DATA 621 - Homework 1

Cheryl Bowersox, Christopher Martin, Robert Sellers, Edwige Talla Badjio

June 19, 2016

```
## Warning: package 'gridExtra' was built under R version 3.2.5
```

```
## Warning: package 'lattice' was built under R version 3.2.5
```

```
## Warning: package 'reshape2' was built under R version 3.2.5
```

1 OBJECTIVE

To build an optimal multiple linear regression model for annual number of wins per team based on predictor values. The data is an annual record of baseball wins per team along with their respective explanatory statistics. The following paper outlines a procedure for the creation of three experimental regression models and explain the selection process of an optimal method.

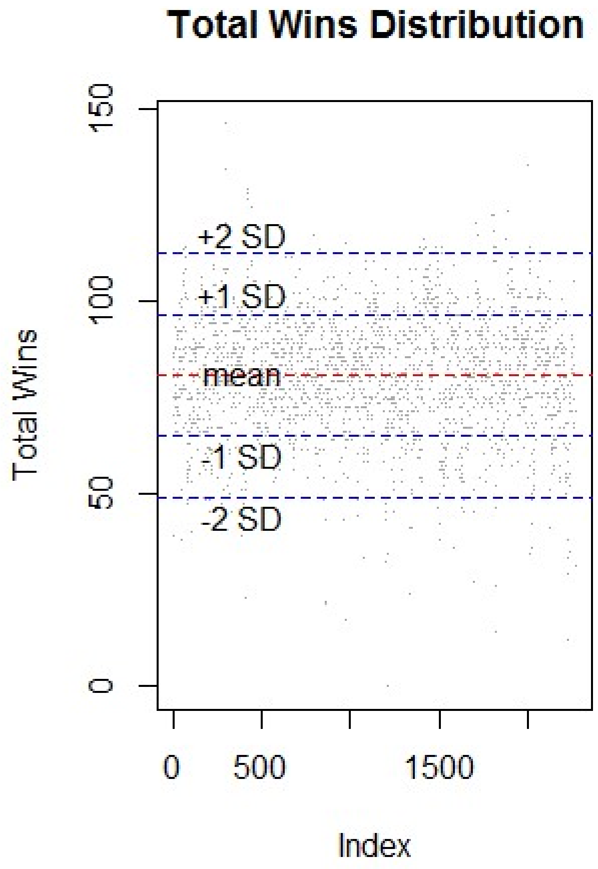
2 DATA EXPLORATION

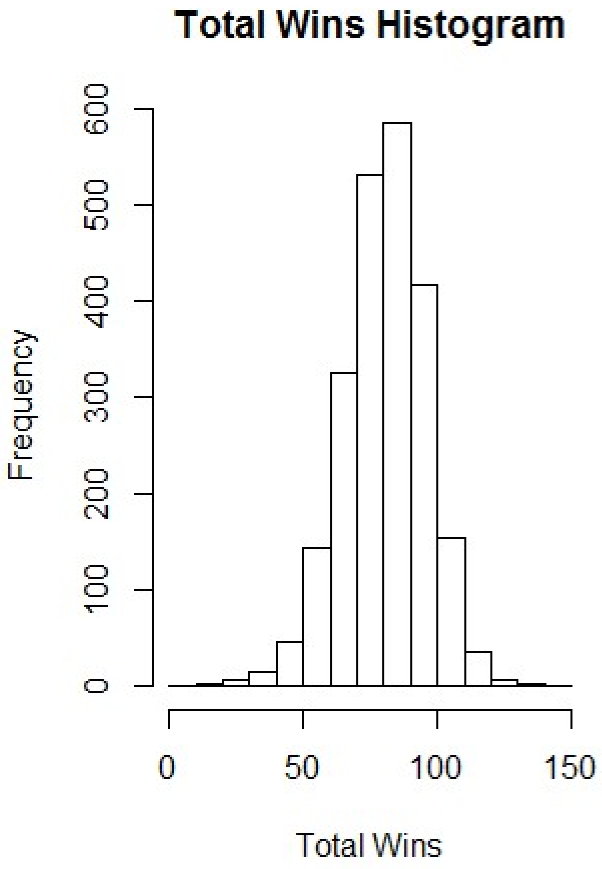
2.1 Summary statistics

The data has 2276 rows, with each record representing 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.

The data has 15 attributes, of which 14 represent our predictor variables comprised of annual hitting, pitching, baserunning, and fielding statistics. Additional predictor variables are created and this is explained in section 3.

The response variable is the number of wins, or TARGET_WINS, which has a mean of 80.8 and standard deviation of 15.8. It follows a relatively normal, uniform distribution and we will not treat any values as outliers.





