

Project1

Sri Seshadri

9/30/2017

1. Introduction

An sports investor is interested in investing in the sport of baseball. The investor is looking at investments ranging from team ownership, collectibles, apparel and legal sports betting. There is interest in understanding what are factors that influence teams' winning and predicting the number of wins in a season, so that an informed decision is made on the investment. To enable decision making, historical performance of baseball teams from the year 1871 to 2006 is analyzed and modeled for predicting number of wins in a season. This report discusses the analysis and the predictive models developed using the historical data. It should be noted that the data scientist who analyzed the data does not have subject matter expertise in the area of baseball. It is expected that the model(s) may need fine tuning based on feedback from subject matter experts.

1.1 Analysis Process

The following process steps were used for building a predictive models:

- Exploratory Data Analysis
 - Perform data quality checks, quantify missing data.
 - Check for systemic loss in data
 - Understand relationships amongst predictors and between target variables and predictors.
 - Create attribute or indicator variables to aid data cleaning.
 - Filter out clean data for feature selection and model building.
- Feature Selection
 - Subset complete records to model wins in season
 - Use different modeling techniques to select candidate predictors.
 - If data is missing for candidate predictors, identify imputing methods.
- Model Building
 - Test models that were build using complete records on the entire data set with imputed data.
 - Compare models based on Adjusted R squared, AIC, MAE
 - Check if models make physical sense.
- Initial model deployment
 - Deploy model to predict wins on out of sample data.
 - Discuss models and results with subject matter experts.
 - Fine tune model and re-test
- Final model deployment
 - Investment decisions.

1.2 Executive summary

The below model was chosen from a candidate pool of 10 models (8 models are discussed in this report) for its simplicity, interpretability and comparable mean absolute error.

$$\begin{aligned} \text{TARGET_WINS} = & 34.12 + 0.01 * \text{TEAM_BATTING_H} + 0.1 * \text{TEAM_BATTING_3B} + \\ & 0.12 * \text{TEAM_BATTING_HR} + 0.04 * \text{TEAM_BATTING_BB} - 0.02 * \text{TEAM_BATTING_SO} \\ & + 0.07 * \text{TEAM_BASERUN_SB} - 5.49 * \text{BatHR_Filter} \end{aligned}$$

The model has an average error (MAE) of ± 11 wins. The data scientist welcomes suggestions from subject matter expertise to further iterate on the model building process to produce a better model.

2. About the Data

The data provided has approximately 2300 rows, each representing a team's performance in the seasons during the years 1871 to 2006. The statistics are adjusted to match a performance of 162 game season. The dictionary of the variables in the data are provided in appendix A.1. Table 1 below shows statistics of the variables in the data.

It can be seen that there are multiple variables with missing data. The systemic nature of missing data is discussed in the next section, the pattern in missing leads us to assume that the missing data is likely from the 18th and the early 19th century, where sophisticated data collection may not have existed. If the variables that contain missing data are deemed important for predicting number of wins, then an appropriate method would be used for data imputation.

Table 1: Summary Stats and missing values

	min	Q1	median	Q3	max	mean	sd	n	missing	cv
INDEX	1	630.75	1270.5	1915.50	2535	1268.46	736.35	2276	0	0.58
TARGET_WINS	0	71.00	82.0	92.00	146	80.79	15.75	2276	0	0.19
TEAM_BATTING_H	891	1383.00	1454.0	1537.25	2554	1469.27	144.59	2276	0	0.10
TEAM_BATTING_2B	69	208.00	238.0	273.00	458	241.25	46.80	2276	0	0.19
TEAM_BATTING_3B	0	34.00	47.0	72.00	223	55.25	27.94	2276	0	0.51
TEAM_BATTING_HR	0	42.00	102.0	147.00	264	99.61	60.55	2276	0	0.61
TEAM_BATTING_BB	0	451.00	512.0	580.00	878	501.56	122.67	2276	0	0.24
TEAM_BATTING_SO	0	548.00	750.0	930.00	1399	735.61	248.53	2174	102	0.34
TEAM_BASERUN_SB	0	66.00	101.0	156.00	697	124.76	87.79	2145	131	0.70
TEAM_BASERUN_CS	0	38.00	49.0	62.00	201	52.80	22.96	1504	772	0.43
TEAM_BATTING_HBP	29	50.50	58.0	67.00	95	59.36	12.97	191	2085	0.22
TEAM_PITCHING_H	1137	1419.00	1518.0	1682.50	30132	1779.21	1406.84	2276	0	0.79
TEAM_PITCHING_HR	0	50.00	107.0	150.00	343	105.70	61.30	2276	0	0.58
TEAM_PITCHING_BB	0	476.00	536.5	611.00	3645	553.01	166.36	2276	0	0.30
TEAM_PITCHING_SO	0	615.00	813.5	968.00	19278	817.73	553.09	2174	102	0.68
TEAM_FIELDING_E	65	127.00	159.0	249.25	1898	246.48	227.77	2276	0	0.92
TEAM_FIELDING_DP	52	131.00	149.0	164.00	228	146.39	26.23	1990	286	0.18

3. Exploratory Data Analysis (EDA)

Figure 1, shows that the missing data is systematic and they correspond to the lower end of the distributions of home runs by batters and strike-outs. The complete cases from the data are used to explore the relationship amongst the predictor variables and the relationship between Wins and the other predictors. The exploratory data analysis is used as a tool for feature selection for model building.

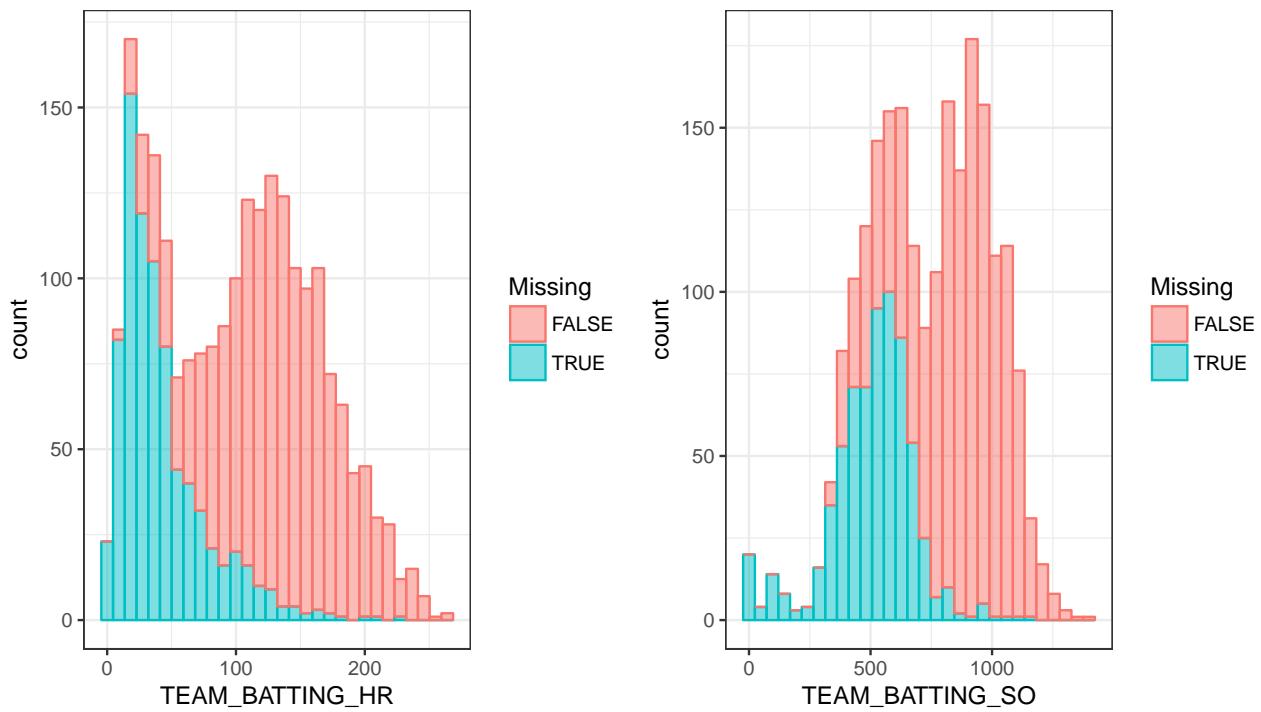
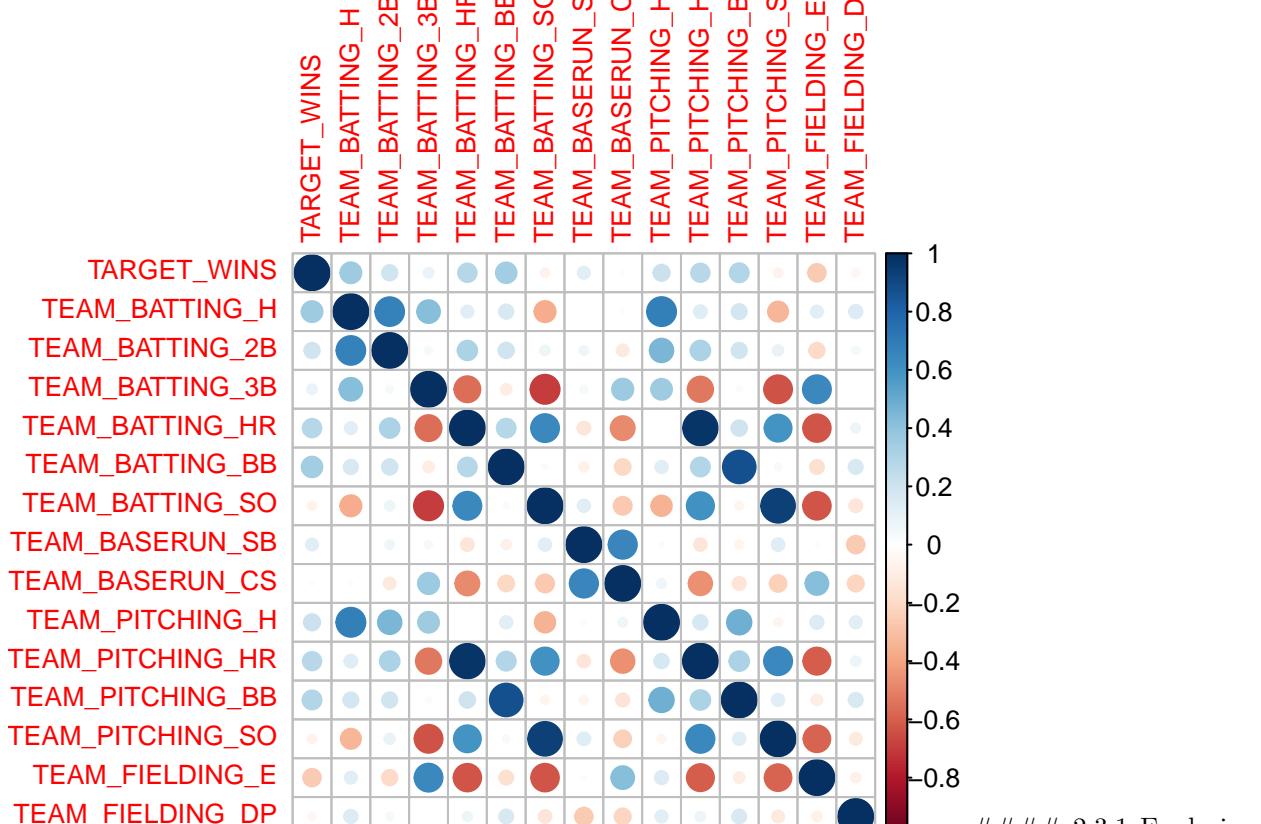
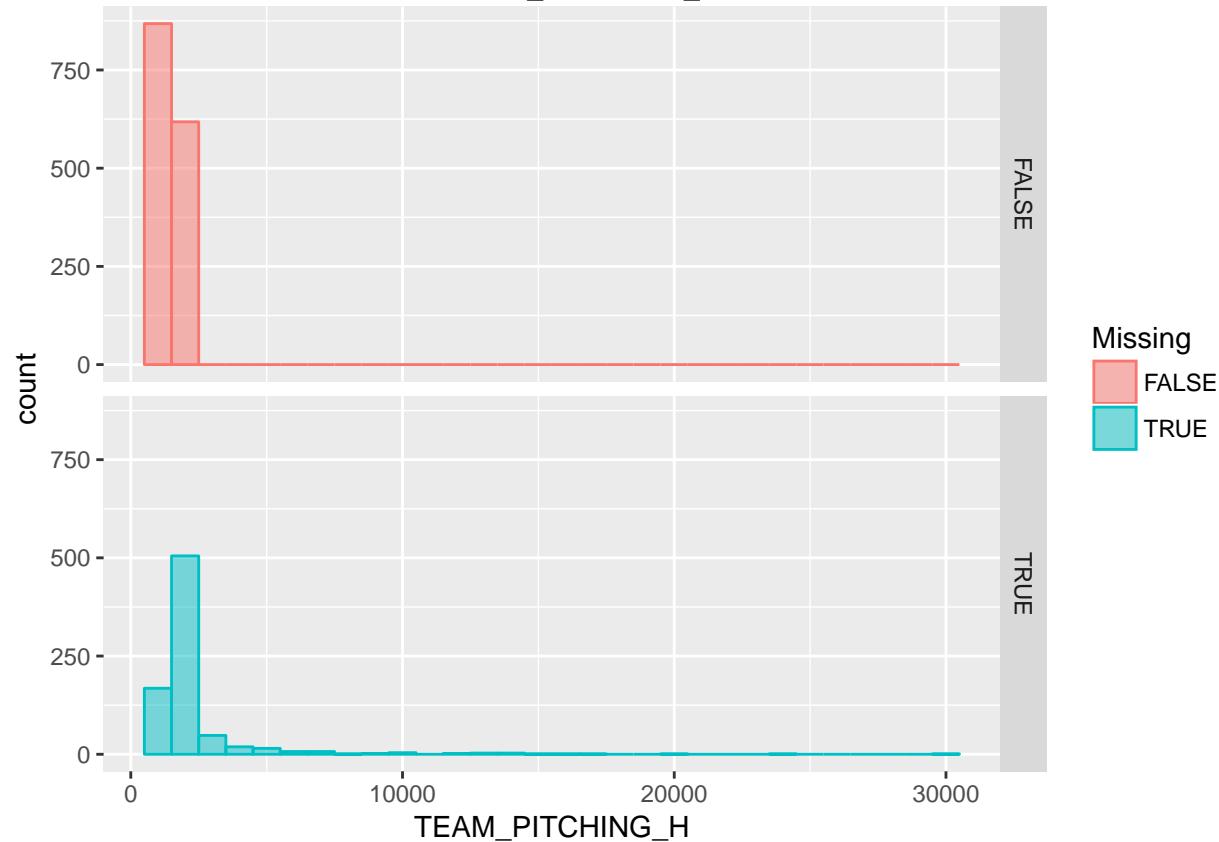
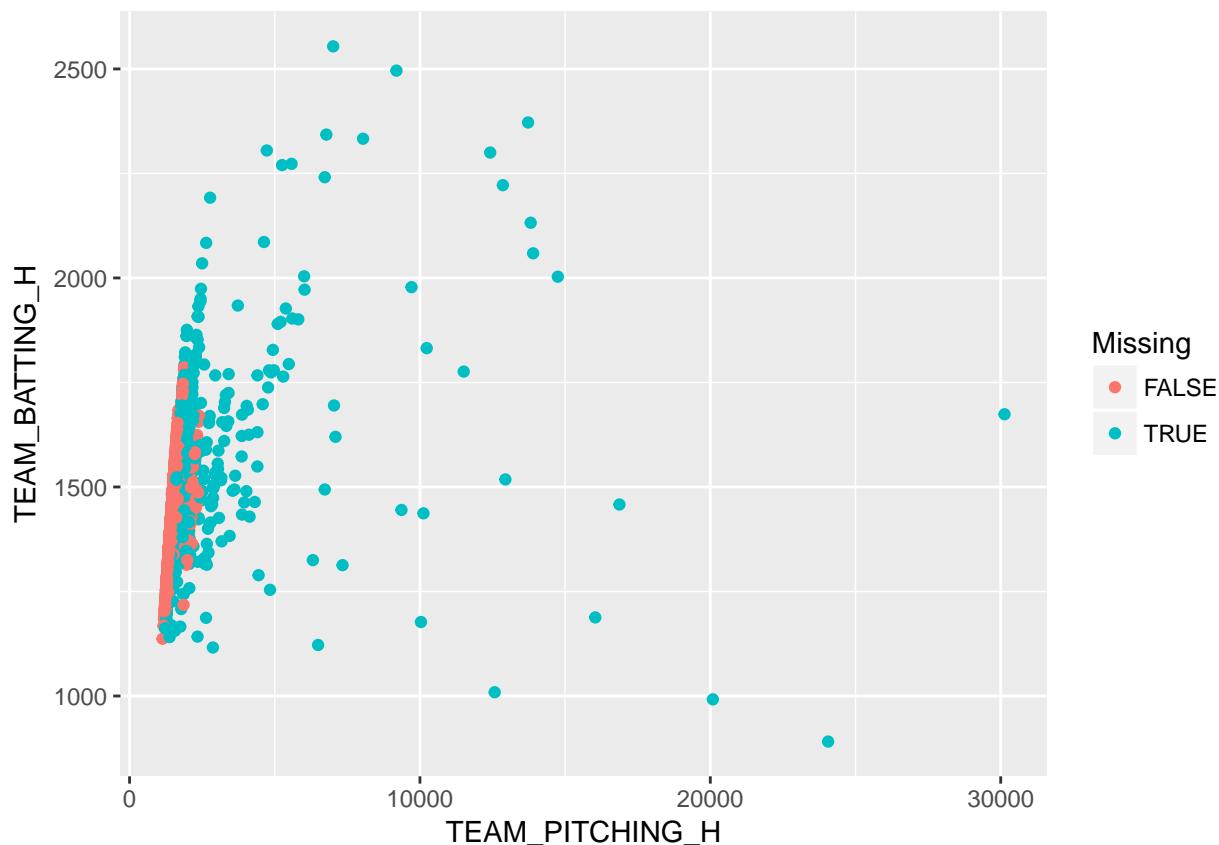


