Data621_Homework_1

Jenny

September 30, 2018

Introduction

```
library(MASS)
library(knitr)
library(ggplot2)
library(grid)
data_train <- read.csv("https://raw.githubusercontent.com/JennierJ/DATA621/master/Homework1/moneyball-t
nrow(data_train)
```

```
## [1] 2276
```

The dataset contains 2276 professional baseball team's performance statistics from the year 1871 to 2006. There are 15 predictor variables including base hits by batters, walks by batters, strikeouts by batters and pitchers, errors, etc. And 1 response variable is number of wins for the 35 years per team.

Data Exploration

Data Summary

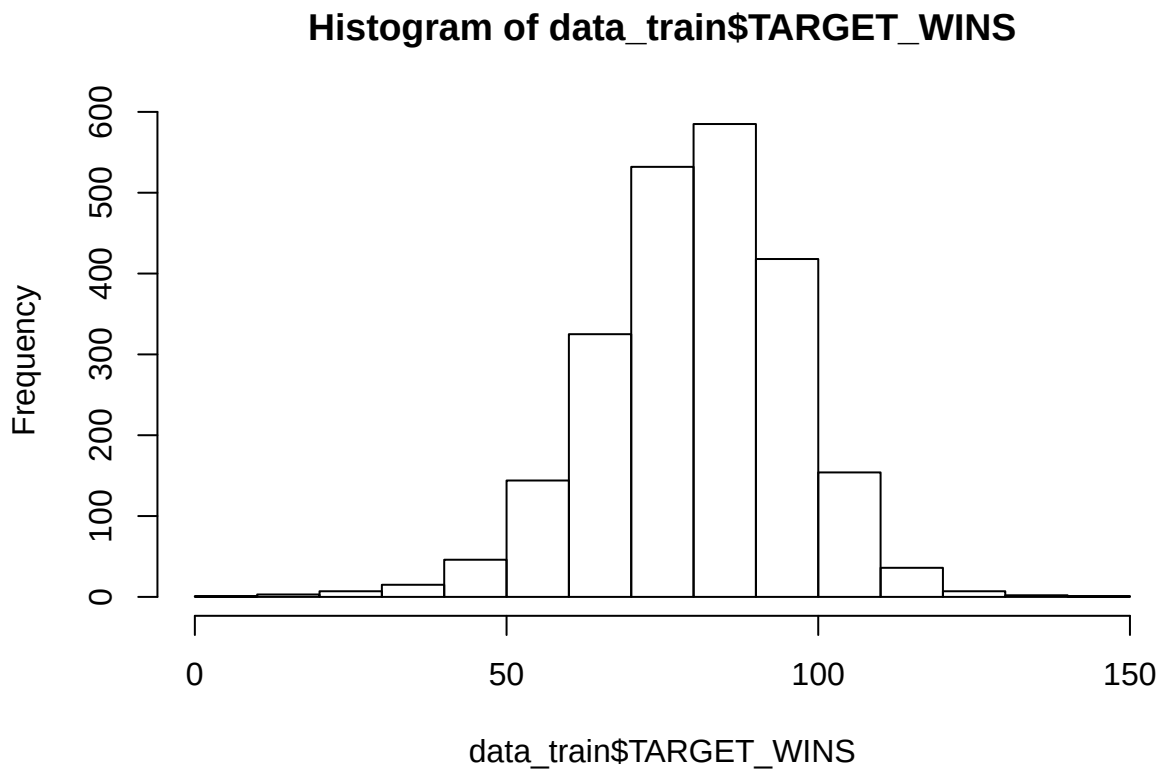
From the dataset summary, we can get a quick look at the training dataset. The best team won 146 games in the 35 years period. And according to the histogram, the number of wins apperas to be normally distributed.

```
summary(data_train)
```

```
##      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
```