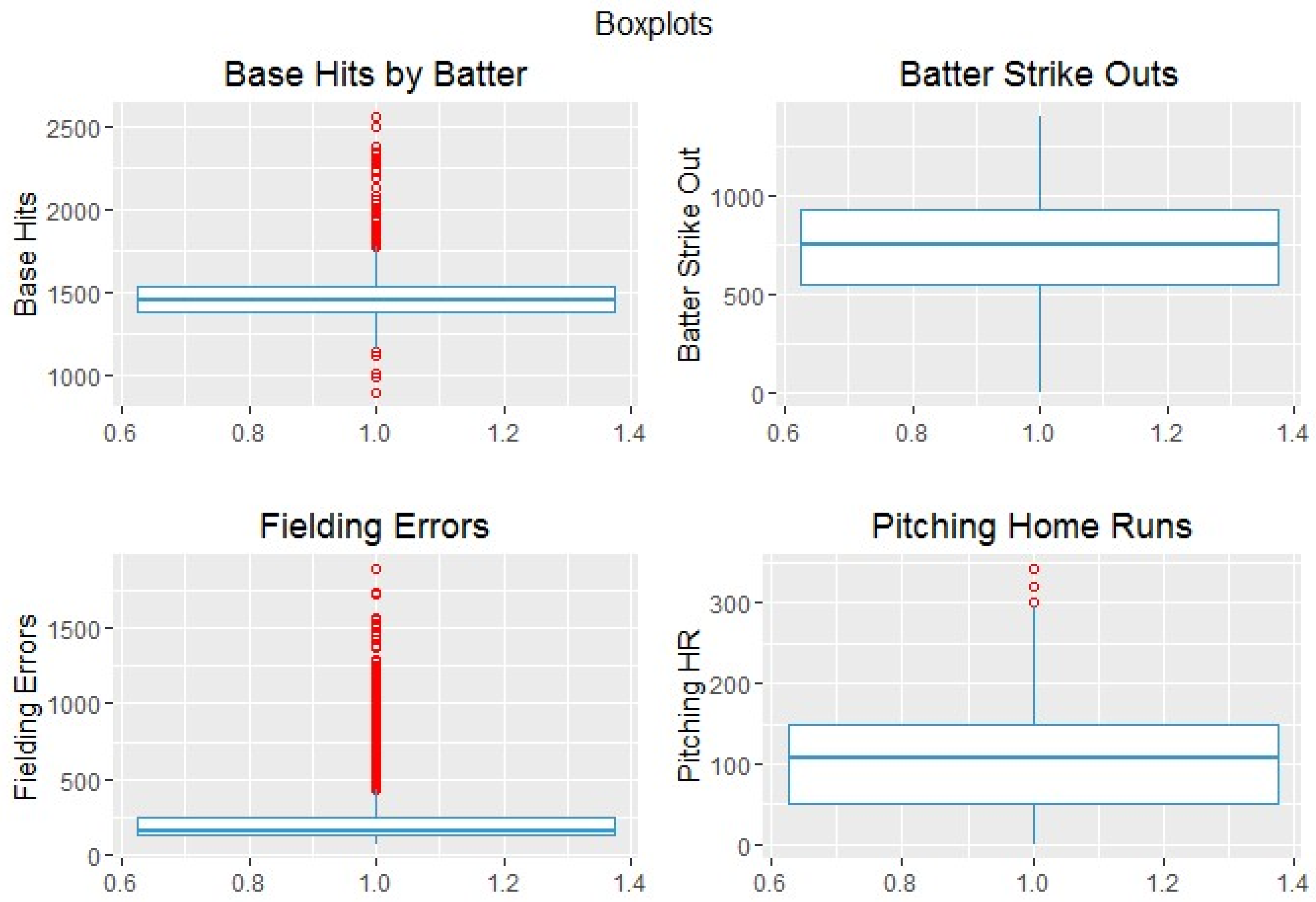
Predictive variables are much more diverse in shape and distribution (see graphical series below). The explanatory/predictor variables are the remaining variables that may have a positive or negative impact on the number of wins. ##Correlations in the data Correlations between variables were calculated, with particular interest to those > 5. One interesting result is the TARGET_WINS variable is not strongly correlated with the data, but the the pitching and batting home runs are very strongly related.

```
##                 Var1             Var2     value
## 1        TARGET_WINS      TARGET_WINS 1.0000000
## 2     TEAM_BATTING_H   TEAM_BATTING_H 1.0000000
## 3    TEAM_BATTING_2B  TEAM_BATTING_2B 1.0000000
## 4    TEAM_BATTING_3B  TEAM_BATTING_3B 1.0000000
## 5    TEAM_BATTING_HR  TEAM_BATTING_HR 1.0000000
## 6    TEAM_BATTING_BB  TEAM_BATTING_BB 1.0000000
## 7    TEAM_BATTING_SO  TEAM_BATTING_SO 1.0000000
## 8    TEAM_BASERUN_SB  TEAM_BASERUN_SB 1.0000000
## 9    TEAM_BASERUN_CS  TEAM_BASERUN_CS 1.0000000
## 10  TEAM_BATTING_HBP TEAM_BATTING_HBP 1.0000000
## 11   TEAM_PITCHING_H  TEAM_PITCHING_H 1.0000000
## 12  TEAM_PITCHING_HR TEAM_PITCHING_HR 1.0000000
## 13  TEAM_PITCHING_BB TEAM_PITCHING_BB 1.0000000
## 14  TEAM_PITCHING_SO TEAM_PITCHING_SO 1.0000000
## 15   TEAM_FIELDING_E  TEAM_FIELDING_E 1.0000000
## 16  TEAM_FIELDING_DP TEAM_FIELDING_DP 1.0000000
## 17  TEAM_PITCHING_HR  TEAM_BATTING_HR 0.9693714
## 18   TEAM_BATTING_HR TEAM_PITCHING_HR 0.9693714
## 19   TEAM_FIELDING_E  TEAM_PITCHING_H 0.6677590
## 20   TEAM_PITCHING_H  TEAM_FIELDING_E 0.6677590
## 21   TEAM_BATTING_2B   TEAM_BATTING_H 0.5628497
## 22    TEAM_BATTING_H  TEAM_BATTING_2B 0.5628497
## 23   TEAM_BATTING_BB  TEAM_BATTING_HR 0.5137348
## 24   TEAM_BATTING_HR  TEAM_BATTING_BB 0.5137348
## 25   TEAM_FIELDING_E  TEAM_BATTING_3B 0.5097784
## 26   TEAM_BATTING_3B  TEAM_FIELDING_E 0.5097784
```

2.2 Data Visualization

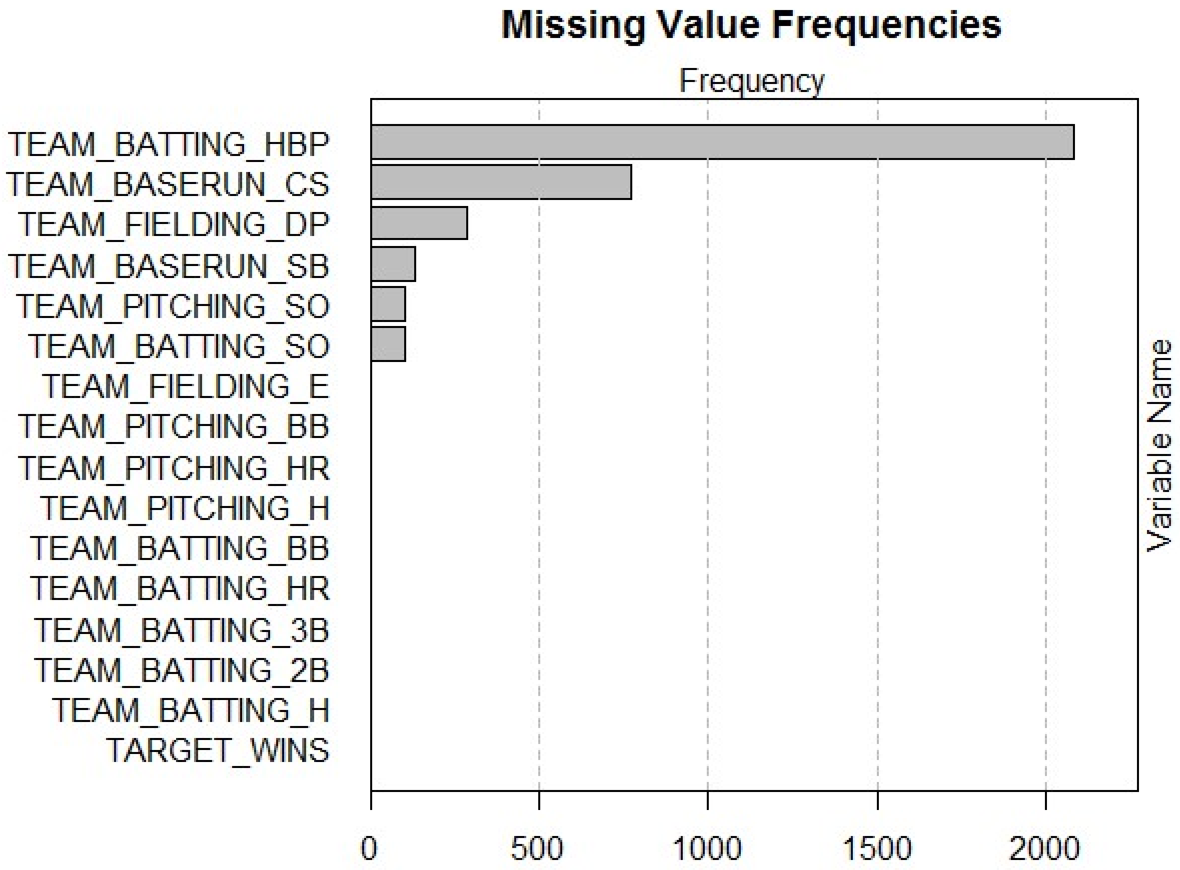
Some predictor variables were chosen to evaluate the outliers and distribution of the data. The high number of 'outliers' may indicate a need for data transformations. In this case the base hits by batter and the field errors have a large amount of variability in the data.

```
## Warning: Removed 102 rows containing non-finite values (stat_boxplot).
```



