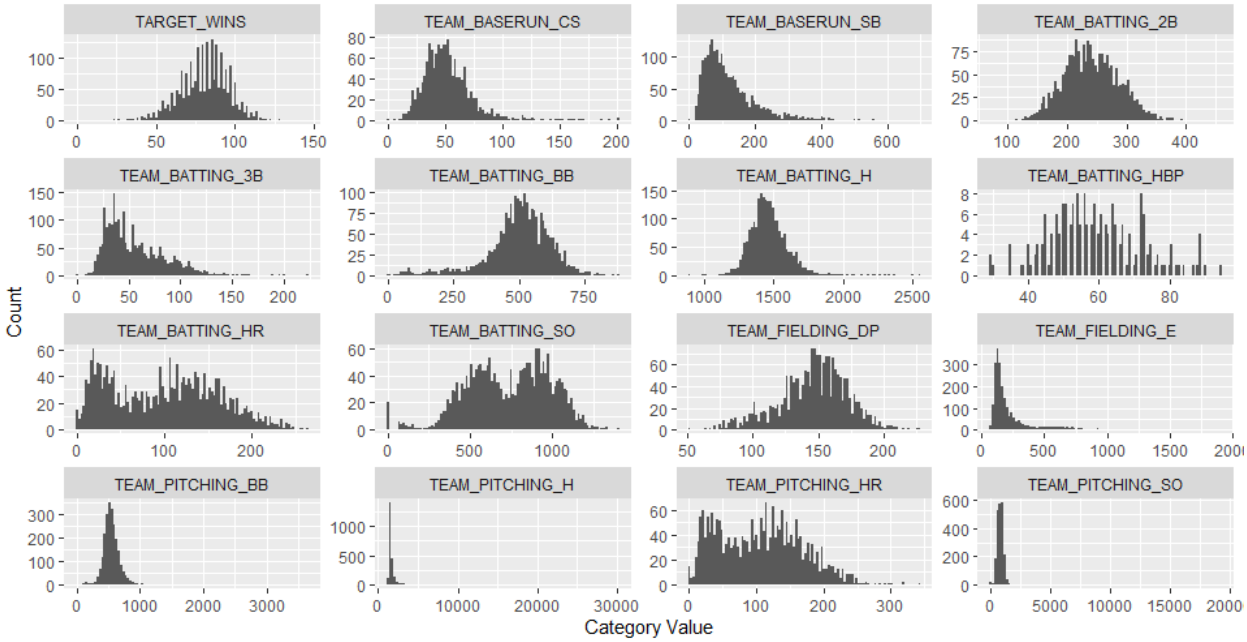
3/1/2021

Data Exploration

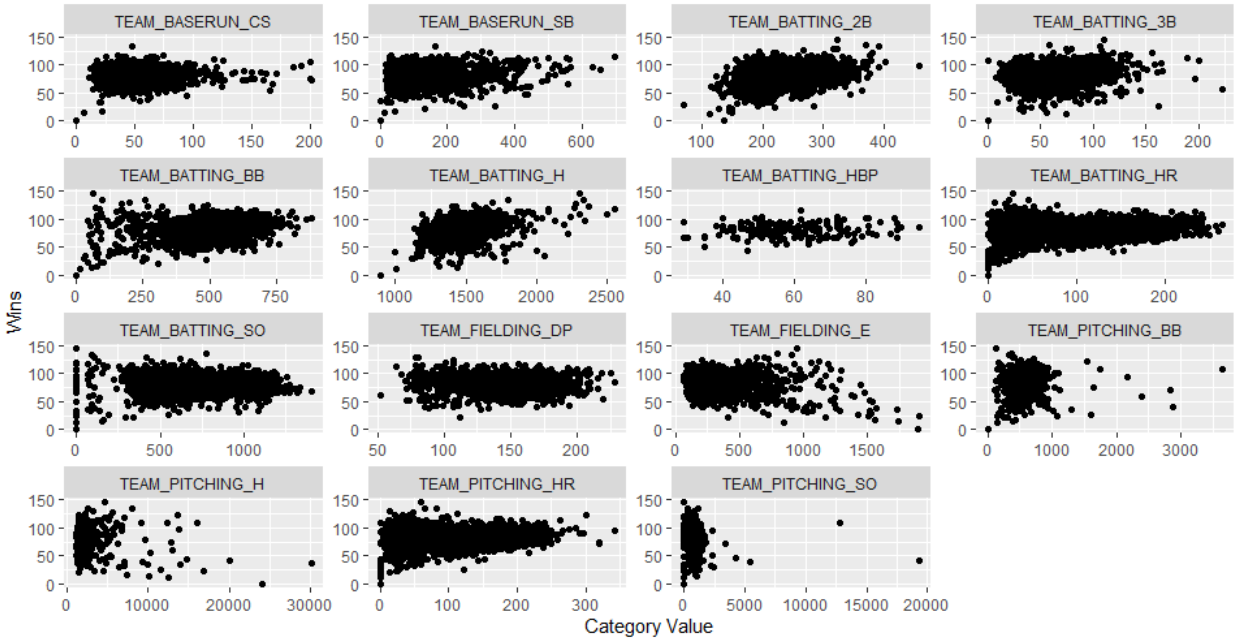
After grabbing the data, I first checked out a summary of the data to see the predictor variables provided along with their summary statistics. This also allowed me to see which predictor variables contained missing data. This summary data can be seen in the table below.

Predictor	Min	Median	Mean	Max	NAs
FIELDING_DP	52	149	146.4	228	286
FIELDING_E	65	159	246.5	1898	0
PITCHING_SO	0	813.5	817.7	19278	102
PITCHING_BB	0	536.5	553	3645	0
PITCHING_HR	0	107	105.7	343	0
PITCHING_H	1137	1518	1779	30132	0
BATTING_HBP	29	58	59.36	95	2085
BATTING_SO	0	750	735.6	1399	102
BATTING_BB	0	512	501.6	878	0
BATTING_HR	0	102	99.61	264	0
BATTING_3B	0	47	55.25	223	0
BATTING_2B	69	238	241.2	458	0
BATTING_H	891	1454	1469	2554	0
BASERUN_CS	0	49	52.8	201	772
BASERUN_SB	0	101	124.8	697	131

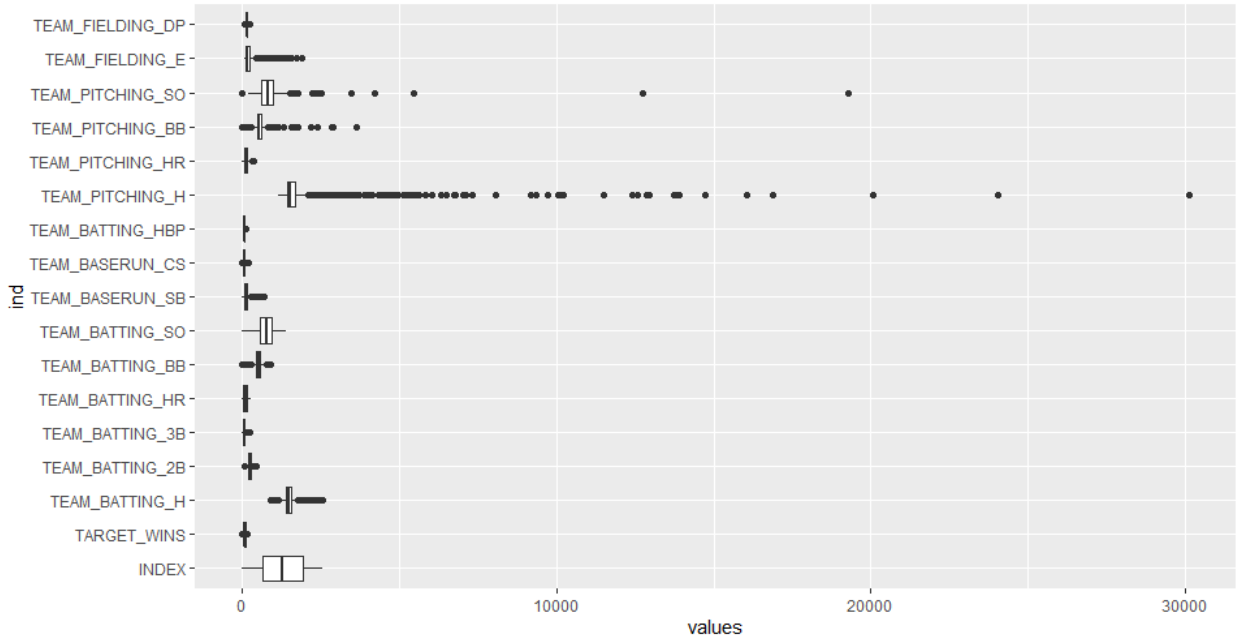
I then created three plots: a small multiples histogram, a small multiples scatterplot, and a boxplot.



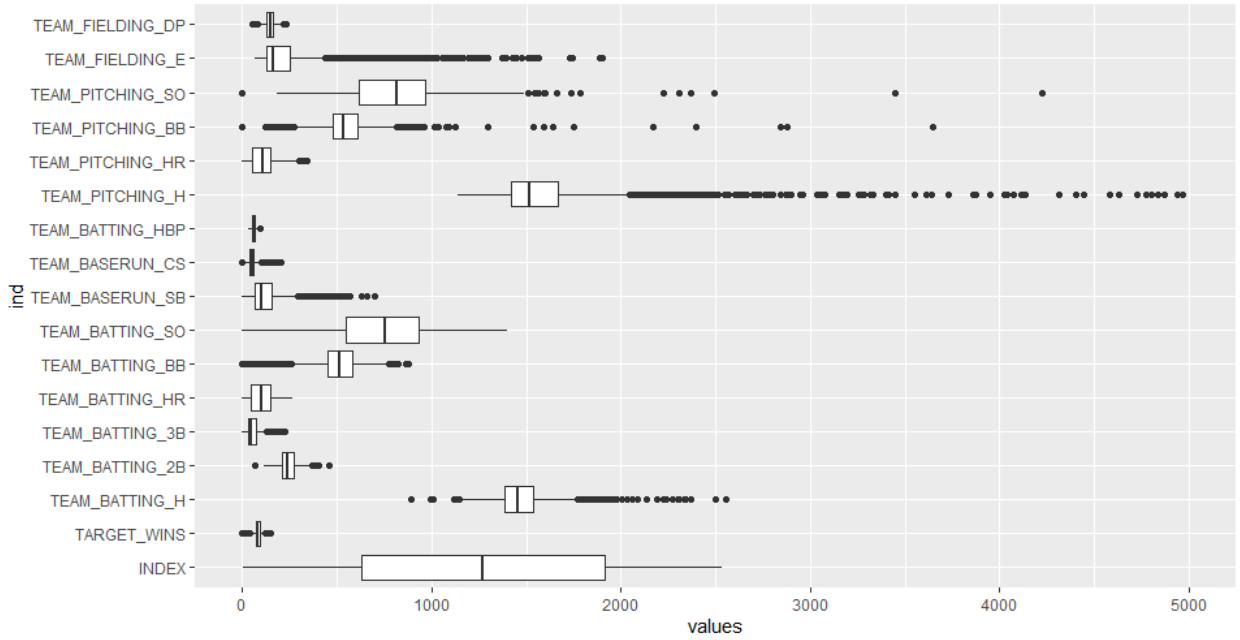
The purpose of the histogram was to get a sense of the normality of each variable. Upon looking at the histogram, it was easy to see that TEAM_BASERUN_CS, TEAM_BASERUN_SB, TEAM_BATTING_3B, TEAM_BATTING_HR, TEAM_FIELDING_E, and TEAM_PITCHING_HR were right skewed and would need to be transformed.