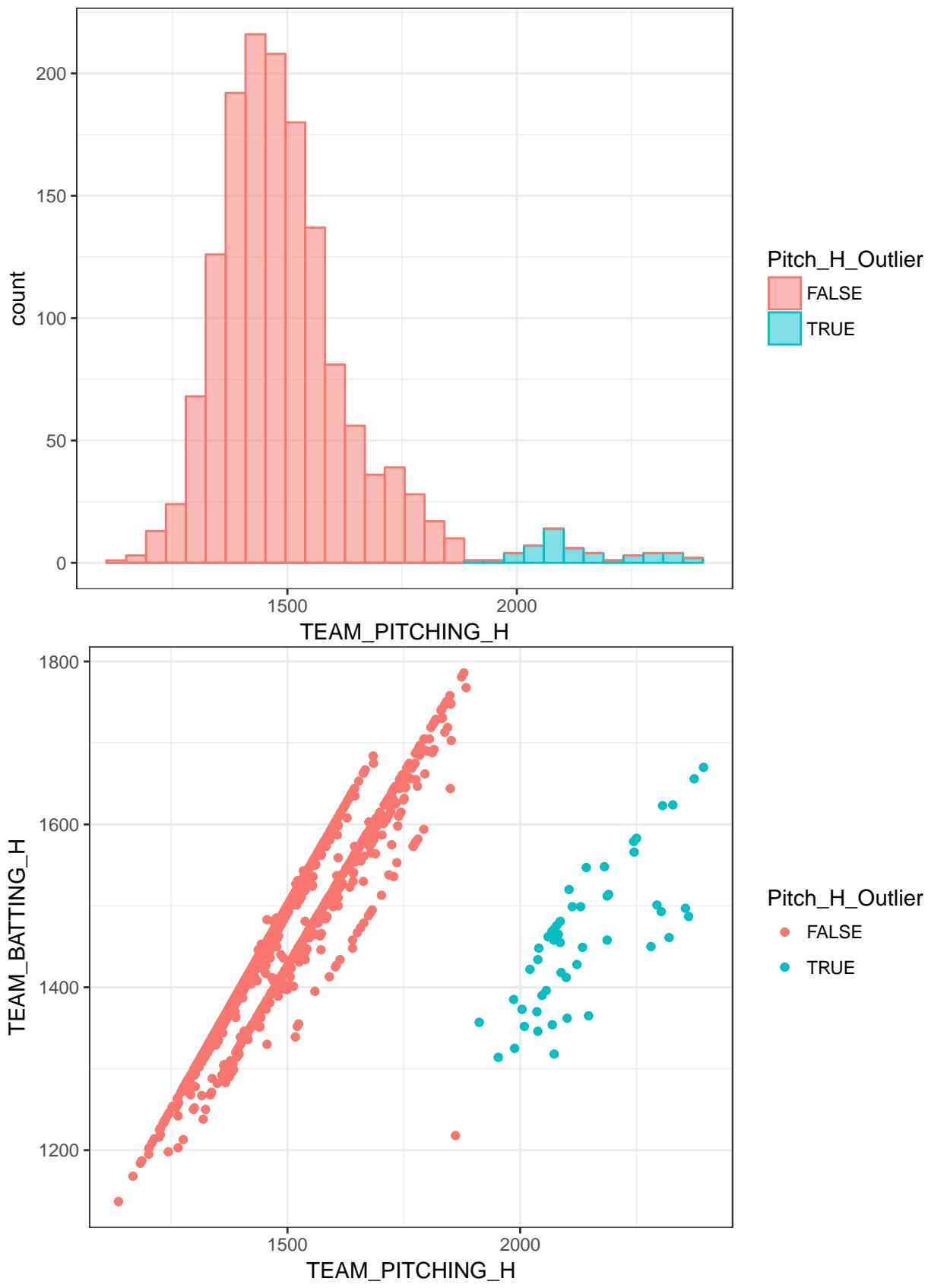
Figure 1: Rows with missing data is from the lower end of the distributions of Home runs and Strike outs

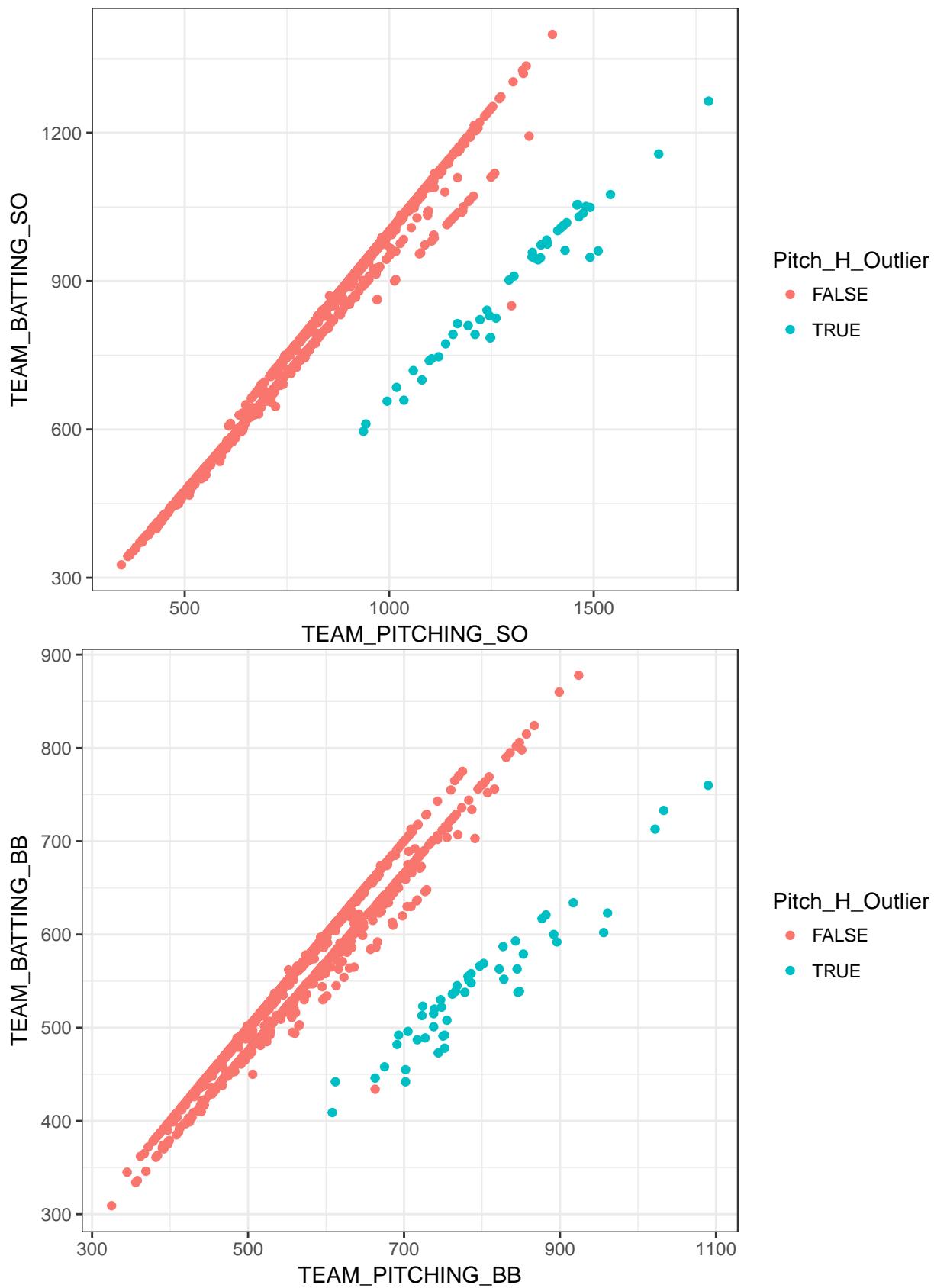


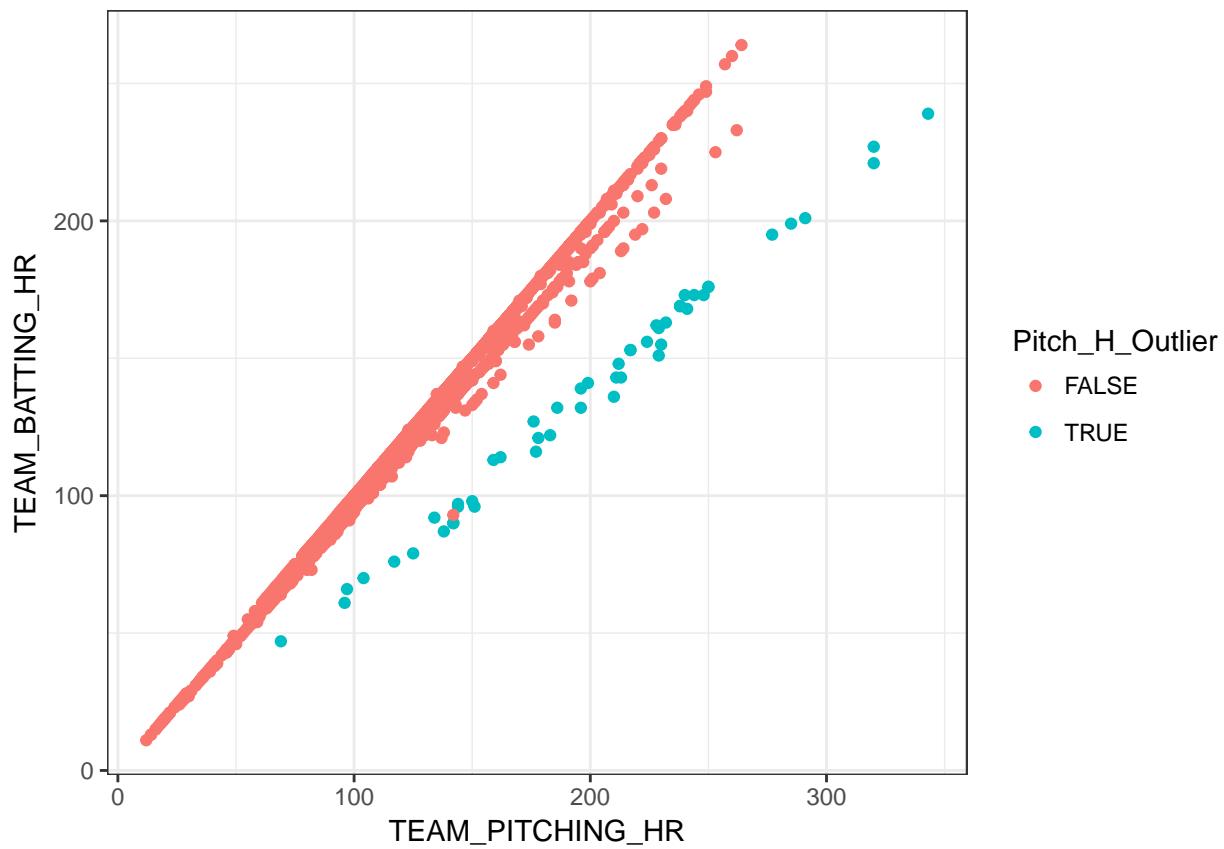
the relationships further

What are the linear relationships with varying slopes - parallel lines?

