Money Ball - Data 621 Project 1

Jack Russo, Javern Wilson, Joseph Simone, Anthony Munoz, Paul Perez

03-01-2020

Contents

Overview

In this homework assignment, we will explore, analyze and model a data set containing approximately 2200 records. This analysis attempts to predict the number of wins for the teams. Each record represents a professional baseball team from the years 1871 to 2006 inclusive. Each record has the performance of the team for the given year, with all of the statistics adjusted to match the performance of a 162 game season.

Part i. DATA EXPLORATION

Preview

Below is a preview of what the dataset contains.

```
## # A tibble: 10 x 17
##    INDEX TARGET_WINS TEAM_BATTING_H TEAM_BATTING_2B TEAM_BATTING_3B
##    <int>       <int>          <int>           <int>           <int>
##  1     1          39           1445             194              39
##  2     2          70           1339             219              22
##  3     3          86           1377             232              35
##  4     4          70           1387             209              38
##  5     5          82           1297             186              27
##  6     6          75           1279             200              36
##  7     7          80           1244             179              54
##  8     8          85           1273             171              37
##  9    11          86           1391             197              40
## 10    12          76           1271             213              18
## # ... with 12 more variables: TEAM_BATTING_HR <int>, TEAM_BATTING_BB <int>,
## #   TEAM_BATTING_SO <int>, TEAM_BASERUN_SB <int>, TEAM_BASERUN_CS <int>,
## #   TEAM_BATTING_HBP <int>, TEAM_PITCHING_H <int>, TEAM_PITCHING_HR <int>,
## #   TEAM_PITCHING_BB <int>, TEAM_PITCHING_SO <int>, TEAM_FIELDING_E <int>,
## #   TEAM_FIELDING_DP <int>
```

Structure of Data

```
## 'data.frame':    2276 obs. of  17 variables:
##  $ INDEX           : int  1 2 3 4 5 6 7 8 11 12 ...
##  $ TARGET_WINS     : int  39 70 86 70 82 75 80 85 86 76 ...
##  $ TEAM_BATTING_H  : int  1445 1339 1377 1387 1297 1279 1244 1273 1391 1271 ...
##  $ TEAM_BATTING_2B : int  194 219 232 209 186 200 179 171 197 213 ...
##  $ TEAM_BATTING_3B : int  39 22 35 38 27 36 54 37 40 18 ...
##  $ TEAM_BATTING_HR : int  13 190 137 96 102 92 122 115 114 96 ...
##  $ TEAM_BATTING_BB : int  143 685 602 451 472 443 525 456 447 441 ...
##  $ TEAM_BATTING_SO : int  842 1075 917 922 920 973 1062 1027 922 827 ...
##  $ TEAM_BASERUN_SB : int  NA 37 46 43 49 107 80 40 69 72 ...
##  $ TEAM_BASERUN_CS : int  NA 28 27 30 39 59 54 36 27 34 ...
##  $ TEAM_BATTING_HBP: int  NA NA NA NA NA NA NA NA NA NA ...
##  $ TEAM_PITCHING_H : int  9364 1347 1377 1396 1297 1279 1244 1281 1391 1271 ...
##  $ TEAM_PITCHING_HR: int  84 191 137 97 102 92 122 116 114 96 ...
##  $ TEAM_PITCHING_BB: int  927 689 602 454 472 443 525 459 447 441 ...
##  $ TEAM_PITCHING_SO: int  5456 1082 917 928 920 973 1062 1033 922 827 ...
##  $ TEAM_FIELDING_E : int  1011 193 175 164 138 123 136 112 127 131 ...
##  $ TEAM_FIELDING_DP: int  NA 155 153 156 168 149 186 136 169 159 ...
```

At first glance, we can see that `TEAM_BATTING_HBP` has a lot of missing data. Let's look at the summary to see if it reveals further information on the data.

Summary of Data

```
##      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
```

Based on the summary above we have our work cut out for us, especially when handling missing values. There are 6 variables with missing values:

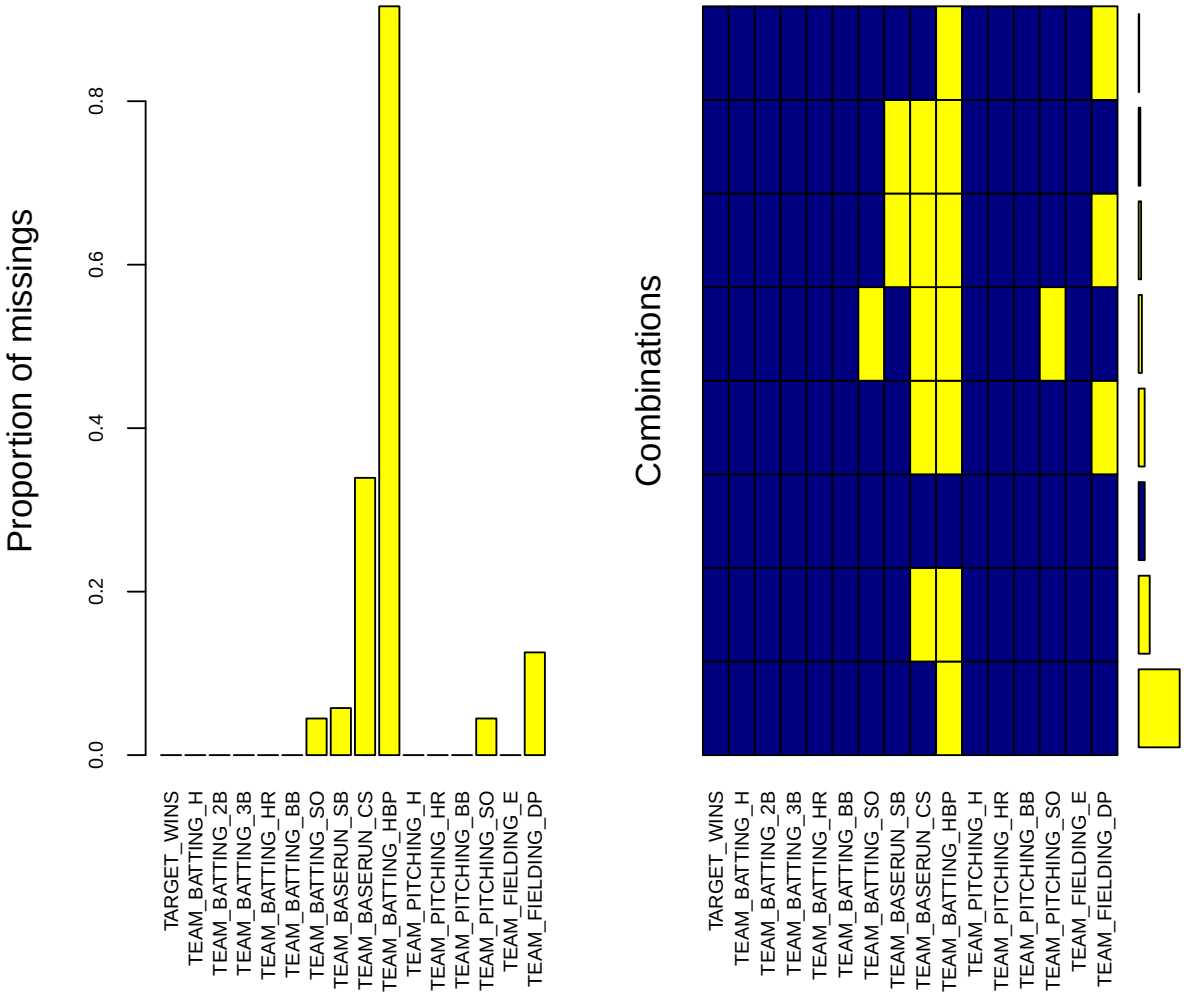
- **TEAM_BATTING_SO**: 102
- **TEAM_BASERUN_SB**: 131
- **TEAM_BASERUN_CS**: 772
- **TEAM_BATTING_HBP**: 2085
- **TEAM_PITCHING_SO**: 102
- **TEAM_FIELDING_DP**: 286

Some variables also have a minumum of 0. Whether or not there values affect our model outcome will be interesting to find out as we move forward.

```
## [1] 8.39
```

About only 8% of the data has complete rows.

On this next plot is a graphic representation of what variable has missing values. As mentioned earlier, the six variables that have missing values can be visually observed here. One of our goal when completing the project is knowing how to handle missing values effectively in order to reduce bias and produce useful and powerful models.



Further Descriptive Analytics

Here we look at more descriptive analytics on the raw dataset.

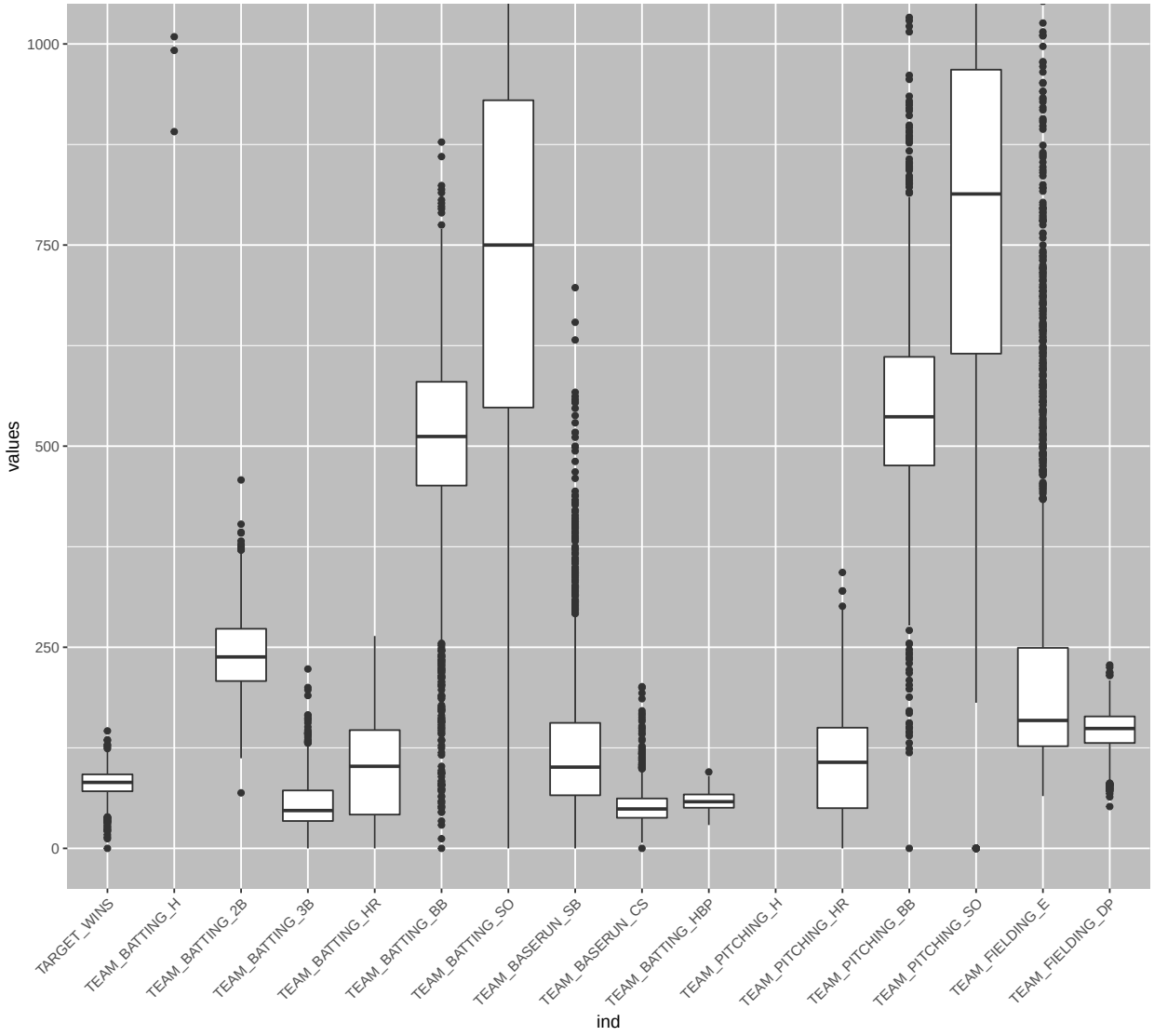
```
## # A tibble: 16 x 13
##     vars     n   mean     sd median trimmed   mad   min   max range   skew
##    <int> <dbl>  <dbl>  <dbl>  <dbl>   <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
##  1     1  2276   80.8   15.8     82    81.3  14.8     0   146   146 -0.399
##  2     2  2276 1469.   145.   1454   1459.  114.    891  2554  1663  1.57
##  3     3  2276  241.    46.8   238    240.   47.4    69   458   389  0.215
##  4     4  2276   55.2   27.9    47     52.2  23.7     0   223   223  1.11
##  5     5  2276   99.6   60.5   102     97.4  78.6     0   264   264  0.186
##  6     6  2276  502.   123.    512    512.   94.9     0   878   878 -1.03
##  7     7  2174  736.   249.    750    742.  285.      0  1399  1399 -0.298
##  8     8  2145  125.    87.8   101    111.   60.8     0   697   697  1.97
##  9     9  1504   52.8   23.0    49     50.4  17.8     0   201   201  1.98
## 10    10   191   59.4   13.0    58     58.9  11.9    29    95    66  0.319
## 11    11  2276 1779.  1407.   1518   1556.  175.   1137 30132 28995 10.3
## 12    12  2276  106.    61.3   107    103.   74.1     0   343   343  0.288
```

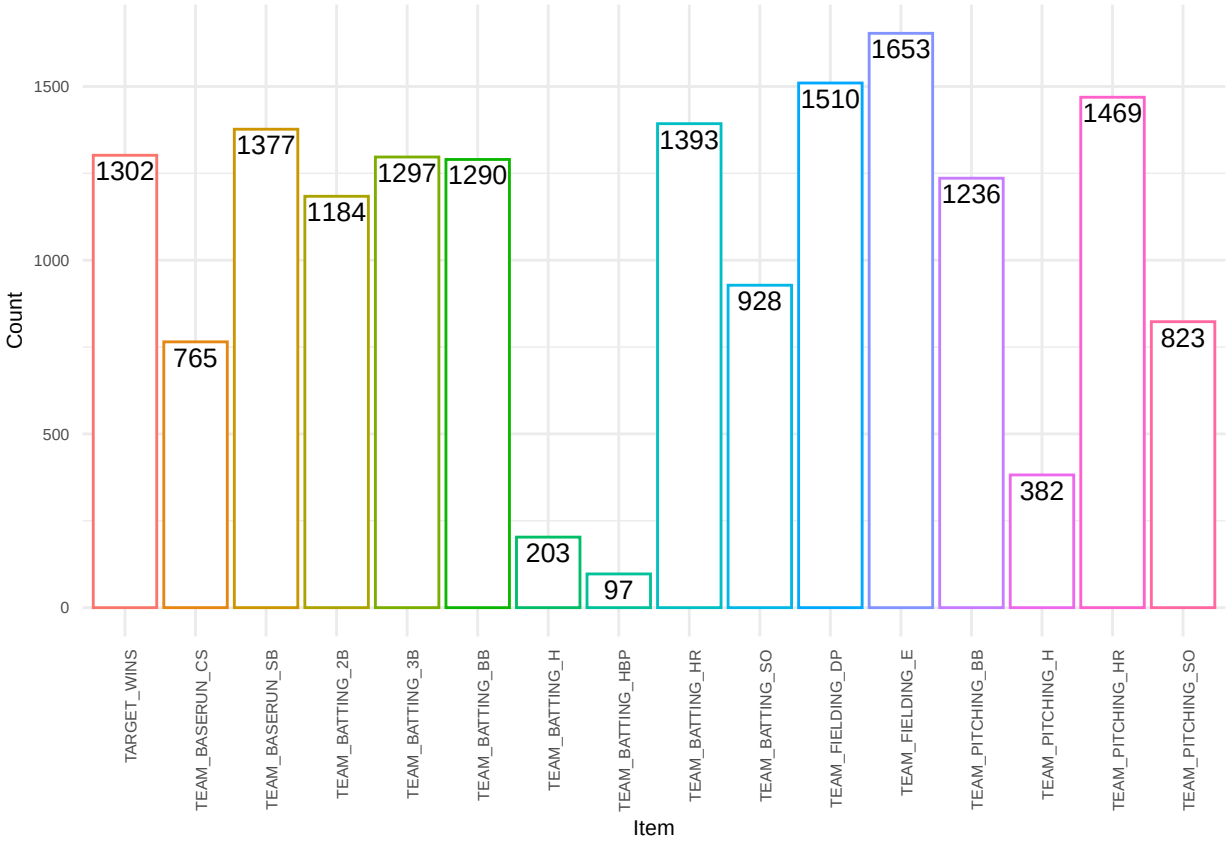
```
## 13    13  2276  553.   166.    536.   543.   98.6     0  3645  3645  6.74
## 14    14  2174  818.   553.    814.   797.  257.      0 19278 19278 22.2
## 15    15  2276  246.   228.    159    193.   62.3    65  1898  1833  2.99
## 16    16  1990  146.    26.2   149    148.   23.7    52   228   176 -0.389
## # ... with 2 more variables: kurtosis <dbl>, se <dbl>
```