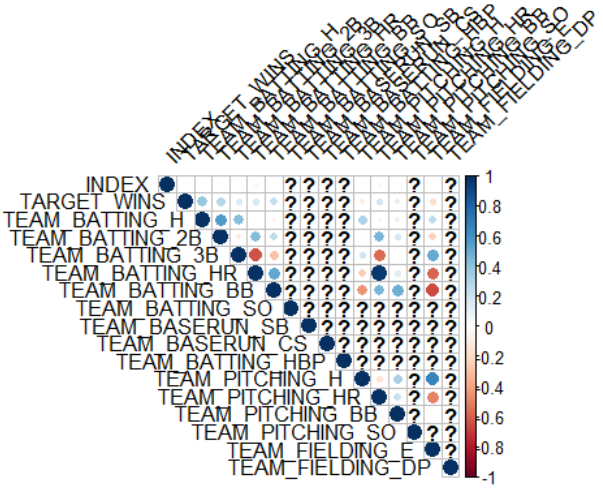
The purpose of this scatterplot was to get a sense of the relationship between each variable and TARGET_WINS. From this, you can see that no predictors have a strong negative relationship to TARGET_WINS, but TEAM_BATTING_H does seem to have a clear positive correlation.



The purpose of the boxplot was to see the data in another light and to get a sense of where there were outliers. It was easy to see at this point that TEAM_PITCHING_H contained a bunch of outliers at the top of the range.



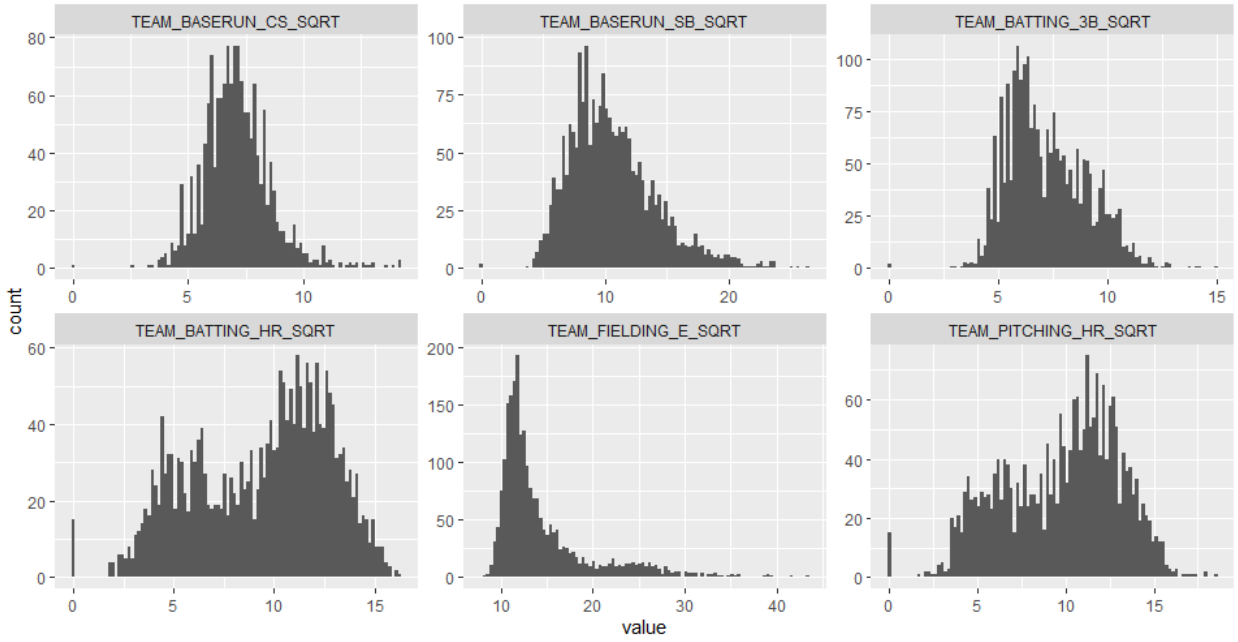
I then zoomed in on the boxplot to get a better sense of outliers in the other predictors.



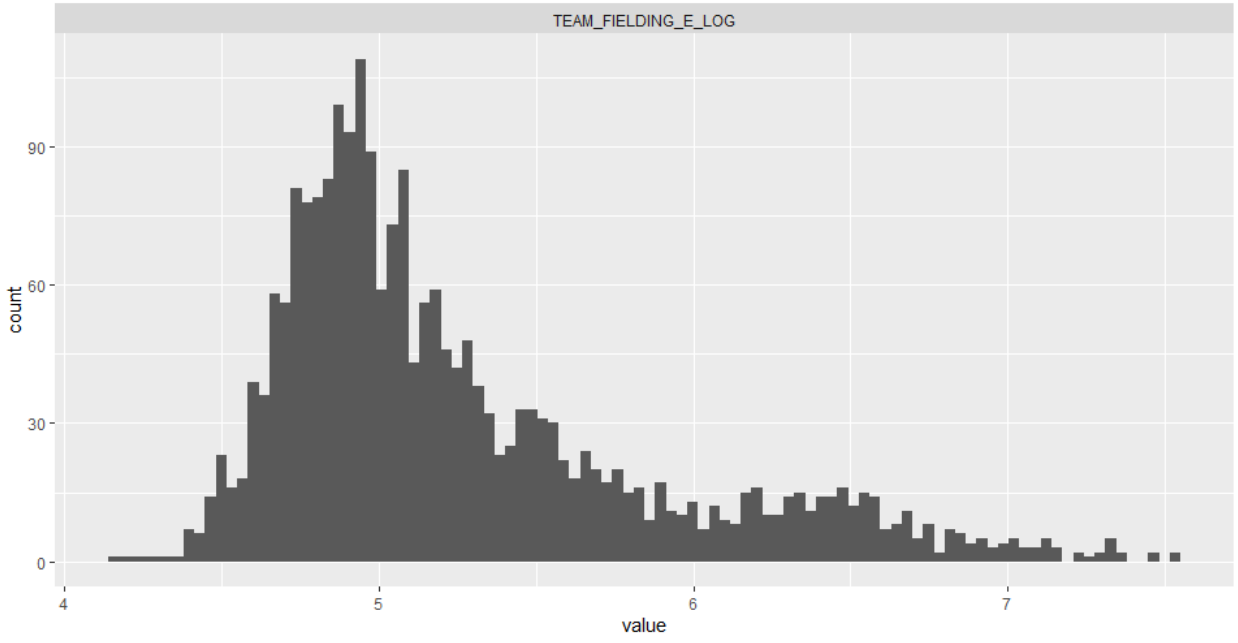
Finally, I created a correlation plot to show how different predictors are related to the target as well as each other. From this plot, it's easy to see that wins is most positively correlated to TEAM_BATTING_H and most negatively correlated to TEAM_FIELDING_E. As expected, other batting categories seem to have positive correlations as well. It is interesting to note that TEAM_PITCHING_HR has a positive correlation too, which is certainly not expected. Some other information that comes out of this visual is a strong correlation between TEAM_BATTING_HR and TEAM_PITCHING_HR and between TEAM_PITCHING_HR and TEAM_FIELDING_E as well as a strong negative correlation between TEAM_BATTING_BB and TEAM_FIELDING_E and between TEAM_FIELDING_E and TEAM_BATTING_HR.

Data Preparation

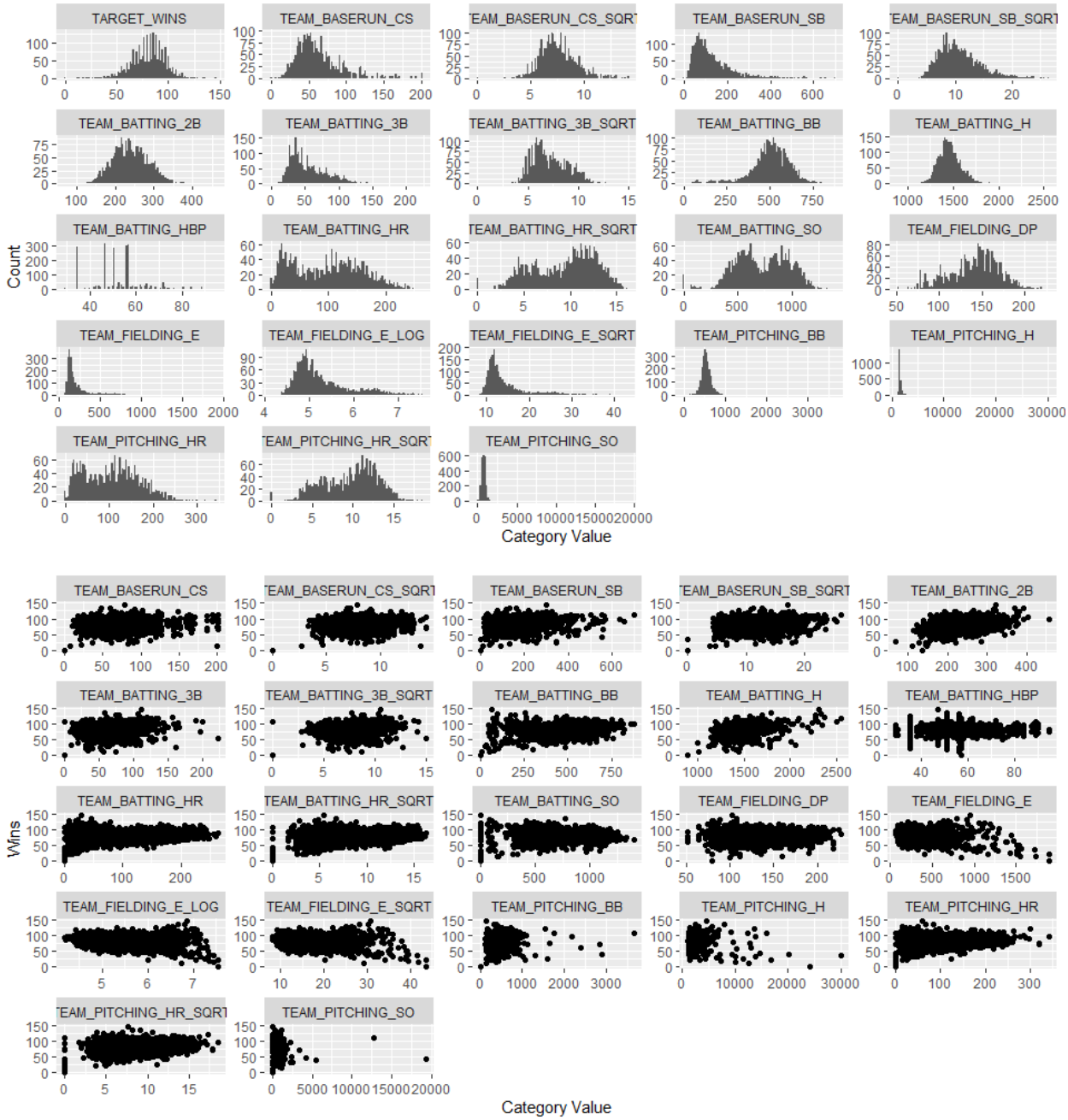
To start data preparation, I performed a few transformations. I did a square root transformation on each of the following variables to correct for their right skew : TEAM_BASERUN_CS, TEAM_BASERUN_SB, TEAM_BATTING_3B, TEAM_BATTING_HR, TEAM_FIELDING_E, and TEAM_PITCHING_HR. I ultimately chose to use a square root transformation instead of a log transformation because many of the variables had large portions of their data with values of 0. This makes log transformations a little bit less useable since you end up with -Inf values.