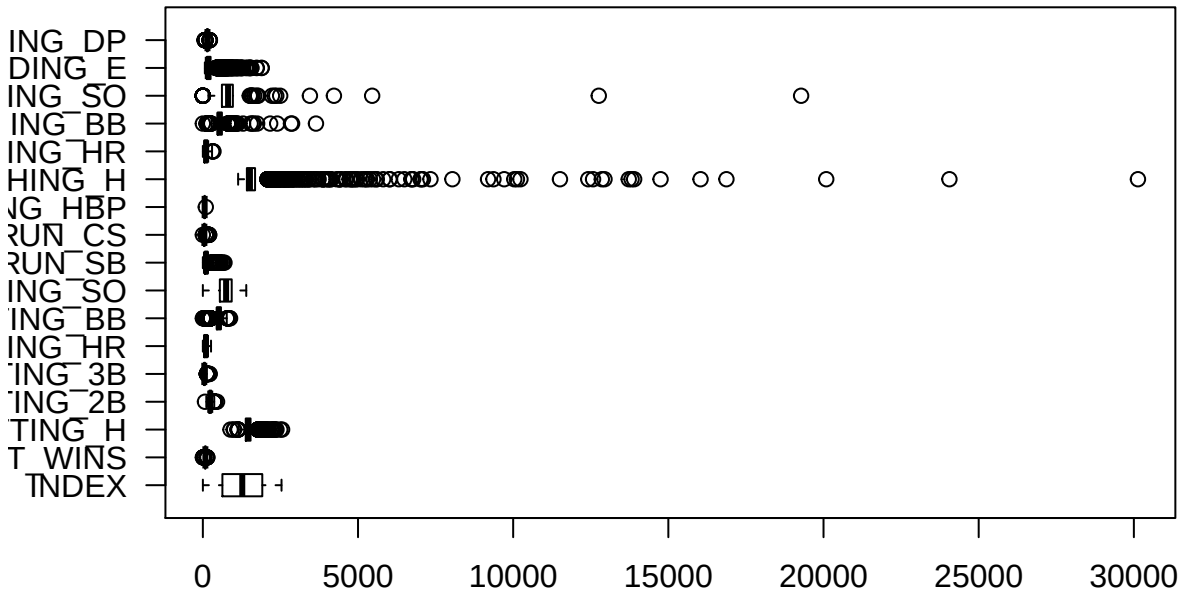
```
hist(data_train$TARGET_WINS)
```



Missing Values

To test how many missing values in the dataset, I would like to find out the numbers of missing values in the variables.There are 102 missing values in Strikeouts by batters, 131 missing values in Stolen bases, 772 missing values in Caught stealings, 2085 missing values in Batters hit by pitch, and 286 missing values in double players.

```
boxplot(data_train, horizontal = TRUE, las=1)
```




```
sum(is.na(data_train$TARGET_WINS))
```

```
## [1] 0
```

```
sum(is.na(data_train$TEAM_BATTING_H))
```

```
## [1] 0
```

```
sum(is.na(data_train$TEAM_BATTING_2B))
```

```
## [1] 0
```

```
sum(is.na(data_train$TEAM_BATTING_3B))
```

```
## [1] 0
```

```
sum(is.na(data_train$TEAM_BATTING_HR))
```

```
## [1] 0
```

```
sum(is.na(data_train$TEAM_BATTING_BB))
```

```
## [1] 0
```

```
sum(is.na(data_train$TEAM_BATTING_SO))
```

```
## [1] 102
```

```
sum(is.na(data_train$TEAM_BASERUN_SB))
```

```
## [1] 131
```

```
sum(is.na(data_train$TEAM_BASERUN_CS))
```

```
## [1] 772
```

```
sum(is.na(data_train$TEAM_BATTING_HBP))
```

```
## [1] 2085
```

```
sum(is.na(data_train$TEAM_PITCHING_H))
```

```
## [1] 0
```

```
sum(is.na(data_train$TEAM_PITCHING_HR))
```

```
## [1] 0
```

```
sum(is.na(data_train$TEAM_PITCHING_BB))
```

```
## [1] 0
```

```
sum(is.na(data_train$TEAM_PITCHING_SO))
```

```
## [1] 102
```

```
sum(is.na(data_train$TEAM_FIELDING_E))
```

```
## [1] 0
```

```
sum(is.na(data_train$TEAM_FIELDING_DP))
```

```
## [1] 286
```