Boxplot: Exploring Outliers

Histogram graphic for each variable

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

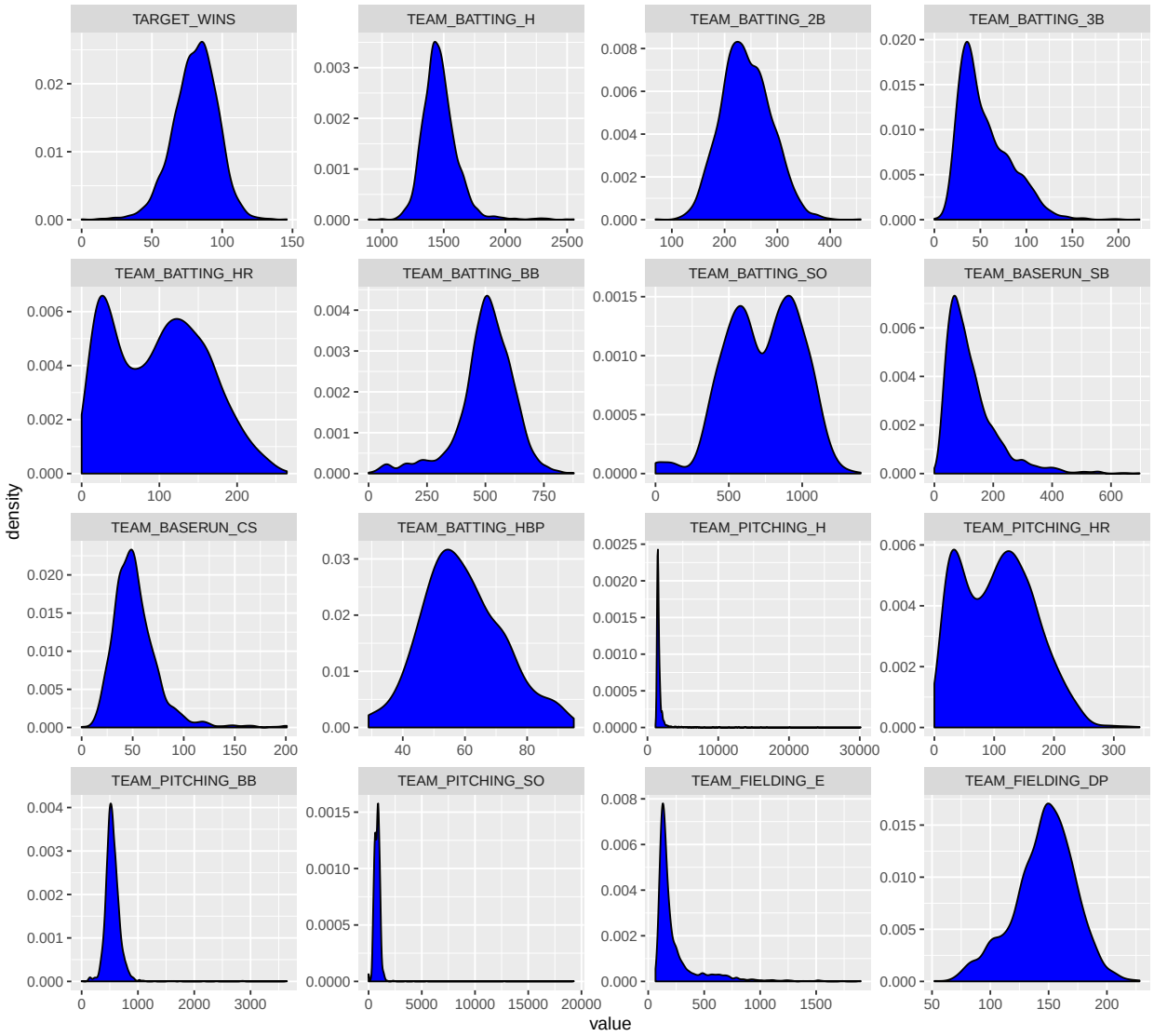




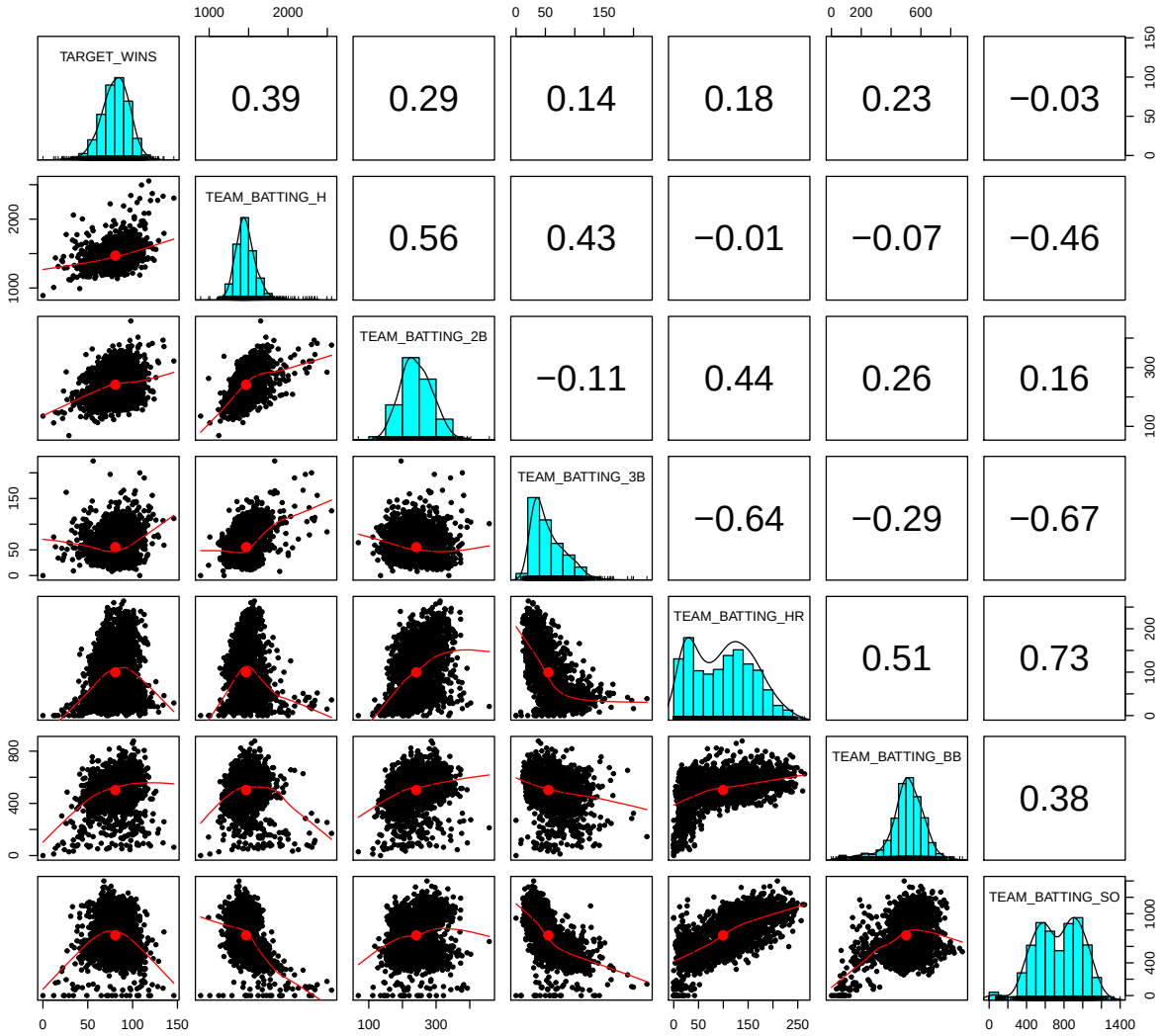
Results show that TEAM_FIELDING_E has the most outliers amongst the predictors and target variable.

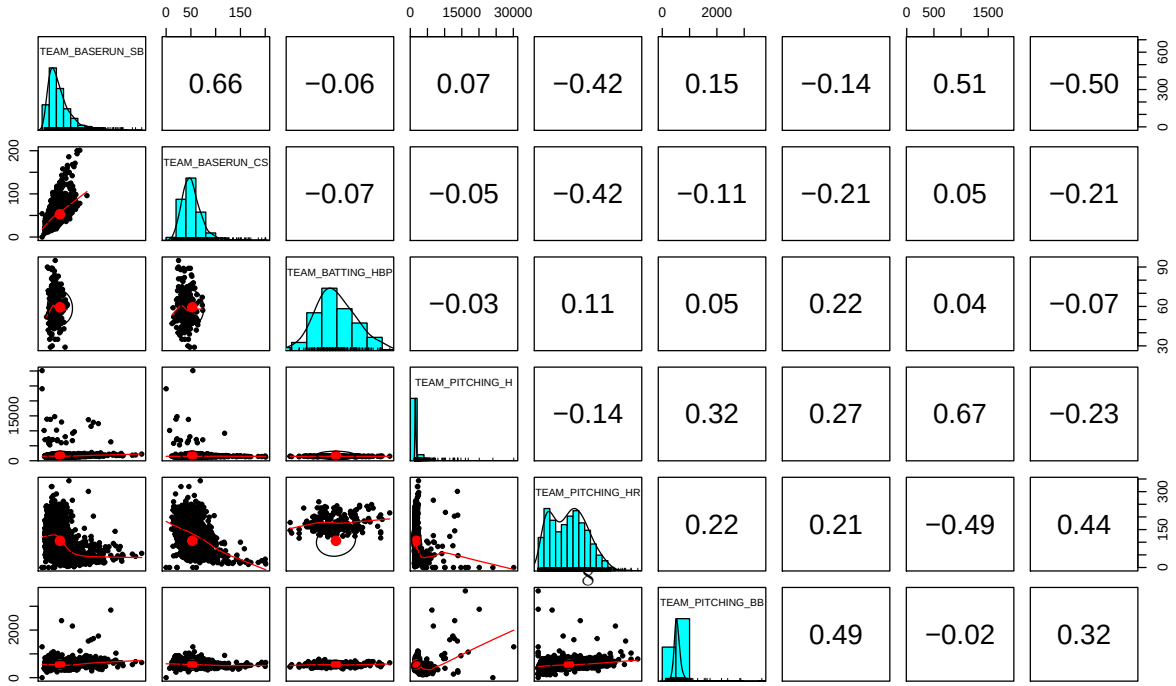
Skewness in Data

Histogram graphic displaying the distribution for each variables.

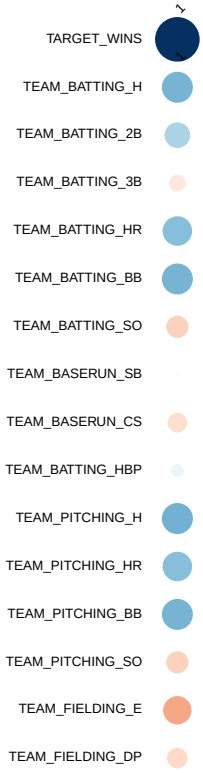

TARGET_WINS
TEAM_BATTING_H
TEAM_BATTING_2B
TEAM_BATTING_3B
TEAM_BATTING_HR
TEAM_BATTING_BB
TEAM_BATTING_SO
TEAM_BASERUN_SB
TEAM_BASERUN_CS
TEAM_BATTING_HBP
TEAM_PITCHING_H
TEAM_PITCHING_HR
TEAM_PITCHING_BB
TEAM_PITCHING_SO
TEAM_FIELDING_E
TEAM_FIELDING_DP
density
value

Correlations





Closer look at correlations with dependant variable.



From this Corrplot, we can see the variables `Team_Batting_H`,`Team_Batting_2B`,`Team_Batting_HR`, `Team_Batting_BB`, `Team_Pitching_H`,`Team_Pitching_HR`, and `Team_Pitching_BB`, all are positively correlated with Target_Wins. Not all variables we'd expect to have a positive contribution do so. For example the number of three base hits is negatively correlated with total wins. Why two base hits contributes positively and three base hits does the opposite is unclear. Walks allowed is also strangely positively correlated with wins.

Part ii. DATA PREPARATION

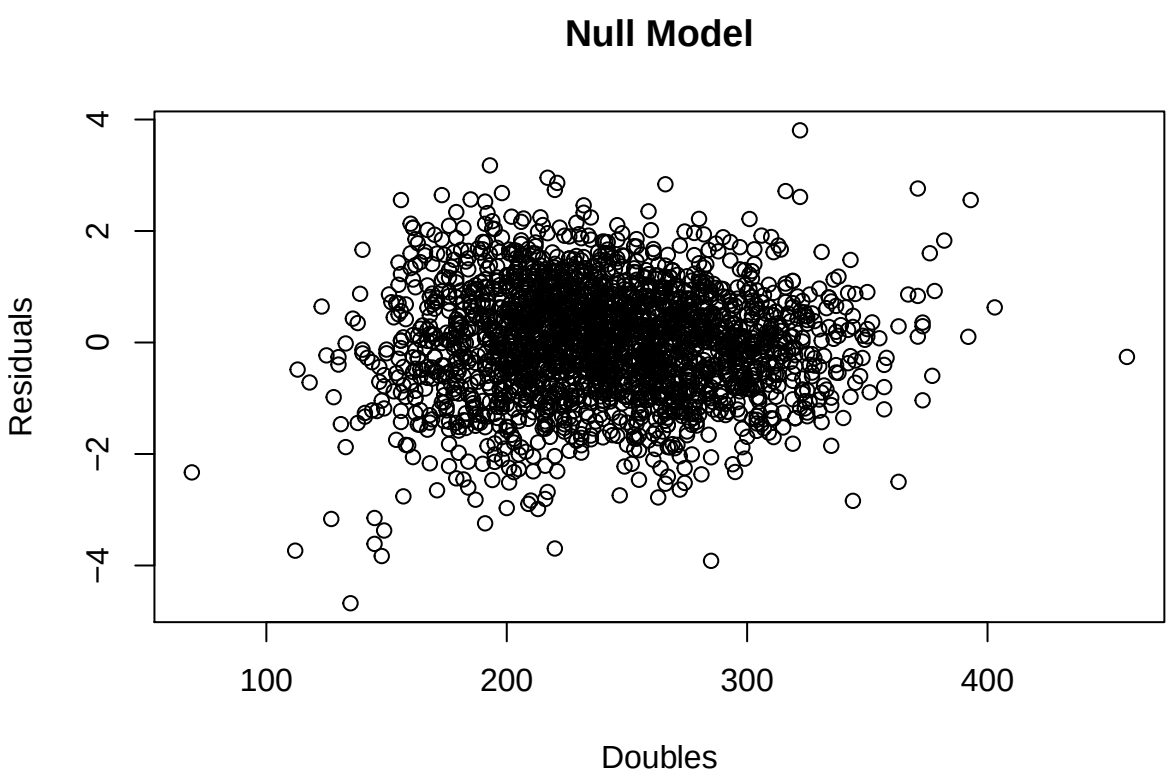
We will prepare our data using two formats: Filtering using booleans and missing value imputation. The outcome of how the models perform will based on the results provided by these two methods.

Objective 1: Ensure each variable can independently explain the variance in total wins. Ideally the residuals are smoothly and independently distributed around 0. Our aim should be to construct a vector of booleans that filters the noise from our explanatory variables.

Objective 2: Missing values can be a problem when trying to do analysis on the data. In most models, missing values are excluded which can limit the amount of information available in the analysis. This is the case why we have to either remove the missing values, impute them or model them. In this example, missing values will be imputed.