I then viewed a histogram of all the transformed predictors that I created. The histogram showed a clear bimodal distribution for TEAM_BATTING_HR and TEAM_PITCHING_HR. It also showed that TEAM_FILEDING_E was still highly right skewed. Due to this, I decided to take a log transform of TEAM_FIELDING_E to check if that would correct the skew. As can be seen below, this log transformation helped, but was not perfect.



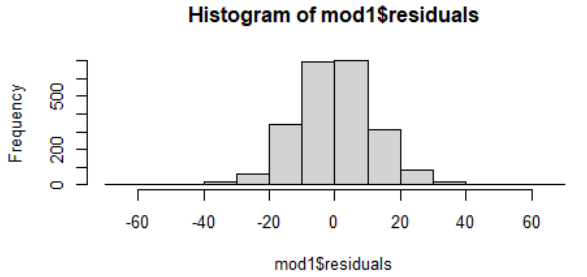
Next, I used the MICE package to impute missing values. I used MICE to implement multiple imputations using predictive mean matching method. After imputing missing values, I created two new plots: a histogram to view normality and a scatterplot to see outliers and correlation.

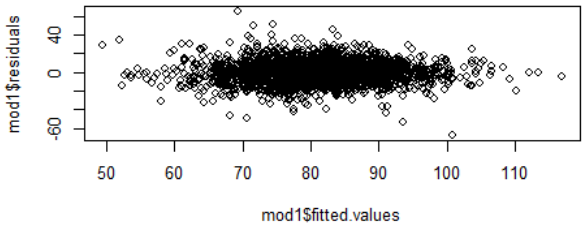


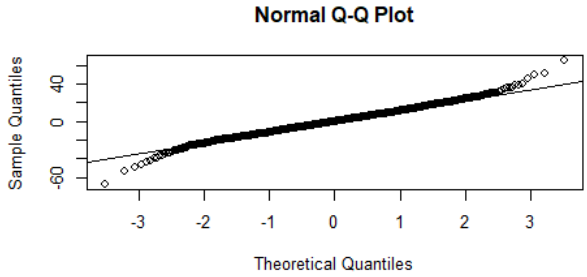
Finally, I created a new predictor, TEAM_BATTING_OB, which was meant to show how often a team got on base and I filtered a few predictors to remove extreme outliers that appeared to have some leverage.

Build Models

The first model I built was simply every variable in the data (excluding variables where I had later taken a transformation).

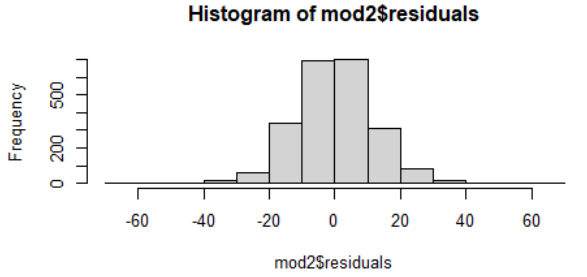


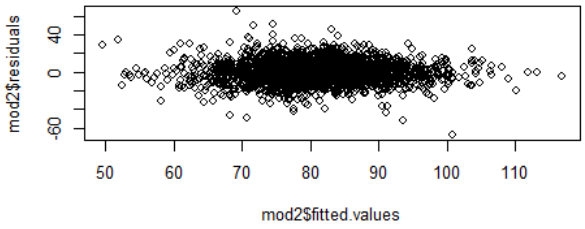


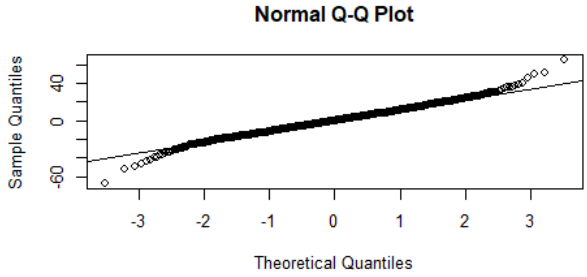


Oddly, this first model found that both TEAM_BATTING_2B and TEAM_FIELDING_DP have a negative coefficient, which suggests increasing them would decrease TARGET_WINS. On the opposite side, TEAM_BASERUN_CS_SQRT and TEAM_PITCHING_H have a positive coefficient, suggesting that they increase TARGET_WINS.

The second model I built was based off the first model, except that I iteratively removed the predictor with the highest p-value until the r-squared value was no longer increasing.

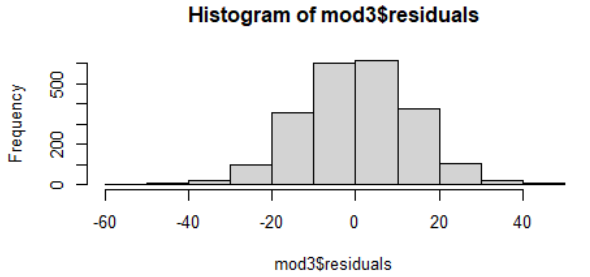


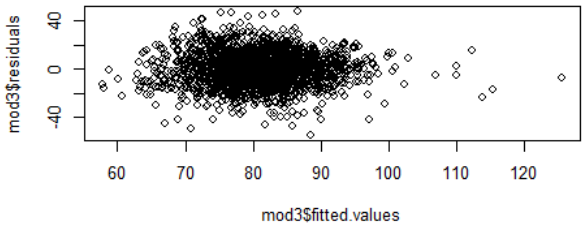


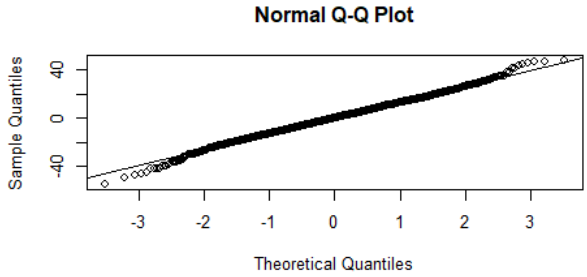


For the second model, TEAM_BATTING_2B still has a negative coefficient and TEAM_PITCHING_H still has a positive coefficient, both of which don't make a ton of immediate sense.

The final model I created was based off of the initial correlation plot I created, using the variables that had the strongest correlation (either positive or negative).







For the third model, all of the slopes make intuitive sense, but the overall fit is rather poor with an adjusted r-squared of 0.197.

Select Models

In this instance, since the only model that doesn't have any counter intuitive coefficients has a significantly lower adjusted r-squared value, I'm going to choose to use the second model, which had many of the same assumptions as the first, but had a better adjusted r-squared.

The MSE is about 145 for this model. The adjusted r-squared value for this model is 0.34 which means 34% of the variation in TARGET_WINS can be accounted for by the model. The F-statistic is 89.34 coupled with a tiny p-value means the independent variables in our model are statistically significant. Finally, as shown above, the histogram of the residuals shows normality, the residuals appear to be randomly distributed, and the qqplot shows a nearly straight line.

Code

```
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------------- tidyverse 1.3.0 --

## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
## v purrr   0.3.4

## -- Conflicts ------------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(mice)
```

```
## Warning: package 'mice' was built under R version 4.0.3
```

```
##
## Attaching package: 'mice'
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following objects are masked from 'package:base':
##
##     cbind, rbind
```

```r
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 4.0.3
```

```
## corrplot 0.84 loaded
```