2.3 Missing Values

Present in the data were some concerns, namely with missing data. In its entirety there are 3478 missing fields. This amounts to 9.55% of the data available. The following graphic shows a high completeness disparity between our values. This will be discussed in greater detail and resolved in section 3 of this document.



3 DATA PREPARATION

For columns with values for HBP (hit by pitch) and DP (double plays), the minimum values are >0 and there are a large number of missing values (10% of DP and 92% of HBP). While many of these values could very well be missing, it is unreasonable to believe that there are 0 values. To simplify this process, a value of 0 has been given to these missing values for the purposes of calculation for new columns, then will be removed with other columns containing missing values (CS, SB, SO) before the model is run.

3.1 Columns added

Additionally, since (in baseball) the attacking team is the team whose turn it is at bat it might be interesting to see their success rates for those occurrences where a successful hit-at-bat occurs:

- TEAM_BATTING_1B was deduced from TEAM_BATTING_H, as the number of singles, calculated through subtracting TEAM_BATTING_3B, TEAM_BATTING_HR, and TEAM_BATTING_2B from TEAM_BATTING_H.
- TOTAL_BASES was added to show the produce of total bases achieved after successfully reaching base, summed together per annual results per team. For instance a triple (TEAM_BATTING_3B) is 3 bases and a walk (TEAM_BATTING_BB) is 1 base, while caught stealing (TEAM_BASERUN_CS) will detract a base.
- TEAM_BATTING_HITSBASE was added to show positive attacking at bat situations (base hits +

homeruns + hit by pitch)

- TEAM_BATTING_ALLPOS was added to show positive attacking situations in batting and stolen bases

- TEAM_BATTING_ALLPOSATTCK was added to show the number of positive attacking situations

The defending team is mostly effective with the pitcher, so some added variables for pitching could be interesting:

- TEAM_PITCHING_BADPITCH was added for negative base pitching situations: hits off a pitch + walks allowed
- TEAM_PITCHING_BASESGIVEN was added for negative pitching situations: hits off a pitch + walks allowed + HRs given
- TEAM_PITCHRATIO was as a relationship between negative pitching outcomes (HR) and positive pitching outcomes (SO)

Other columns were added to see a form of success ratio:

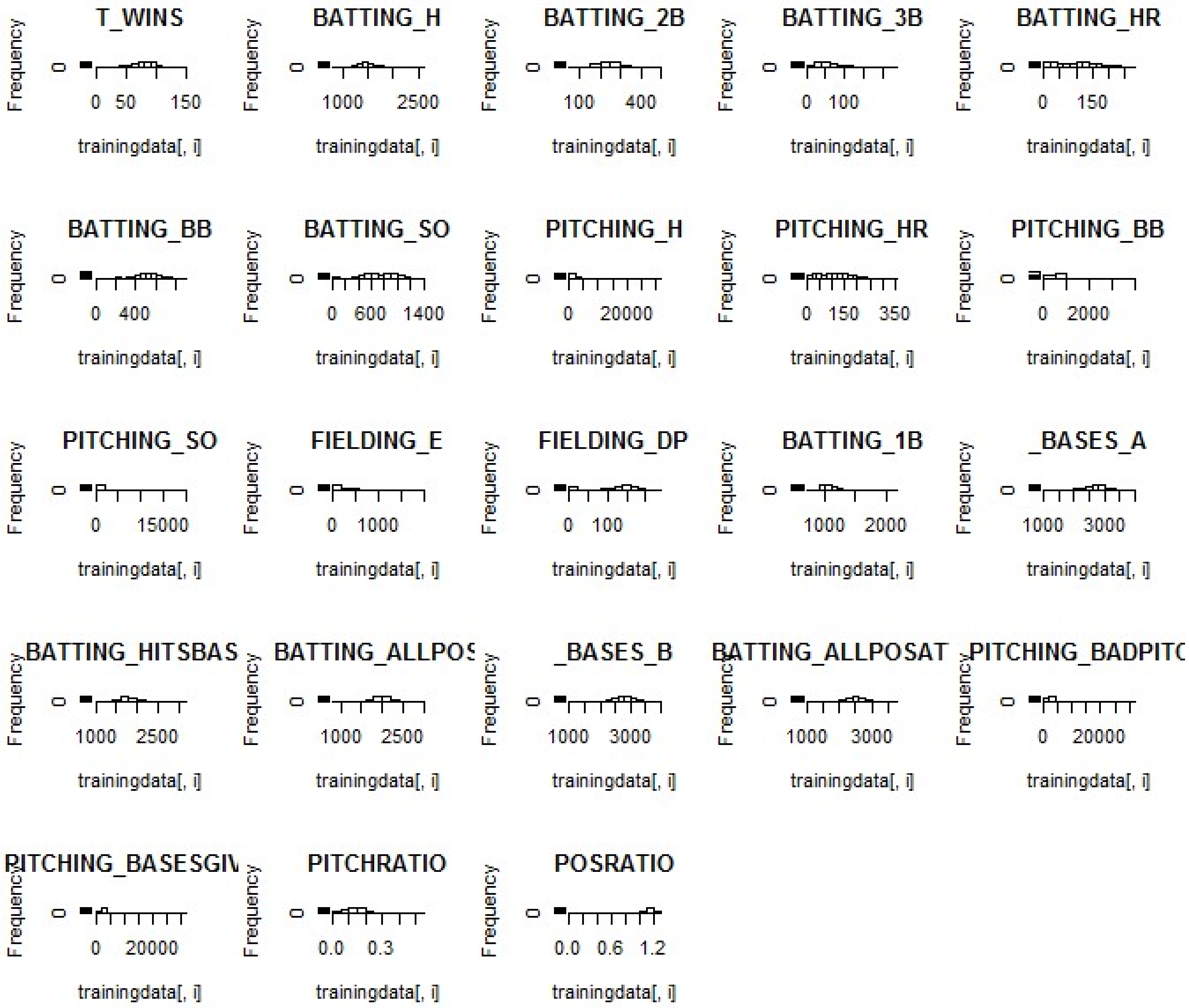
- TEAM_POSRATIO shows the success ratio: successes in attacking (batting) vs failures in pitching
- SBPERCENT was added for the stolen base percentage: bases stolen vs bases caught stealing (removed from data set)