Data Exploration

Missing Values. Some variables have the NA's. There are too many missing values in the field of batters hit by pitch, which rarely happends in the game. So I decided to remove this variable from the dataset, along with index.

```
data_train1 <- subset(data_train, select = -c(TEAM_BATTING_HBP, INDEX))
summary(data_train1)
```

```
##   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.: 548.0   1st Qu.: 66.0
##  Median :102.00   Median :512.0   Median : 750.0   Median :101.0
##  Mean   : 99.61   Mean   :501.6   Mean   : 735.6   Mean   :124.8
##  3rd Qu.:147.00   3rd Qu.:580.0   3rd Qu.: 930.0   3rd Qu.:156.0
##  Max.   :264.00   Max.   :878.0   Max.   :1399.0   Max.   :697.0
##                                   NA's   :102      NA's   :131
##  TEAM_BASERUN_CS TEAM_PITCHING_H TEAM_PITCHING_HR TEAM_PITCHING_BB
##  Min.   :  0.0   Min.   : 1137   Min.   :  0.0    Min.   :   0.0
##  1st Qu.: 38.0   1st Qu.: 1419   1st Qu.: 50.0    1st Qu.: 476.0
##  Median : 49.0   Median : 1518   Median :107.0    Median : 536.5
##  Mean   : 52.8   Mean   : 1779   Mean   :105.7    Mean   : 553.0
##  3rd Qu.: 62.0   3rd Qu.: 1682   3rd Qu.:150.0    3rd Qu.: 611.0
##  Max.   :201.0   Max.   :30132   Max.   :343.0    Max.   :3645.0
##  NA's   :772
##  TEAM_PITCHING_SO  TEAM_FIELDING_E  TEAM_FIELDING_DP
##  Min.   :    0.0   Min.   :  65.0   Min.   : 52.0
##  1st Qu.:  615.0   1st Qu.: 127.0   1st Qu.:131.0
##  Median :  813.5   Median : 159.0   Median :149.0
##  Mean   :  817.7   Mean   : 246.5   Mean   :146.4
##  3rd Qu.:  968.0   3rd Qu.: 249.2   3rd Qu.:164.0
##  Max.   :19278.0   Max.   :1898.0   Max.   :228.0
##  NA's   :102                        NA's   :286
```

For the rest variables wiht NAs, I decided to replace NAs with mean values of the variabls.

```
data_train1$TEAM_BATTING_SO[which(is.na(data_train1$TEAM_BATTING_SO))] <- mean(data_train1$TEAM_BATTING_
data_train1$TEAM_BASERUN_SB[which(is.na(data_train1$TEAM_BASERUN_SB))] <- mean(data_train1$TEAM_BASERUN_
data_train1$TEAM_BASERUN_CS[which(is.na(data_train1$TEAM_BASERUN_CS))] <- mean(data_train1$TEAM_BASERUN_
data_train1$TEAM_PITCHING_SO[which(is.na(data_train1$TEAM_PITCHING_SO))] <- mean(data_train1$TEAM_PITCHI
data_train1$TEAM_FIELDING_DP[which(is.na(data_train1$TEAM_FIELDING_DP))] <- mean(data_train1$TEAM_FIELDI
```

No Nas in the training dataset

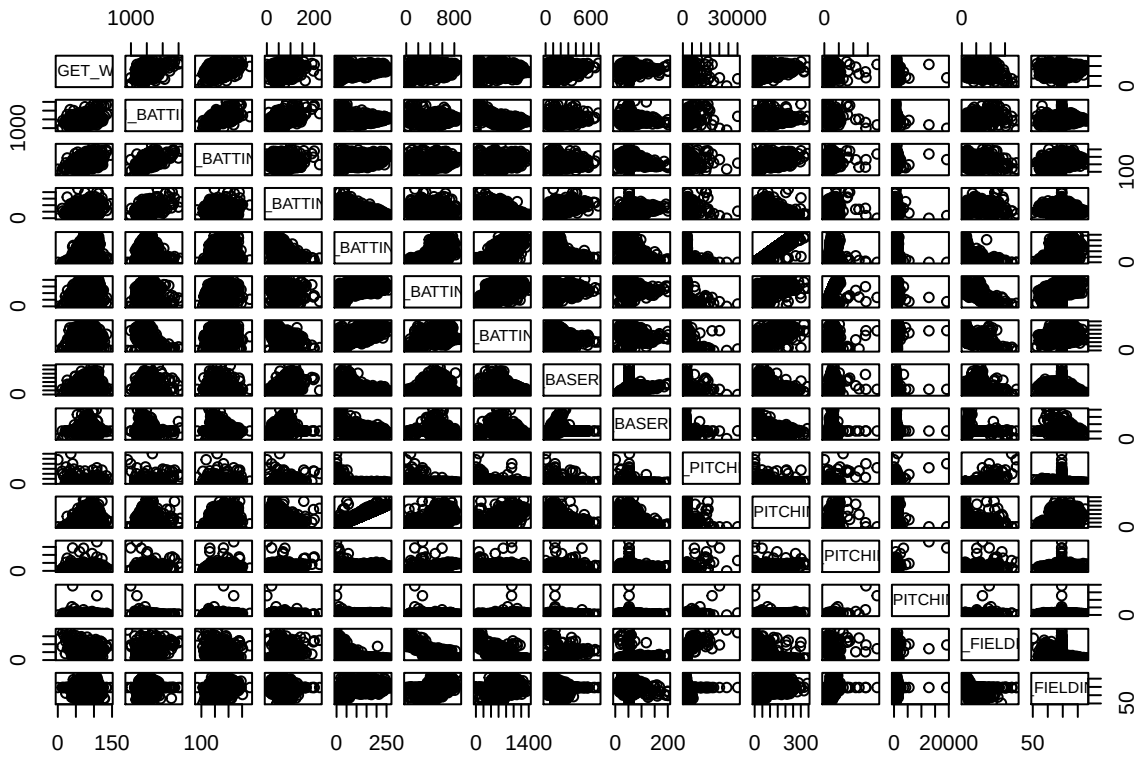
```
sum(is.na(data_train1))
```

```
## [1] 0
```