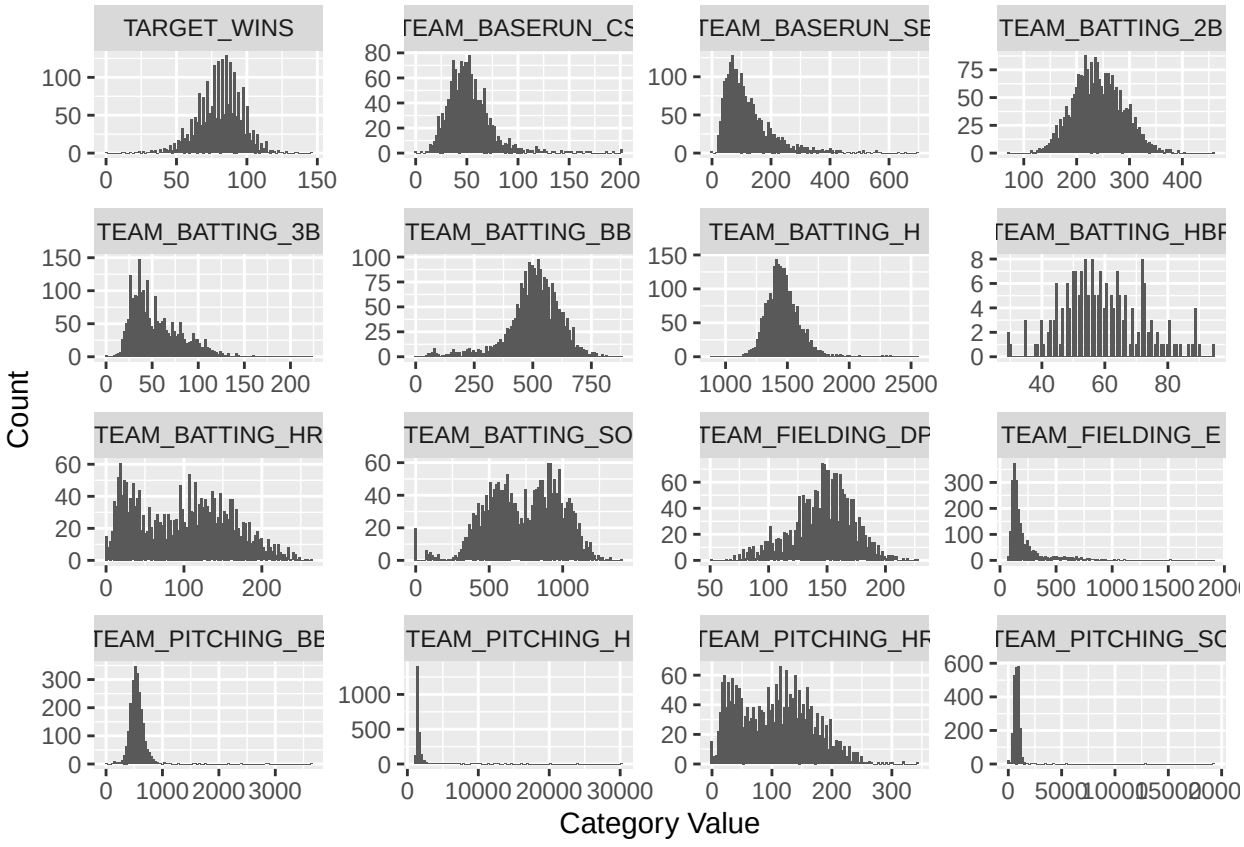
```r
# Read in the data
mlb <- read.csv("https://raw.githubusercontent.com/dmoste/DATA621/master/hw1/moneyball-training-data.cs
summary(mlb)
```

```
##      INDEX         TARGET_WINS     TEAM_BATTING_H TEAM_BATTING_2B
##  Min.   :   1.0   Min.   :  0.00   Min.   : 891   Min.   : 69.0
##  1st Qu.: 630.8   1st Qu.: 71.00   1st Qu.:1383   1st Qu.:208.0
##  Median :1270.5   Median : 82.00   Median :1454   Median :238.0
##  Mean   :1268.5   Mean   : 80.79   Mean   :1469   Mean   :241.2
##  3rd Qu.:1915.5   3rd Qu.: 92.00   3rd Qu.:1537   3rd Qu.:273.0
##  Max.   :2535.0   Max.   :146.00   Max.   :2554   Max.   :458.0
##
##  TEAM_BATTING_3B  TEAM_BATTING_HR  TEAM_BATTING_BB TEAM_BATTING_SO
##  Min.   :  0.00   Min.   :  0.00   Min.   :  0.0   Min.   :   0.0
##  1st Qu.: 34.00   1st Qu.: 42.00   1st Qu.:451.0   1st Qu.: 548.0
##  Median : 47.00   Median :102.00   Median :512.0   Median : 750.0
##  Mean   : 55.25   Mean   : 99.61   Mean   :501.6   Mean   : 735.6
##  3rd Qu.: 72.00   3rd Qu.:147.00   3rd Qu.:580.0   3rd Qu.: 930.0
##  Max.   :223.00   Max.   :264.00   Max.   :878.0   Max.   :1399.0
##                                                    NA's   :102
##  TEAM_BASERUN_SB TEAM_BASERUN_CS TEAM_BATTING_HBP TEAM_PITCHING_H
##  Min.   :  0.0   Min.   :  0.0   Min.   :29.00    Min.   : 1137
##  1st Qu.: 66.0   1st Qu.: 38.0   1st Qu.:50.50    1st Qu.: 1419
##  Median :101.0   Median : 49.0   Median :58.00    Median : 1518
##  Mean   :124.8   Mean   : 52.8   Mean   :59.36    Mean   : 1779
##  3rd Qu.:156.0   3rd Qu.: 62.0   3rd Qu.:67.00    3rd Qu.: 1682
##  Max.   :697.0   Max.   :201.0   Max.   :95.00    Max.   :30132
##  NA's   :131     NA's   :772     NA's   :2085
##  TEAM_PITCHING_HR TEAM_PITCHING_BB TEAM_PITCHING_SO  TEAM_FIELDING_E
```

```
##  Min.   :  0.0    Min.   :   0.0   Min.   :    0.0   Min.   :  65.0
##  1st Qu.: 50.0    1st Qu.: 476.0   1st Qu.:  615.0   1st Qu.: 127.0
##  Median :107.0    Median : 536.5   Median :  813.5   Median : 159.0
##  Mean   :105.7    Mean   : 553.0   Mean   :  817.7   Mean   : 246.5
##  3rd Qu.:150.0    3rd Qu.: 611.0   3rd Qu.:  968.0   3rd Qu.: 249.2
##  Max.   :343.0    Max.   :3645.0   Max.   :19278.0   Max.   :1898.0
##                                    NA's   :102
##  TEAM_FIELDING_DP
##  Min.   : 52.0
##  1st Qu.:131.0
##  Median :149.0
##  Mean   :146.4
##  3rd Qu.:164.0
##  Max.   :228.0
##  NA's   :286
```

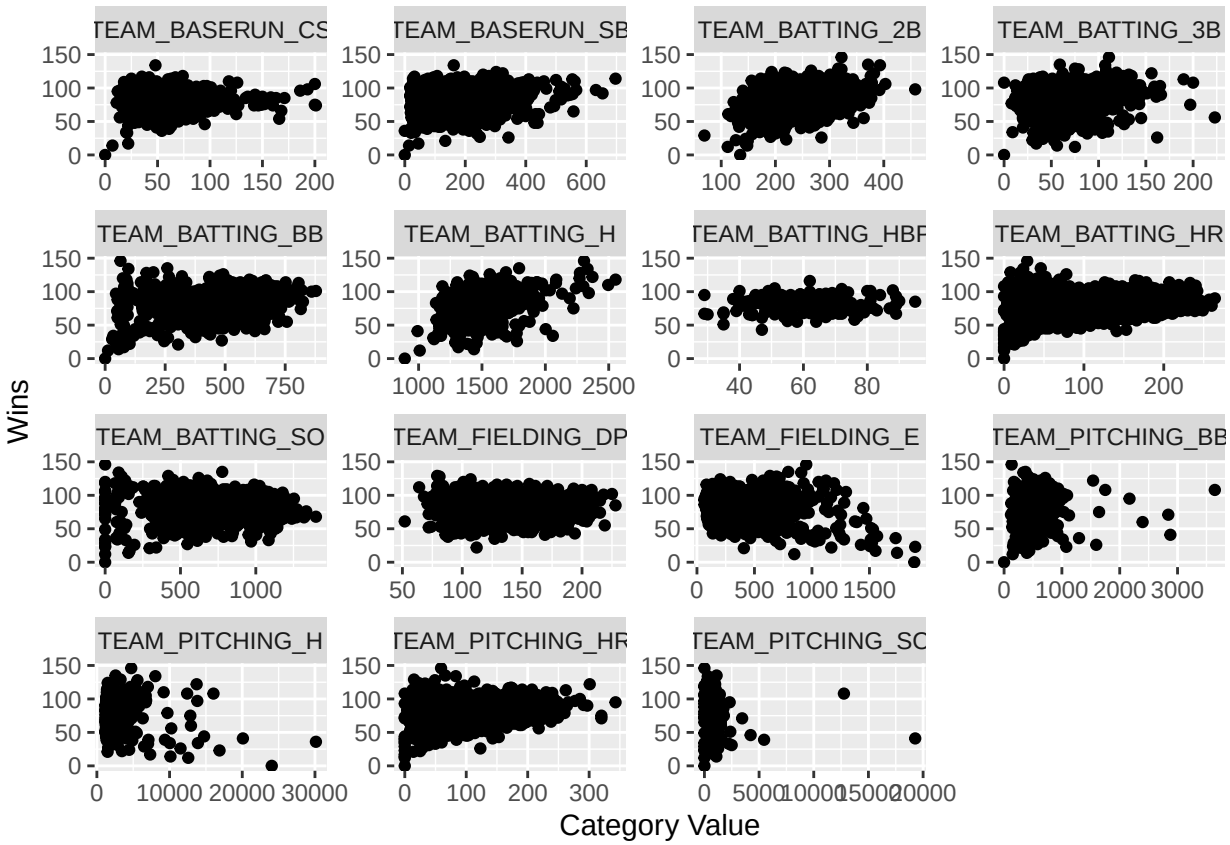
```
# Plot a histogram for each statistic to get a sense of the distributions
mlb %>%
  gather(-INDEX, key = "var", value = "value") %>%
  ggplot(aes(x = value)) +
  geom_histogram(bins = 100) +
  facet_wrap(~var, scales = "free") +
  xlab("Category Value") +
  ylab("Count")
```

```
## Warning: Removed 3478 rows containing non-finite values (stat_bin).
```



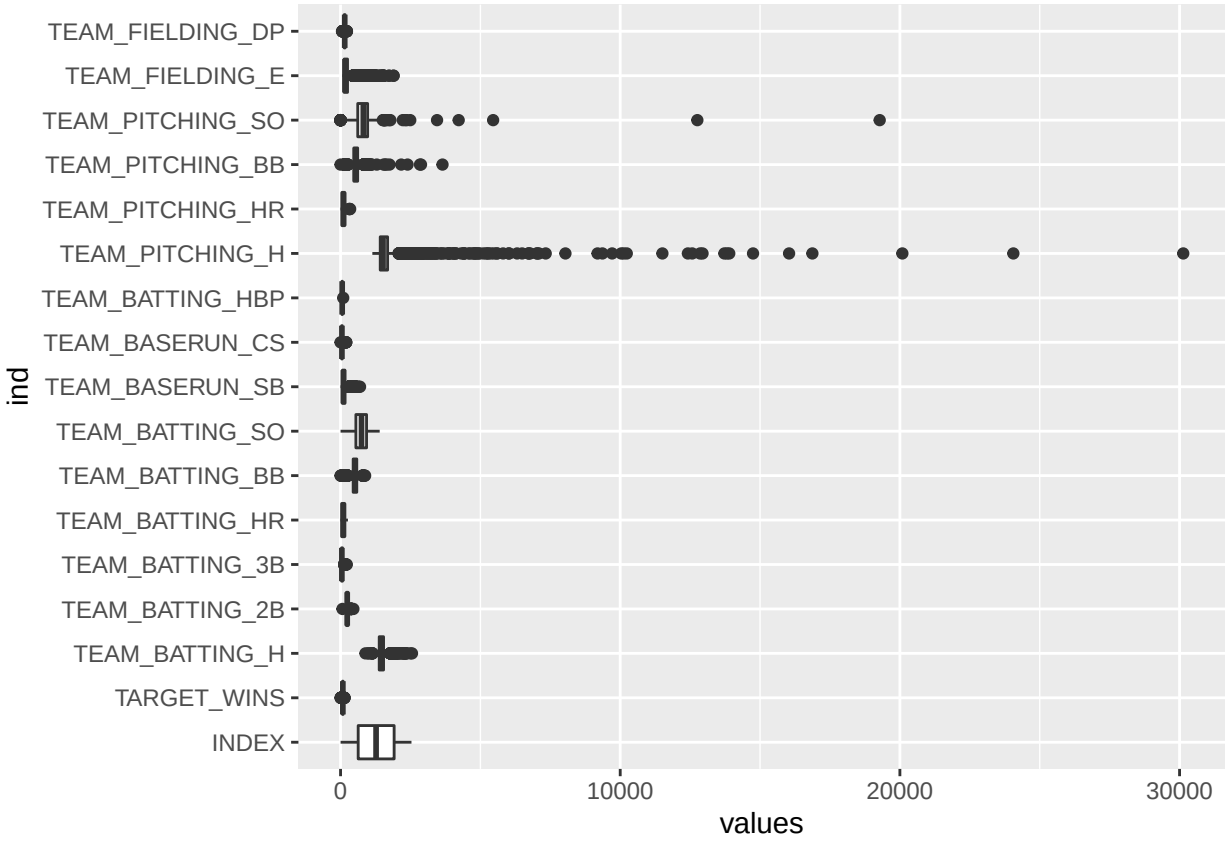
```
# Plot a scatterplot for each predictor to get a sense of relationships to TARGET_WINS
mlb %>%
  gather(-TARGET_WINS, -INDEX, key = "var", value = "value") %>%
  ggplot(aes(x = value, y = TARGET_WINS)) +
  geom_point() +
  facet_wrap(~var, scales = "free") +
  xlab("Category Value") +
  ylab("Wins")
```

```
## Warning: Removed 3478 rows containing missing values (geom_point).
```