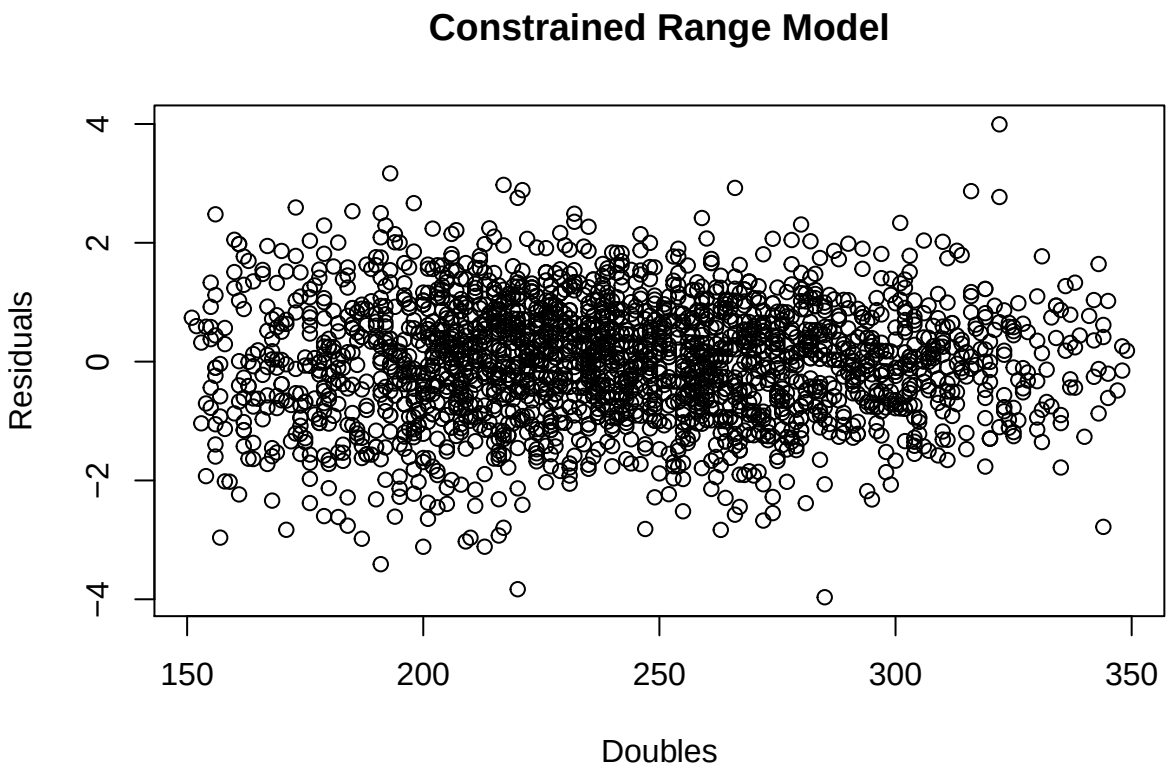
Filtering Noise

Doubles vs Residuals



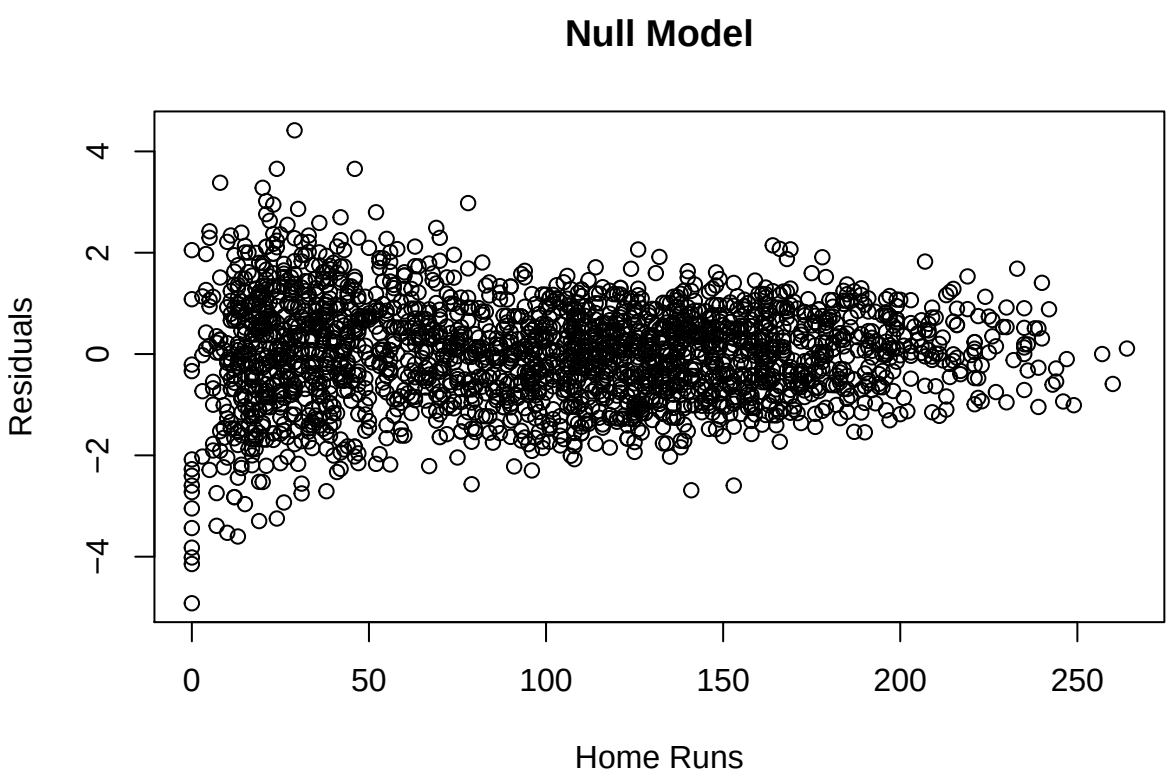
For a range from 150 to 350 doubles, the explanatory variable doubles apears to satisfy our assumptions (independence, homoscedasity) well. Most of our doubles data points are confined to this range. Therefore our transformation will consist of constraining the range of our explanatory variable.

Doubles Range (150 -350) vs Residuals



This worked very well. We can now create modified doubles and wins variables with only this range to use in our final model.

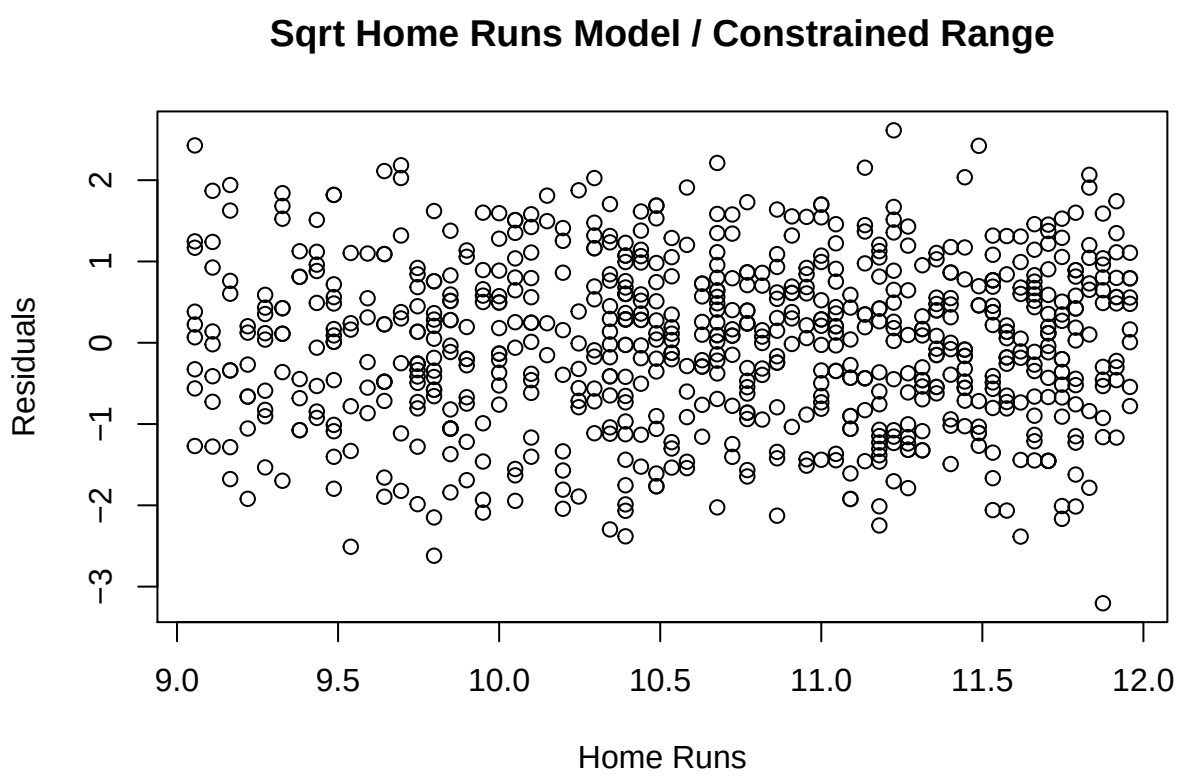
Home Runs vs Residuals



The explainitory variable Home Runs appears to satisfy our assumptions (independence, homoscedasity) well.There does appear to be some curvature for the home run range less than 50. We can take the square root of the explanatory variable to flatten this fish tail in the data

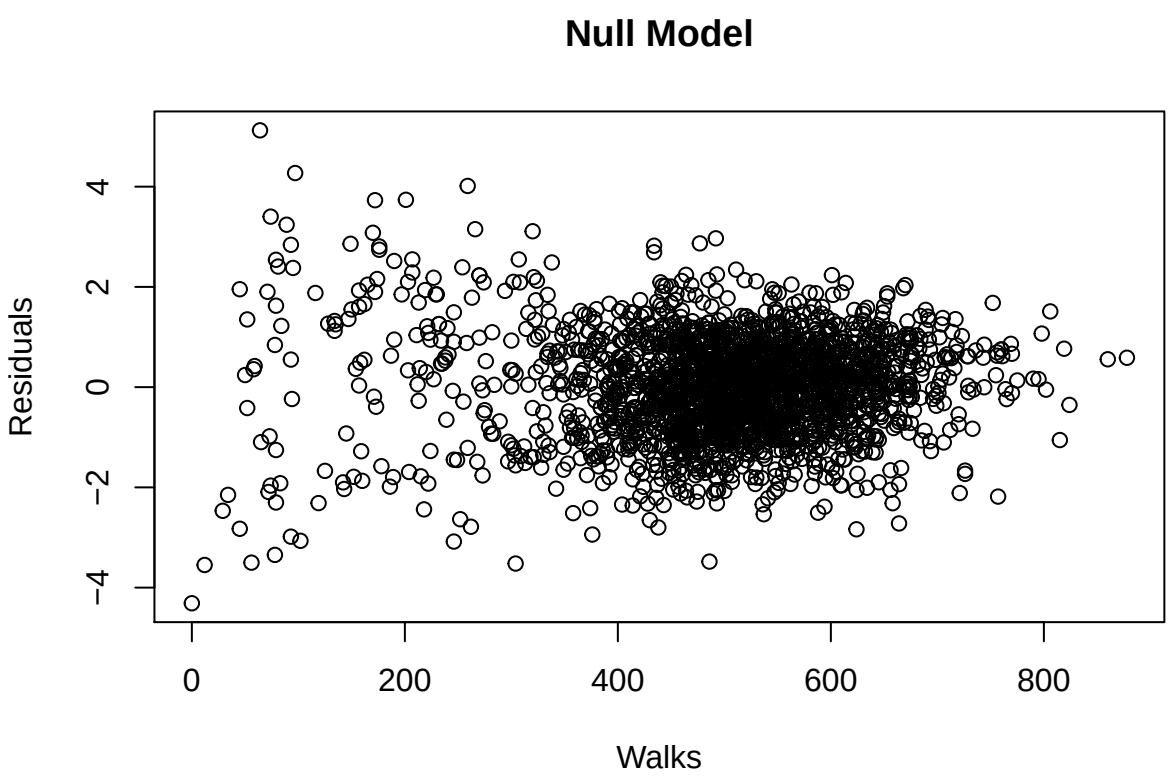
Home Runs vs Residuals

We can see this transformation has flattened the residuals overall but has left a gap in the data points at the lower end of Home Runs. Again we can constrain our range to observations greater than square root 7 and less than 12.

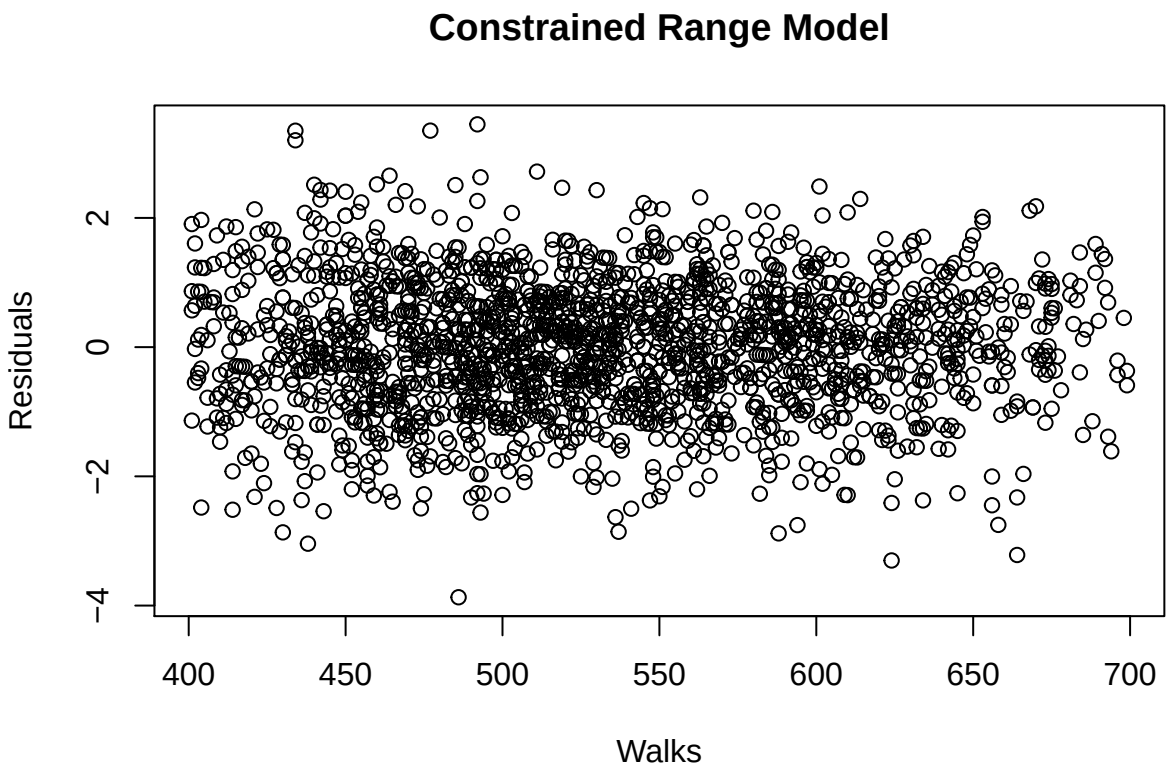


As can be seen from the range of residuals, the preceeding transformations have provided an explanatory variable that satisfies our assumptions. We can now further filter our data with a new explanatory variable and filtered target variable.

Walks vs Residuals

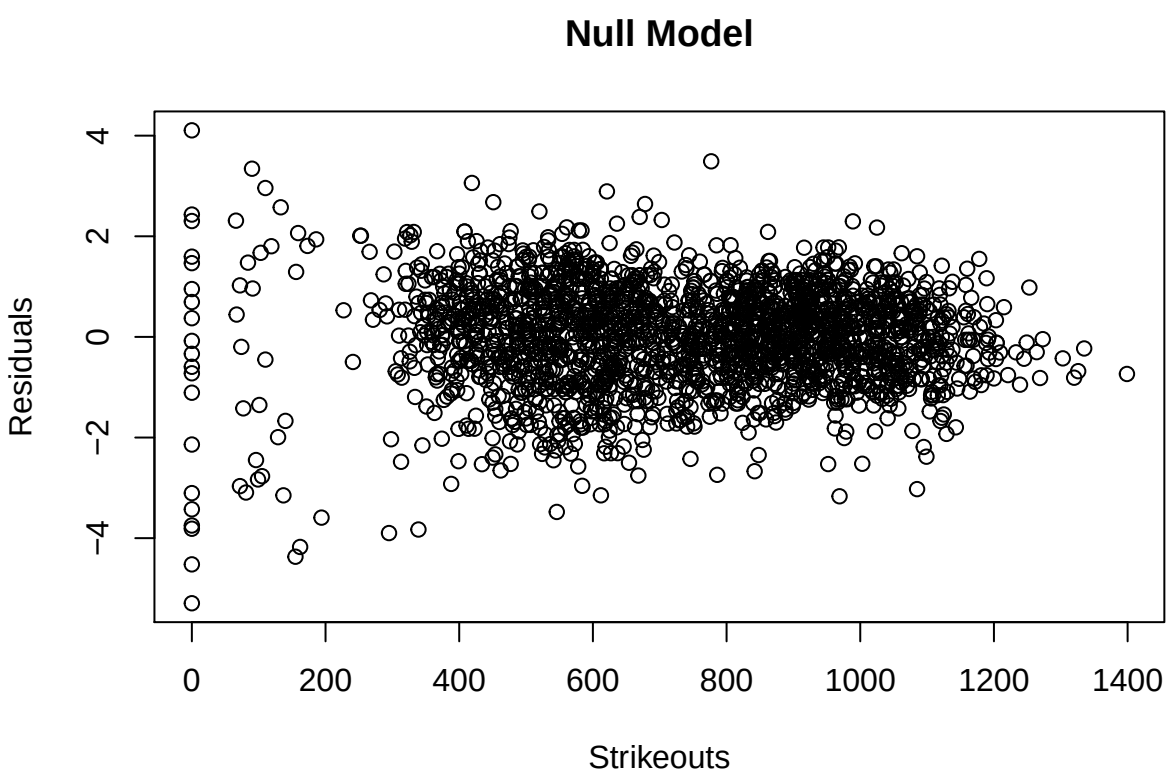


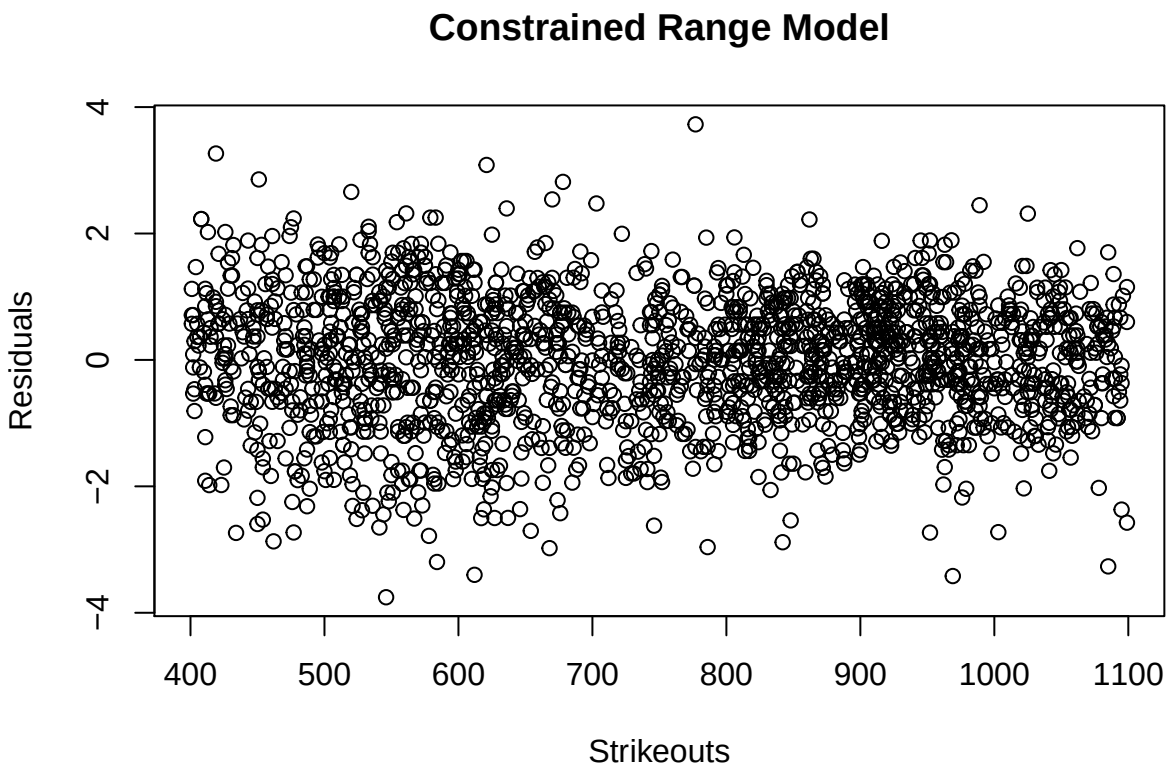
For walks we can see a situation similar to doubles. Let's zoom into the range between 400 and 700