Build Models

Before beginning model development, it is useful to get a visual sense of the relationship within the data.

```
pairs(data_train1, gap = 0.5)
```



Let's try out the models by backward elimination. I am using all the variables left to build the full multiple regression model.

```
full.model <- lm(TARGET_WINS ~., data = data_train1 )
summary(full.model)
```

```
##
## Call:
## lm(formula = TARGET_WINS ~ ., data = data_train1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -49.994  -8.576   0.136   8.345  58.628
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       2.502e+01  5.397e+00   4.636 3.75e-06 ***
## TEAM_BATTING_H    4.824e-02  3.687e-03  13.085  < 2e-16 ***
## TEAM_BATTING_2B  -2.006e-02  9.152e-03  -2.192 0.028486 *
## TEAM_BATTING_3B   6.047e-02  1.676e-02   3.608 0.000315 ***
## TEAM_BATTING_HR   5.299e-02  2.743e-02   1.932 0.053488 .
## TEAM_BATTING_BB   1.042e-02  5.818e-03   1.790 0.073544 .
## TEAM_BATTING_SO  -9.349e-03  2.551e-03  -3.665 0.000253 ***
## TEAM_BASERUN_SB   2.949e-02  4.462e-03   6.610 4.78e-11 ***
## TEAM_BASERUN_CS  -1.188e-02  1.614e-02  -0.736 0.461905
```

```
## TEAM_PITCHING_H  -7.342e-04  3.676e-04  -1.997 0.045946 *
## TEAM_PITCHING_HR  1.480e-02  2.432e-02   0.609 0.542877
## TEAM_PITCHING_BB  8.891e-05  4.145e-03   0.021 0.982891
## TEAM_PITCHING_SO  2.843e-03  9.187e-04   3.095 0.001994 **
## TEAM_FIELDING_E  -2.112e-02  2.480e-03  -8.516  < 2e-16 ***
## TEAM_FIELDING_DP -1.210e-01  1.302e-02  -9.297  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.04 on 2261 degrees of freedom
## Multiple R-squared:  0.3189, Adjusted R-squared:  0.3147
## F-statistic: 75.63 on 14 and 2261 DF,  p-value: < 2.2e-16
```

```
# From the summary of the full.model, we can see that the R-squre is 0.3147. To continue developing the
full.model <- update(full.model, .~. - TEAM_PITCHING_BB)
summary(full.model)
```

```
##
## 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_SO + TEAM_FIELDING_E + TEAM_FIELDING_DP, data = data_train1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -49.994  -8.576   0.136   8.345  58.626
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      25.0145796  5.3904993   4.640 3.67e-06 ***
## TEAM_BATTING_H    0.0482393  0.0036807  13.106  < 2e-16 ***
## TEAM_BATTING_2B  -0.0200575  0.0091490  -2.192 0.028457 *
## TEAM_BATTING_3B   0.0604730  0.0167556   3.609 0.000314 ***
## TEAM_BATTING_HR   0.0527106  0.0240710   2.190 0.028641 *
## TEAM_BATTING_BB   0.0105175  0.0033664   3.124 0.001805 **
## TEAM_BATTING_SO  -0.0093631  0.0024585  -3.809 0.000144 ***
## TEAM_BASERUN_SB   0.0295055  0.0044087   6.693 2.76e-11 ***
## TEAM_BASERUN_CS  -0.0118872  0.0161276  -0.737 0.461155
## TEAM_PITCHING_H  -0.0007306  0.0003283  -2.225 0.026147 *
## TEAM_PITCHING_HR  0.0150659  0.0209923   0.718 0.473025
## TEAM_PITCHING_SO  0.0028567  0.0006717   4.253 2.20e-05 ***
## TEAM_FIELDING_E  -0.0211192  0.0024784  -8.521  < 2e-16 ***
## TEAM_FIELDING_DP -0.1210298  0.0130139  -9.300  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.04 on 2262 degrees of freedom
## Multiple R-squared:  0.3189, Adjusted R-squared:  0.315
## F-statistic: 81.49 on 13 and 2262 DF,  p-value: < 2.2e-16
```

```
# Remove TEAM_PITCHING_HR
full.model <- update(full.model, .~. -TEAM_PITCHING_HR)
summary(full.model)
```

```
##
```