3.2 Columns removed

Some columns have been removed from the analysis: TEAM_BASERUN_SB, TEAM_BASERUN_CS, and TEAM_BATTING_HBP. There are a significant number of NA's (90% of the values are missing) in the HBP column and applying a zero value to this figure is misleading. By including the values that are present in our newly created columns, we can preserve the values of gaining a base on an attacking play by being hit by a ball without the noise. A similar methodology and thought process was used for SB and CS

3.3 Diagnostic plots

Diagnostic plots of the variables were used to discover if any variables needed a mathematical transformation (i.e. log or square roots):


T_WINS
BATTING_H
BATTING_2B
BATTING_3B
BATTING_HR
BATTING_BB
BATTING_SO
PITCHING_H
PITCHING_HR
PITCHING_BB
PITCHING_SO
FIELDING_E
FIELDING_DP
BATTING_1B
_BASES_A
BATTING_HITSBAS
BATTING_ALLPOS
_BASES_B
BATTING_ALLPOSAT
PITCHING_BADPITC
PITCHING_BASESGIV
PITCHRATIO
POSRATIO
Frequency
0
trainingdata[, i]
0 50 150
1000 2500
100 400
0 100
0 150
0 400
0 600 1400
0 20000
0 150 350
0 2000
0 15000
0 1000
0 100
1000 2000
1000 3000
1000 2500
1000 2500
1000 3000
1000 3000
0 20000
0 20000
0.0 0.3
0.0 0.6 1.2

3.4 Mathematical transformations

Skewed variables needed to be adjusted, so the log form of the following variables was used to normalize the data:

- TEAM_BATTING_H
- TEAM_BATTING_3B
- TEAM_PITCHING_BB
- TEAM_FIELDING_E
- TEAM_BATTING_HITSBASE
- TEAM_BATTING_1B

The following variables are skewed but were not adjusted: - TEAM_PITCHING_H #PITCHING_H still has too many high values - TEAM_PITCHING_BADPITCH #high values skew - TEAM_PITCHING_BASESGIVEN #high values skew - TEAM_POSRATIO #low values skew

These fields need to be fixed: - TEAM_FIELDING_DP (0 values skew) - TOTAL_BASES (0 values are skewing– should remove 0's) - SBPERFECT (0 values are skewing)

Remaining 'na' values were omited from the data set before models were created.

4 BUILD MODELS

4.1 Forward Stepwise Method

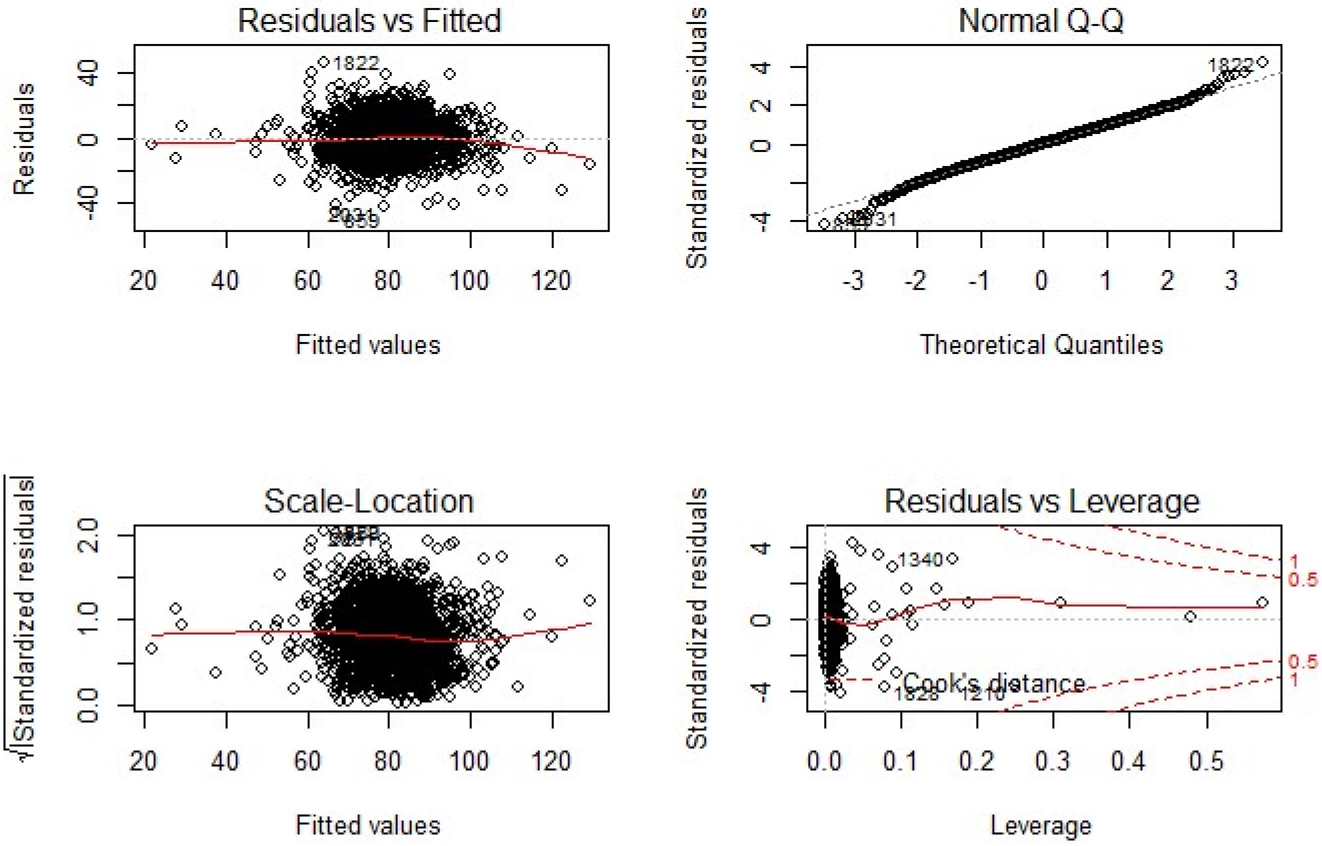
As a measure of automation accuracy and a test against more traditional models, a stepwise approach was performed using a forward AIC (Akaike information criterion) method.