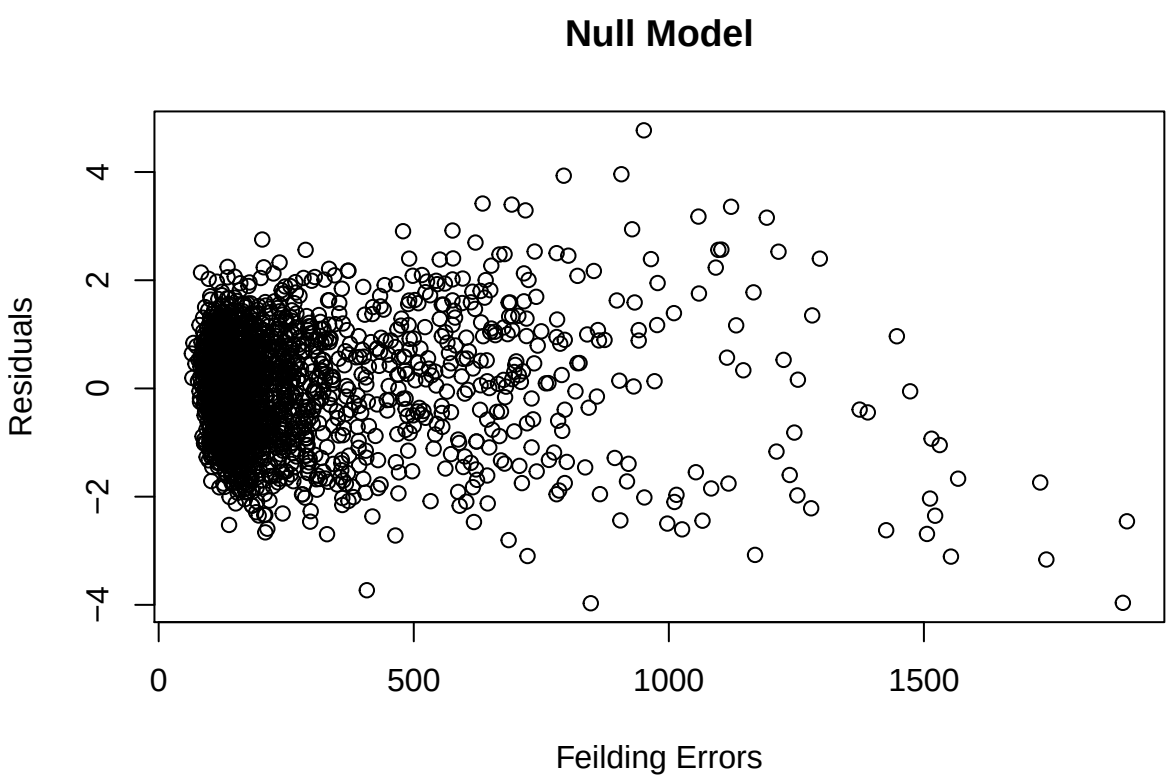
Much better! Let's store the boolean vector for walks.

Strikeouts

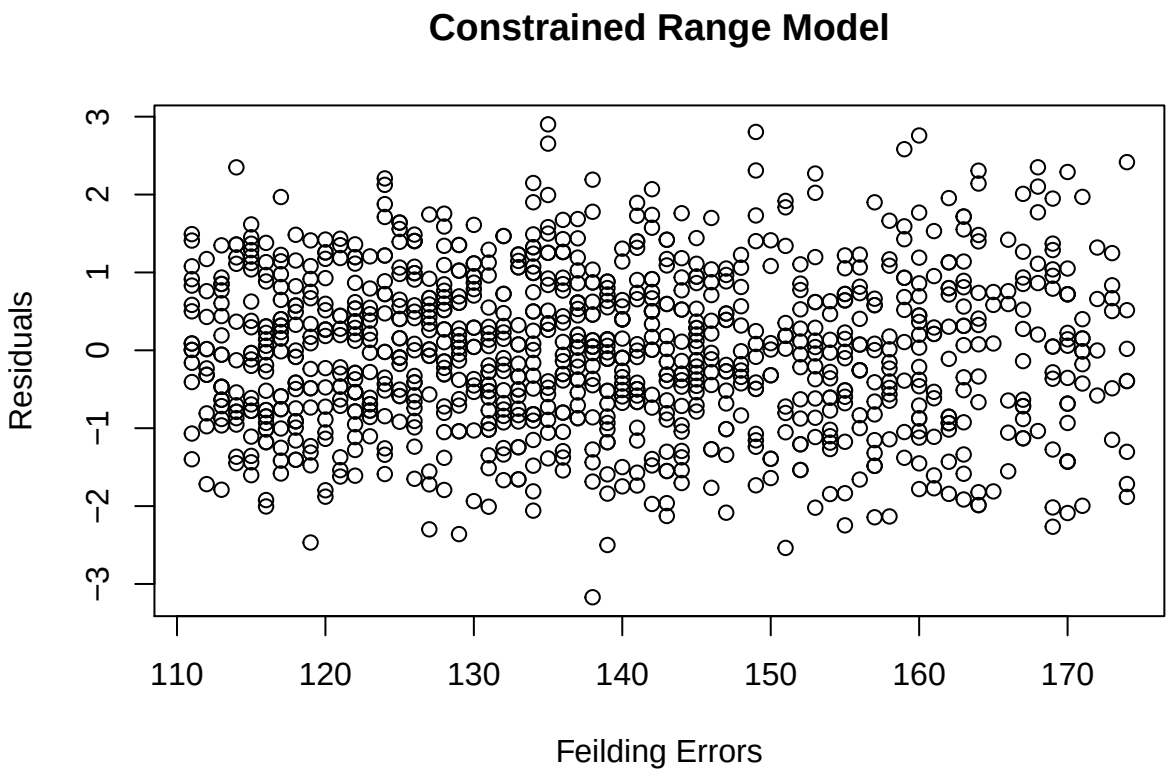

Null Model
Residuals
4
2
0
−2
−4
0
200
400
600
800
1000
1200
1400
Strikeouts



Fielding Errors



Feilding Errors vs Residuals



Combine Vectors and Filter Training Dataset.

Impute Missing data

```
## Warning: Number of logged events: 50
```

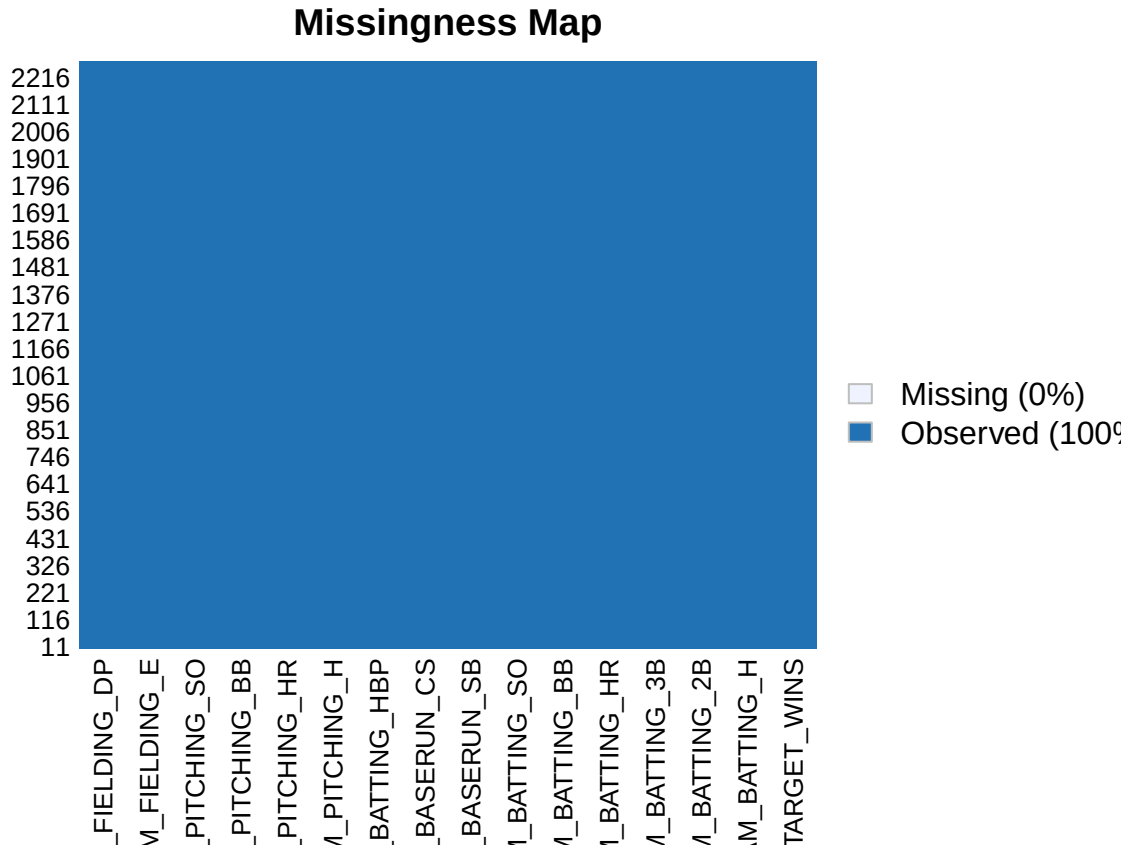
Preview

Look especially at the the variable TEAM_BATTING_HBP compared to original data it has 90% of its values missing.

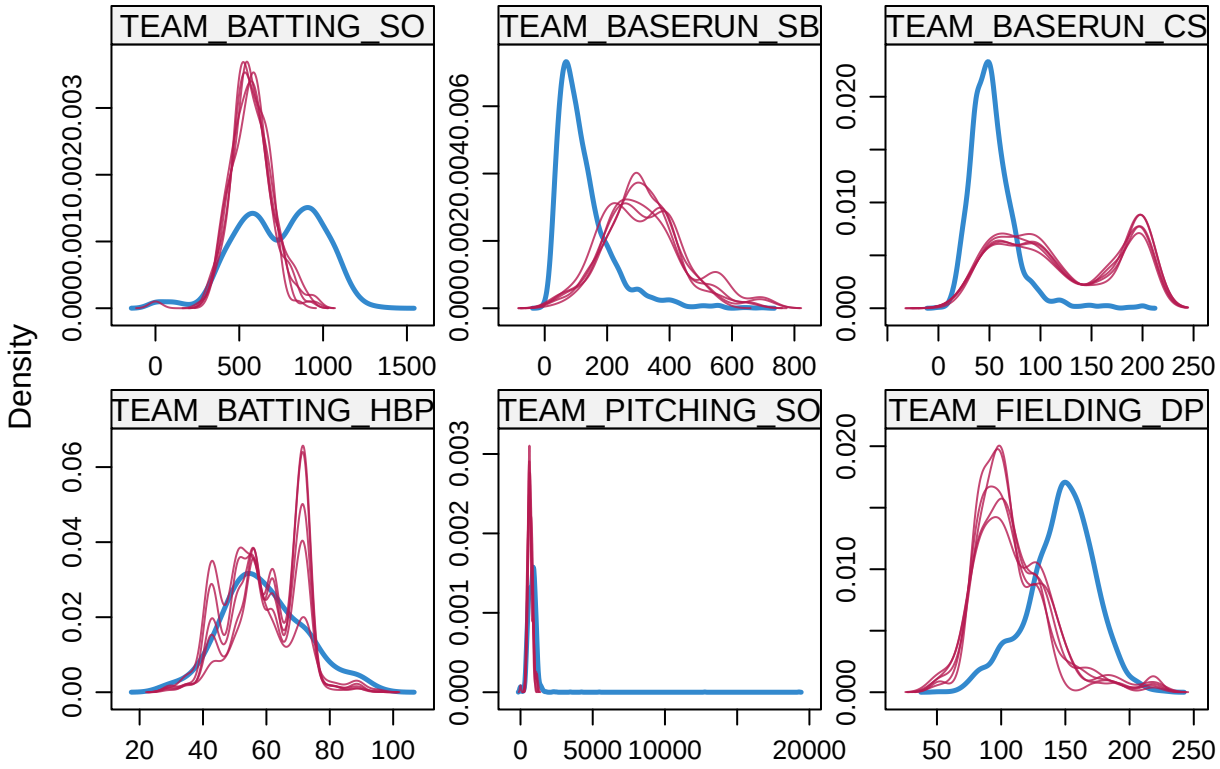
```
## # A tibble: 10 x 16
##    TARGET_WINS TEAM_BATTING_H TEAM_BATTING_2B TEAM_BATTING_3B TEAM_BATTING_HR
##          <int>          <int>           <int>           <int>           <int>
##  1          39           1445             194              39              13
##  2          70           1339             219              22             190
##  3          86           1377             232              35             137
##  4          70           1387             209              38              96
##  5          82           1297             186              27             102
##  6          75           1279             200              36              92
##  7          80           1244             179              54             122
##  8          85           1273             171              37             115
##  9          86           1391             197              40             114
## 10          76           1271             213              18              96
```

```
## # ... with 11 more variables: TEAM_BATTING_BB <int>, TEAM_BATTING_SO <int>,
## #   TEAM_BASERUN_SB <int>, TEAM_BASERUN_CS <int>, TEAM_BATTING_HBP <int>,
## #   TEAM_PITCHING_H <int>, TEAM_PITCHING_HR <int>, TEAM_PITCHING_BB <int>,
## #   TEAM_PITCHING_SO <int>, TEAM_FIELDING_E <int>, TEAM_FIELDING_DP <int>
```

Visual of complete dataset

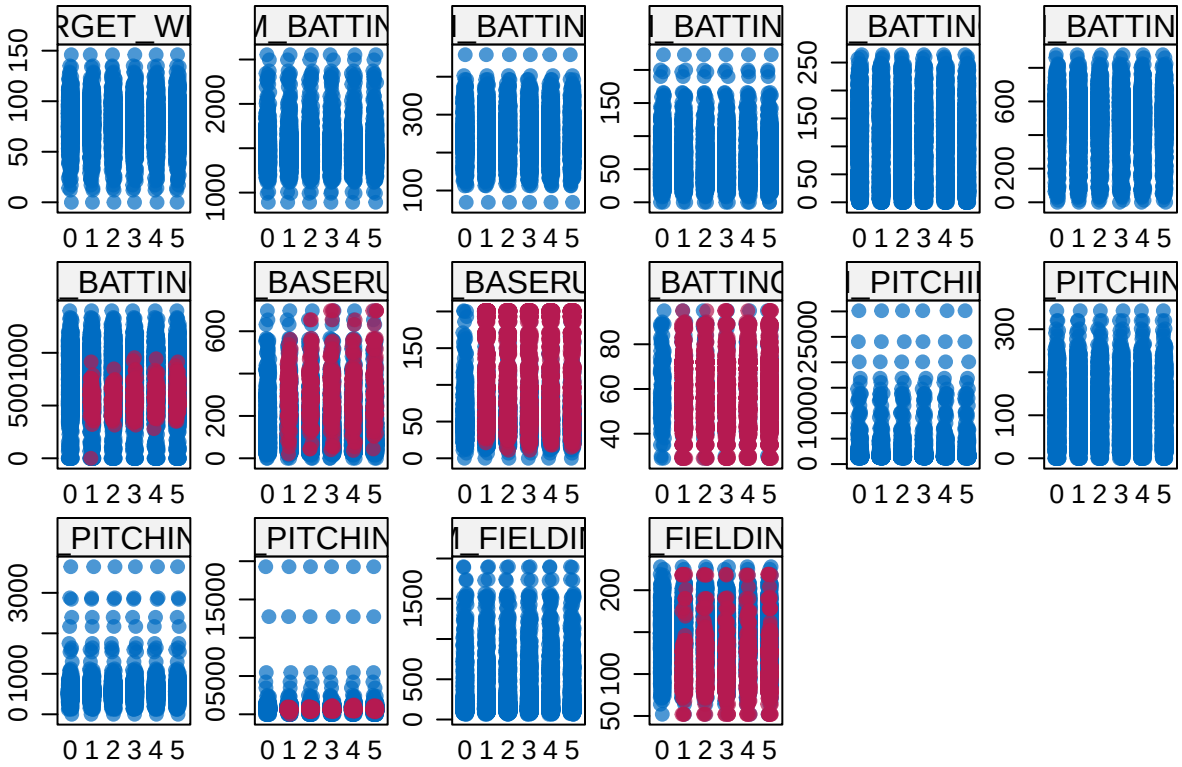


The dataset now consist of only complete rows where each missing value is replaced via the predictive mean method.



The imputed points are red and the obserced are blue. The matching shape of each distribution would tell us that the imputed values are plausible enough.

The Stripplot shows where the missing values were imputed based on the variables.