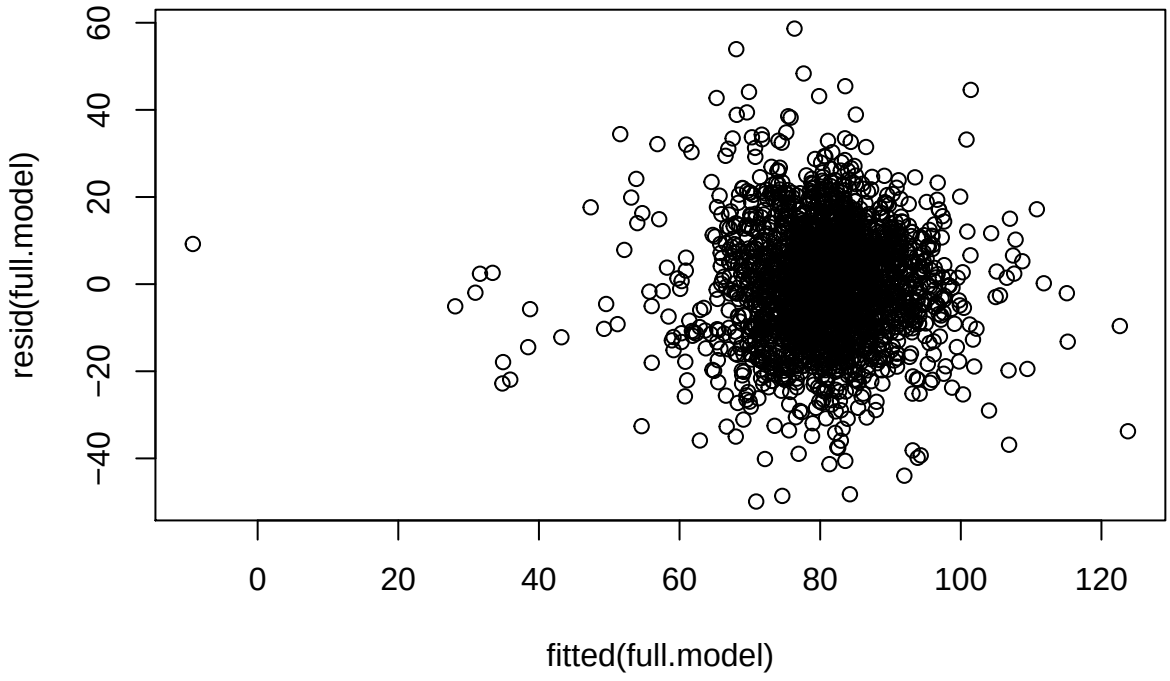
```
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_2B +
##     TEAM_BATTING_3B + TEAM_BATTING_HR + TEAM_BATTING_BB + TEAM_BATTING_SO +
##     TEAM_BASERUN_SB + TEAM_BASERUN_CS + TEAM_PITCHING_H + TEAM_PITCHING_SO +
##     TEAM_FIELDING_E + TEAM_FIELDING_DP, data = data_train1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -49.986  -8.604   0.092   8.351  58.685
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      24.4155968  5.3249267   4.585 4.78e-06 ***
## TEAM_BATTING_H    0.0484960  0.0036629  13.240  < 2e-16 ***
## TEAM_BATTING_2B  -0.0202501  0.0091440  -2.215 0.026889 *
## TEAM_BATTING_3B   0.0621392  0.0165923   3.745 0.000185 ***
## TEAM_BATTING_HR   0.0684918  0.0097916   6.995 3.48e-12 ***
## TEAM_BATTING_BB   0.0105028  0.0033660   3.120 0.001830 **
## TEAM_BATTING_SO  -0.0093218  0.0024575  -3.793 0.000153 ***
## TEAM_BASERUN_SB   0.0294999  0.0044082   6.692 2.77e-11 ***
## TEAM_BASERUN_CS  -0.0116212  0.0161216  -0.721 0.471078
## TEAM_PITCHING_H  -0.0006827  0.0003214  -2.124 0.033769 *
## TEAM_PITCHING_SO  0.0028801  0.0006708   4.293 1.84e-05 ***
## TEAM_FIELDING_E  -0.0209829  0.0024709  -8.492  < 2e-16 ***
## TEAM_FIELDING_DP -0.1208771  0.0130108  -9.291  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.04 on 2263 degrees of freedom
## Multiple R-squared:  0.3188, Adjusted R-squared:  0.3152
## F-statistic: 88.25 on 12 and 2263 DF,  p-value: < 2.2e-16
```

```
# Remove TEAM_BASERUN_CS
full.model <- update(full.model, .~. -TEAM_BASERUN_CS)
summary(full.model)
```