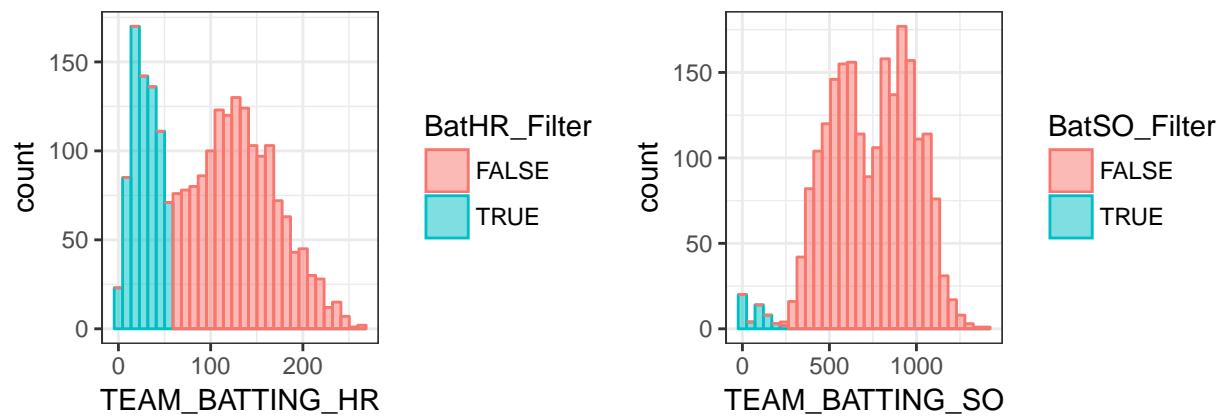


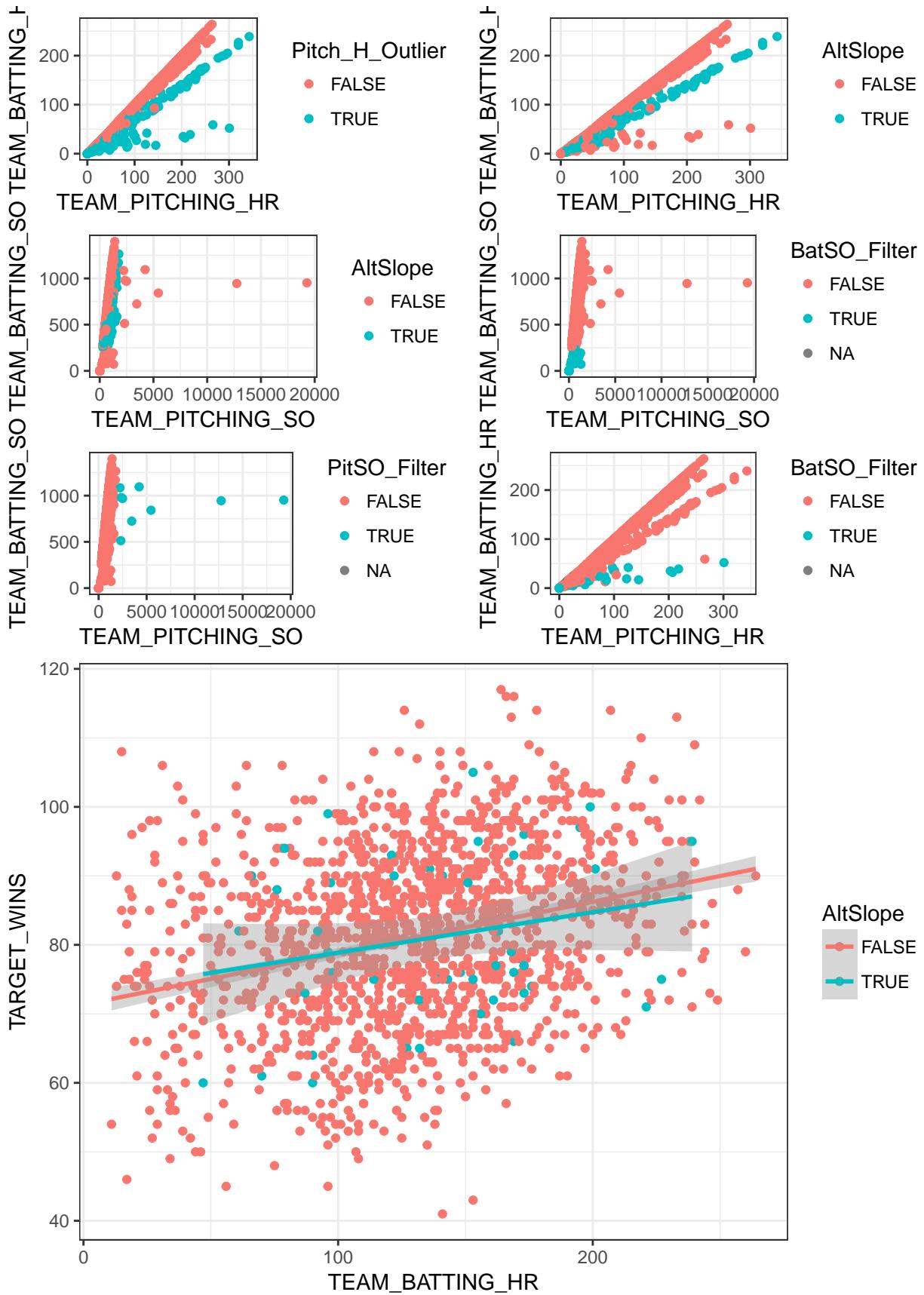


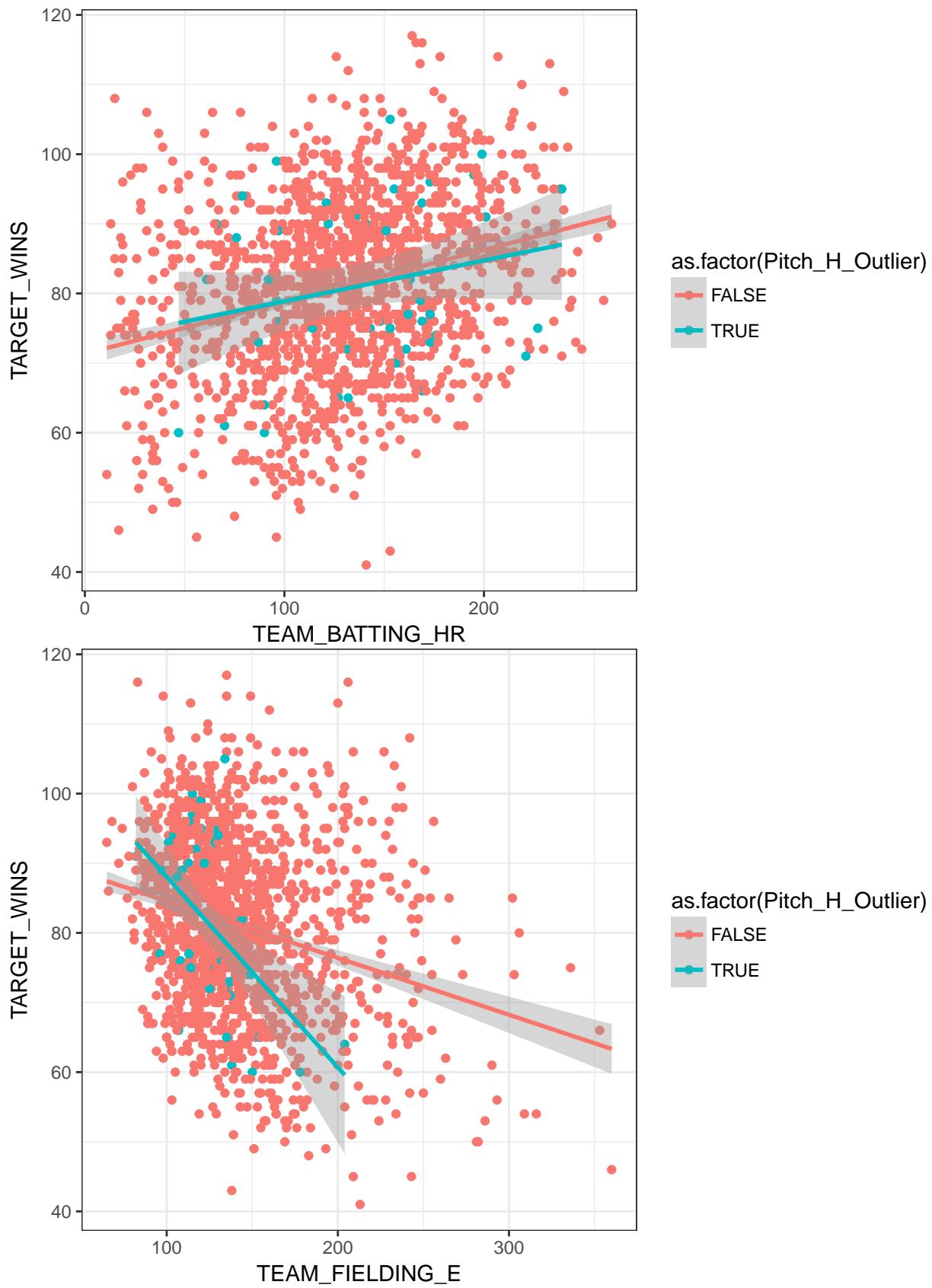


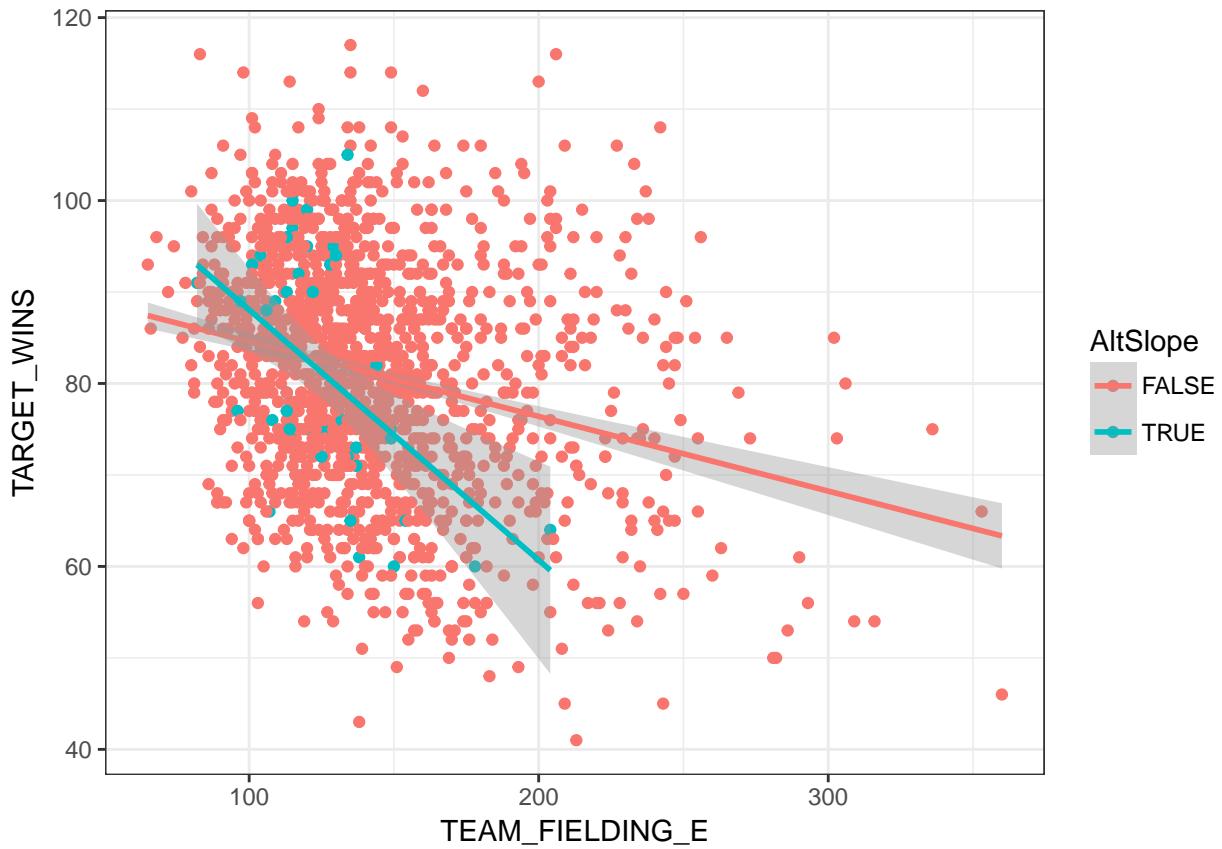


2.3.2 Indicator variables









3.0 Feature selections

What are the important features amongst complete cases?

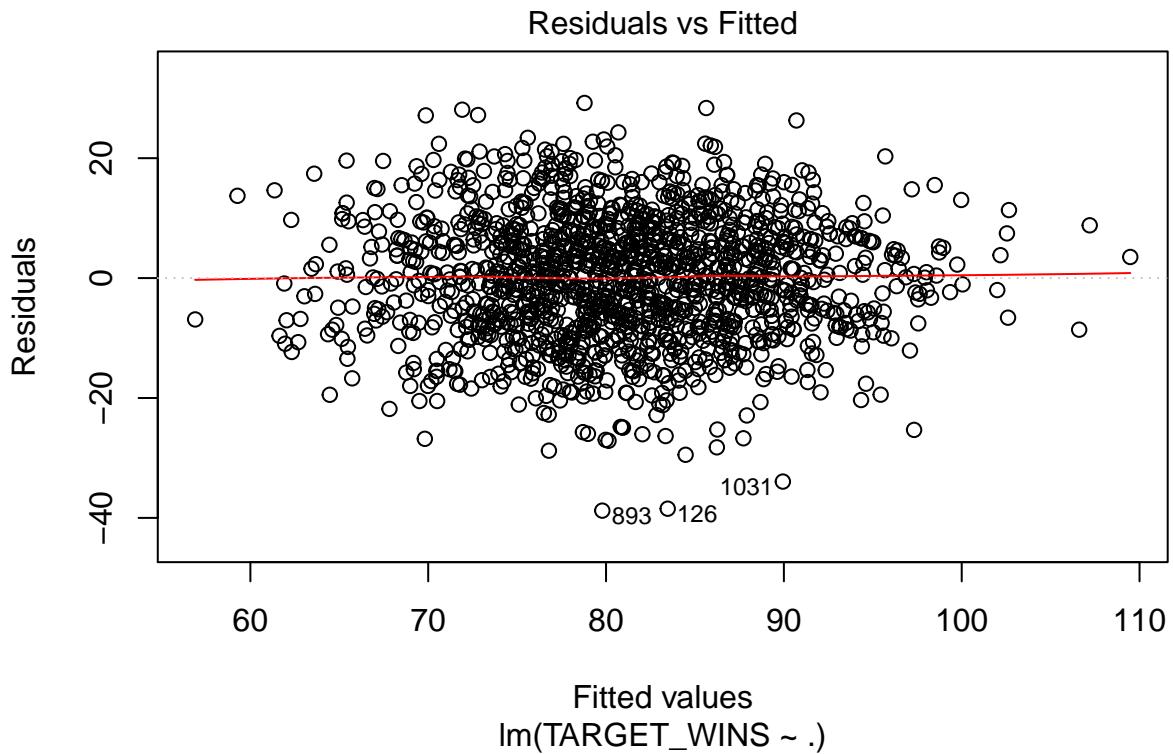
Would a simple model work?