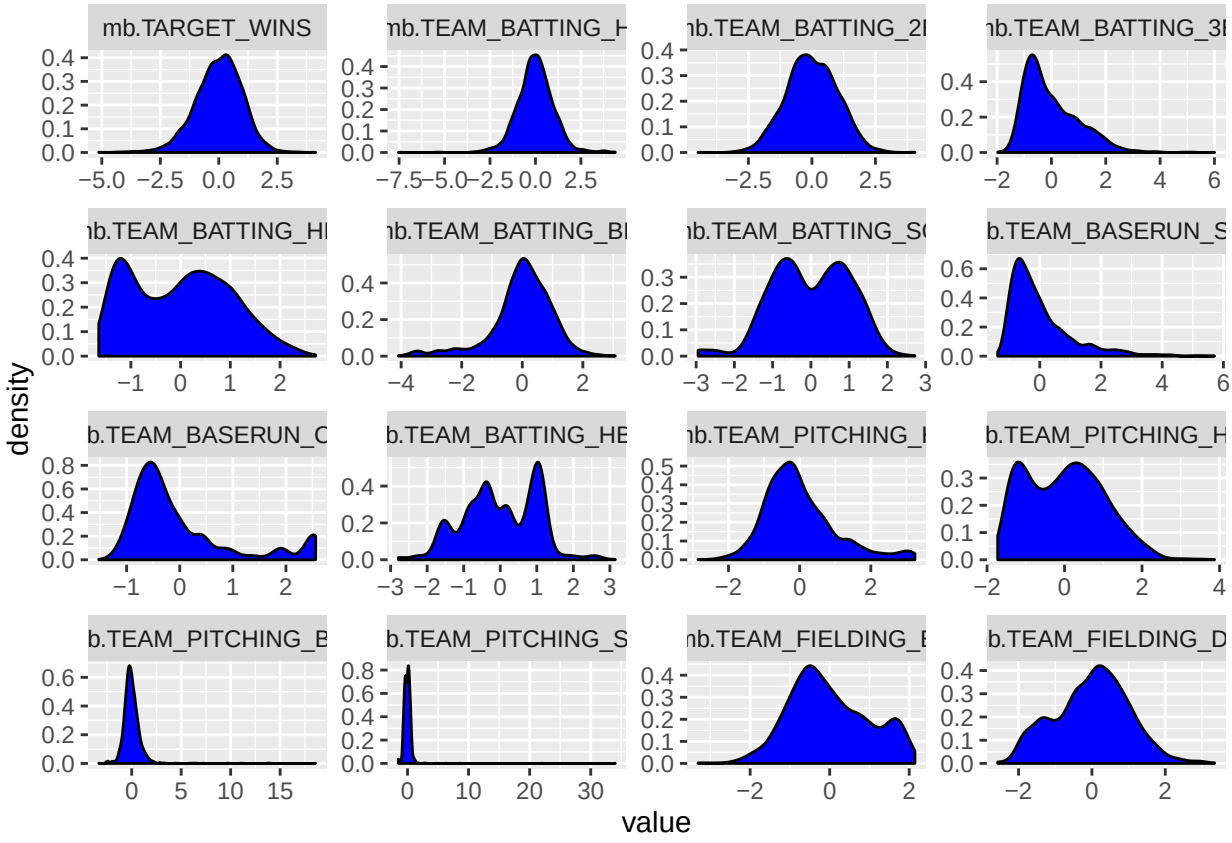


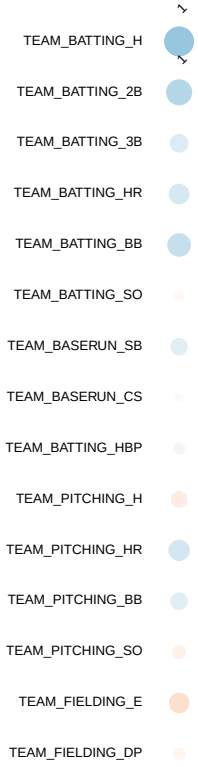
After imputation we can see that every variable has a value in each row and the NAs are gone.

```
##   TARGET_WINS     TEAM_BATTING_H TEAM_BATTING_2B TEAM_BATTING_3B
##  Min.   :  0.00   Min.   : 891   Min.   : 69.0   Min.   :  0.00
##  1st Qu.: 71.00   1st Qu.:1383   1st Qu.:208.0   1st Qu.: 34.00
##  Median : 82.00   Median :1454   Median :238.0   Median : 47.00
##  Mean   : 80.79   Mean   :1469   Mean   :241.2   Mean   : 55.25
##  3rd Qu.: 92.00   3rd Qu.:1537   3rd Qu.:273.0   3rd Qu.: 72.00
##  Max.   :146.00   Max.   :2554   Max.   :458.0   Max.   :223.00
##  TEAM_BATTING_HR  TEAM_BATTING_BB TEAM_BATTING_SO  TEAM_BASERUN_SB
##  Min.   :  0.00   Min.   :  0.0   Min.   :   0.0   Min.   :  0.0
##  1st Qu.: 42.00   1st Qu.:451.0   1st Qu.: 542.0   1st Qu.: 67.0
##  Median :102.00   Median :512.0   Median : 734.5   Median :105.5
##  Mean   : 99.61   Mean   :501.6   Mean   : 728.0   Mean   :135.2
##  3rd Qu.:147.00   3rd Qu.:580.0   3rd Qu.: 925.0   3rd Qu.:170.0
##  Max.   :264.00   Max.   :878.0   Max.   :1399.0   Max.   :697.0
##  TEAM_BASERUN_CS  TEAM_BATTING_HBP TEAM_PITCHING_H TEAM_PITCHING_HR
##  Min.   :  0.00   Min.   :29.00    Min.   : 1137   Min.   :  0.0
##  1st Qu.: 43.00   1st Qu.:51.00    1st Qu.: 1419   1st Qu.: 50.0
##  Median : 57.00   Median :61.00    Median : 1518   Median :107.0
##  Mean   : 75.05   Mean   :60.00    Mean   : 1779   Mean   :105.7
##  3rd Qu.: 90.00   3rd Qu.:70.25    3rd Qu.: 1682   3rd Qu.:150.0
##  Max.   :201.00   Max.   :95.00    Max.   :30132   Max.   :343.0
##  TEAM_PITCHING_BB TEAM_PITCHING_SO  TEAM_FIELDING_E  TEAM_FIELDING_DP
##  Min.   :   0.0   Min.   :    0.0   Min.   :  65.0   Min.   : 52.0
```

```
##  1st Qu.: 476.0   1st Qu.:  607.8   1st Qu.: 127.0   1st Qu.:124.0
##  Median : 536.5   Median :  802.5   Median : 159.0   Median :145.0
##  Mean   : 553.0   Mean   :  809.2   Mean   : 246.5   Mean   :141.3
##  3rd Qu.: 611.0   3rd Qu.:  957.2   3rd Qu.: 249.2   3rd Qu.:162.0
##  Max.   :3645.0   Max.   :19278.0   Max.   :1898.0   Max.   :228.0
```

Transformation: Centering and Scaling





We have more postive correlation with the target variable than the previous correlation plot.

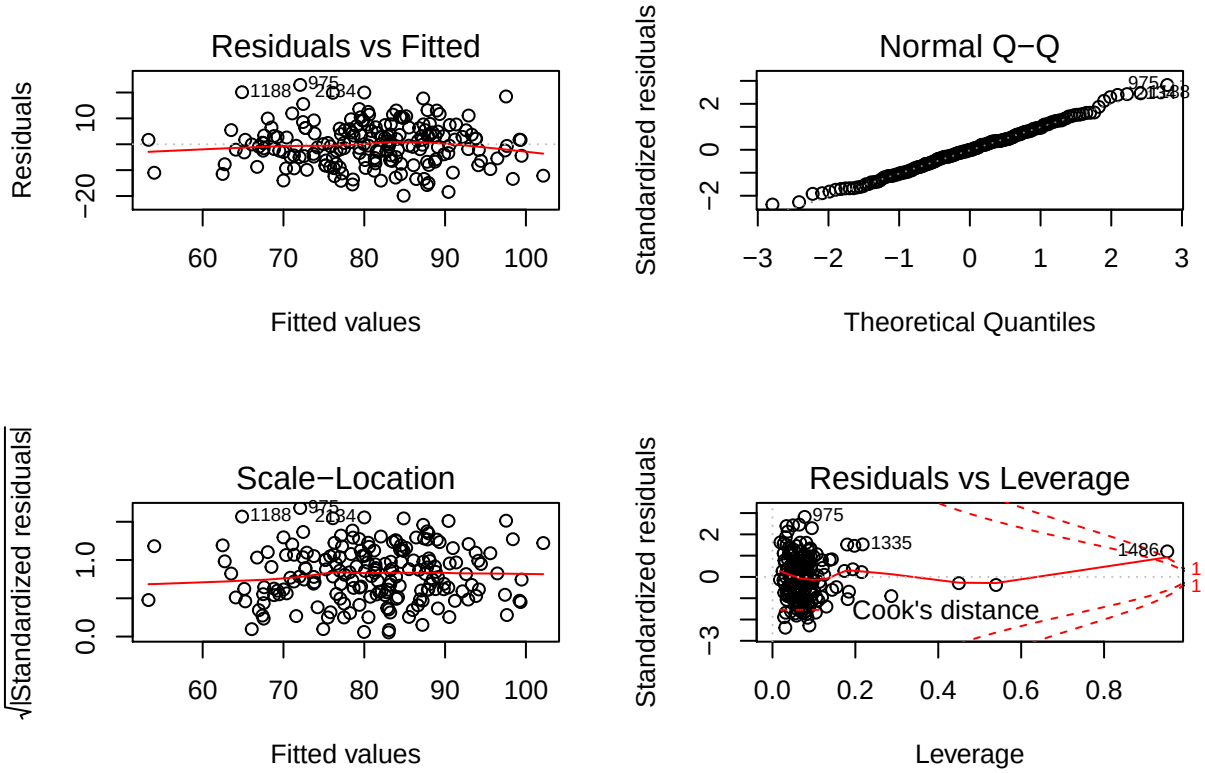
Part iii. BUILD MODELS

Model 1

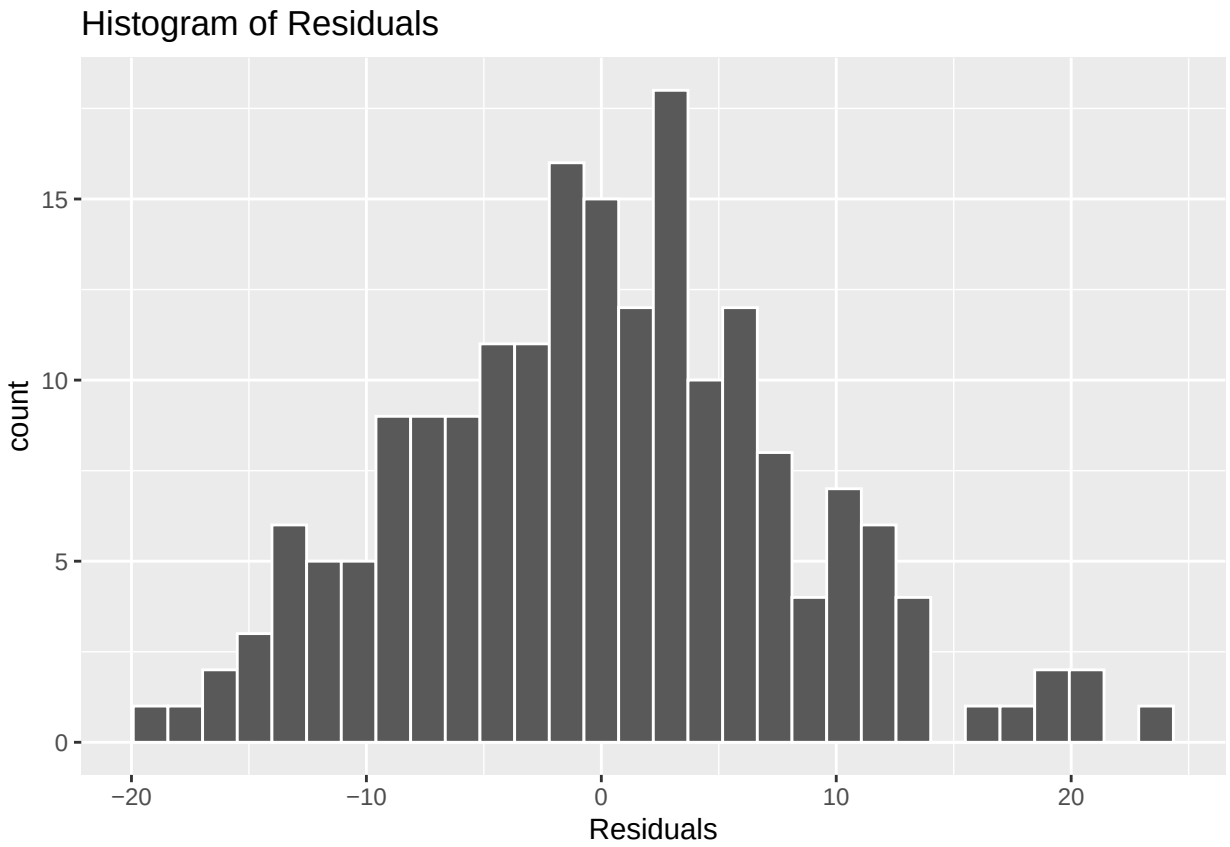
For the first model approach, we decide to create a regression on the raw data using all the variables on the data set. As we can observe we have many variables that don't have a good significance level. In our first model attempt, we obtain an R Squared value of 0.5501 and an adjusted R-square value of 0.5116, noticing that the difference maybe because of the numbers of variables in the regression that doesn't have a significance level. Also, we obtain a low F-statistic result.

```
##
## Call:
## lm(formula = TARGET_WINS ~ ., data = moneyball_training_data[,
##     -1])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.8708  -5.6564  -0.0599   5.2545  22.9274
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      60.28826   19.67842   3.064  0.00253 **
## TEAM_BATTING_H    1.91348    2.76139   0.693  0.48927
```

```
## TEAM_BATTING_2B   0.02639    0.03029   0.871  0.38484
## TEAM_BATTING_3B  -0.10118    0.07751  -1.305  0.19348
## TEAM_BATTING_HR  -4.84371   10.50851  -0.461  0.64542
## TEAM_BATTING_BB  -4.45969    3.63624  -1.226  0.22167
## TEAM_BATTING_SO   0.34196    2.59876   0.132  0.89546
## TEAM_BASERUN_SB   0.03304    0.02867   1.152  0.25071
## TEAM_BASERUN_CS  -0.01104    0.07143  -0.155  0.87730
## TEAM_BATTING_HBP  0.08247    0.04960   1.663  0.09815 .
## TEAM_PITCHING_H  -1.89096    2.76095  -0.685  0.49432
## TEAM_PITCHING_HR  4.93043   10.50664   0.469  0.63946
## TEAM_PITCHING_BB  4.51089    3.63372   1.241  0.21612
## TEAM_PITCHING_SO -0.37364    2.59705  -0.144  0.88577
## TEAM_FIELDING_E  -0.17204    0.04140  -4.155 5.08e-05 ***
## TEAM_FIELDING_DP -0.10819    0.03654  -2.961  0.00349 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.467 on 175 degrees of freedom
##   (2085 observations deleted due to missingness)
## Multiple R-squared:  0.5501, Adjusted R-squared:  0.5116
## F-statistic: 14.27 on 15 and 175 DF,  p-value: < 2.2e-16
```




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



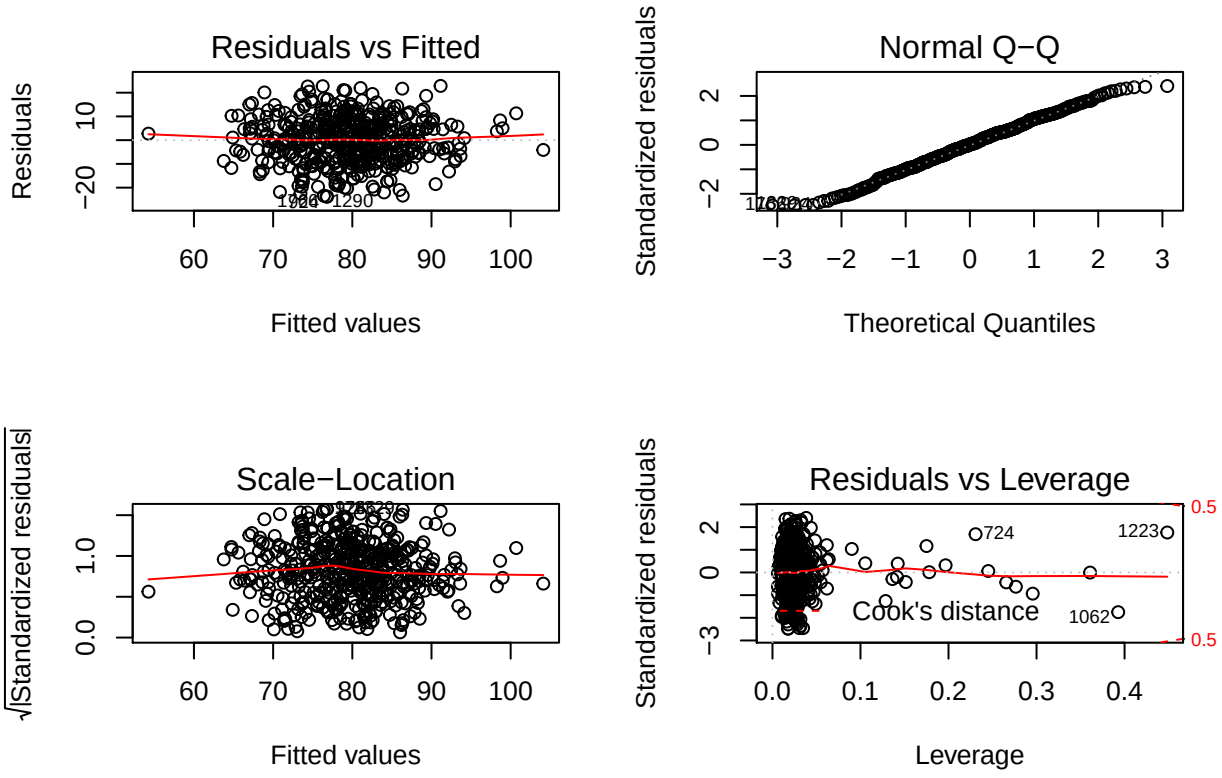
On a whole, the model is significant. However 2085 observations were removed due to missingness which makes the model skeptical. The residual plot seems normal with the points distributed randomly. We should be somewhat concerned with the outliers in the qq-plot which caused the tails of the plot to turn into the opposite direction. The variation in the third plot (bottom-left) seems to display homoscedasticity.