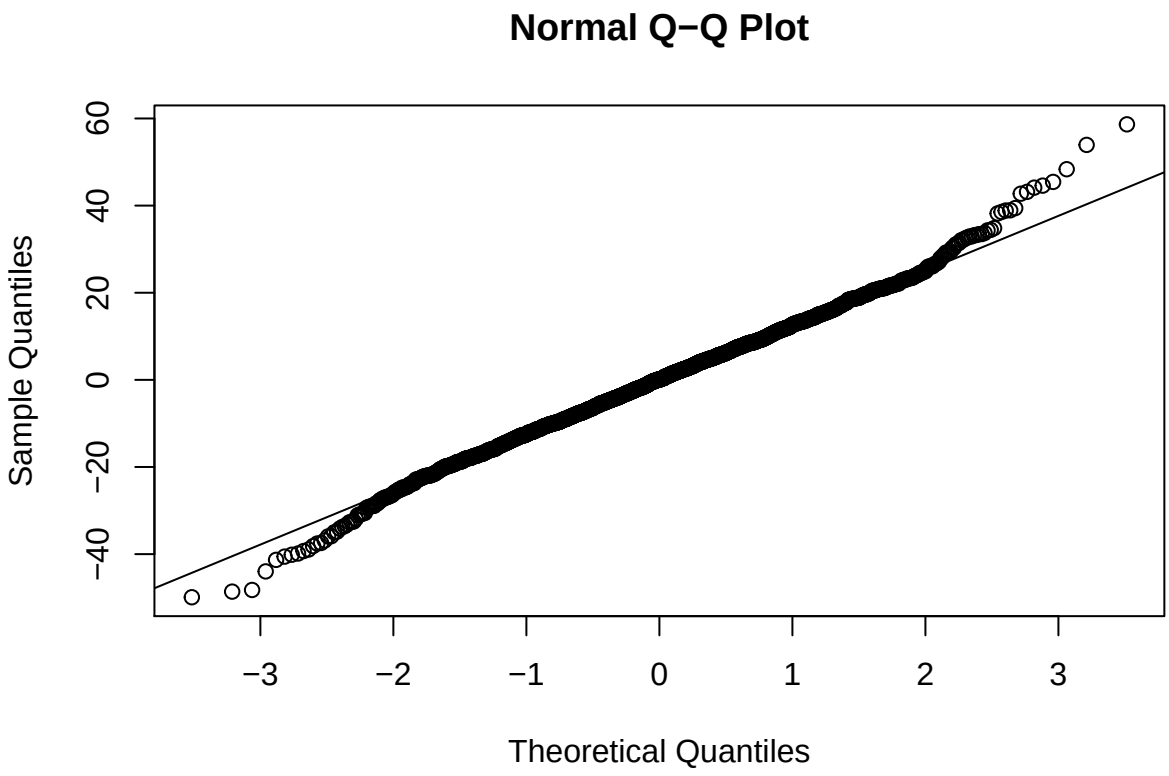
```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_2B +
##     TEAM_BATTING_3B + TEAM_BATTING_HR + TEAM_BATTING_BB + TEAM_BATTING_SO +
##     TEAM_BASERUN_SB + TEAM_PITCHING_H + TEAM_PITCHING_SO + TEAM_FIELDING_E +
##     TEAM_FIELDING_DP, data = data_train1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -49.899  -8.568   0.091   8.397  58.651
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      23.6666983  5.2220414   4.532 6.14e-06 ***
## TEAM_BATTING_H    0.0484570  0.0036621  13.232  < 2e-16 ***
## TEAM_BATTING_2B  -0.0205123  0.0091358  -2.245 0.024847 *
## TEAM_BATTING_3B   0.0624661  0.0165843   3.767 0.000170 ***
## TEAM_BATTING_HR   0.0697785  0.0096266   7.249 5.75e-13 ***
## TEAM_BATTING_BB   0.0107446  0.0033489   3.208 0.001354 **
```

```
## TEAM_BATTING_SO  -0.0093019  0.0024571  -3.786 0.000157 ***
## TEAM_BASERUN_SB   0.0287708  0.0042901   6.706 2.51e-11 ***
## TEAM_PITCHING_H  -0.0006920  0.0003211  -2.155 0.031253 *
## TEAM_PITCHING_SO  0.0028867  0.0006707   4.304 1.75e-05 ***
## TEAM_FIELDING_E  -0.0205973  0.0024120  -8.540  < 2e-16 ***
## TEAM_FIELDING_DP -0.1210083  0.0130082  -9.302  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.03 on 2264 degrees of freedom
## Multiple R-squared:  0.3186, Adjusted R-squared:  0.3153
## F-statistic: 96.25 on 11 and 2264 DF,  p-value: < 2.2e-16
```

```
# Residual Analysis
plot(fitted(full.model), resid(full.model))
```




```
qqnorm(resid(full.model))
qqline(resid(full.model))
```