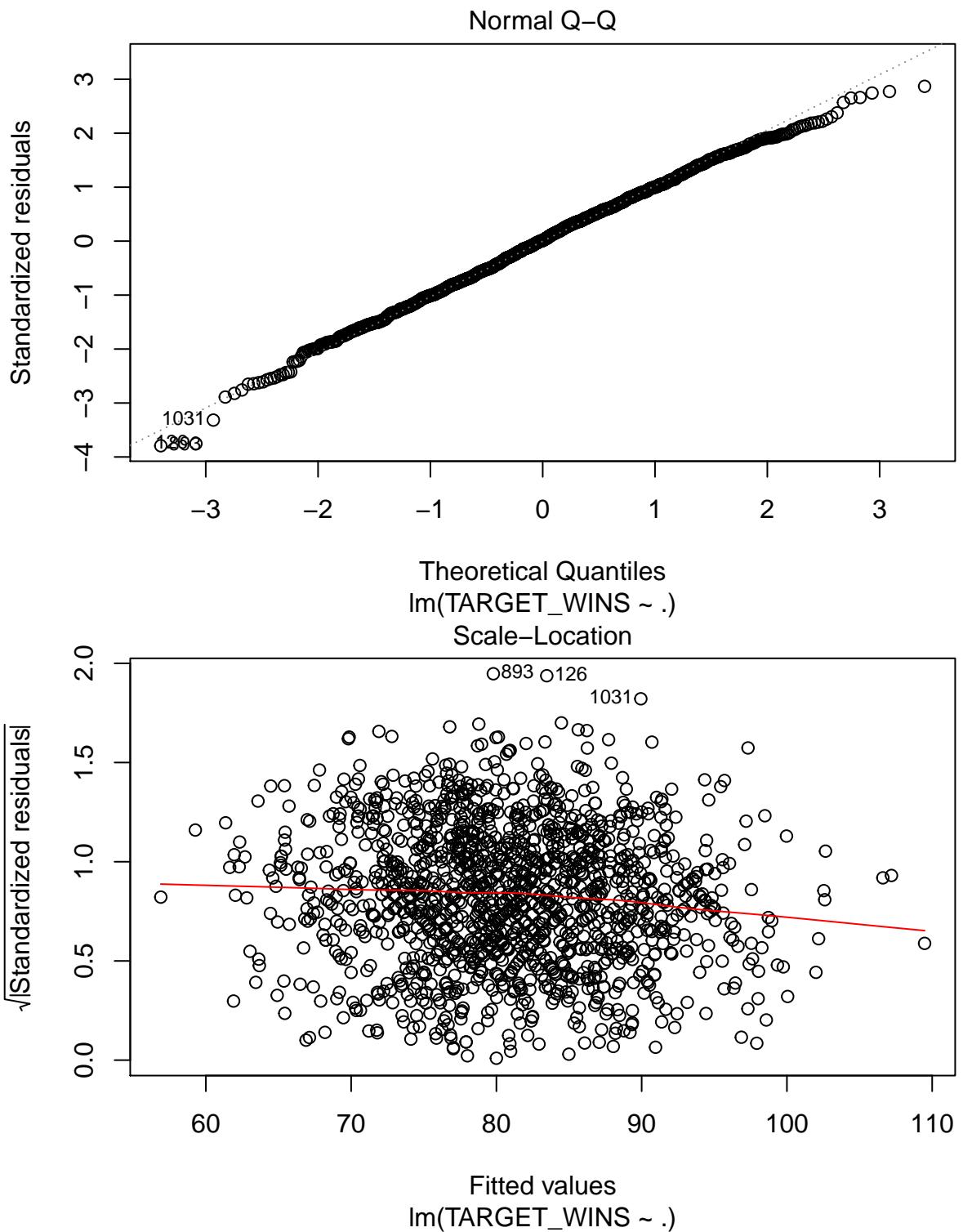
```
##
## Call:
## lm(formula = TARGET_WINS ~ ., data = cordata[, simple.Pred])
##
## Residuals:
##    Min     1Q Median     3Q    Max 
## -38.789 -7.134  0.131  7.060 29.210 
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)            30.521206   6.851571  4.455 9.04e-06 ***
## TEAM_BATTING_H          0.032833   0.005038  6.517 9.83e-11 ***
## TEAM_BATTING_2B         -0.051953   0.010036 -5.177 2.57e-07 ***
## TEAM_BATTING_3B         0.075513   0.023054  3.276 0.00108 ** 
## TEAM_BATTING_HR         0.111551   0.010290 10.841 < 2e-16 ***
## TEAM_BATTING_BB         0.039478   0.003606 10.948 < 2e-16 ***
## TEAM_BATTING_SO        -0.017120   0.002589 -6.611 5.31e-11 ***
## TEAM_BASERUN_SB         0.066541   0.009068  7.338 3.58e-13 ***
## TEAM_BASERUN_CS        -0.013322   0.019192 -0.694  0.48770  
## TEAM_FIELDING_DP        -0.091301   0.014024 -6.511 1.02e-10 ***
## Pitch_H_OutlierTRUE    -0.866432   1.475723 -0.587  0.55721  
##
```

```

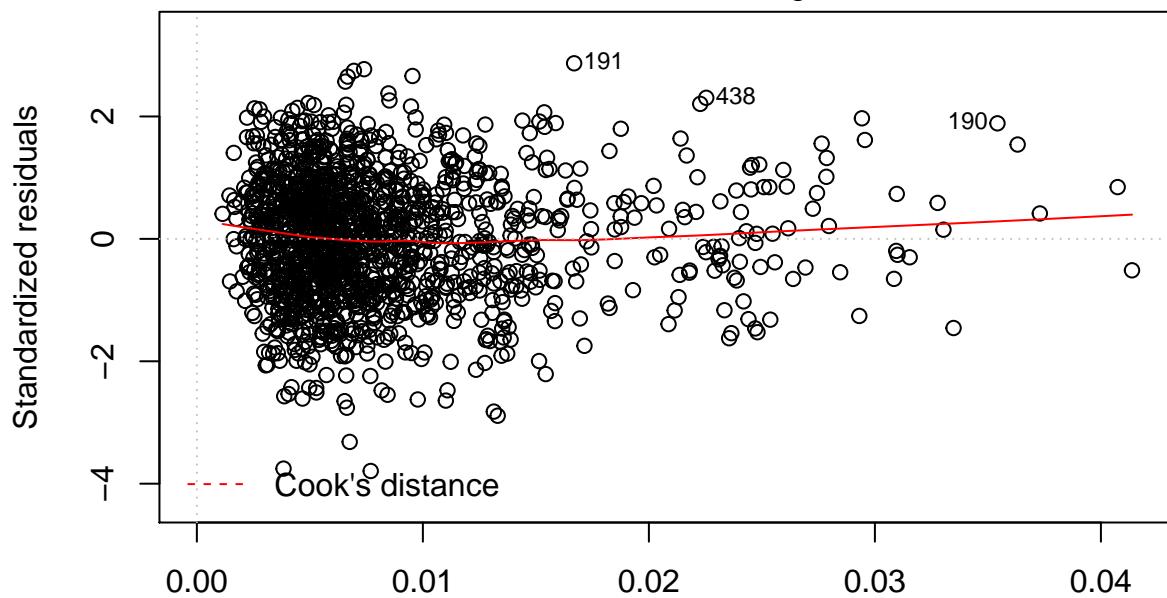
## BatHR_FilterTRUE    -5.402819   1.259058  -4.291 1.89e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.27 on 1474 degrees of freedom
## Multiple R-squared:  0.3505, Adjusted R-squared:  0.3456
## F-statistic: 72.31 on 11 and 1474 DF,  p-value: < 2.2e-16
##   TEAM_BATTING_H  TEAM_BATTING_2B  TEAM_BATTING_3B  TEAM_BATTING_HR
##   3.891751       2.505281       2.603314       3.523489
##   TEAM_BATTING_BB TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_BASERUN_CS
##   1.188891       3.791054       2.277732       2.708859
##   TEAM_FIELDING_DP Pitch_H_Outlier      BatHR_Filter
##   1.143868       1.017259       1.795982

```

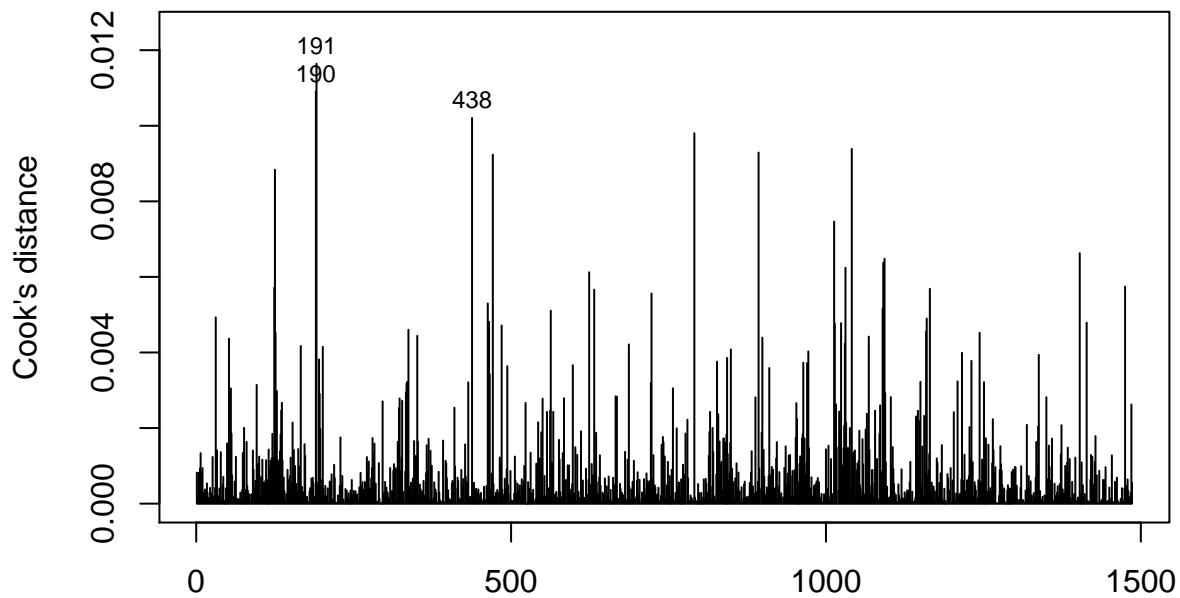




Residuals vs Leverage



Leverage
 $\text{lm}(\text{TARGET_WINS} \sim .)$
 Cook's distance



Obs. number
 $\text{lm}(\text{TARGET_WINS} \sim .)$

```
##  

## Call:  

## lm(formula = TARGET_WINS ~ ., data = cordata[, simple.rev.Pred])  

##  

## Residuals:  

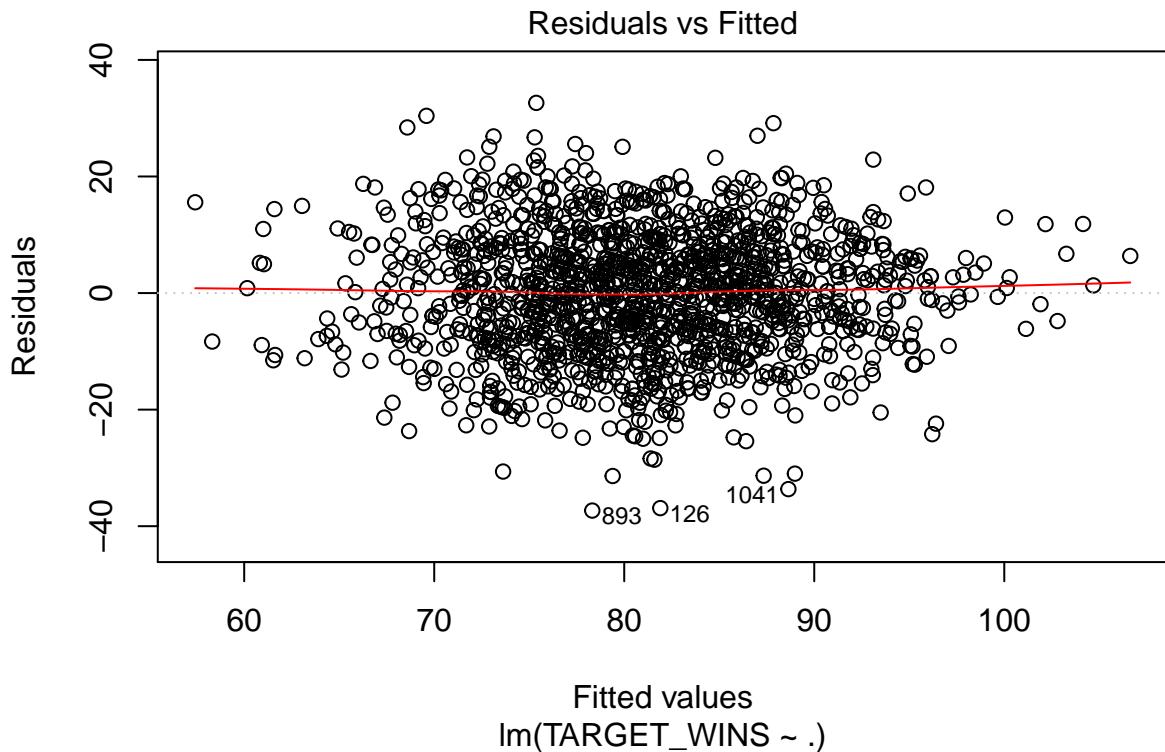
##      Min       1Q   Median       3Q      Max  

##
```

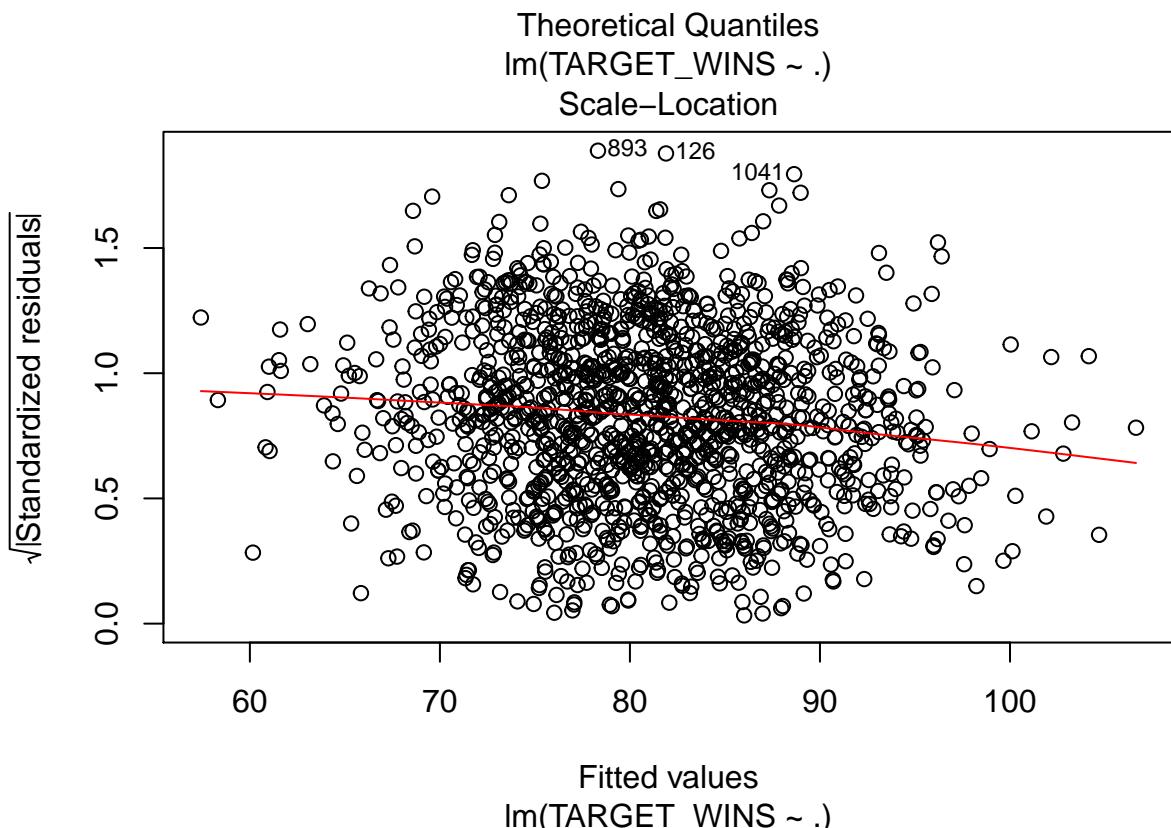
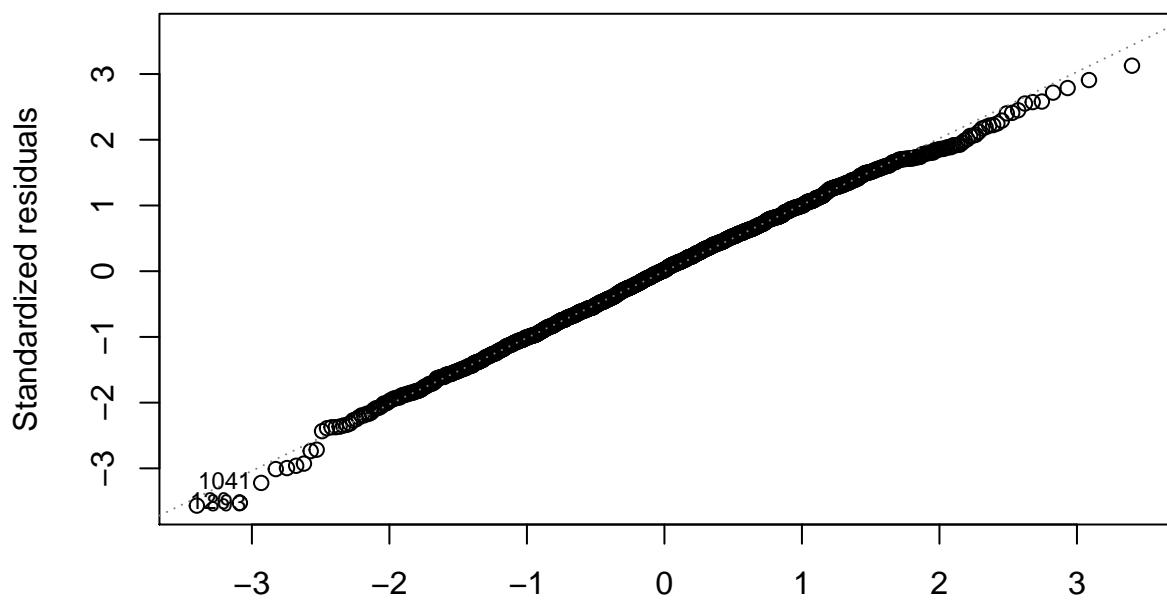
```

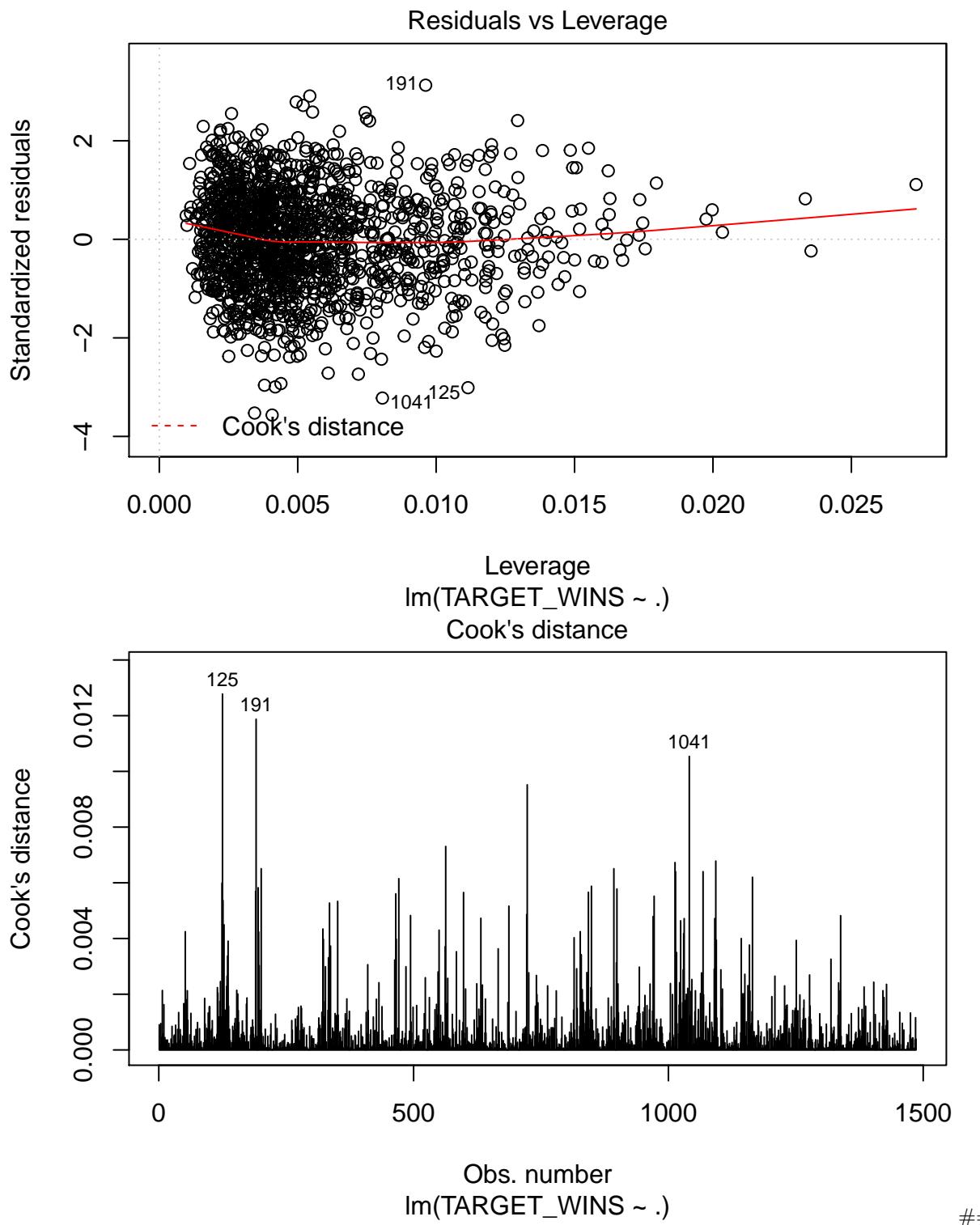
## -37.318   -7.166    0.248    7.110   32.629
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 34.124308  5.721631  5.964 3.07e-09 ***
## TEAM_BATTING_H  0.012656  0.003639  3.478 0.00052 *** 
## TEAM_BATTING_3B  0.099892  0.023008  4.342 1.51e-05 ***
## TEAM_BATTING_HR  0.116681  0.010430 11.187 < 2e-16 ***
## TEAM_BATTING_BB  0.035491  0.003624  9.794 < 2e-16 ***
## TEAM_BATTING_SO -0.019570  0.002469 -7.926 4.43e-15 ***
## TEAM_BASERUN_SB  0.070751  0.006691 10.574 < 2e-16 ***
## BathR_FilterTRUE -5.489045  1.245679 -4.406 1.13e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.48 on 1478 degrees of freedom
## Multiple R-squared:  0.321, Adjusted R-squared:  0.3178
## F-statistic: 99.81 on 7 and 1478 DF, p-value: < 2.2e-16
##   TEAM_BATTING_H TEAM_BATTING_3B TEAM_BATTING_HR TEAM_BATTING_BB
##   1.947511      2.486960      3.472341      1.151583
## TEAM_BATTING_SO TEAM_BASERUN_SB BathR_Filter
## 3.306176      1.189305      1.686170