Model 2

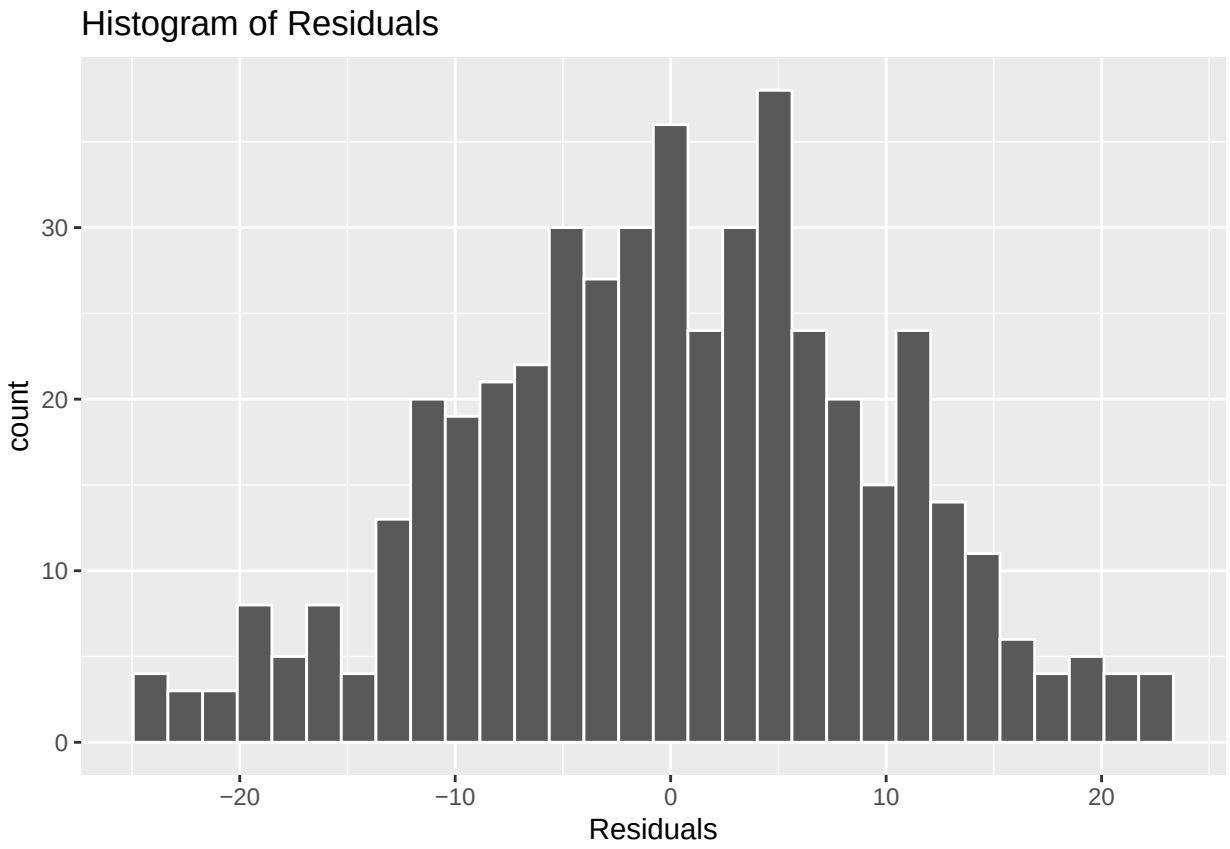
With this model we started doing an approach by filtering some of the noise in our variables (Doubles, Home runs, Walks, Strikeouts, fielding error). On this model we didn't see too much improvement, R square went down comparing to the first model but the F-statistics went slightly higher. As stated earlier, this model was built on a filtering method which kept 506 observations for the model to work with. An additional 33 were removed due to missingness.

```
##
## Call:
## lm(formula = 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,
##     data = meta_filter)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -23.8165  -6.6792   0.1261   6.2821  22.8485
##
```

```
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)      31.5992851 13.9216579   2.270  0.02368 *
## TEAM_BATTING_H    0.0514584  0.0377933   1.362  0.17400
## TEAM_BATTING_2B  -0.0565008  0.0185335  -3.049  0.00243 **
## TEAM_BATTING_3B   0.1392117  0.0423792   3.285  0.00110 **
## TEAM_BATTING_HR  -0.0282346  0.2850461  -0.099  0.92114
## TEAM_BATTING_BB  -0.1062370  0.0956209  -1.111  0.26714
## TEAM_BATTING_SO   0.0913935  0.0467464   1.955  0.05118 .
## TEAM_BASERUN_SB   0.0312539  0.0164747   1.897  0.05844 .
## TEAM_BASERUN_CS  -0.0312946  0.0395656  -0.791  0.42938
## TEAM_PITCHING_H  -0.0008172  0.0357390  -0.023  0.98177
## TEAM_PITCHING_HR  0.0952730  0.2736531   0.348  0.72788
## TEAM_PITCHING_BB  0.1375139  0.0929499   1.479  0.13971
## TEAM_PITCHING_SO -0.0953209  0.0444738  -2.143  0.03261 *
## TEAM_FIELDING_E  -0.1623533  0.0297495  -5.457 7.91e-08 ***
## TEAM_FIELDING_DP -0.1016161  0.0247806  -4.101 4.87e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.688 on 461 degrees of freedom
##   (33 observations deleted due to missingness)
## Multiple R-squared:  0.3219, Adjusted R-squared:  0.3014
## F-statistic: 15.63 on 14 and 461 DF,  p-value: < 2.2e-16
```




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

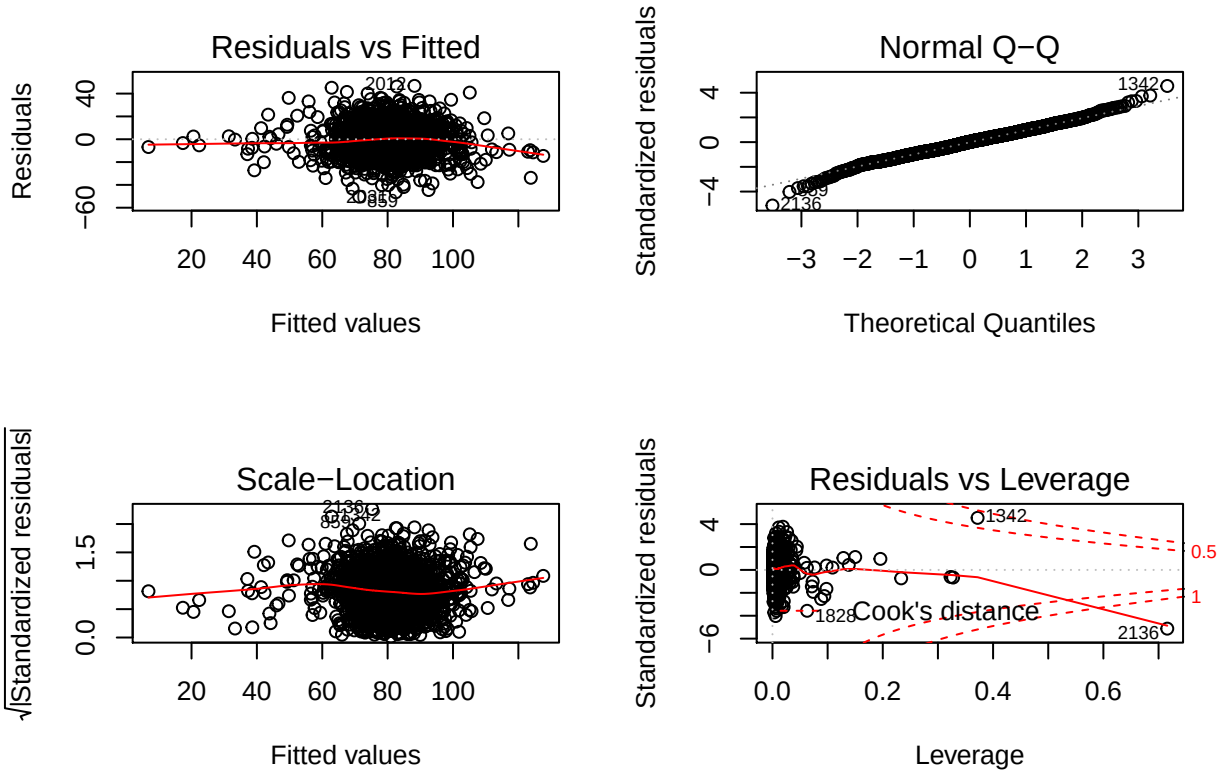


Model 3

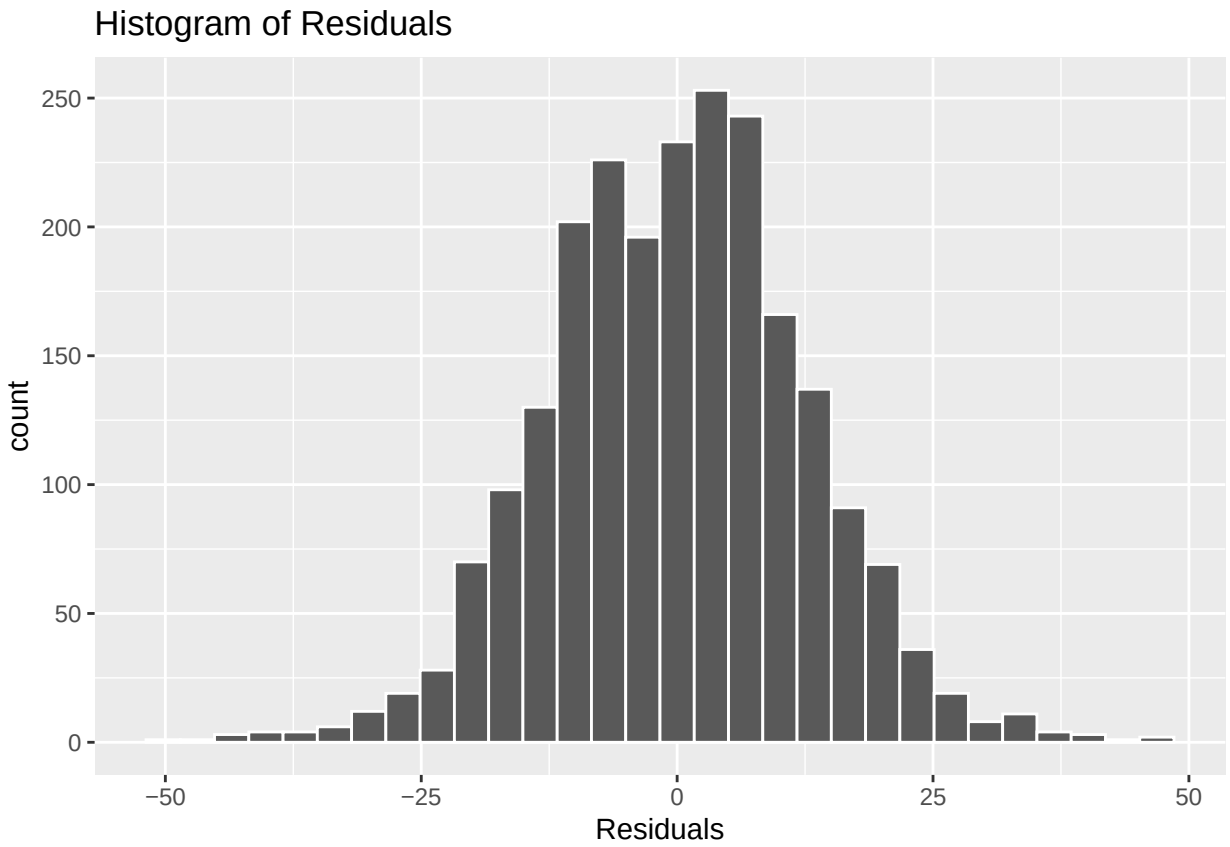
We observed that we have a lot missing data in our raw dataset and to solve this and hope to see an improvement in our model we decide to work by doing multiple imputations in our training data and after that create a new model.

```
##
## Call:
## lm(formula = TARGET_WINS ~ ., data = imputed_train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -50.308  -8.429   0.262   8.022  46.776
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      32.4215968  5.3247937   6.089 1.33e-09 ***
## TEAM_BATTING_H    0.0429219  0.0035525  12.082  < 2e-16 ***
## TEAM_BATTING_2B  -0.0217313  0.0088293  -2.461 0.013918 *
## TEAM_BATTING_3B   0.0313910  0.0163183   1.924 0.054522 .
## TEAM_BATTING_HR   0.0606282  0.0263058   2.305 0.021271 *
## TEAM_BATTING_BB   0.0193784  0.0056640   3.421 0.000634 ***
## TEAM_BATTING_SO  -0.0163874  0.0024659  -6.646 3.77e-11 ***
## TEAM_BASERUN_SB   0.0599810  0.0052927  11.333  < 2e-16 ***
## TEAM_BASERUN_CS  -0.0184157  0.0105683  -1.743 0.081548 .
```

```
## TEAM_BATTING_HBP  0.0632476  0.0252139   2.508 0.012196 *
## TEAM_PITCHING_H   0.0014835  0.0003808   3.896 0.000101 ***
## TEAM_PITCHING_HR  0.0168792  0.0234027   0.721 0.470829
## TEAM_PITCHING_BB -0.0088860  0.0040068  -2.218 0.026673 *
## TEAM_PITCHING_SO  0.0031946  0.0008956   3.567 0.000369 ***
## TEAM_FIELDING_E  -0.0421098  0.0026422 -15.937  < 2e-16 ***
## TEAM_FIELDING_DP -0.1221000  0.0126844  -9.626  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.53 on 2260 degrees of freedom
## Multiple R-squared:  0.3716, Adjusted R-squared:  0.3674
## F-statistic:  89.1 on 15 and 2260 DF,  p-value: < 2.2e-16
```




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



In this model, we definitely obtained more statistical significance variables and a good increase on f-statistics. This is an improvement compared to model 2 because the missing values were imputed. This gave the model more options to work with.

Model 4

This model is built on removing multicollinearity from the model 3 for improved results. The rule of thumb is to remove any variables with a score of more than 5.

```
##   TEAM_BATTING_H  TEAM_BATTING_2B  TEAM_BATTING_3B  TEAM_BATTING_HR
##         3.824178         2.474938         3.012718        36.769145
##  TEAM_BATTING_BB  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS
##         6.997200         5.365888         3.931167         3.922233
## TEAM_BATTING_HBP  TEAM_PITCHING_H TEAM_PITCHING_HR TEAM_PITCHING_BB
##         1.140037         4.160152        29.828536         6.440010
## TEAM_PITCHING_SO  TEAM_FIELDING_E TEAM_FIELDING_DP
##         3.426995         5.249712         2.051073
```

TEAM_BATTING_HR has a high VIF (highly correlated) score so it will be the first to be removed from the model.

```
##   TEAM_BATTING_H  TEAM_BATTING_2B  TEAM_BATTING_3B  TEAM_BATTING_BB
##         3.812960         2.474181         2.884116         5.615529
##  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS TEAM_BATTING_HBP
##         5.340989         3.928034         3.921867         1.136316
```

```
##  TEAM_PITCHING_H TEAM_PITCHING_HR TEAM_PITCHING_BB TEAM_PITCHING_SO
##         4.138367         3.802642         5.000993         3.175985
##  TEAM_FIELDING_E TEAM_FIELDING_DP
##         5.244291         2.051018
```

Next we remove TEAM_BATTING_BB

```
##   TEAM_BATTING_H  TEAM_BATTING_2B  TEAM_BATTING_3B  TEAM_BATTING_SO
##         3.811750         2.469475         2.882555         5.135989
##  TEAM_BASERUN_SB  TEAM_BASERUN_CS TEAM_BATTING_HBP  TEAM_PITCHING_H
##         3.927827         3.921833         1.112140         3.437098
## TEAM_PITCHING_HR TEAM_PITCHING_BB TEAM_PITCHING_SO  TEAM_FIELDING_E
##         3.802252         1.846735         2.306046         4.978480
## TEAM_FIELDING_DP
##         1.980628
```

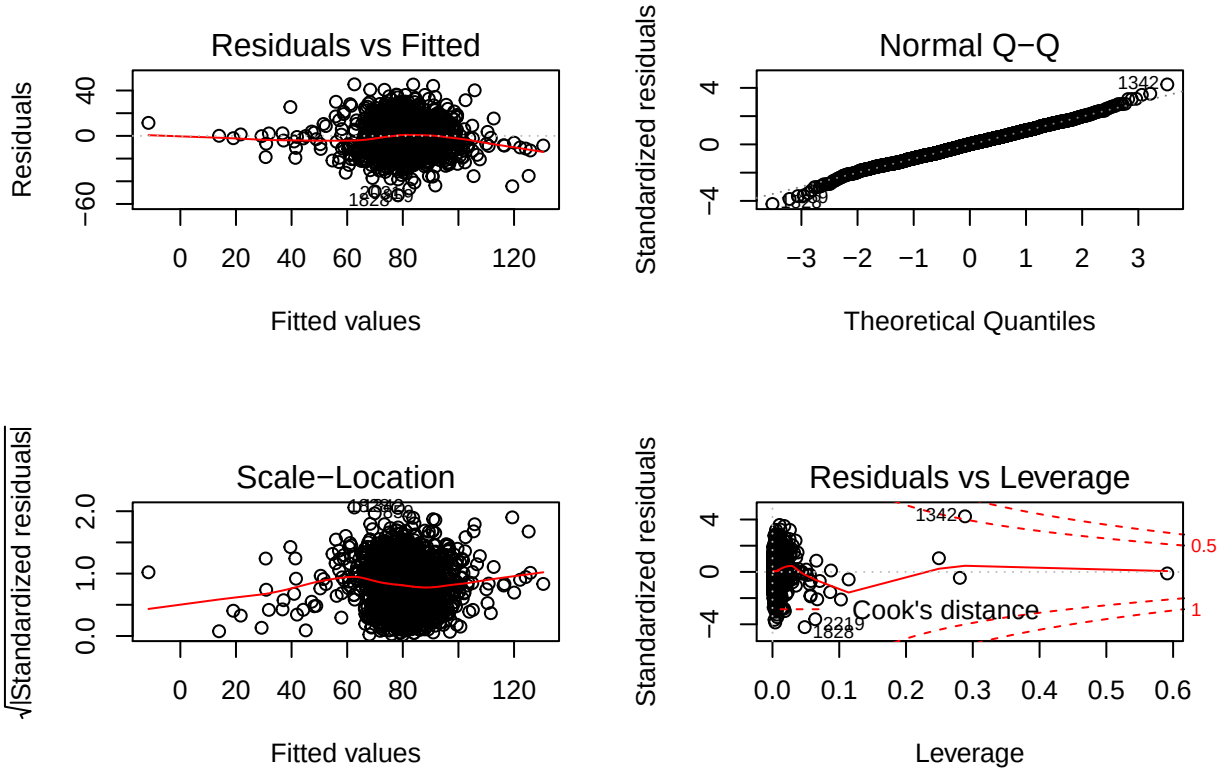
TEAM_BATTING_SO

```
##   TEAM_BATTING_H  TEAM_BATTING_2B  TEAM_BATTING_3B  TEAM_BASERUN_SB
##         3.157901         2.359989         2.815128         3.720868
##  TEAM_BASERUN_CS TEAM_BATTING_HBP  TEAM_PITCHING_H TEAM_PITCHING_HR
##         3.908211         1.107066         3.400599         2.441104
## TEAM_PITCHING_BB TEAM_PITCHING_SO  TEAM_FIELDING_E TEAM_FIELDING_DP
##         1.689236         1.872926         4.791321         1.966047
```