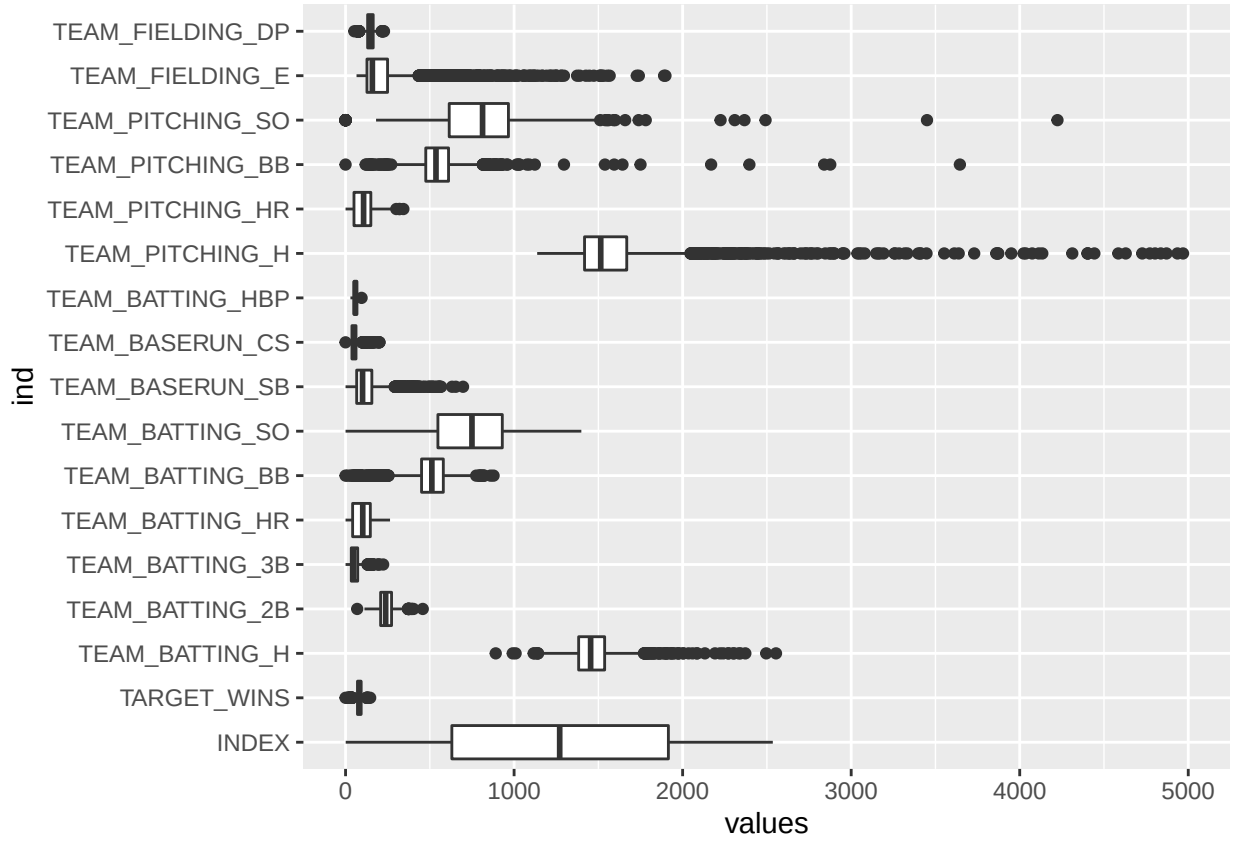

```
# Plot a boxplot for each statistic to get a sense of the outliers
ggplot(stack(mlb), aes(x = ind, y = values)) +
  geom_boxplot() +
  coord_flip()
```

```
## Warning: Removed 3478 rows containing non-finite values (stat_boxplot).
```

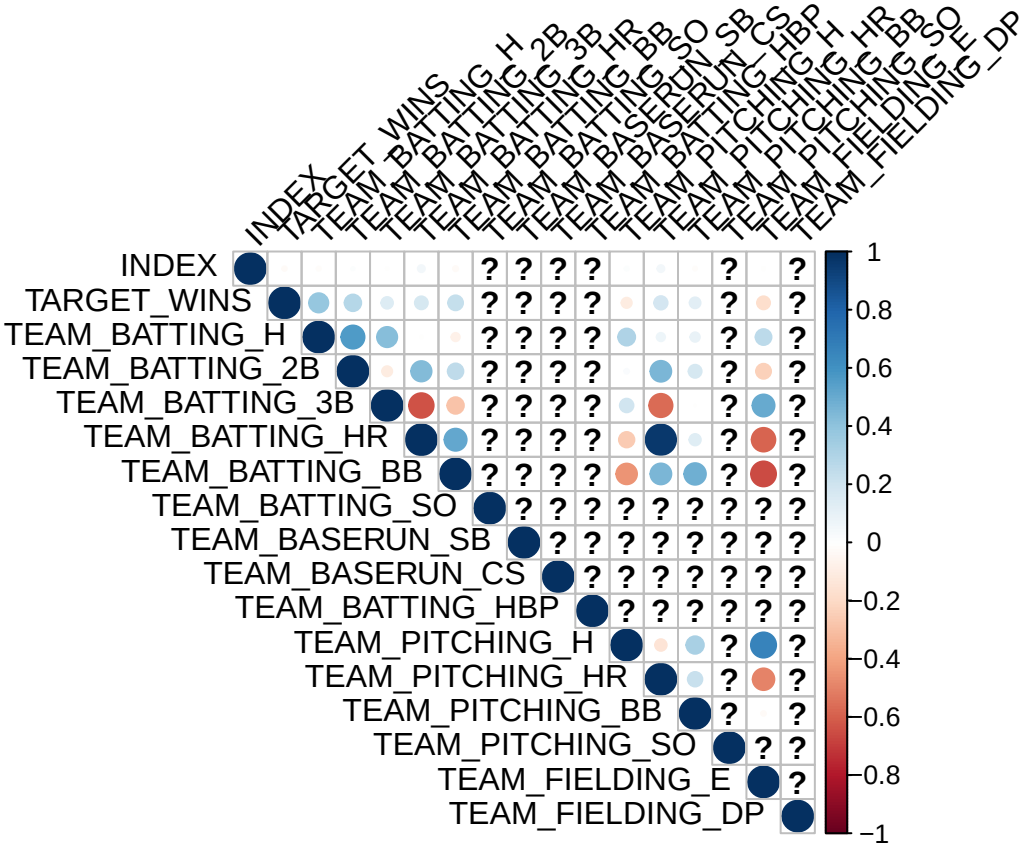



```
# Zoom in on the boxplot
ggplot(stack(mlb), aes(x = ind, y = values)) +
  geom_boxplot() +
  coord_flip() +
  ylim(0,5000)
```

```
## Warning: Removed 3522 rows containing non-finite values (stat_boxplot).
```

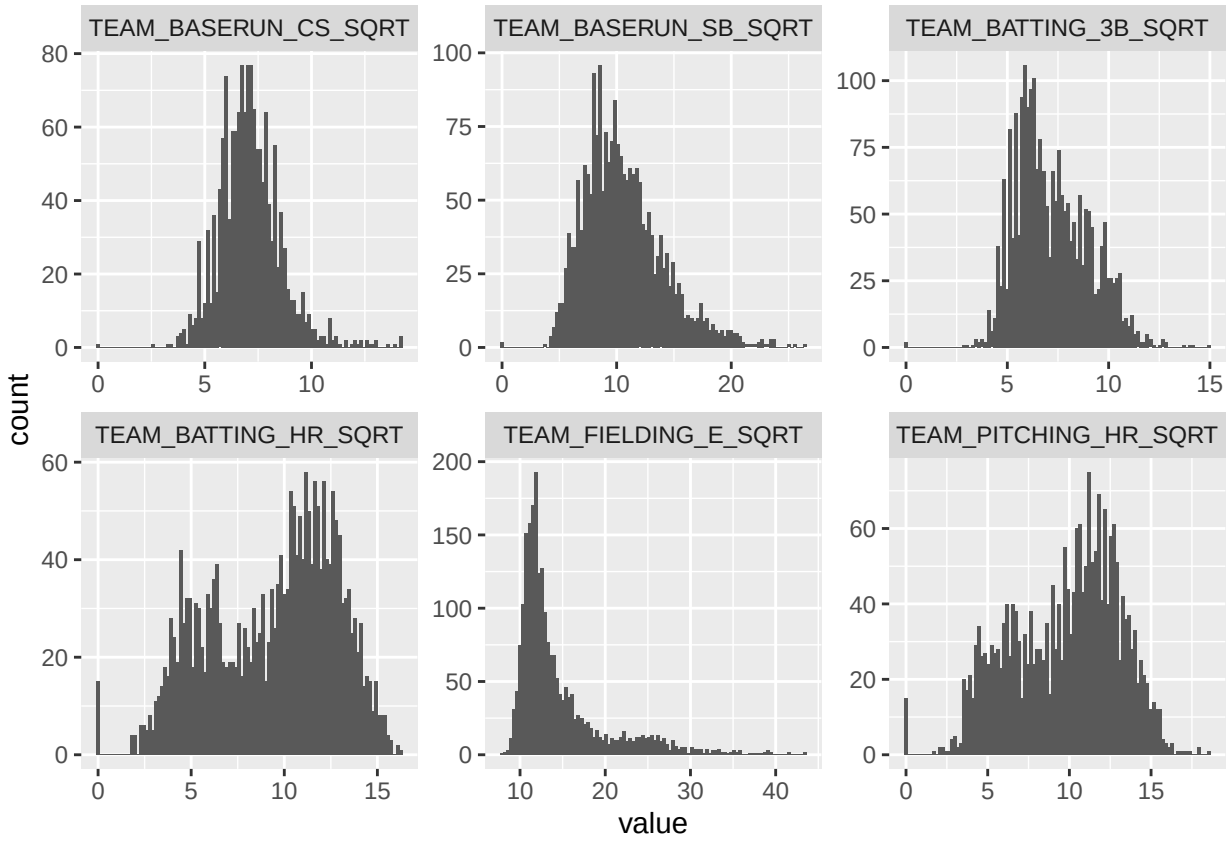



```r
# Show the correlation between the predictors and the target
corrplot(cor(mlb), method = "circle",
         type = "upper",
         tl.col = "black",
         tl.srt = 45)
```