```



Normal Q–Q





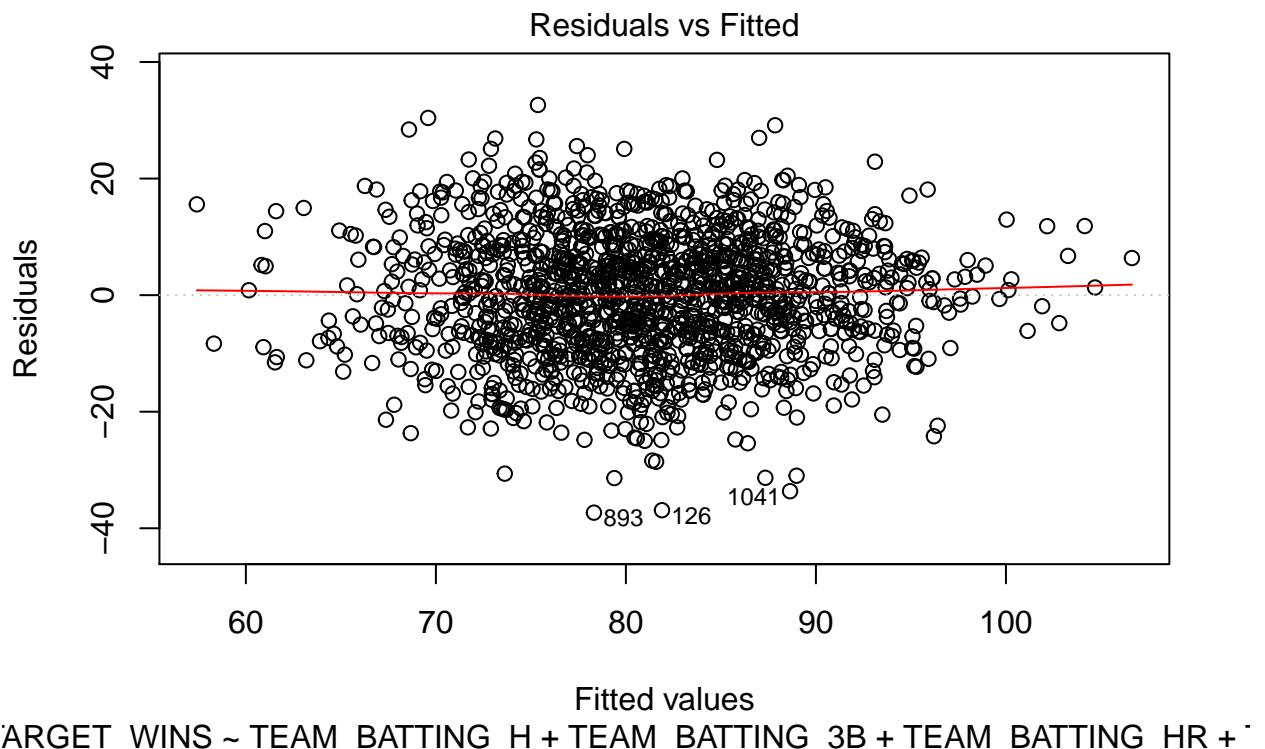
3.1 Automated Variable selection

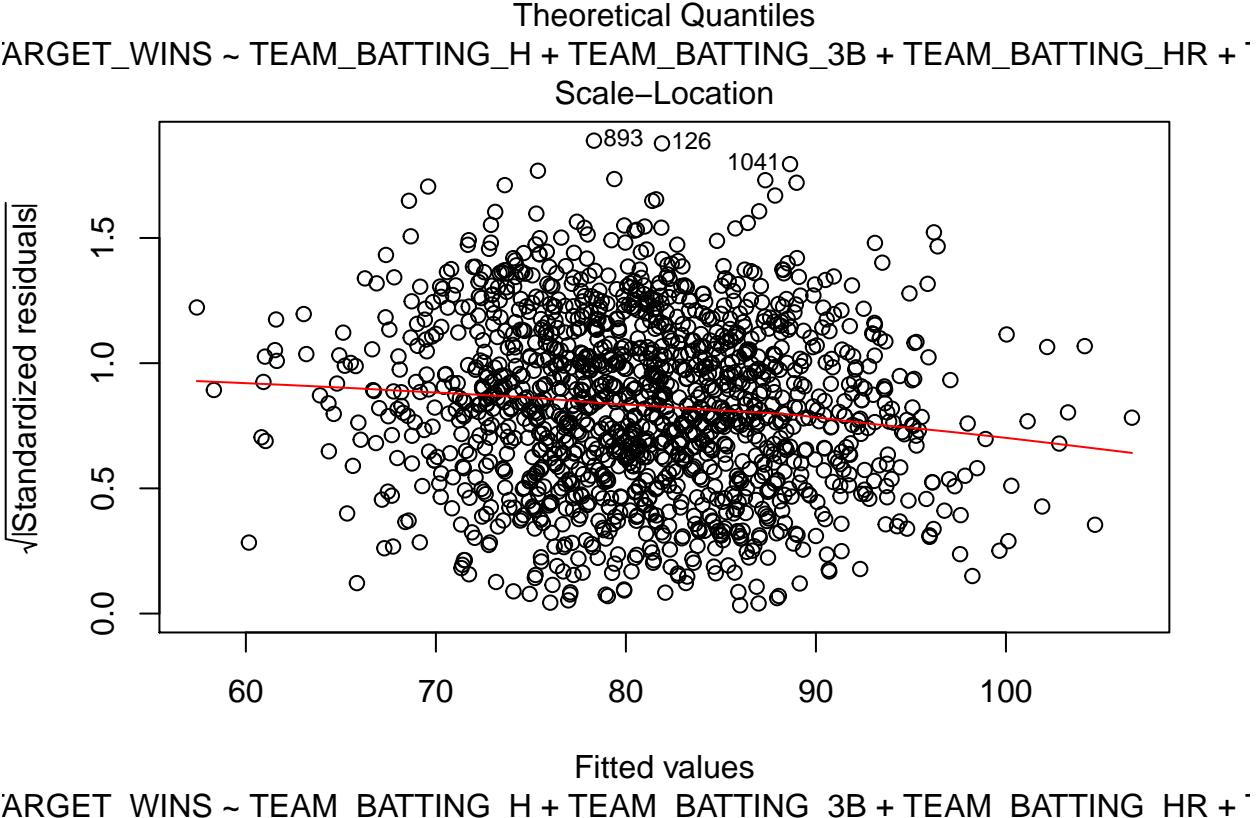
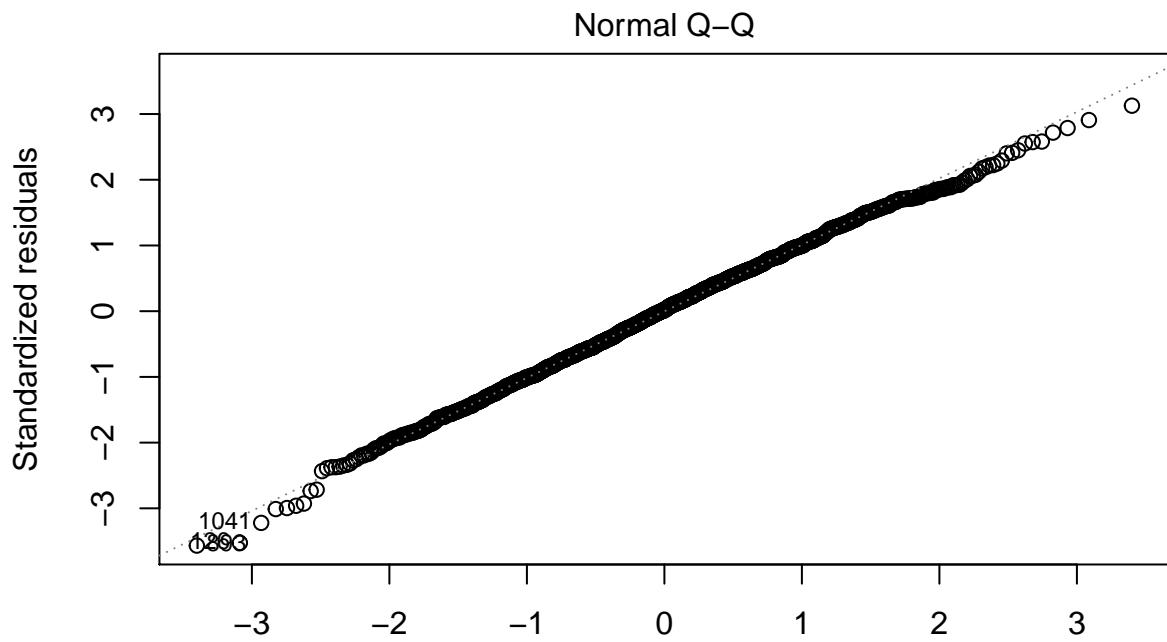
```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_3B +
##     TEAM_BATTING_HR + TEAM_BATTING_BB + TEAM_BATTING_SO + TEAM_BASERUN_SB +
##     BatHR_Filter, data = cordata[, simple.rev.Pred])
```

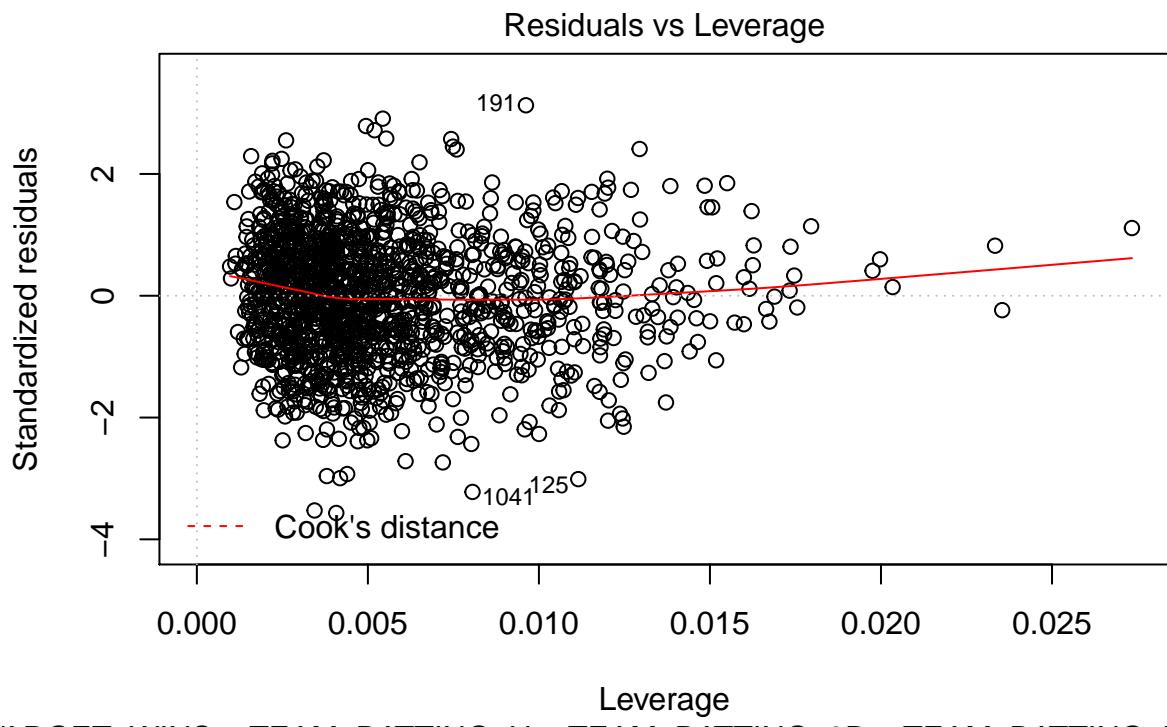
```

##
## Residuals:
##      Min       1Q   Median      3Q      Max
## -37.318  -7.166   0.248   7.110  32.629
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)            34.124308   5.721631   5.964 3.07e-09 ***
## TEAM_BATTING_H          0.012656   0.003639   3.478  0.00052 ***
## TEAM_BATTING_3B         0.099892   0.023008   4.342 1.51e-05 ***
## TEAM_BATTING_HR         0.116681   0.010430  11.187 < 2e-16 ***
## TEAM_BATTING_BB         0.035491   0.003624   9.794 < 2e-16 ***
## TEAM_BATTING_SO        -0.019570   0.002469  -7.926 4.43e-15 ***
## TEAM_BASERUN_SB         0.070751   0.006691  10.574 < 2e-16 ***
## BatHR_FilterTRUE      -5.489045   1.245679  -4.406 1.13e-05 ***
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.48 on 1478 degrees of freedom
## Multiple R-squared:  0.321, Adjusted R-squared:  0.3178
## F-statistic: 99.81 on 7 and 1478 DF, p-value: < 2.2e-16