Select Models

We are to use the model to predict the number of wins with evaluation data. And the predicted number of wins are listed in the below dataset of predicted, with index on the first column.

```
data_pre <- read.csv("https://raw.githubusercontent.com/JennierJ/DATA621/master/Homework1/moneyball-eval
head(data_pre)
```

```
##   INDEX TEAM_BATTING_H TEAM_BATTING_2B TEAM_BATTING_3B TEAM_BATTING_HR
## 1     9           1209             170              33              83
## 2    10           1221             151              29              88
## 3    14           1395             183              29              93
## 4    47           1539             309              29             159
## 5    60           1445             203              68               5
## 6    63           1431             236              53              10
##   TEAM_BATTING_BB TEAM_BATTING_SO TEAM_BASERUN_SB TEAM_BASERUN_CS
## 1             447            1080              62              50
## 2             516             929              54              39
## 3             509             816              59              47
## 4             486             914             148              57
## 5              95             416              NA              NA
## 6             215             377              NA              NA
##   TEAM_BATTING_HBP TEAM_PITCHING_H TEAM_PITCHING_HR TEAM_PITCHING_BB
## 1               NA            1209               83              447
```

```
## 2               NA            1221               88              516
## 3               NA            1395               93              509
## 4               42            1539              159              486
## 5               NA            3902               14              257
## 6               NA            2793               20              420
##   TEAM_PITCHING_SO TEAM_FIELDING_E TEAM_FIELDING_DP
## 1             1080              140              156
## 2              929              135              164
## 3              816              156              153
## 4              914              124              154
## 5             1123              616              130
## 6              736              572              105
```

```
predicted.win <- predict(full.model, newdata = data_pre)
predicted <- data.frame(predicted.win)
summary(predicted)
```

```
##  predicted.win
##  Min.   : 61.21
##  1st Qu.: 77.33
##  Median : 81.38
##  Mean   : 81.62
##  3rd Qu.: 85.76
##  Max.   :107.10
##  NA's   :54
```

```
head(predicted)
```

```
##   predicted.win
## 1      63.67770
## 2      65.35447
## 3      75.05018
## 4      86.17206
## 5            NA
## 6            NA
```