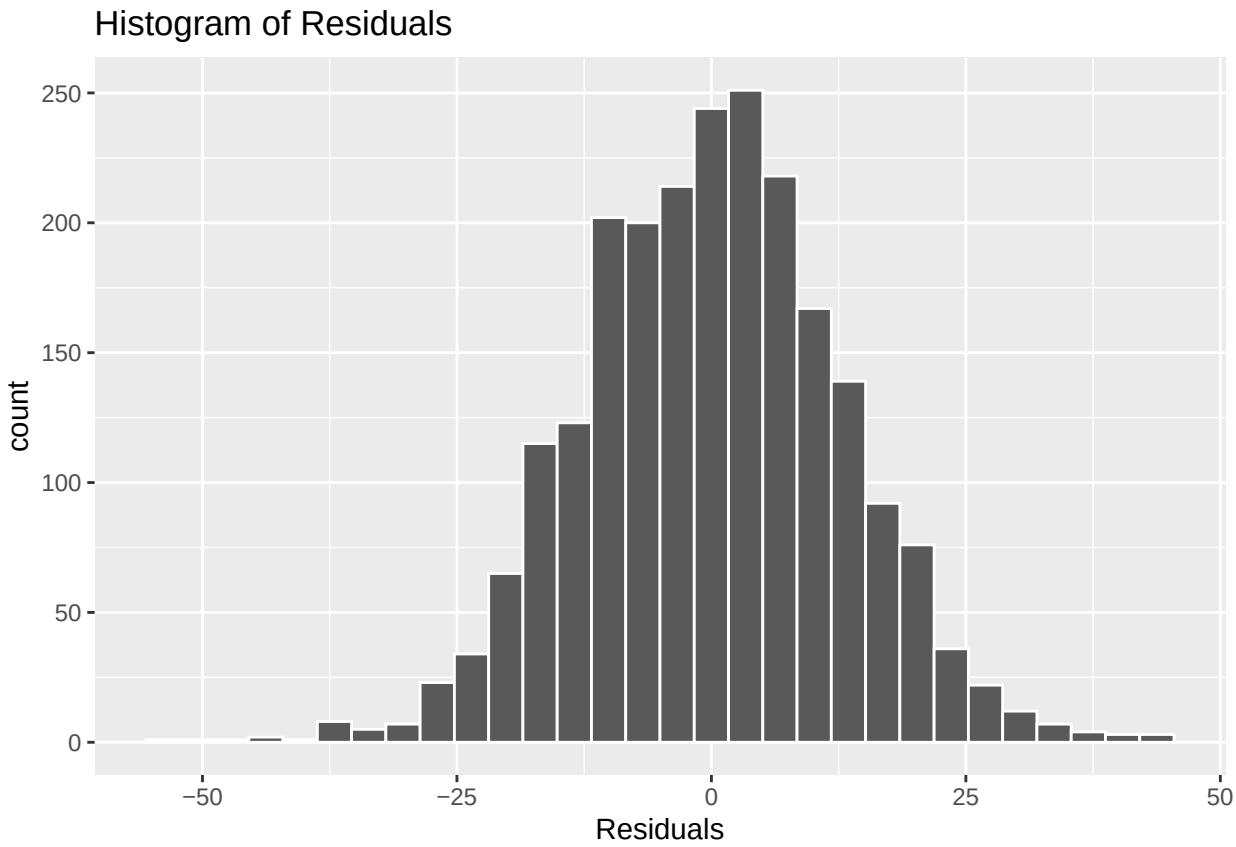
Multicollinearity is no longer present in this model. We can establish that `TEAM_BATTING_BB, TEAM_BATTING_SO and TEAM_BATTING_HR` are all dependent on other predictor variables and so they were removed from the model.

```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_2B +
##     TEAM_BATTING_3B + 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, data = imputed_train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -52.212  -8.669   0.270   8.274  45.415
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      17.8086199  3.9013570   4.565 5.27e-06 ***
## TEAM_BATTING_H    0.0519650  0.0032698  15.892  < 2e-16 ***
## TEAM_BATTING_2B  -0.0299403  0.0087330  -3.428 0.000618 ***
## TEAM_BATTING_3B   0.0355485  0.0159776   2.225 0.026187 *
## TEAM_BASERUN_SB   0.0529770  0.0052156  10.157  < 2e-16 ***
## TEAM_BASERUN_CS  -0.0152922  0.0106854  -1.431 0.152531
## TEAM_BATTING_HBP  0.0511955  0.0251671   2.034 0.042046 *
## TEAM_PITCHING_H   0.0009723  0.0003487   2.788 0.005345 **
## TEAM_PITCHING_HR  0.0394807  0.0067812   5.822 6.64e-09 ***
## TEAM_PITCHING_BB  0.0041987  0.0020786   2.020 0.043504 *
```

```
## TEAM_PITCHING_SO -0.0002819  0.0006706  -0.420 0.674231
## TEAM_FIELDING_E  -0.0424416  0.0025568 -16.600  < 2e-16 ***
## TEAM_FIELDING_DP -0.1042555  0.0125789  -8.288  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.69 on 2263 degrees of freedom
## Multiple R-squared:  0.3544, Adjusted R-squared:  0.351
## F-statistic: 103.5 on 12 and 2263 DF,  p-value: < 2.2e-16
```




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



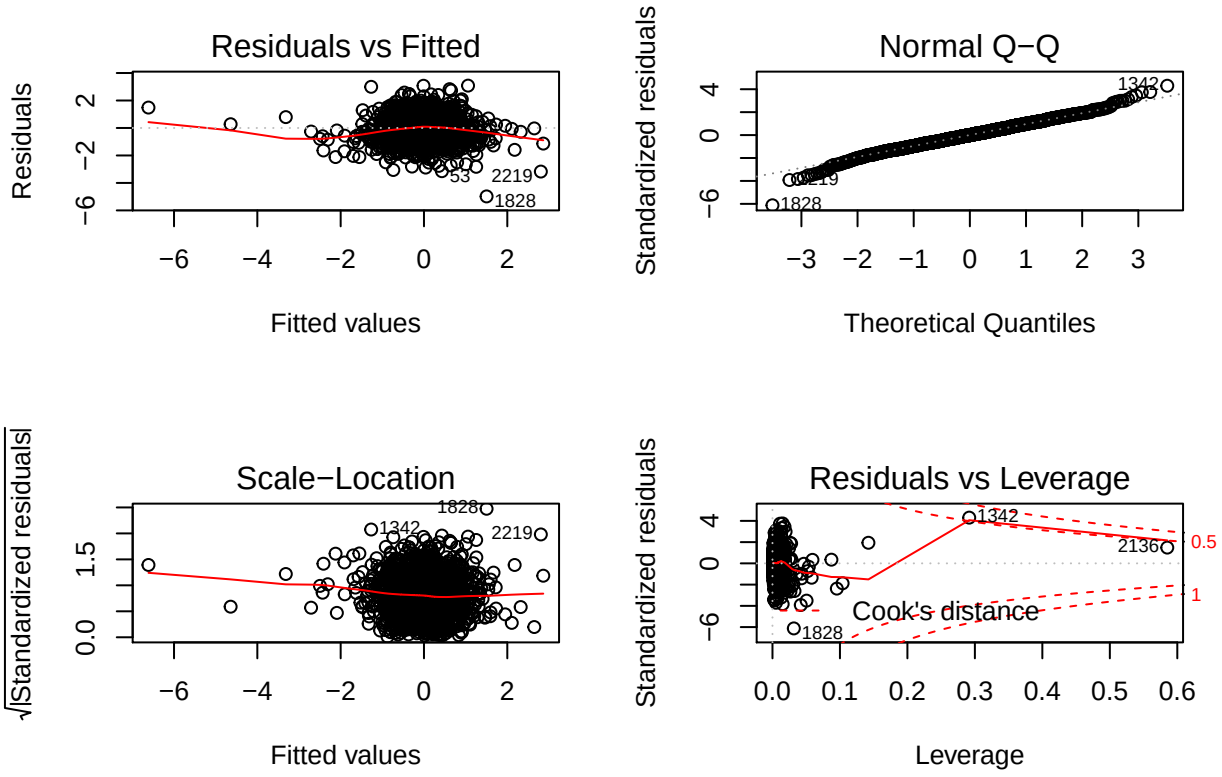
Even with multicollinearity removed, the Adj R-Squared did not increase. There is curvature in the scale-location plot which indicates non-contant variance.

Model 5

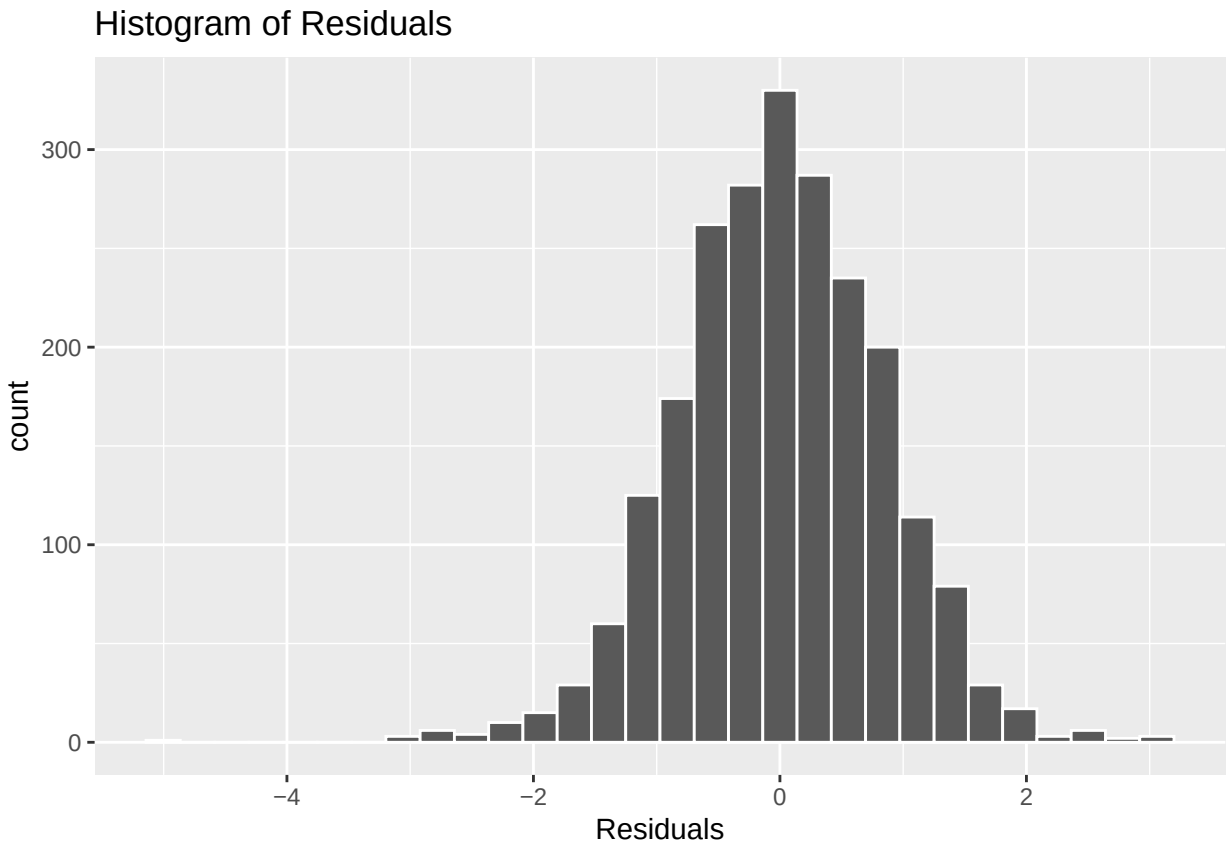
Using Transformed data and removing collinearity

```
##
## Call:
## lm(formula = mb.TARGET_WINS ~ mb.TEAM_BATTING_H + mb.TEAM_BATTING_2B +
##     mb.TEAM_BATTING_3B + mb.TEAM_BASERUN_SB + mb.TEAM_BASERUN_CS +
##     mb.TEAM_BATTING_HBP + mb.TEAM_PITCHING_H + mb.TEAM_PITCHING_HR +
##     mb.TEAM_PITCHING_BB + mb.TEAM_PITCHING_SO + mb.TEAM_FIELDING_E +
##     mb.TEAM_FIELDING_DP, data = moneyball_transformed)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.9810 -0.5259 -0.0020  0.5336  3.0816
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          1.728e-11  1.733e-02   0.000 1.000000
## mb.TEAM_BATTING_H    5.128e-01  3.520e-02  14.567  < 2e-16 ***
## mb.TEAM_BATTING_2B  -7.042e-02  2.736e-02  -2.574 0.010115 *
## mb.TEAM_BATTING_3B   1.836e-01  2.991e-02   6.137 9.91e-10 ***
```

```
## mb.TEAM_BASERUN_SB   2.296e-01  3.203e-02   7.168 1.02e-12 ***
## mb.TEAM_BASERUN_CS   3.162e-03  3.423e-02   0.092 0.926399
## mb.TEAM_BATTING_HBP  6.625e-02  1.854e-02   3.573 0.000361 ***
## mb.TEAM_PITCHING_H  -2.089e-01  3.230e-02  -6.467 1.22e-10 ***
## mb.TEAM_PITCHING_HR  6.543e-02  2.952e-02   2.216 0.026779 *
## mb.TEAM_PITCHING_BB  1.121e-01  2.132e-02   5.255 1.62e-07 ***
## mb.TEAM_PITCHING_SO -3.694e-03  2.352e-02  -0.157 0.875183
## mb.TEAM_FIELDING_E  -4.421e-01  3.813e-02 -11.595  < 2e-16 ***
## mb.TEAM_FIELDING_DP -1.630e-01  2.351e-02  -6.931 5.42e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8267 on 2263 degrees of freedom
## Multiple R-squared:  0.3202, Adjusted R-squared:  0.3166
## F-statistic: 88.83 on 12 and 2263 DF,  p-value: < 2.2e-16
```




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



The previous models showed a much wider spread however the residuals in this model are closer to 0. The plots seem normal. There are outliers noted but not to the point where we have to worry too much as we have more than 2000 observations to consider.