```



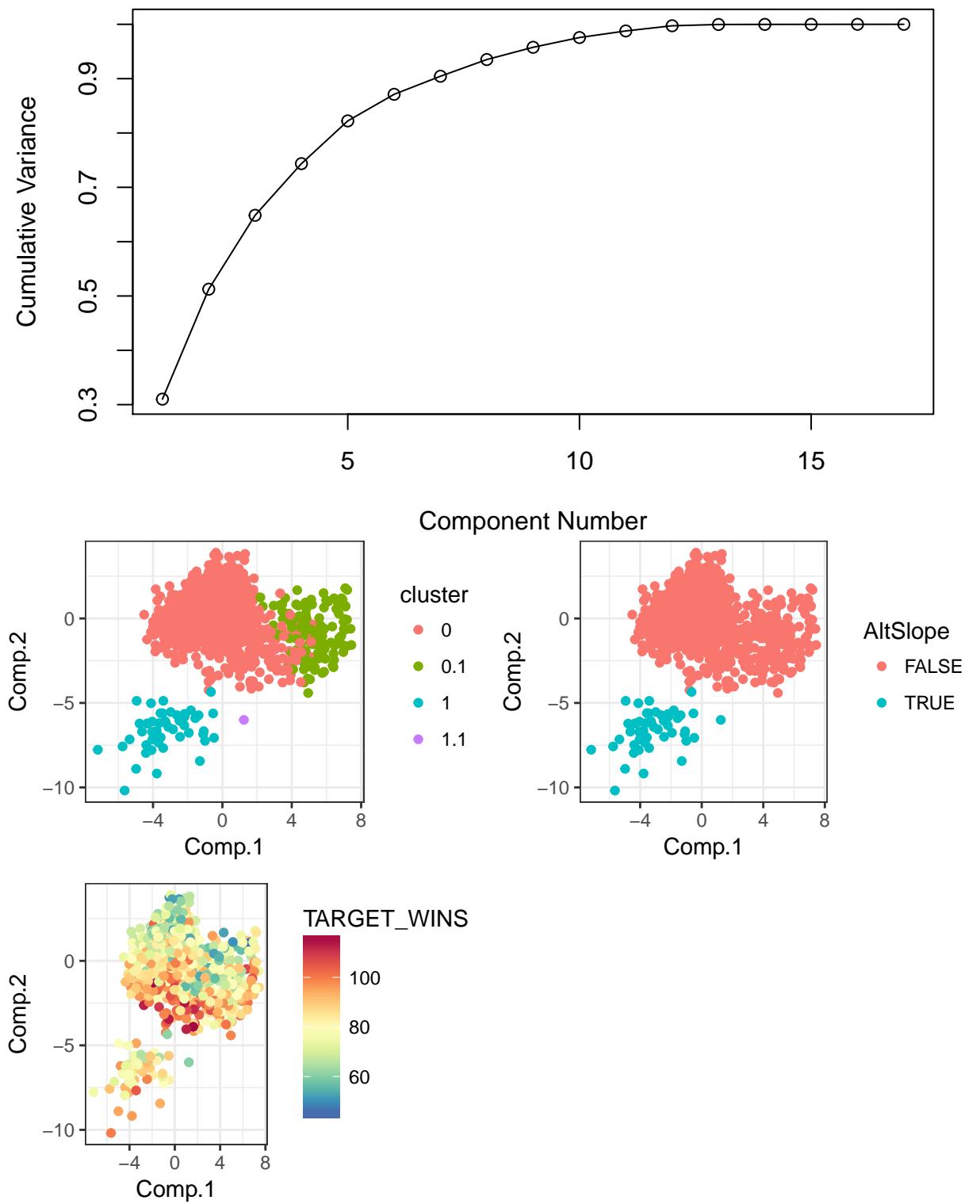




ARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_3B + TEAM_BATTING_HR +

3.2 PCA

```
## Importance of components:
##                         Comp.1    Comp.2    Comp.3    Comp.4    Comp.5
## Standard deviation     2.2961654 1.8549068 1.5203573 1.27137742 1.15632108
## Proportion of Variance 0.3101397 0.2023929 0.1359698 0.09508239 0.07865167
## Cumulative Proportion  0.3101397 0.5125326 0.6485024 0.74358479 0.82223647
##                         Comp.6    Comp.7    Comp.8    Comp.9
## Standard deviation     0.91188783 0.75215016 0.72295091 0.61905022
## Proportion of Variance 0.04891408 0.03327823 0.03074459 0.02254254
## Cumulative Proportion  0.87115055 0.90442878 0.93517337 0.95771591
##                         Comp.10   Comp.11   Comp.12   Comp.13
## Standard deviation     0.55246457 0.45403972 0.402273828 0.201145815
## Proportion of Variance 0.01795395 0.01212659 0.009519073 0.002379979
## Cumulative Proportion  0.97566985 0.98779644 0.997315517 0.999695496
##                         Comp.14   Comp.15   Comp.16   Comp.17
## Standard deviation     0.0518176699 3.903839e-02 3.110471e-02 6.756102e-09
## Proportion of Variance 0.0001579453 8.964682e-05 5.691192e-05 2.684995e-18
## Cumulative Proportion  0.9998534413 9.999431e-01 1.000000e+00 1.000000e+00
```