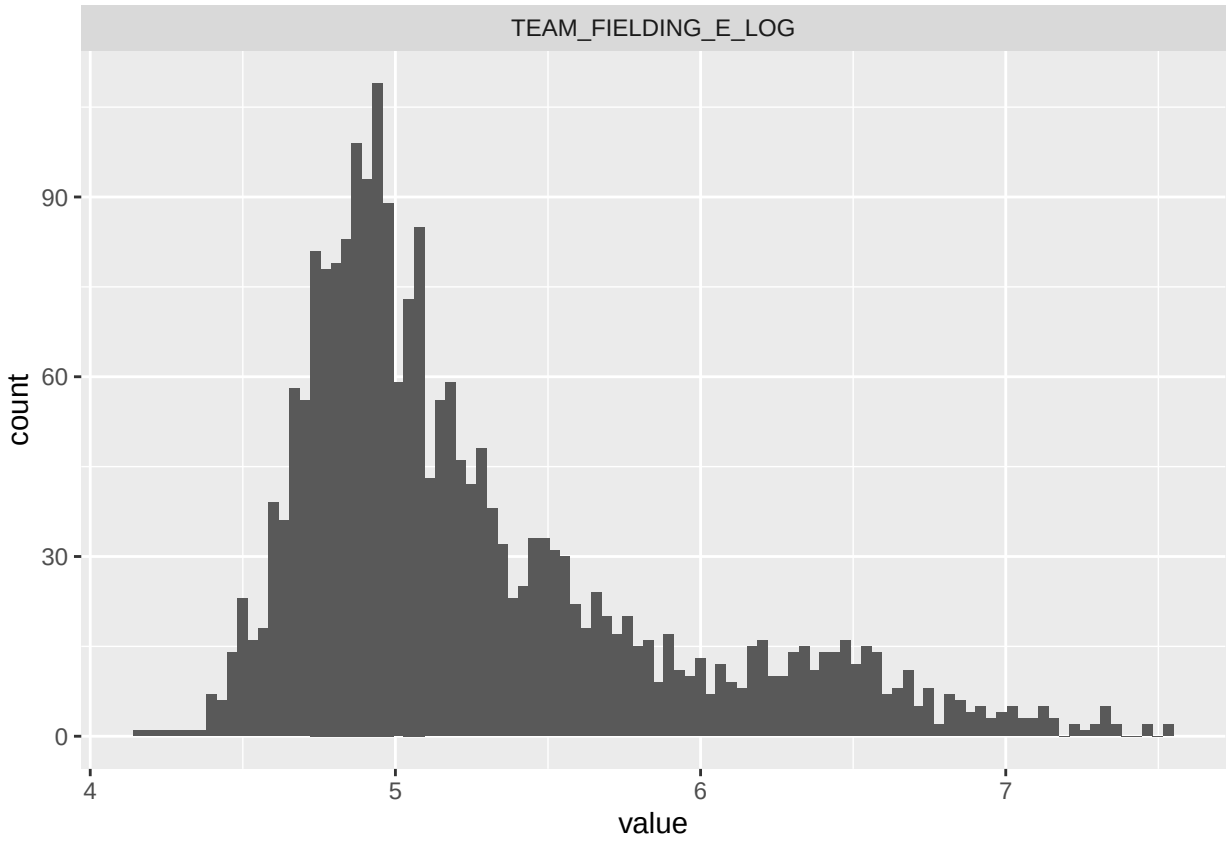



```
# Fix the skew of the variable by doing a square root transformation
mlb <- mlb %>%
  mutate(TEAM_BATTING_3B_SQRT = sqrt(TEAM_BATTING_3B)) %>%
  mutate(TEAM_BATTING_HR_SQRT = sqrt(TEAM_BATTING_HR)) %>%
  mutate(TEAM_FIELDING_E_SQRT = sqrt(TEAM_FIELDING_E)) %>%
  mutate(TEAM_BASERUN_CS_SQRT = sqrt(TEAM_BASERUN_CS)) %>%
  mutate(TEAM_BASERUN_SB_SQRT = sqrt(TEAM_BASERUN_SB)) %>%
  mutate(TEAM_PITCHING_HR_SQRT = sqrt(TEAM_PITCHING_HR))

ggplot(gather(mlb[18:23], cols, value), aes(x = value)) +
  geom_histogram(bins = 100) +
  facet_wrap(~cols, scales = "free")
```

```
## Warning: Removed 903 rows containing non-finite values (stat_bin).
```




```
mlb <- mlb %>%
  mutate(TEAM_FIELDING_E_LOG = log(TEAM_FIELDING_E))

ggplot(gather(mlb[24], cols, value), aes(x = value)) +
  geom_histogram(bins = 100) +
  facet_wrap(~cols, scales = "free")
```



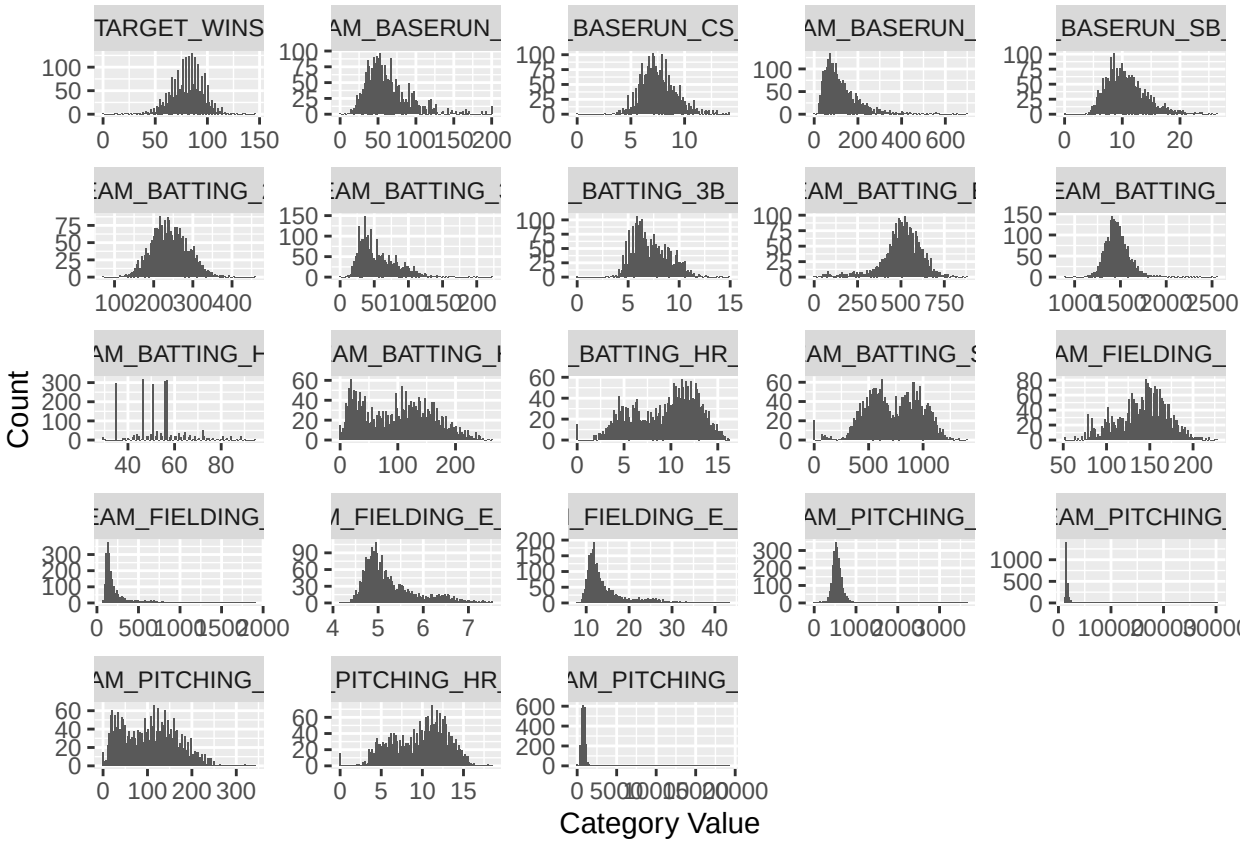

```
# Use the mice package (predictive mean matching method) to impute any missing
# values. Then normalize the data set and plot a series of histograms.
mlb_temp <- mice(mlb,
                 m = 5,
                 maxit = 10,
                 method = "pmm",
                 seed = 1234)
```

```
##
##  iter imp variable
##    1   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    1   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    1   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    1   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    1   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    2   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    2   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    2   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    2   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    2   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    3   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    3   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    3   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    3   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    3   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    4   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
```

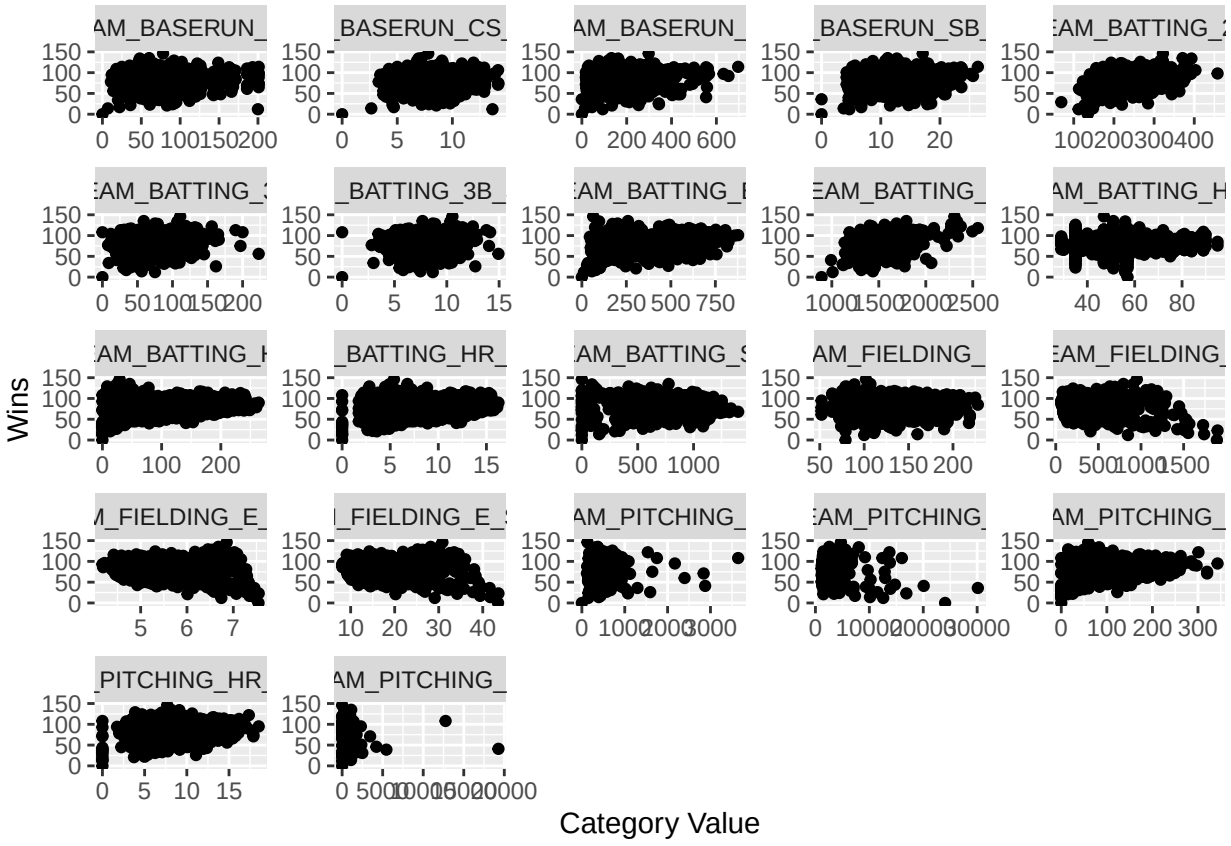
```
##   4   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   4   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   4   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   4   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   5   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   5   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   5   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   5   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   5   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   6   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   6   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   6   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   6   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   6   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   7   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   7   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   7   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   7   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   7   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   8   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   8   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   8   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   8   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   8   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   9   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   9   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   9   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   9   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##   9   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM_
##  10   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##  10   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##  10   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##  10   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##  10   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
```

```
## Warning: Number of logged events: 400
```

```r
complete_mlb <- complete(mlb_temp,1)

complete_mlb %>%
  gather(-INDEX, key = "var", value = "value") %>%
  ggplot(aes(x = value)) +
  geom_histogram(bins = 100) +
  facet_wrap(~var, scales = "free") +
  xlab("Category Value") +
  ylab("Count")
```




```
complete_mlb %>%
  gather(-TARGET_WINS, -INDEX, key = "var", value = "value") %>%
  ggplot(aes(x = value, y = TARGET_WINS)) +
  geom_point() +
  facet_wrap(~var, scales = "free") +
  xlab("Category Value") +
  ylab("Wins")
```




```
# Create a new variable: ON BASE
complete_mlb <- complete_mlb %>%
  mutate(TEAM_BATTING_OB = TEAM_BATTING_H + TEAM_BATTING_BB + TEAM_BATTING_HBP) %>%
  filter(TEAM_PITCHING_H < 7500) %>%
  filter(TEAM_PITCHING_BB < 1500) %>%
  filter(TEAM_PITCHING_SO < 2000) %>%
  filter(TEAM_PITCHING_SO > 0) %>%
  filter(TEAM_PITCHING_HR_SQRT > 0) %>%
  filter(TEAM_BATTING_HR_SQRT > 0) %>%
  filter(TEAM_BATTING_SO > 0)

# Model with all features included
mod1 <- lm(TARGET_WINS ~ TEAM_BASERUN_CS_SQRT + TEAM_BASERUN_SB_SQRT
           + TEAM_BATTING_2B + TEAM_BATTING_3B_SQRT + TEAM_BATTING_BB + TEAM_BATTING_HR_SQRT
           + TEAM_BATTING_HBP + TEAM_BATTING_H + TEAM_BATTING_SO
           + TEAM_FIELDING_DP + TEAM_FIELDING_E_LOG
           + TEAM_PITCHING_BB + TEAM_PITCHING_H + TEAM_PITCHING_HR_SQRT + TEAM_PITCHING_SO
           + TEAM_BATTING_OB, data = complete_mlb)
summary(mod1)
```

```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BASERUN_CS_SQRT + TEAM_BASERUN_SB_SQRT +
##     TEAM_BATTING_2B + TEAM_BATTING_3B_SQRT + TEAM_BATTING_BB +
##     TEAM_BATTING_HR_SQRT + TEAM_BATTING_HBP + TEAM_BATTING_H +
##     TEAM_BATTING_SO + TEAM_FIELDING_DP + TEAM_FIELDING_E_LOG +
```