Current optimal results from stepwise method with diagnostic plots

```
#stepResults
sumary(stepResults)
```

```
##                          Estimate  Std. Error  t value  Pr(>|t|)
## (Intercept)            300.9533710  48.3434790   6.2253 5.825e-10
## TOTAL_BASES_B            0.0617399   0.0048655  12.6894 < 2.2e-16
## TEAM_PITCHING_SO        -0.0083047   0.0027682  -3.0000 0.0027323
## TEAM_FIELDING_E        -23.4927159   1.2899569 -18.2120 < 2.2e-16
## TEAM_BATTING_2B         -0.0933401   0.0100098  -9.3249 < 2.2e-16
## TEAM_BATTING_HR         -0.1768531   0.0209915  -8.4250 < 2.2e-16
## TEAM_BATTING_BB          0.0341314   0.0112872   3.0239 0.0025266
## TEAM_PITCHRATIO         41.4345456  12.2679678   3.3775 0.0007454
## TEAM_FIELDING_DP        -0.0408027   0.0095856  -4.2566 2.170e-05
## TEAM_BATTING_ALLPOS     -0.0332843   0.0069695  -4.7757 1.919e-06
## SBPERCENT                6.5380272   1.8148520   3.6025 0.0003228
## TEAM_POSRATIO          -22.1353822   5.7268715  -3.8652 0.0001145
## TEAM_PITCHING_BB       -26.8874857   7.8596741  -3.4209 0.0006364
## TEAM_PITCHING_BADPITCH   0.0118748   0.0075544   1.5719 0.1161273
## TEAM_BATTING_3B          1.8213870   1.0330535   1.7631 0.0780327
## TEAM_PITCHING_H         -0.0107521   0.0076242  -1.4103 0.1586149
##
## n = 2042, p = 16, Residual SE = 11.27498, R-Squared = 0.39
```

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

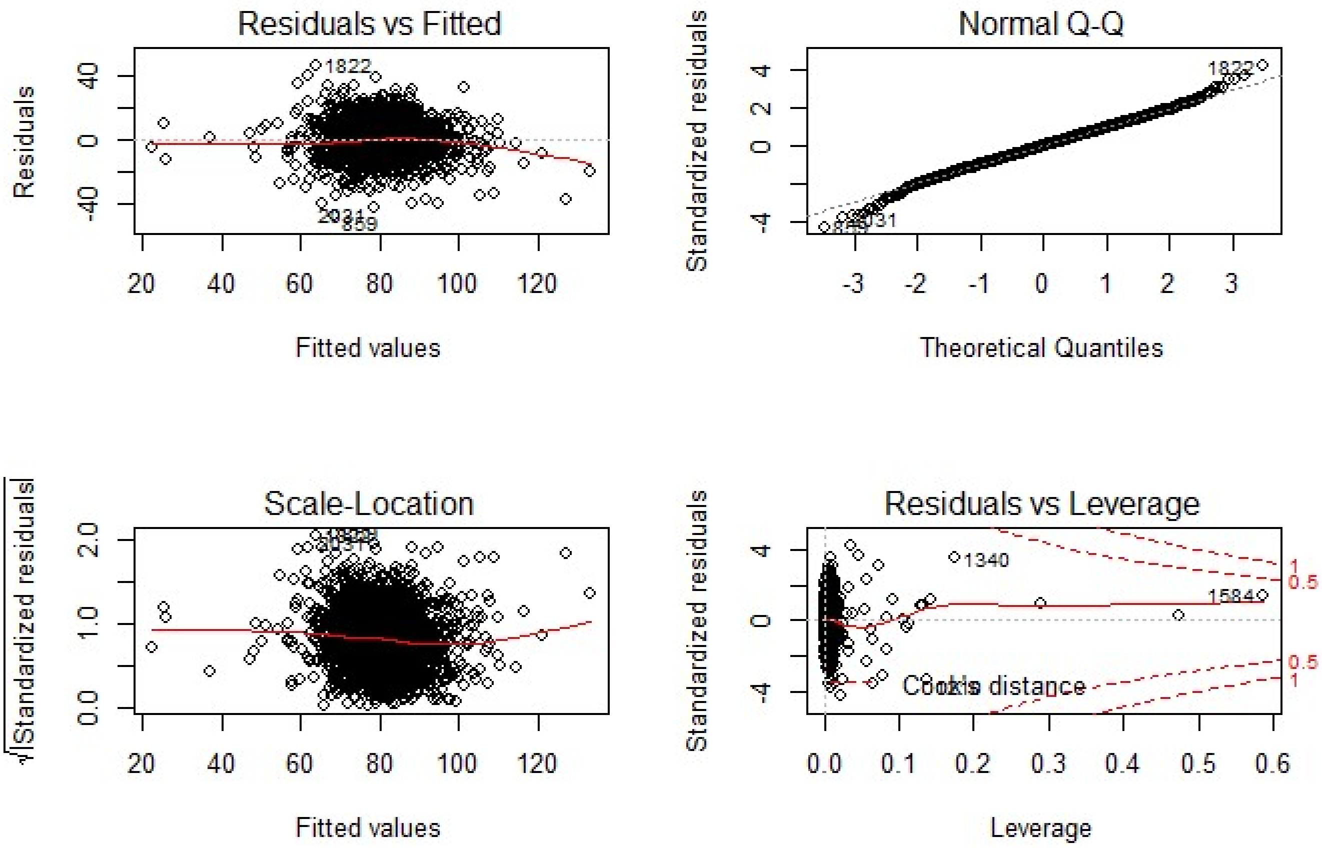


Discussion of coefficients: In the stepwise generated model many variables were selected that are linearly related to each other. The first 12 variables are statistically significant despite the multicollinearity. An argument can be made for the final 3 that they are not highly significant and could be removed. The relationships between the predictor variables result in some surprising coefficients that are difficult to directly interpret. For example The Total Bases variable has an expected positive impact on score, but the number of 2nd bases has a negative impact. Another example is the positive coefficient for the variable Pitch Ratio - this variable measures bad/good pitching outcomes, and a higher amount means a greater number of 'bad' outcomes (home runs allowed). The positive relationship however is probably related to the strange fact that pitching home runs allowed is highly, positively, related to the number of batting home runs - a counterintuitive relationship in the raw data.