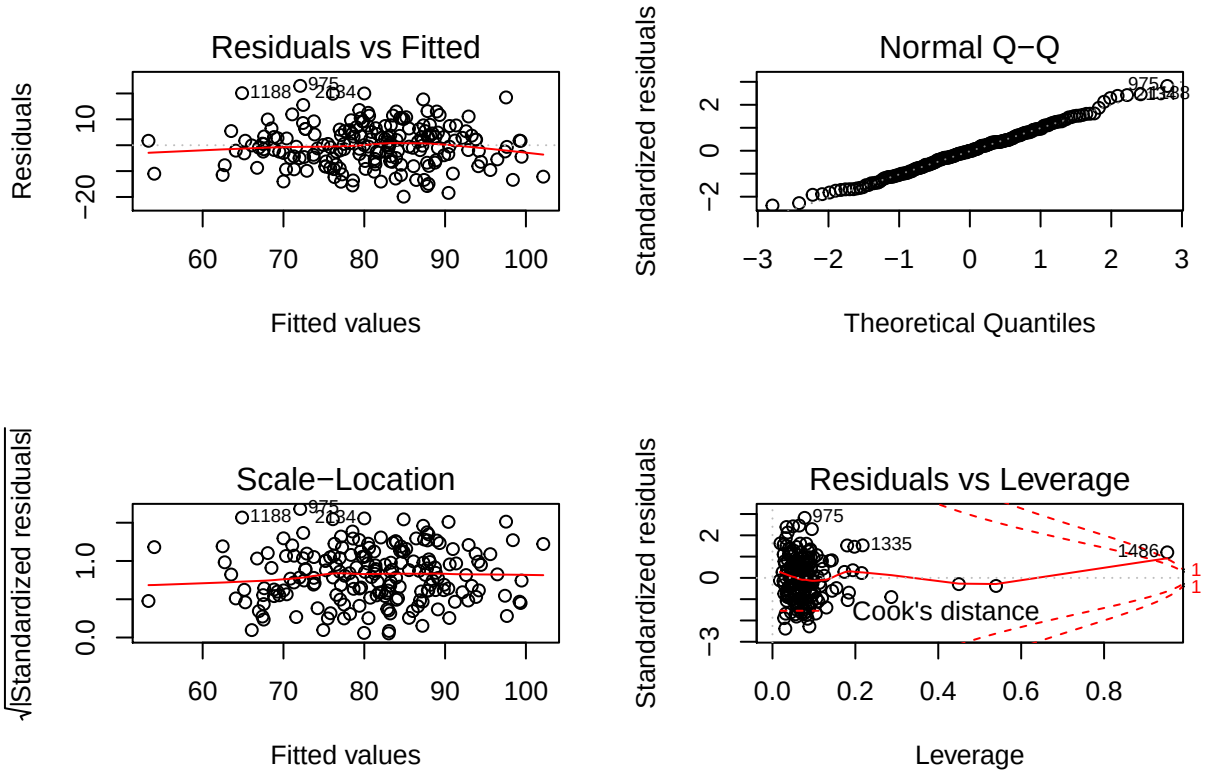
Model 6

Backward selection

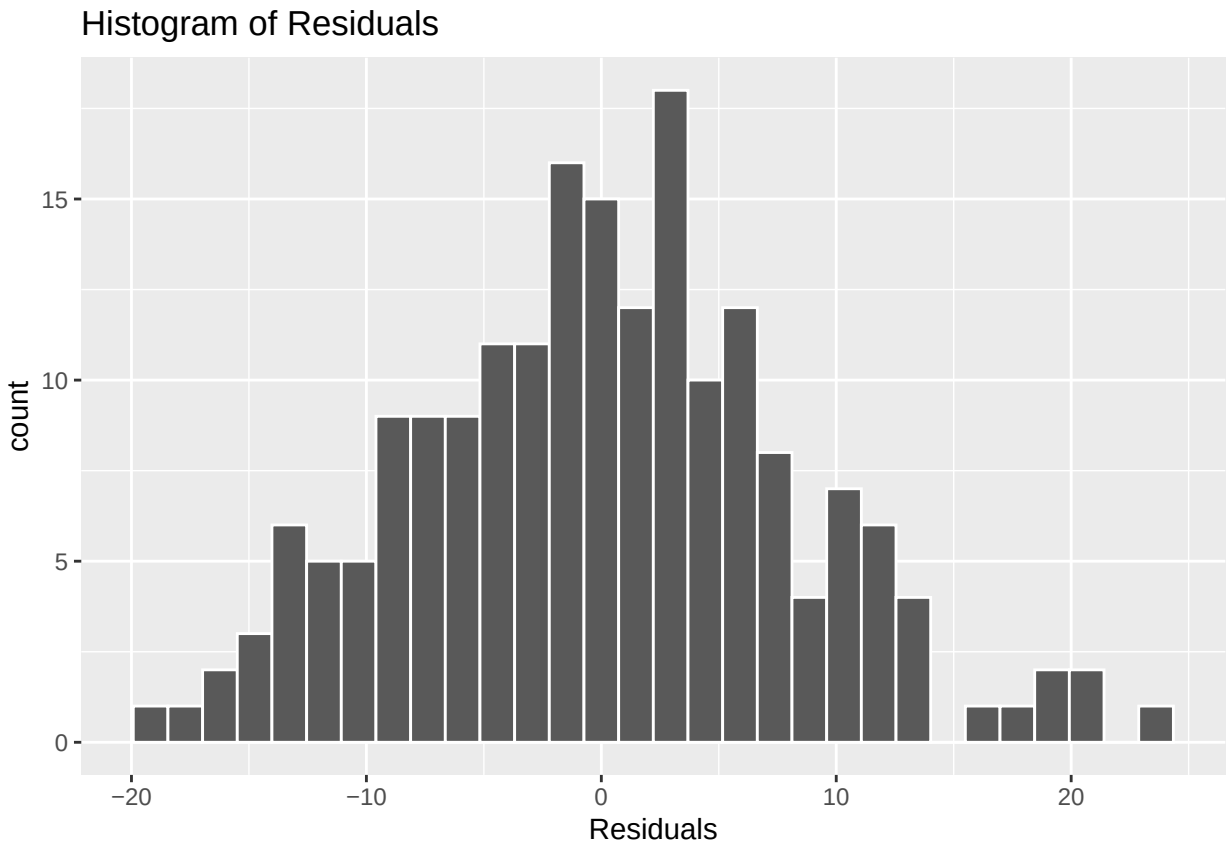
This model we going to work with Backward selection. We run a backward selection on the raw data in order to find the most significant variables. We get a high R square 0.5345 but a decrease in the F-statistic.

```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_HBP +
##     TEAM_PITCHING_HR + TEAM_PITCHING_BB + TEAM_PITCHING_SO +
##     TEAM_FIELDING_E + TEAM_FIELDING_DP, data = moneyball_training_data[,
##     -1])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -20.2248  -5.6294  -0.0212   5.0439  21.3065
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      60.95454   19.10292   3.191 0.001670 **
## TEAM_BATTING_H    0.02541    0.01009   2.518 0.012648 *
```

```
## TEAM_BATTING_HBP  0.08712    0.04852   1.796 0.074211 .
## TEAM_PITCHING_HR  0.08945    0.02394   3.736 0.000249 ***
## TEAM_PITCHING_BB  0.05672    0.00940   6.034 8.66e-09 ***
## TEAM_PITCHING_SO -0.03136    0.00728  -4.308 2.68e-05 ***
## TEAM_FIELDING_E  -0.17218    0.03970  -4.338 2.38e-05 ***
## TEAM_FIELDING_DP -0.11904    0.03516  -3.386 0.000869 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.422 on 183 degrees of freedom
##   (2085 observations deleted due to missingness)
## Multiple R-squared:  0.5345, Adjusted R-squared:  0.5167
## F-statistic: 30.02 on 7 and 183 DF,  p-value: < 2.2e-16
```




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



This model can be compared with Model 1. The difference is that only the statistically significant predictors remain which indeed improved the output of the model.

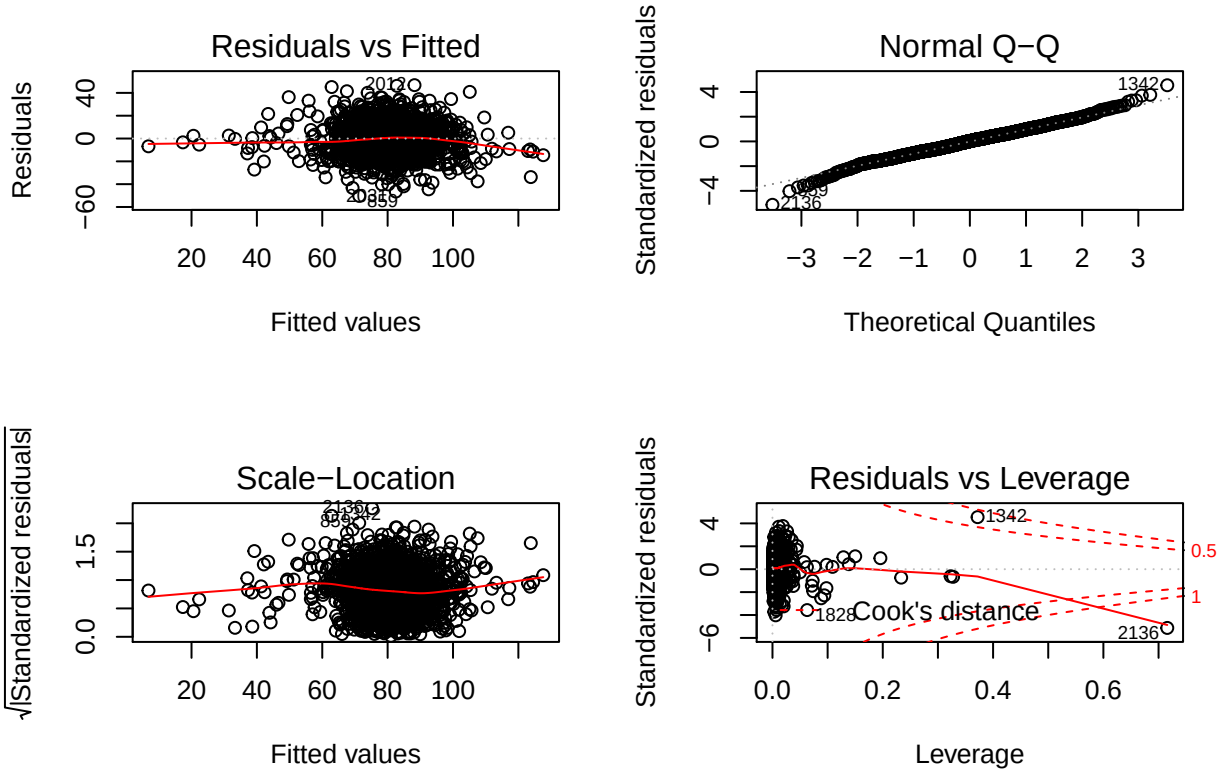
Model 7

Stepwise selection

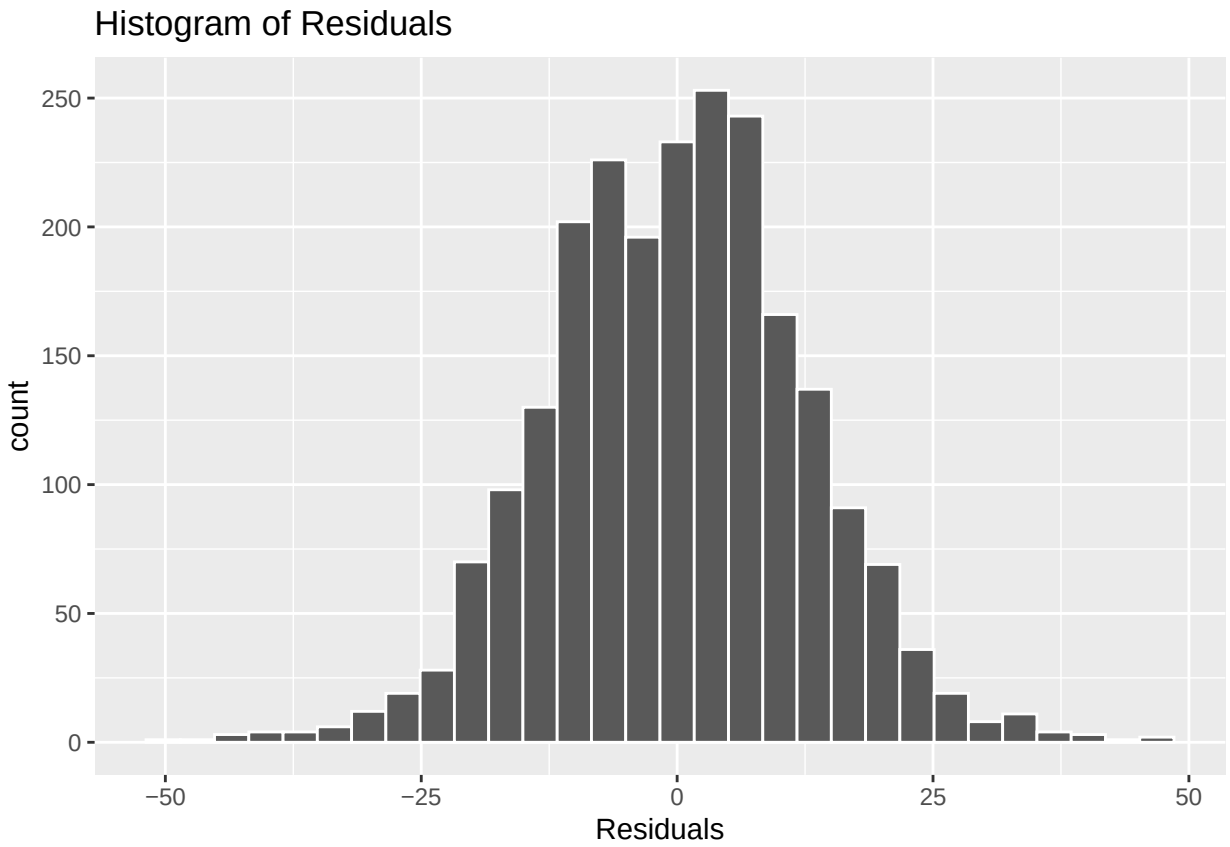
This model we work with stepwise selection both (forward and backward). We used the imputed data and comparing with model 6.

```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_H + TEAM_FIELDING_E +
##     TEAM_BASERUN_SB + TEAM_FIELDING_DP + TEAM_BATTING_SO + TEAM_PITCHING_H +
##     TEAM_BATTING_HR + TEAM_BATTING_BB + TEAM_BATTING_HBP + TEAM_PITCHING_SO +
##     TEAM_BATTING_2B + TEAM_PITCHING_BB + TEAM_BATTING_3B + TEAM_BASERUN_CS,
##     data = imputed_train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -50.301  -8.447   0.274   8.014  46.897
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      31.9812180  5.2891103   6.047 1.73e-09 ***
## TEAM_BATTING_H    0.0431774  0.0035344  12.216  < 2e-16 ***
```

```
## TEAM_FIELDING_E  -0.0420179  0.0026389 -15.923  < 2e-16 ***
## TEAM_BASERUN_SB   0.0599939  0.0052921  11.337  < 2e-16 ***
## TEAM_FIELDING_DP -0.1218711  0.0126791  -9.612  < 2e-16 ***
## TEAM_BATTING_SO  -0.0161301  0.0024397  -6.612 4.73e-11 ***
## TEAM_PITCHING_H   0.0014640  0.0003798   3.855 0.000119 ***
## TEAM_BATTING_HR   0.0783506  0.0093914   8.343  < 2e-16 ***
## TEAM_BATTING_BB   0.0176960  0.0051608   3.429 0.000617 ***
## TEAM_BATTING_HBP  0.0645168  0.0251498   2.565 0.010373 *
## TEAM_PITCHING_SO  0.0029988  0.0008534   3.514 0.000450 ***
## TEAM_BATTING_2B  -0.0219183  0.0088245  -2.484 0.013071 *
## TEAM_PITCHING_BB -0.0074438  0.0034718  -2.144 0.032135 *
## TEAM_BATTING_3B   0.0327671  0.0162047   2.022 0.043286 *
## TEAM_BASERUN_CS  -0.0187900  0.0105544  -1.780 0.075161 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.53 on 2261 degrees of freedom
## Multiple R-squared:  0.3715, Adjusted R-squared:  0.3676
## F-statistic: 95.45 on 14 and 2261 DF,  p-value: < 2.2e-16
```




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Definitely an improvement due to an increase in both the Adj R-Squared and F-statistic compared to models 3 and 6. The plots follow some assumptions of the model such as normality.

Part iv. SELECT MODELS

After running 7 different models, we've decided to use **Model 7**. The approach taken for this model was that of stepwise which took steps in adding a variable, and then evaluated each of them to determine their significance to the model. The data selected was that of a cleaned up imputed data set. Our primary criteria for selecting the this model was comparing the Adjusted R-Squared values and F-Statistics. While looking at each of the models Adjusted R-Squared values, we noticed that Models 1 and 6 had the hightest values of 0.5116 and 0.5167 respectively. Additionally, looking at there F-Statistic values, Model 1 had an F-Statistic of 14.27, and Model 6 had an F-Statistic of 30.02. While the Adjusted R-Squared values were the highest, we opted not to select these models because their F-Statistic values were too low, most of the observations were removed due to missingness and the data set for the model was that of the raw data.

Being left with Models 2, 3, 4, 5, and 7, we can loosely look at Model 2, which used a filtered data set, and eliminate it for it had the lowest Adjusted R-Squared value of 0.3014 and a low F-Statistic value of 15.63. Model 3 was one of the favorite models because it had a higher Adjusted R-Squared value of 0.3674 and F-Statistic of 89.1. The method taken here was apply multiple imputations in our training data set and hope to see better results. This held true, but we took additional measures in Model 4 to improve Model 3. We wanted to remove the multicollinearity that was in Model 3, so we evaluate the Variance Inflation Factor (VIF) for each variable in the model. If the VIF was above 5, we could say that the variable had a correlation with another variable. Model 4 removed 3 variables (TEAM_BATTING_BB, TEAM_BATTING_SO and TEAM_BATTING_HR), as they were all dependent on other predictor variables. While doing this, we did not see an improvement in our Adjusted R-Squared value, 0.351, but did see an increase in our F-Statistic, now above 100, at 103.5. Model 5 used the transformed data and also removed the variables associated with

multicollinearity, and again, we saw a lower Adjusted R-Squared value of 0.3166 and a lower F-Statistic of 88.83.

Comparing all of our models to Model 7, we saw the greatest Adjusted R-Squared value of 0.3676 and the second highest F-Statistic of 95.45. We valued the Adjusted R-Squared value for these models more than the F-Statistic, even though the F-Statistic was important.

```
## Warning: Number of logged events: 50
```

Let's see the predicted wins for the teams based on the the model we chose.

```
##         fit      lwr       upr
## 1  61.53147 36.90491  86.15802
## 2  66.05619 41.41377  90.69862
## 3  73.16340 48.54653  97.78028
## 4  86.78046 62.16465 111.39628
## 5  59.57531 34.89319  84.25743
## 6  67.46261 42.79323  92.13198
## 7  81.16960 56.50145 105.83776
## 8  77.77784 53.14352 102.41217
## 9  70.30781 45.69187  94.92376
## 10 74.50157 49.89114  99.11200
## 11 68.97133 44.34881  93.59385
## 12 82.56822 57.95149 107.18494
## 13 81.75140 57.12043 106.38236
## 14 84.28518 59.65652 108.91385
## 15 86.58895 61.95208 111.22583
## 16 78.86785 54.22822 103.50747
## 17 73.81928 49.22330  98.41526
## 18 77.88048 53.29065 102.47032
## 19 72.68869 48.05343  97.32395
## 20 90.02777 65.40021 114.65533
## 21 80.81645 56.19392 105.43897
## 22 82.56889 57.95040 107.18737
## 23 77.98142 53.36607 102.59677
## 24 71.32904 46.73034  95.92775
## 25 82.52217 57.92136 107.12298
## 26 89.92955 65.31472 114.54439
## 27 63.30306 38.16660  88.43951
## 28 74.91196 50.29321  99.53071
## 29 82.26683 57.61431 106.91935
## 30 74.85095 50.20168  99.50022
```

Appendix

Rcode: Github

Predicted Wins: Github

HTML View: nbviewer

PDF: nbviewer