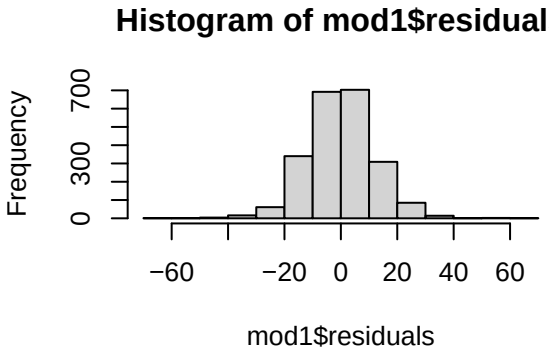
```
##     TEAM_PITCHING_BB + TEAM_PITCHING_H + TEAM_PITCHING_HR_SQRT +
##     TEAM_PITCHING_SO + TEAM_BATTING_OB, data = complete_mlb)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -67.784  -7.631  -0.004   7.700  65.674
##
## Coefficients: (1 not defined because of singularities)
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)           103.853668   8.384941  12.386  < 2e-16 ***
## TEAM_BASERUN_CS_SQRT    0.014783   0.246337   0.060 0.952153
## TEAM_BASERUN_SB_SQRT    1.077604   0.133691   8.060 1.23e-15 ***
## TEAM_BATTING_2B        -0.030680   0.009101  -3.371 0.000761 ***
## TEAM_BATTING_3B_SQRT    2.208252   0.278680   7.924 3.61e-15 ***
## TEAM_BATTING_BB         0.093414   0.015788   5.917 3.79e-09 ***
## TEAM_BATTING_HR_SQRT    5.412761   1.613913   3.354 0.000810 ***
## TEAM_BATTING_HBP        0.021143   0.024205   0.874 0.382480
## TEAM_BATTING_H          0.026502   0.004881   5.430 6.26e-08 ***
## TEAM_BATTING_SO        -0.068389   0.006303 -10.850  < 2e-16 ***
## TEAM_FIELDING_DP       -0.147335   0.012750 -11.556  < 2e-16 ***
## TEAM_FIELDING_E_LOG   -17.788285   1.095292 -16.241  < 2e-16 ***
## TEAM_PITCHING_BB       -0.062901   0.013889  -4.529 6.25e-06 ***
## TEAM_PITCHING_H         0.010008   0.001388   7.208 7.74e-13 ***
## TEAM_PITCHING_HR_SQRT  -3.999254   1.532972  -2.609 0.009147 **
## TEAM_PITCHING_SO        0.050284   0.005274   9.535  < 2e-16 ***
## TEAM_BATTING_OB               NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.09 on 2214 degrees of freedom
## Multiple R-squared:  0.3439, Adjusted R-squared:  0.3394
## F-statistic: 77.36 on 15 and 2214 DF,  p-value: < 2.2e-16
```

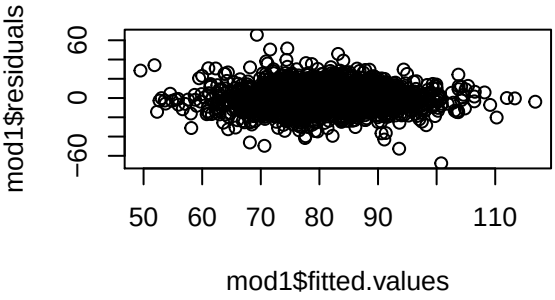
```
par(mfrow=c(2,2))
hist(mod1$residuals)
plot(mod1$residuals ~ mod1$fitted.values)
qqnorm(mod1$residuals)
qqline(mod1$residuals)

# Model based on removing high p-value predictors until best r-squared is reached
mod2 <- lm(TARGET_WINS ~ TEAM_BASERUN_SB_SQRT
           + TEAM_BATTING_2B + TEAM_BATTING_3B_SQRT + TEAM_BATTING_BB + TEAM_BATTING_HR_SQRT
           + TEAM_BATTING_SO
           + TEAM_FIELDING_DP + TEAM_FIELDING_E_LOG
           + TEAM_PITCHING_BB + TEAM_PITCHING_H + TEAM_PITCHING_HR_SQRT + TEAM_PITCHING_SO
           + TEAM_BATTING_OB, data = complete_mlb)
summary(mod2)
```

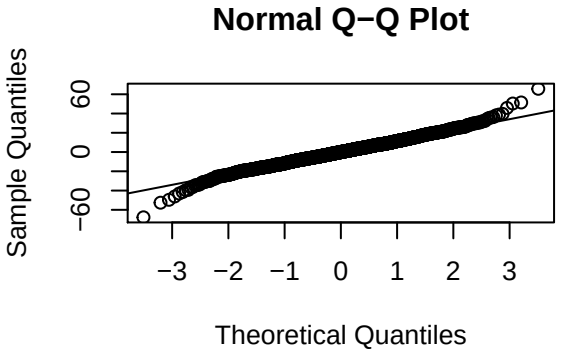
```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BASERUN_SB_SQRT + TEAM_BATTING_2B +
##     TEAM_BATTING_3B_SQRT + TEAM_BATTING_BB + TEAM_BATTING_HR_SQRT +
##     TEAM_BATTING_SO + TEAM_FIELDING_DP + TEAM_FIELDING_E_LOG +
##     TEAM_PITCHING_BB + TEAM_PITCHING_H + TEAM_PITCHING_HR_SQRT +
```

```
##     TEAM_PITCHING_SO + TEAM_BATTING_OB, data = complete_mlb)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -67.873  -7.652   0.000   7.721  65.742
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)           103.869073   8.007890  12.971  < 2e-16 ***
## TEAM_BASERUN_SB_SQRT    1.081756   0.109966   9.837  < 2e-16 ***
## TEAM_BATTING_2B        -0.030425   0.008999  -3.381 0.000735 ***
## TEAM_BATTING_3B_SQRT    2.218894   0.274272   8.090 9.71e-16 ***
## TEAM_BATTING_BB         0.067091   0.016422   4.085 4.56e-05 ***
## TEAM_BATTING_HR_SQRT    5.444632   1.598869   3.405 0.000673 ***
## TEAM_BATTING_SO        -0.068479   0.006285 -10.896  < 2e-16 ***
## TEAM_FIELDING_DP       -0.147139   0.012703 -11.583  < 2e-16 ***
## TEAM_FIELDING_E_LOG   -17.784976   1.080573 -16.459  < 2e-16 ***
## TEAM_PITCHING_BB       -0.062787   0.013870  -4.527 6.31e-06 ***
## TEAM_PITCHING_H         0.010042   0.001370   7.333 3.15e-13 ***
## TEAM_PITCHING_HR_SQRT  -4.030094   1.521833  -2.648 0.008150 **
## TEAM_PITCHING_SO        0.050260   0.005267   9.543  < 2e-16 ***
## TEAM_BATTING_OB         0.026237   0.004707   5.574 2.79e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.09 on 2216 degrees of freedom
## Multiple R-squared:  0.3439, Adjusted R-squared:   0.34
## F-statistic: 89.34 on 13 and 2216 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,2))
```