4.2 Backward Elimination Model

Using backward elimination, and removing high p-value variables to get the highest adjusted R^2, the final regression model is:

```
##                              Estimate  Std. Error  t value  Pr(>|t|)
## (Intercept)                8.0021e+02  1.0418e+02   7.6811 2.438e-14
## TEAM_BATTING_H             8.7263e+01  1.9424e+01   4.4926 7.431e-06
## TEAM_BATTING_3B            4.6571e+00  1.3191e+00   3.5304 0.0004242
## TEAM_BATTING_HR           -2.3127e-01  3.4381e-02  -6.7268 2.250e-11
## TEAM_PITCHING_H           -1.2302e-02  7.4217e-03  -1.6576 0.0975558
## TEAM_PITCHING_BB          -2.6626e+01  5.9743e+00  -4.4568 8.773e-06
## TEAM_PITCHING_SO          -1.0034e-02  2.8204e-03  -3.5575 0.0003830
## TEAM_FIELDING_E           -2.3383e+01  1.2864e+00 -18.1764 < 2.2e-16
## TEAM_FIELDING_DP          -4.1636e-02  9.6607e-03  -4.3098 1.712e-05
## TEAM_BATTING_HITSBASE     -1.6331e+02  1.8108e+01  -9.0184 < 2.2e-16
## TOTAL_BASES_B              9.8376e-02  1.6043e-02   6.1322 1.039e-09
## TEAM_BATTING_ALLPOSATTCK -3.7203e-02  1.6391e-02  -2.2697 0.0233317
## TEAM_PITCHING_BADPITCH     1.3279e-02  7.3068e-03   1.8174 0.0693041
## TEAM_PITCHRATIO            2.8194e+01  1.2390e+01   2.2755 0.0229800
## TEAM_POSRATIO             -2.1759e+01  5.5920e+00  -3.8911 0.0001030
## SBPERCENT                  7.1149e+00  1.7895e+00   3.9759 7.257e-05
##
## n = 2042, p = 16, Residual SE = 11.26632, R-Squared = 0.39
```



Discussion of coefficients All remaining variables after elimination are statistically significant to the model. The overall R^2 decreased by 1 point, with the elimination of 7 variables. TOTAL_BASES: The final model indicates a positive impact on total wins of .09 per-bases-gained in batting situations because of the relationship between the TOTAL_BASES variables and other base-gain variables such as TEAM_BATTING_HR. And TEAM_BATTING_HITSBAS and TEAM_BATTING_ALLPOSATTCK we see unexpected negative coefficients in those areas. FIELDING_DP: This is a unexpected negative impact on total wins of .004 per double play.

4.3 Simple Model

The 'simple' model was designed to include a limited number of input variables using a more intuitive approach to the selection as a comparison to the backwards elimination or stepwise approach. This model was created by examining the correlation between variables, choosing those that have high relationships with the target variable and lower relationships with each other. There are 3 general types of variables to choose from - batting, pitching and fielding. A single variable was chosen from each category to represent these factors.

Variables chosen: TOTAL_BASES - a linear combination of the batting variables, weighted by base number lending greater significance to higher bases. TEAM_FIELDING_E - the log of the # of fielding errors. TEAM_PITCHRATIO - This is the ratio of the number of home runs allowed versus the number of strikeouts allowed

Model Output

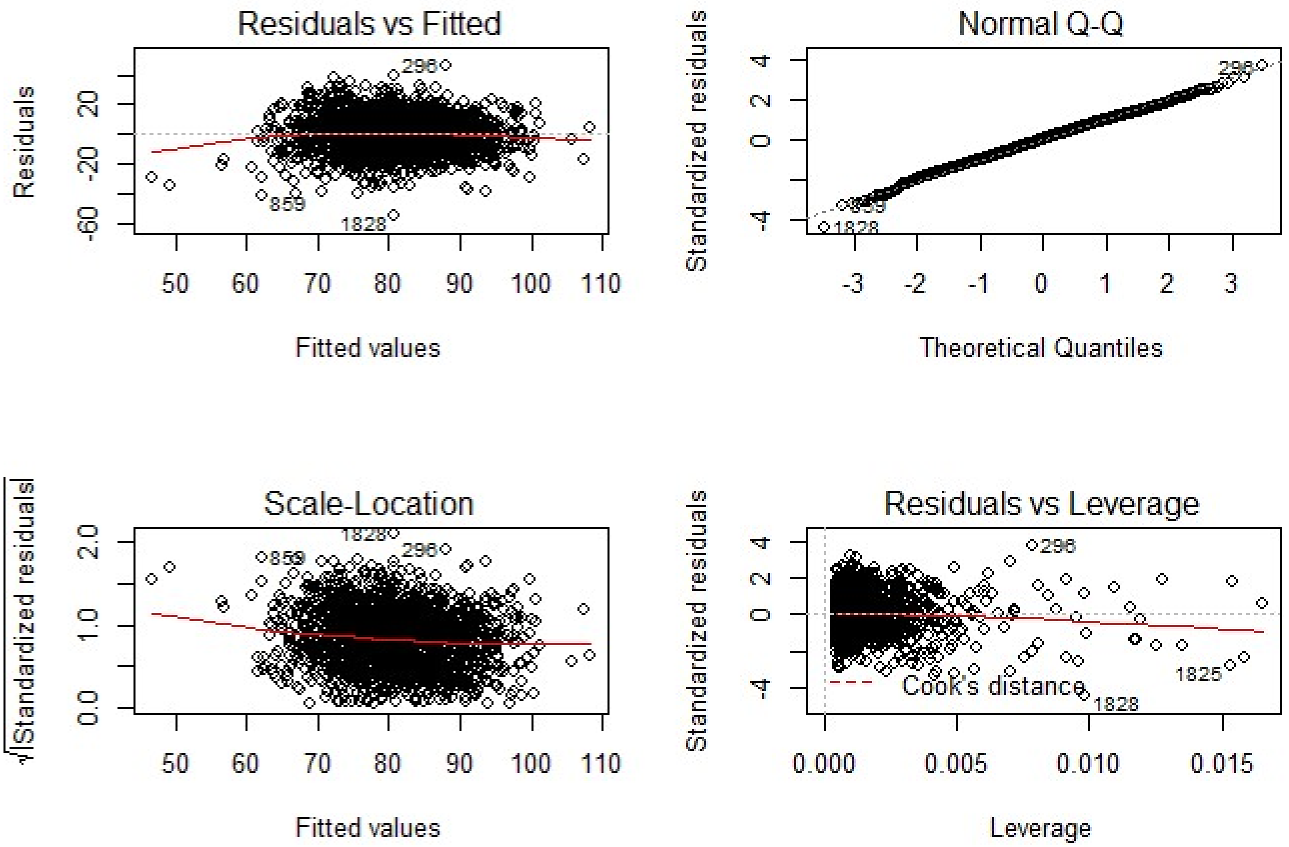
```
##                    Estimate Std. Error t value  Pr(>|t|)
## (Intercept)      26.3600281  4.2072616  6.2654 4.525e-10
## TOTAL_BASES_B     0.0248483  0.0012942 19.2004 < 2.2e-16
## TEAM_FIELDING_E -2.8387979  0.5977432 -4.7492 2.185e-06
## TEAM_PITCHRATIO -2.1453440  6.4311239 -0.3336    0.7387
##
## n = 2042, p = 4, Residual SE = 12.48170, R-Squared = 0.25
```