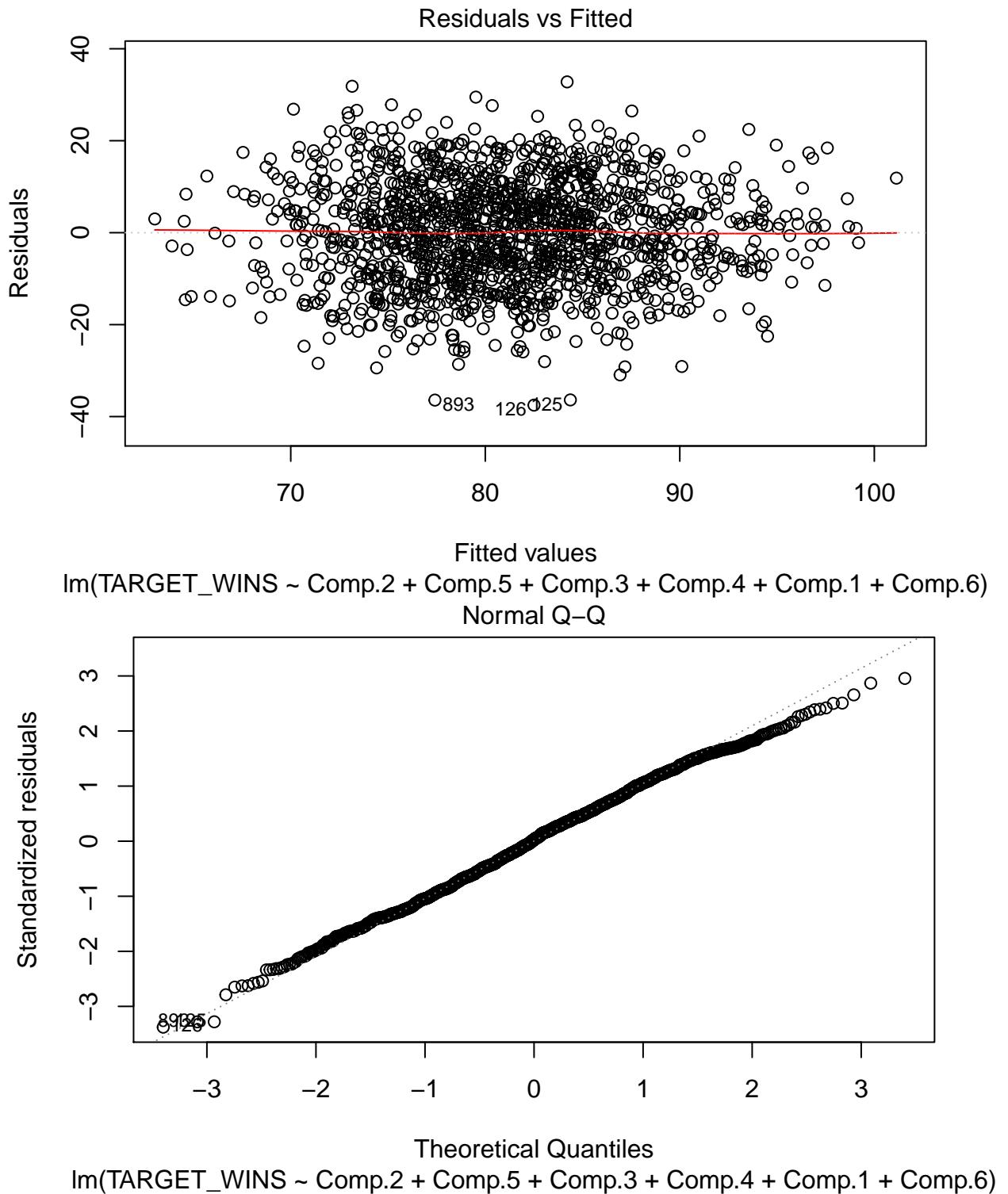


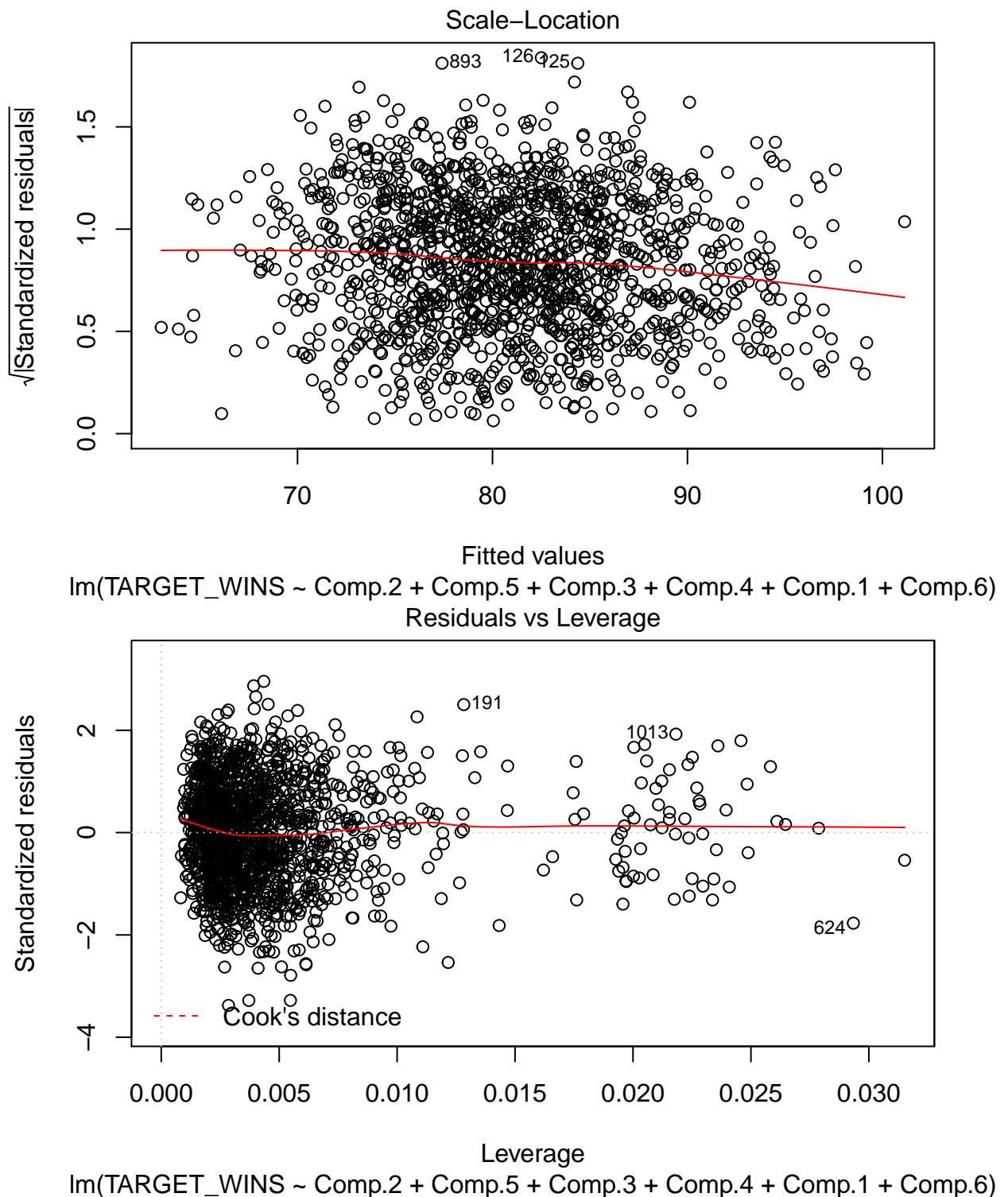
```
##
## Call:
## lm(formula = TARGET_WINS ~ Comp.2 + Comp.5 + Comp.3 + Comp.4 +
##     Comp.1 + Comp.6, data = dfclust[, c(1:6, 10)])
##
## Residuals:
```

```

##      Min     1Q Median     3Q    Max
## -37.505 -7.796  0.259  7.867 32.792
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 80.9966   0.2885 280.778 < 2e-16 ***
## Comp.2      -1.8439   0.1555 -11.856 < 2e-16 ***
## Comp.5      -2.4834   0.2495 -9.954 < 2e-16 ***
## Comp.3       1.7658   0.1897  9.306 < 2e-16 ***
## Comp.4       2.0729   0.2269  9.136 < 2e-16 ***
## Comp.1      -0.8234   0.1256 -6.554 7.73e-11 ***
## Comp.6      -0.5721   0.3163 -1.808  0.0708 .
##
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.12 on 1479 degrees of freedom
## Multiple R-squared: 0.2356, Adjusted R-squared: 0.2325
## F-statistic: 75.99 on 6 and 1479 DF, p-value: < 2.2e-16
##
## Analysis of Variance Table
##
## Response: TARGET_WINS
##           Df Sum Sq Mean Sq F value    Pr(>F)
## Comp.2      1 17383 17383.4 140.5754 < 2.2e-16 ***
## Comp.5      1 12253 12253.4  99.0902 < 2.2e-16 ***
## Comp.3      1 10710 10709.5  86.6055 < 2.2e-16 ***
## Comp.4      1 10321 10321.3  83.4662 < 2.2e-16 ***
## Comp.1      1  5311  5311.4  42.9521 7.726e-11 ***
## Comp.6      1   404   404.4   3.2702  0.07075 .
##
## Residuals 1479 182892   123.7
##
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```



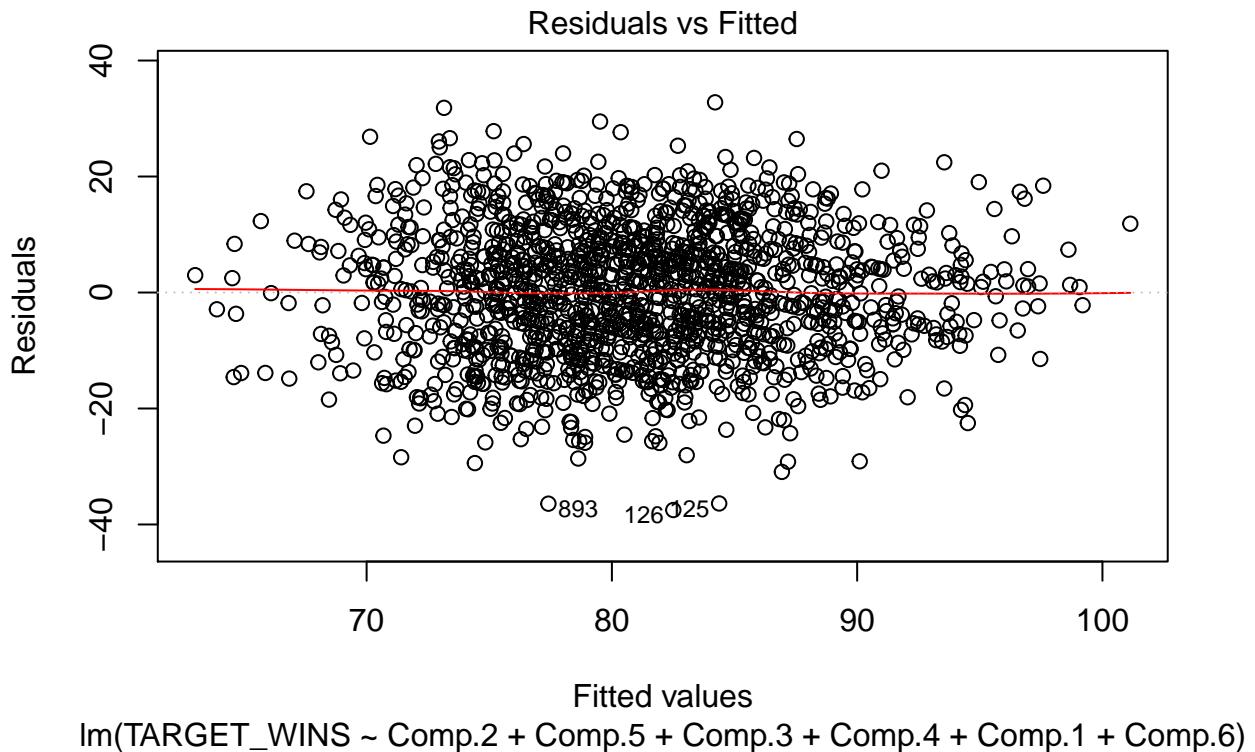


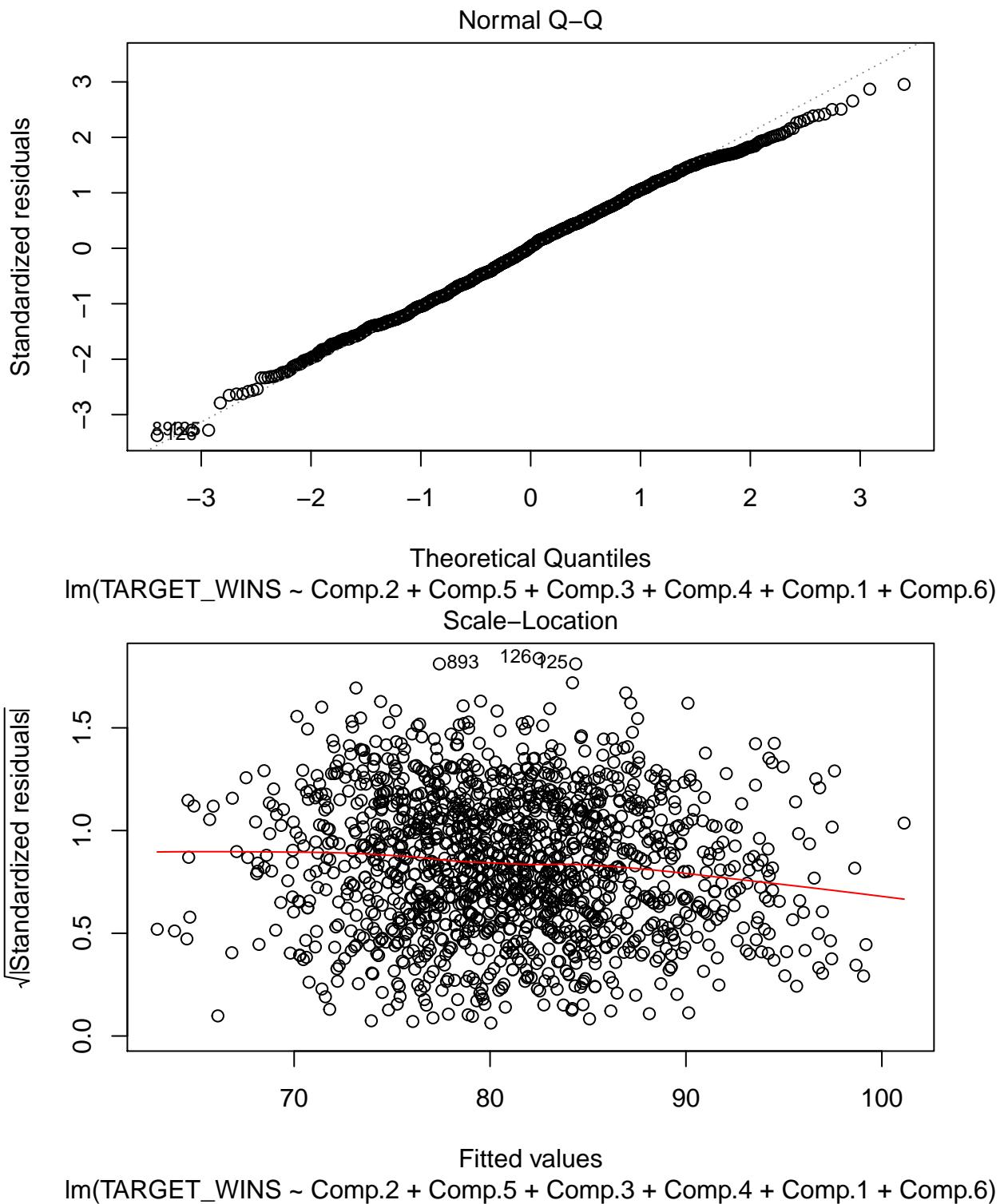
```
## 
## Call:
## lm(formula = TARGET_WINS ~ Comp.2 + Comp.5 + Comp.3 + Comp.4 +
##     Comp.1 + Comp.6, data = dfclust[, c(1:6, 10)])
## 
## Residuals:
```

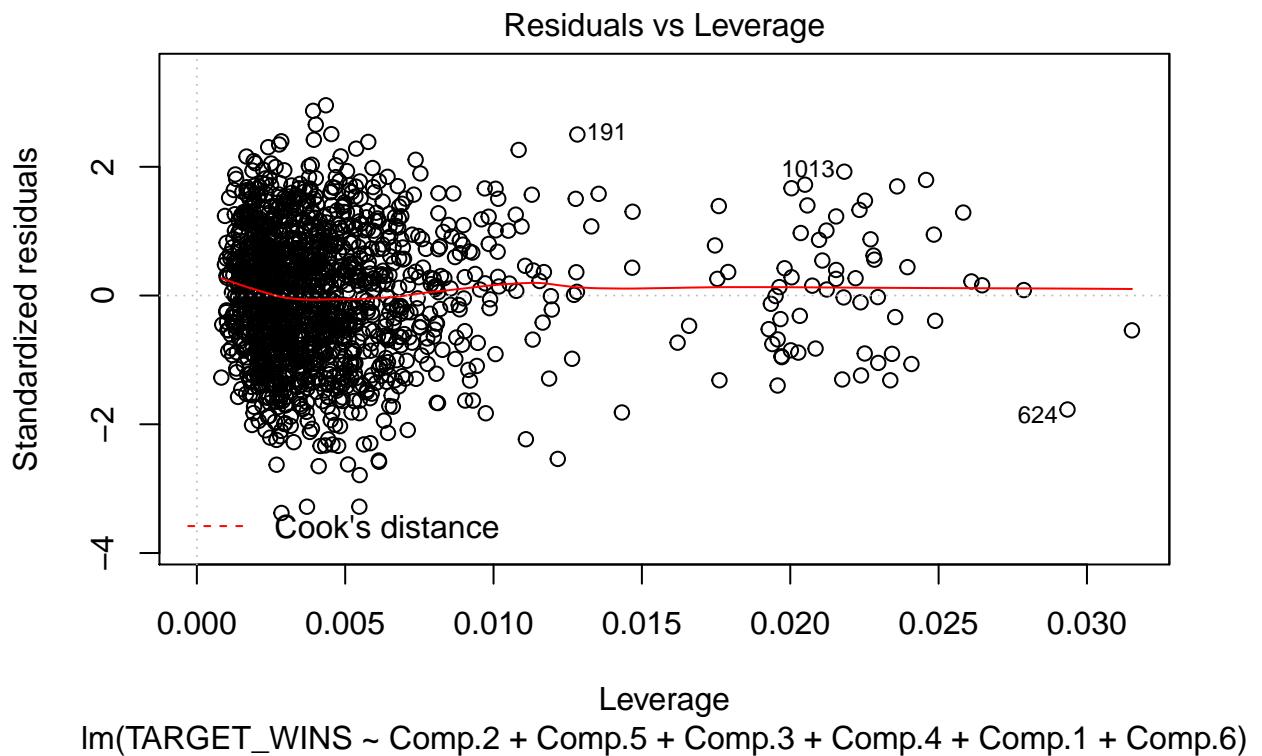
```

##      Min     1Q   Median     3Q    Max
## -37.505 -7.796   0.259   7.867 32.792
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 80.9966   0.2885 280.778 < 2e-16 ***
## Comp.2      -1.8439   0.1555 -11.856 < 2e-16 ***
## Comp.5      -2.4834   0.2495 -9.954 < 2e-16 ***
## Comp.3       1.7658   0.1897  9.306 < 2e-16 ***
## Comp.4       2.0729   0.2269  9.136 < 2e-16 ***
## Comp.1      -0.8234   0.1256 -6.554 7.73e-11 ***
## Comp.6      -0.5721   0.3163 -1.808   0.0708 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.12 on 1479 degrees of freedom
## Multiple R-squared:  0.2356, Adjusted R-squared:  0.2325
## F-statistic: 75.99 on 6 and 1479 DF, p-value: < 2.2e-16

```

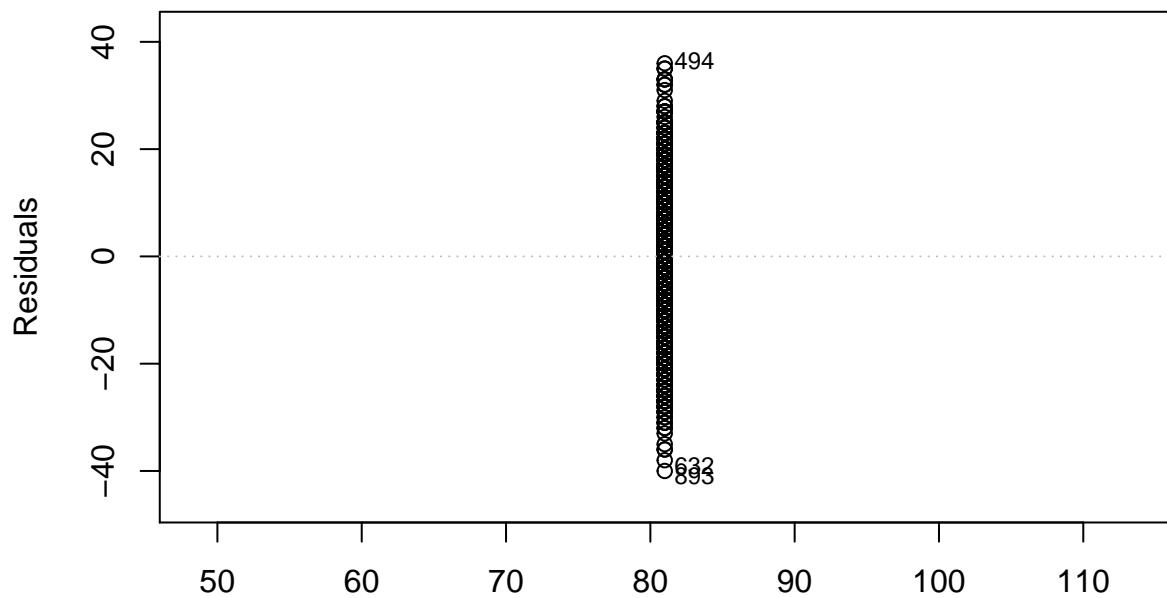




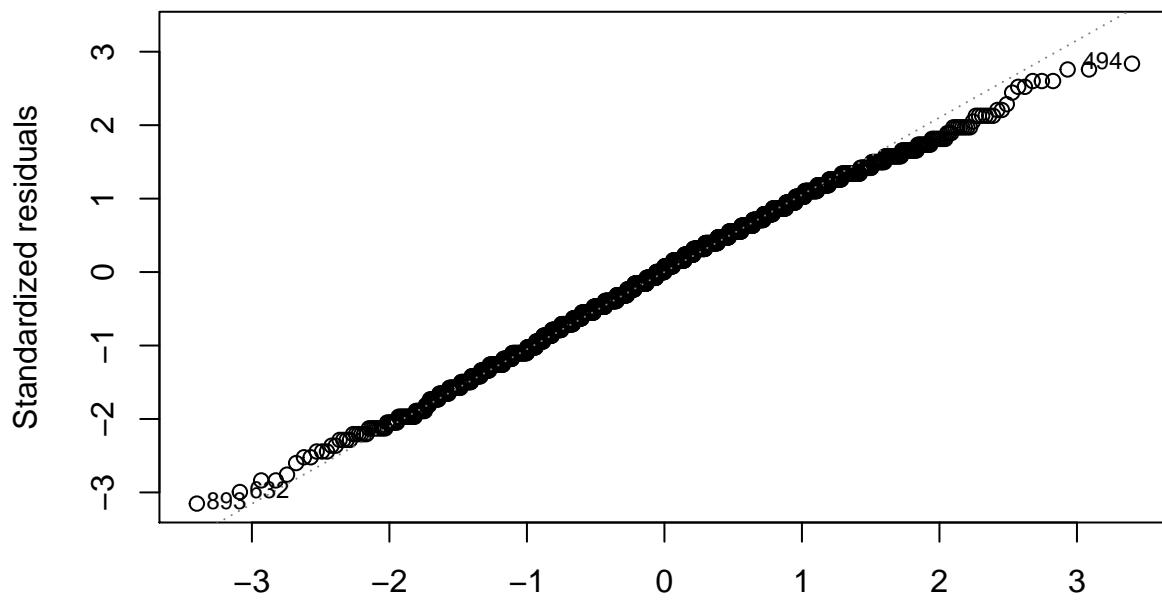