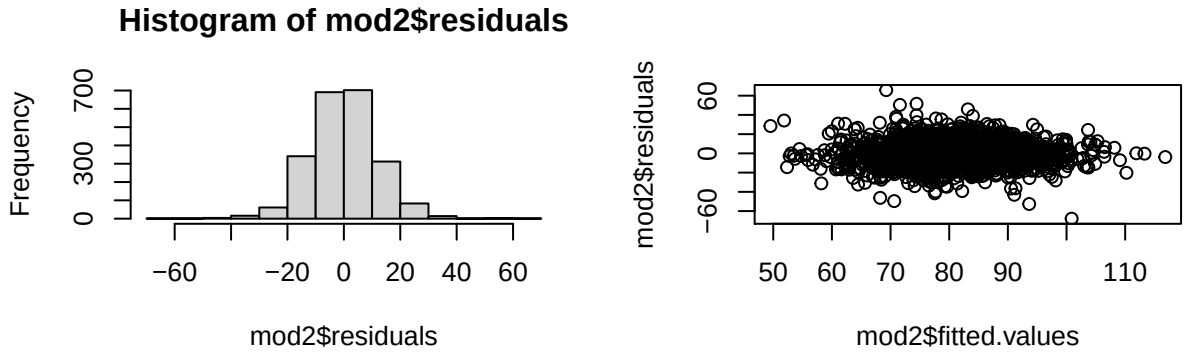


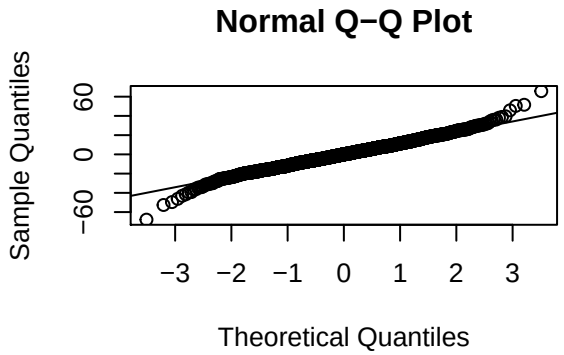

```
hist(mod2$residuals)
plot(mod2$residuals ~ mod2$fitted.values)
qqnorm(mod2$residuals)
qqline(mod2$residuals)

# Model based on features with high correlation to TARGET_WINS
mod3 <- lm(TARGET_WINS ~ TEAM_BATTING_OB + TEAM_BATTING_H
           + TEAM_FIELDING_E_LOG + TEAM_PITCHING_H, data = complete_mlb)
summary(mod3)
```

```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_OB + TEAM_BATTING_H +
##     TEAM_FIELDING_E_LOG + TEAM_PITCHING_H, data = complete_mlb)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -55.513  -8.887   0.318   8.838  48.465
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         21.251116   5.203205   4.084 4.58e-05 ***
## TEAM_BATTING_OB      0.022323   0.003243   6.884 7.52e-12 ***
## TEAM_BATTING_H       0.019011   0.004986   3.813 0.000141 ***
## TEAM_FIELDING_E_LOG -3.295462   0.673465  -4.893 1.06e-06 ***
```

```
## TEAM_PITCHING_H      0.002435   0.000950   2.563 0.010447 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.35 on 2225 degrees of freedom
## Multiple R-squared:  0.197,  Adjusted R-squared:  0.1956
## F-statistic: 136.5 on 4 and 2225 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,2))
```






```
hist(mod3$residuals)
plot(mod3$residuals ~ mod3$fitted.values)
qqnorm(mod3$residuals)
qqline(mod3$residuals)

# Predicting on evaluation data using model 2
eval <- read.csv("https://raw.githubusercontent.com/dmoste/DATA621/master/hw1/moneyball-evaluation-data

eval <- eval %>%
  mutate(TEAM_BATTING_3B_SQRT = sqrt(TEAM_BATTING_3B)) %>%
  mutate(TEAM_BATTING_HR_SQRT = sqrt(TEAM_BATTING_HR)) %>%
  mutate(TEAM_FIELDING_E_LOG = log(TEAM_FIELDING_E)) %>%
  mutate(TEAM_BASERUN_CS_SQRT = sqrt(TEAM_BASERUN_CS)) %>%
  mutate(TEAM_BASERUN_SB_SQRT = sqrt(TEAM_BASERUN_SB)) %>%
  mutate(TEAM_PITCHING_HR_SQRT = sqrt(TEAM_PITCHING_HR))
```

```
eval_temp <- mice(eval,
                  m = 5,
                  maxit = 10,
                  method = "pmm",
                  seed = 1234)
```

```
##
##  iter imp variable
##    1   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    1   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    1   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    1   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    1   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    2   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    2   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    2   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    2   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    2   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    3   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    3   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    3   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    3   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    3   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    4   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    4   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    4   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    4   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    4   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    5   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    5   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    5   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    5   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    5   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    6   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    6   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    6   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    6   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    6   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    7   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    7   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    7   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    7   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    7   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    8   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    8   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    8   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    8   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    8   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    9   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    9   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    9   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    9   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##    9   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
```

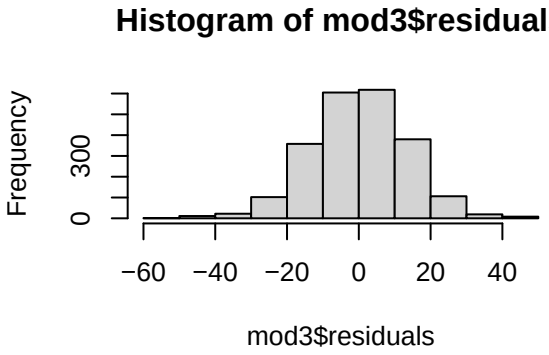
```
##   10   1  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##   10   2  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##   10   3  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##   10   4  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
##   10   5  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS  TEAM_BATTING_HBP  TEAM_PITCHING_SO  TEAM
```

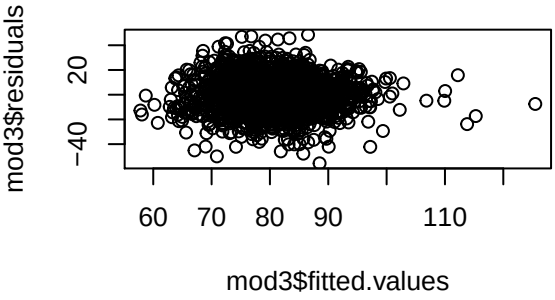
```
## Warning: Number of logged events: 405
```

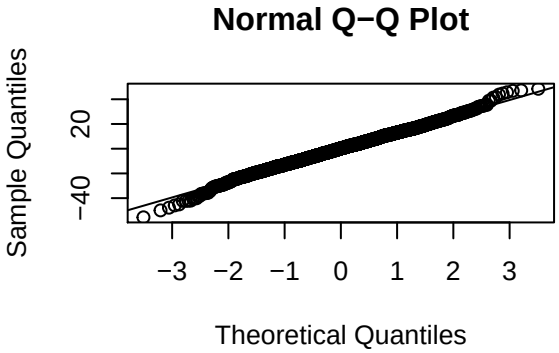
```
complete_eval <- complete(eval_temp,1)

complete_eval <- complete_eval %>%
  mutate(TEAM_BATTING_OB = TEAM_BATTING_H + TEAM_BATTING_BB + TEAM_BATTING_HBP) %>%
  filter(TEAM_PITCHING_H < 7500) %>%
  filter(TEAM_PITCHING_BB < 1500) %>%
  filter(TEAM_PITCHING_SO < 2000) %>%
  filter(TEAM_PITCHING_SO > 0) %>%
  filter(TEAM_PITCHING_HR_SQRT > 0) %>%
  filter(TEAM_BATTING_HR_SQRT > 0) %>%
  filter(TEAM_BATTING_SO > 0)

predict(mod2,complete_eval)
```

```
##        1        2        3        4        5        6        7        8
## 61.05058 65.40862 72.26636 84.15120 90.22513 68.54808 79.59470 71.09724
##        9       10       11       12       13       14       15       16
## 71.35738 71.40956 64.26482 81.24515 81.85096 79.08800 87.93104 74.17939
##       17       18       19       20       21       22       23       24
## 71.17725 76.61824 78.04764 87.30323 84.85916 82.58025 81.86401 67.19682
##       25       26       27       28       29       30       31       32
## 80.20004 86.68044 70.12486 83.47598 68.19544 91.49050 87.26062 84.52088
##       33       34       35       36       37       38       39       40
## 82.13953 79.92918 83.58573 76.56494 88.69839 81.50179 88.76895 80.81031
##       41       42       43       44       45       46       47       48
## 98.10622 106.88197 90.65467 91.57671 93.76122 75.45044 66.18836 79.14116
##       49       50       51       52       53       54       55       56
## 74.64636 83.37731 74.20361 75.86334 70.99021 80.63052 89.91527 74.28080
##       57       58       59       60       61       62       63       64
## 62.14898 77.44567 88.42108 80.97452 87.96039 81.84009 83.95839 97.94577
##       65       66       67       68       69       70       71       72
## 70.71094 74.78308 82.95516 89.56711 87.01648 71.53916 78.59868 89.61024
##       73       74       75       76       77       78       79       80
## 74.58358 78.54745 88.12403 82.61968 60.54596 76.46977 81.84473 86.07005
##       81       82       83       84       85       86       87       88
## 94.41097 71.50677 85.70241 78.01740 85.27186 87.10397 95.56252 90.24420
##       89       90       91       92       93       94       95       96
## 80.10476 69.32305 84.69420 87.38001 70.88188 92.03555 103.61585 88.22023
##       97       98       99      100      101      102      103      104
## 88.56956 78.87392 71.84992 83.08293 84.73919 72.37036 67.23077 54.07637
##      105      106      107      108      109      110      111      112
## 74.78838 90.00867 58.57340 88.74152 87.96389 95.56055 92.71210 80.69740
##      113      114      115      116      117      118      119      120
## 78.96322 85.18347 81.95571 70.57306 74.11059 95.85095 63.15170 73.81018
##      121      122      123      124      125      126      127      128
```

```
##  75.48836  68.71230  85.32363  88.15123  76.42848  93.40664  89.22381  83.29810
##       129       130       131       132       133       134       135       136
##  81.44061  73.18407  84.08980  93.29032  71.19795  76.57250  75.96058  93.72966
##       137       138       139       140       141       142       143       144
##  80.50792  58.56718  76.90421  92.02140  70.12983  75.17416  71.12289  75.31741
##       145       146       147       148       149       150       151       152
##  81.65218  78.75430  85.98703  81.91302  83.74949  62.52675  80.20501  66.96516
##       153       154       155       156       157       158       159       160
##  91.56293  69.13666  90.56294  68.61189 104.83151 110.95916  98.17556 104.55799
##       161       162       163       164       165       166       167       168
##  99.53608  94.53942  83.92324  84.59193  70.09413  79.62705  94.88515  91.04155
##       169       170       171       172       173       174       175       176
##  81.81993  95.34161  80.71752  76.67967  81.72883  67.23015  73.12035  80.12171
##       177       178       179       180       181       182       183       184
##  90.13580  87.75314  87.65468  85.89150  89.61380  83.33875  61.76211  55.37195
##       185       186       187       188       189       190       191       192
## 103.94202  61.27701  76.32364  71.58526  72.64872  72.45795  61.75573  74.94887
##       193       194       195       196       197       198       199       200
##  92.75305  82.42007  86.17507  72.18206  81.83104  78.50816  90.57293  81.94252
##       201       202       203       204       205       206       207       208
##  87.77161  80.01039  80.16328  78.93984  72.69002  97.02720  91.23246  84.80254
##       209       210       211       212       213       214       215       216
##  61.88303  66.73451  82.66636  78.27830  93.35274  77.30730  83.06044  76.29707
##       217       218       219       220       221       222       223       224
##  70.11069  77.93703  72.44045  97.20000  79.23794  79.69873  77.57326  86.16767
##       225       226       227       228       229       230       231       232
##  80.19797  81.94618  94.24220  81.23797  87.42346  80.21803  77.01091  78.23654
##       233       234       235       236       237       238       239       240
##  81.94544  94.68268  72.11268  88.46732  90.42693  84.47731  81.80645  58.52746
##       241       242       243       244       245       246       247       248
##  84.44943  79.64248  83.87861  73.10640  83.66294  79.60975  68.20665  92.48896
##       249       250       251       252       253
##  69.76667  78.11372  84.00121  82.72730  72.41195
```


Histogram of mod3$residuals
Frequency
0
300
−60
−40
−20
0
20
40
mod3$residuals


mod3$residuals
−40
20
60
70
80
90
110
mod3$fitted.values


Normal Q−Q Plot
Sample Quantiles
−40
20
−3
−2
−1
0
1
2
3
Theoretical Quantiles