Coefficients Two out of three of the coefficients selected for the model were statistically significant. TOTAL_BASES: Wins increase by .0250 for each base per batter. TOTAL_BASES is a linear combination of all bases gained, weighted for the base number. Because of the high values (mean of 2814 and standard deviation of 257) the impact of TOTAL_BASES on the number of wins is not negligible. TEAM_FIELDING_E: Wins are decreased as the number fielding errors increase. Because our variable is the the log of the errors it is not a linear relationship, but rather -2.702 for each log of the field errors. TEAM_PITCHRATIO: This variable was not helpful in explaining the model, having a high p-value and was removed from the final version shown below.

Model was re-evaluated excluding the PitchRatio variable, which slightly alters the coefficients.

Revised Model Output

```
##                     Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)      26.41700087  4.20287748  6.2855 3.986e-10
## TOTAL_BASES_B     0.02457139  0.00099255 24.7558 < 2.2e-16
## TEAM_FIELDING_E -2.75847597  0.54698313 -5.0431 4.986e-07
##
## n = 2042, p = 3, Residual SE = 12.47898, R-Squared = 0.25
```



Equation with transformed variables: Wins = .0246* TOTAL_BASES - 2.7584 * TEAM_FIELDING_E

Equation with original variables:

Wins = .0246 * (TEAM_BATTING_BB + TEAM_BATTING_1B + 2* TEAM_BATTING_2B + 3TEAM_BATTING_3B+ 4 TEAM_BATTING_HR + BASERUN_SB) -2.7584* log(TEAM_FIELDING_E)

5 SELECT MODELS

A comparison of the stepwise, elimination, and simple models included evaluating the R^2 values, the Aikake Information Criterion (AIC), and the Bayesian Information Criterion (BIC).

5.1 Model selection with the adjusted R squared

Assuming the model with the highest adjusted R squared is the “best” model.

The Adjusted R squared for each model is 0.3872123, 0.3881538, 0.2493513. Based on this criteria, the first model, Stepwise, should be selected as the best model. Let’s see if this result can be confirmed by another model selection method. We will focus in the next section on Aikake Information Criterion.

5.2 Model selection with Aikake Information Criterion (AIC)

The idea here is that the model with the smallest AIC value is the "best". We will be interested in the difference of AIC between fitted models.

The AIC for each model is given by 1.570672410^{4}, 1.570358410^{4}, 1.610814610^{4}.
According to the AIC criterion, the first model also has the smallest AIC. Based on this result, we can use the stepwise as our best model. We can use the BIC double check the results with the BIC criterion.

5.3 Model selection with Bayesian Information Criterion (BIC)

BIC and AIC work the same. The model with the smallest BIC is the "best" in the set of models fitted.

The BIC for each model is given by 1.580229210^{4}, 1.579915310^{4}, 1.613063210^{4} The BIC criterion is also showing the Stepwise model as the one with the smallest BIC difference among fitted model.

5.4 Predicting TARGET_WINS using the evaluation data

5.4.1 Data preparation

Evaluation data was treated to the same variable transformations and handeling of 'na' values as the original training data set.

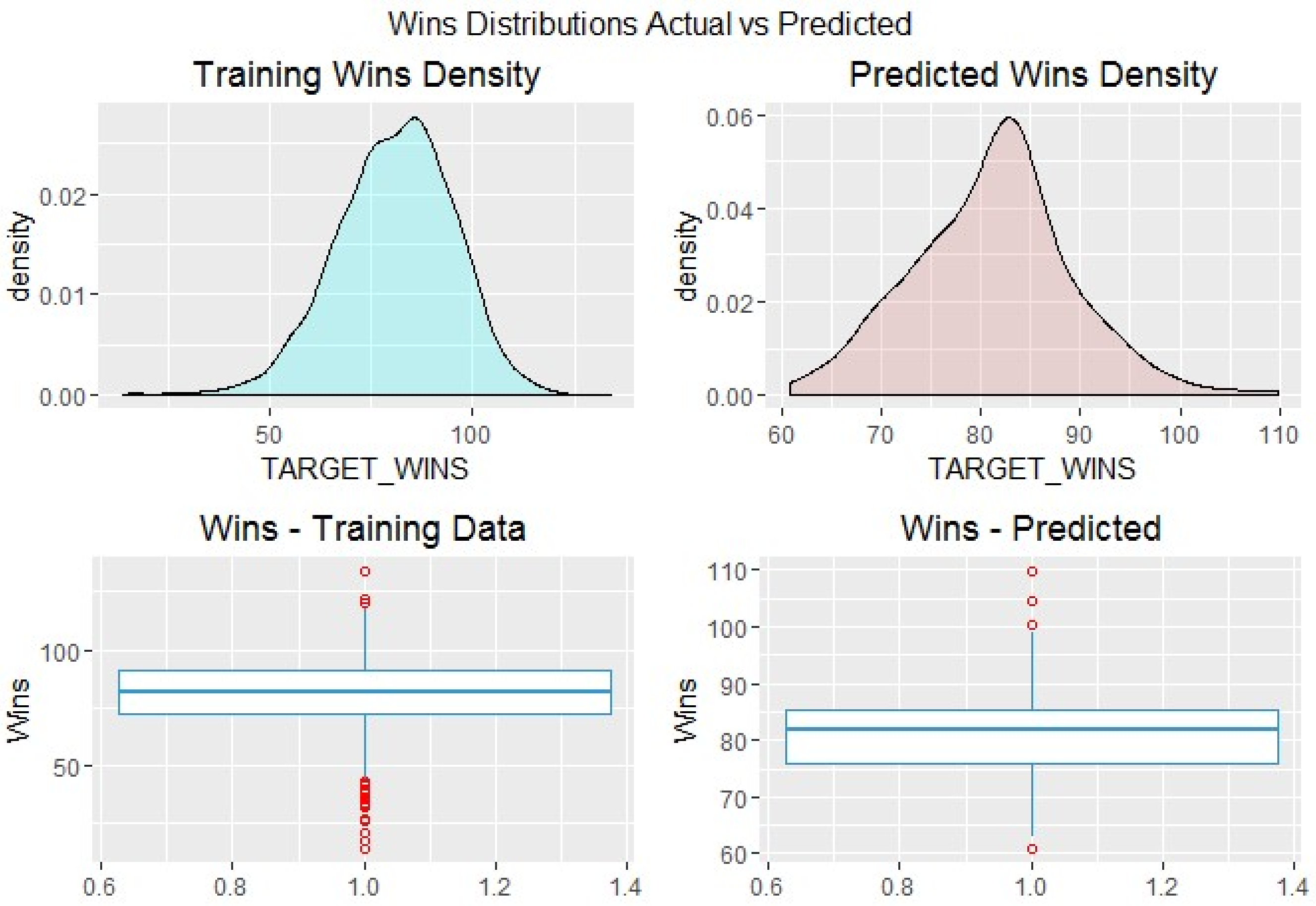
```
evalData<-evaluationData
#applying same logic used for training data to deal w/ na's
evalData$TEAM_BATTING_HBP[is.na(evalData$TEAM_BATTING_HBP)] <- 0
evalData$TEAM_FIELDING_DP[is.na(evalData$TEAM_FIELDING_DP)] <- 0

evalData$TEAM_BATTING_1B<-evalData$TEAM_BATTING_H- evalData$TEAM_BATTING_2B- evalData
$TEAM_BATTING_3B- evalData$TEAM_BATTING_HR

temp1B<-evalData$TEAM_BATTING_1B;temp1B[is.na(temp1B)]<-0
temp2B<-evalData$TEAM_BATTING_2B*2;temp2B[is.na(temp2B)]<-0
temp3B<-evalData$TEAM_BATTING_3B*3;temp3B[is.na(temp3B)]<-0
tempHR<-evalData$TEAM_BATTING_HR*4;tempHR[is.na(tempHR)]<-0
tempSB<-as.numeric(evalData$TEAM_BASERUN_SB);tempSB[is.na(tempSB)]<-0
tempCS<-evalData$TEAM_BASERUN_CS*-1;tempCS[is.na(tempCS)]<-0
tempBB<-as.numeric(evalData$TEAM_BATTING_BB);tempBB[is.na(tempBB)]<-0
tempHBP<-as.numeric(evalData$TEAM_BATTING_HBP);tempHBP[is.na(tempHBP)]<-0
evalData$TOTAL_BASES_A<-temp1B+temp2B+temp3B+tempHR+tempSB+tempCS+tempBB+tempHBP

evalData$TOTAL_BASES_B<-(evalData$TEAM_BATTING_2B*2)+(evalData$TEAM_BATTING_3B*3)+(ev
alData$TEAM_BATTING_HR*4)+ evalData$TEAM_BASERUN_SB+evalData$TEAM_BATTING_BB+evalData
$TEAM_BATTING_1B

evalData$TEAM_BATTING_HITSBASE <- evalData$TEAM_BATTING_H + evalData$TEAM_BATTING_2B
+ evalData$TEAM_BATTING_3B


evalData$TEAM_BATTING_ALLPOS <- evalData$TEAM_BATTING_H + evalData$TEAM_BATTING_BB +
evalData$TEAM_BATTING_HBP


evalData$TEAM_BATTING_ALLPOSATTCK <- evalData$TEAM_BATTING_H + evalData$TEAM_BATTING_
2B + evalData$TEAM_BATTING_3B + evalData$TEAM_BATTING_HR + evalData$TEAM_BATTING_BB +
 evalData$TEAM_BATTING_HBP + evalData$TEAM_BASERUN_SB

evalData$TEAM_PITCHING_BADPITCH <- evalData$TEAM_PITCHING_H + evalData$TEAM_PITCHING_
BB


evalData$TEAM_PITCHING_BASESGIVEN <- evalData$TEAM_PITCHING_H + evalData$TEAM_PITCHIN
G_BB + evalData$TEAM_PITCHING_HR

evalData$TEAM_PITCHRATIO <- evalData$TEAM_PITCHING_HR / evalData$TEAM_PITCHING_SO

evalData$TEAM_POSRATIO <- evalData$TEAM_BATTING_ALLPOSATTCK / evalData$TEAM_PITCHING_
BASESGIVEN

evalData$SBPERCENT <- evalData$TEAM_BASERUN_SB / (evalData$TEAM_BASERUN_SB + evalData
$TEAM_BASERUN_CS)

evalData$TEAM_BATTING_H <- log(evalData$TEAM_BATTING_H)
```

```
evalData$TEAM_BATTING_3B <- log(evalData$TEAM_BATTING_3B)
evalData$TEAM_PITCHING_BB <- log(evalData$TEAM_PITCHING_BB)
evalData$TEAM_FIELDING_E <- log(evalData$TEAM_FIELDING_E)
evalData$TEAM_BATTING_HITSBASE <- log(evalData$TEAM_BATTING_HITSBASE)
evalData$TEAM_BATTING_1B <- log(evalData$TEAM_BATTING_1B)

evalData<-evalData%>%select(-TEAM_BATTING_HBP)
evalData$TEAM_BASERUN_SB <- NULL
evalData$TEAM_BASERUN_CS <- NULL



# variables actually need for our model, creating subset so we don't worry about the
na's in other fields

evalDatasub <- evalData%>%select(TOTAL_BASES_B, TEAM_PITCHING_SO, TEAM_FIELDING_E, TE
AM_BATTING_2B, TEAM_BATTING_HR,TEAM_BATTING_BB, TEAM_PITCHRATIO,TEAM_FIELDING_DP,TEAM
_BATTING_ALLPOS,TEAM_POSRATIO,TEAM_PITCHING_BB,TEAM_PITCHING_BADPITCH,TEAM_PITCHING_H
,TEAM_BATTING_3B,SBPERCENT)



# 259 obs,31 are NA

evalDatasub<-na.omit(evalDatasub)
```

5.4.2 Prediction

```
evalDatasub$TARGET_WINS<-predict(best_model, newdata=evalDatasub, type='response')
```

We were able to use our selected model to predict Target Wins for the evaluation data for 228 complete observations out of the 259 total available. We found the mean of the predicted values was81.2145352 with a standard deviation of 8.0708116. This can be compared to the mean and standard deviation of the training data set used to create the model of 80.7908612 and 15.7521525. Comparing density of actual wins vs. predict wins


Wins Distributions Actual vs Predicted
Training Wins Density
density
0.00
0.01
0.02
50
100
TARGET_WINS
Predicted Wins Density
density
0.00
0.02
0.04
0.06
60
70
80
90
100
110
TARGET_WINS
Wins - Training Data
Wins
50
100
0.6
0.8
1.0
1.2
1.4
Wins - Predicted
Wins
60
70
80
90
100
110
0.6
0.8
1.0
1.2
1.4

***

6 APPENDIX