```
##
## Call:
## lm(formula = TARGET_WINS ~ 1, data = dfclust[, c(1:6, 10)])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -39.997  -8.997   0.503   9.003  36.003
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 80.9966    0.3293   246    <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.69 on 1485 degrees of freedom
```

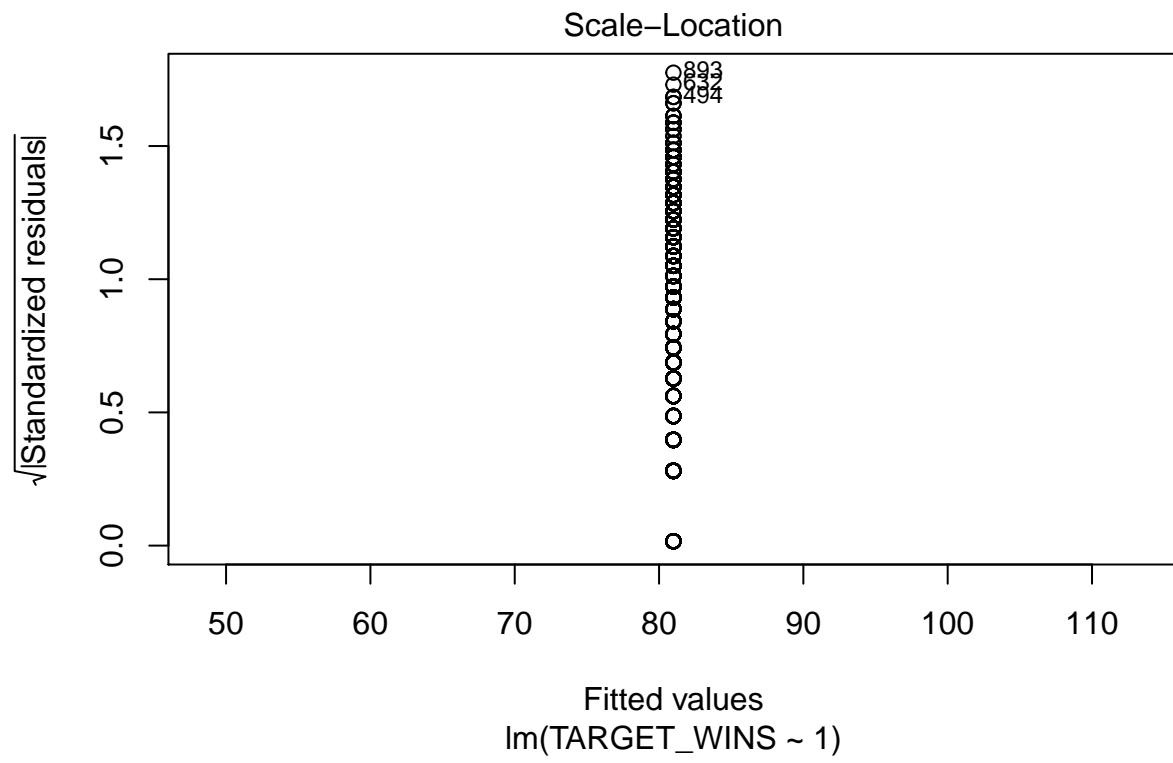
Residuals vs Fitted



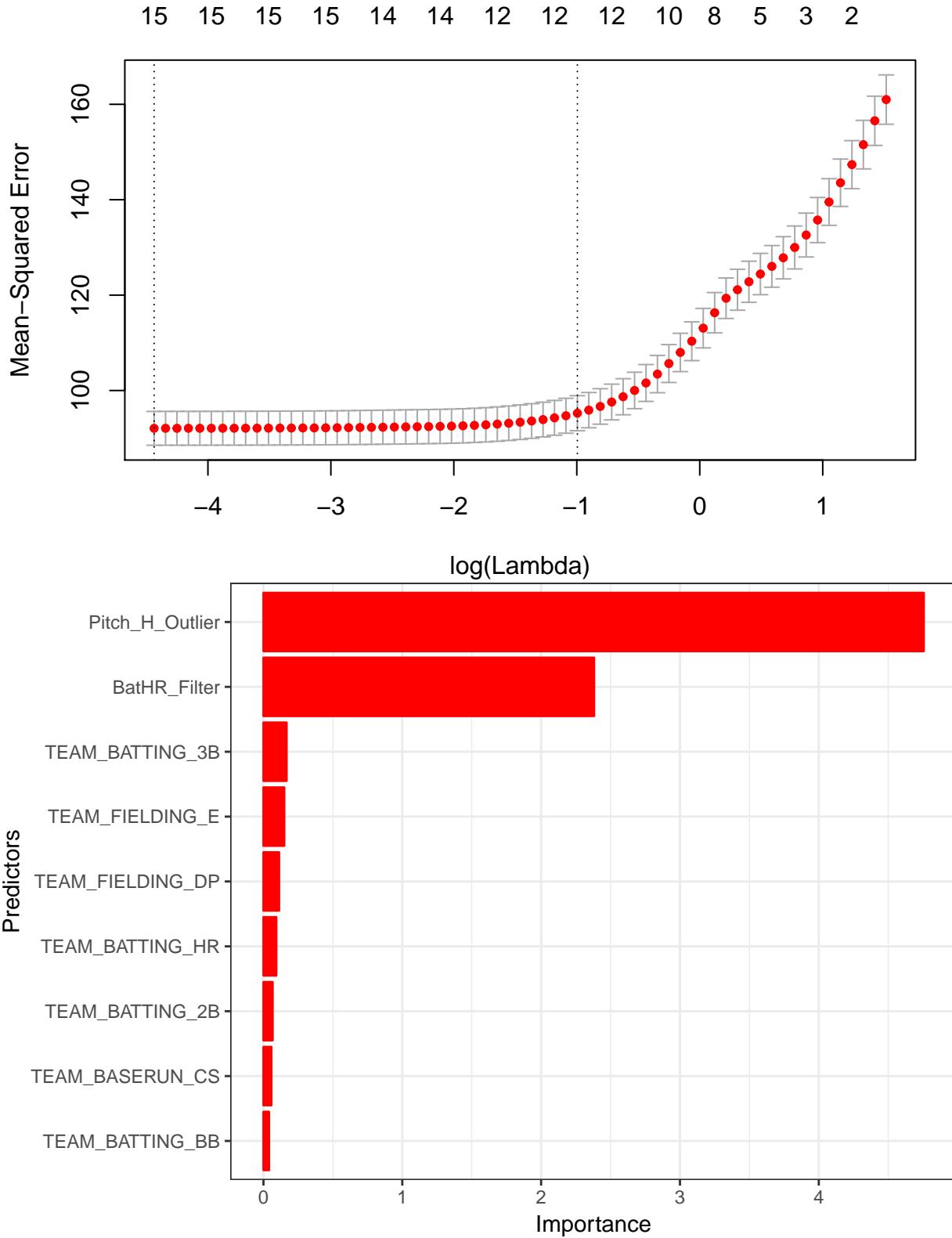
Fitted values
Im(TARGET_WINS ~ 1)
Normal Q-Q



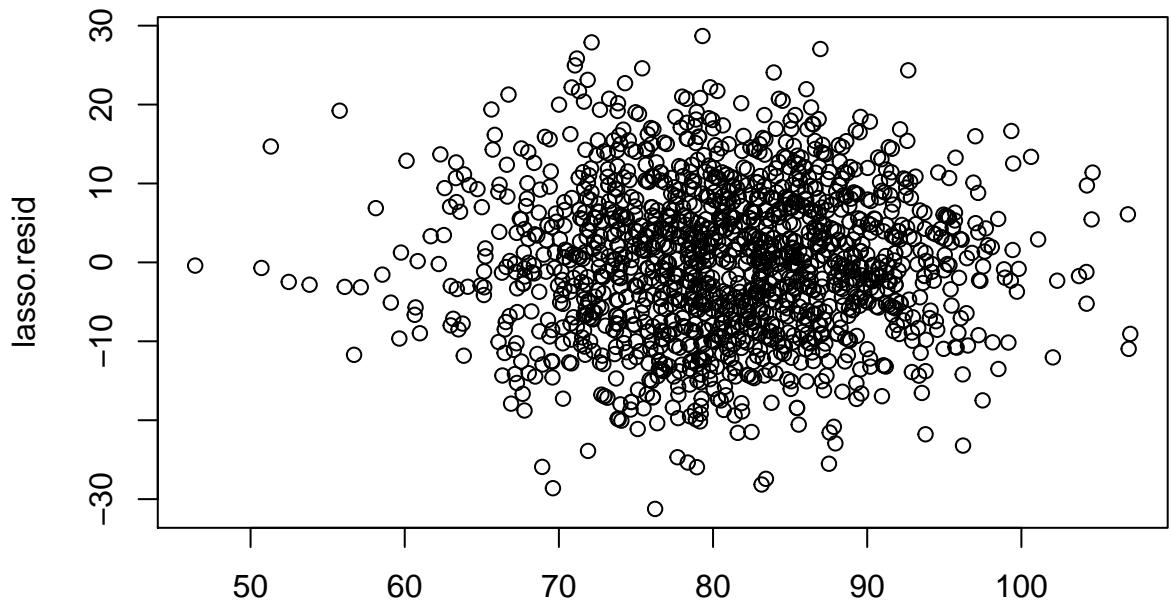
Theoretical Quantiles
Im(TARGET_WINS ~ 1)



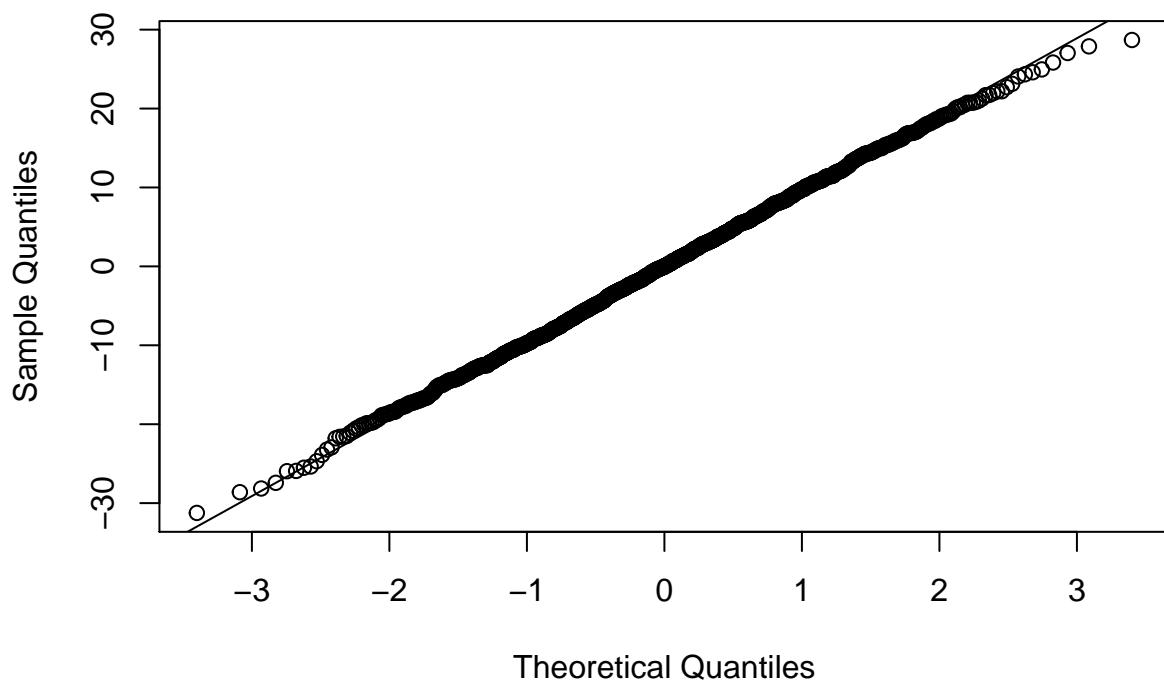
3.2 Lasso



```
## [1] 0.4405071
```



`lasso.pred`
Normal Q-Q Plot



```
##  
## Call:  
## lm(formula = TARGET_WINS ~ ., data = cordata[, c("TARGET_WINS",  
##     Candidate_Lasso)])  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max  
## -34.266  -7.471   0.105    7.259  28.584
```

```

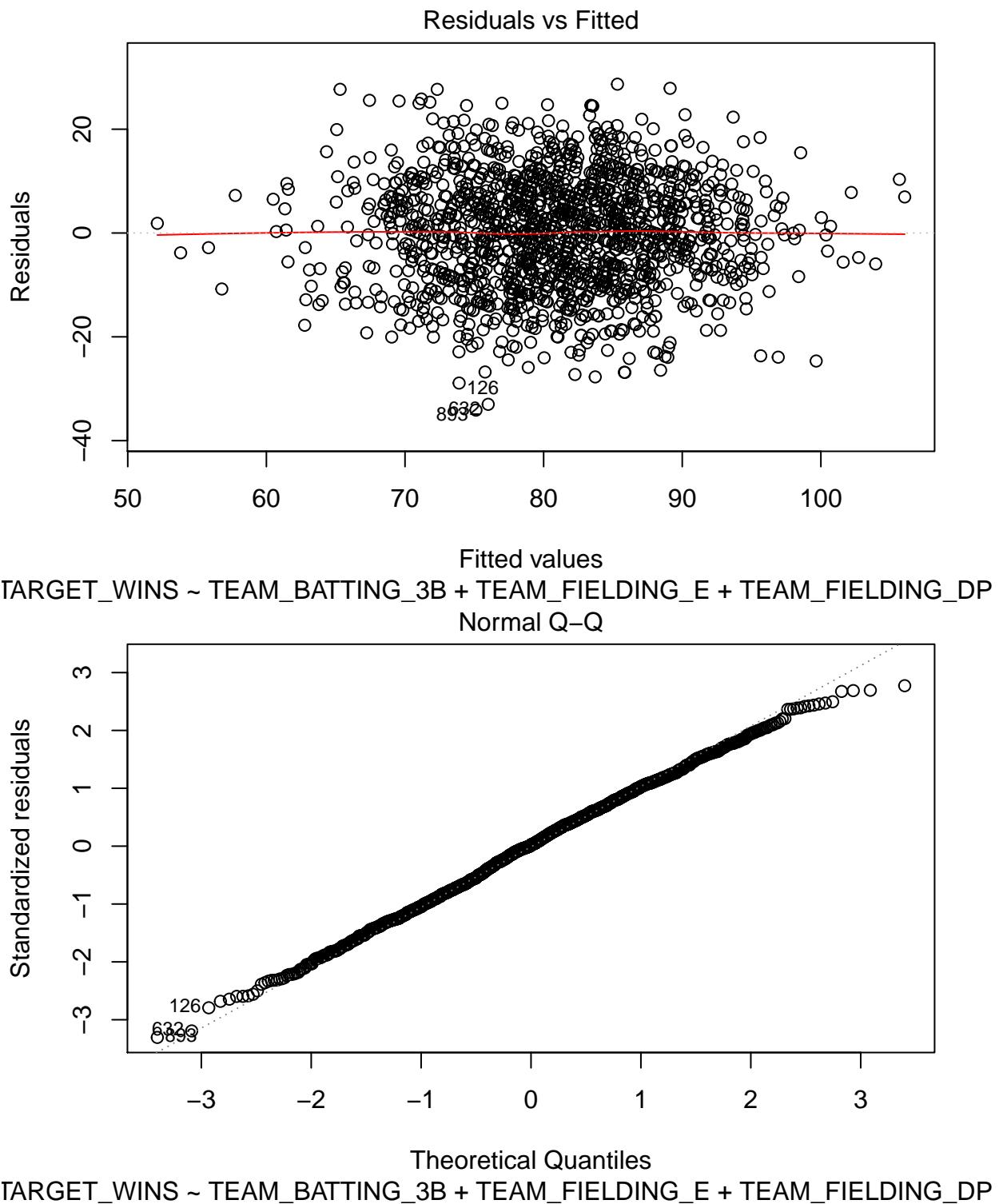
## 
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)      59.425382   3.641236 16.320 < 2e-16 ***
## Pitch_H_OutlierTRUE -1.529950   1.491255 -1.026   0.305    
## BathR_FilterTRUE -1.718587   1.282512 -1.340   0.180    
## TEAM_BATTING_3B    0.337979   0.020725 16.308 < 2e-16 ***  
## TEAM_FIELDING_E   -0.133816   0.010366 -12.909 < 2e-16 ***  
## TEAM_FIELDING_DP  -0.076028   0.013759 -5.526 3.87e-08 ***  
## TEAM_BATTING_HR    0.076531   0.008587  8.913 < 2e-16 ***  
## TEAM_BATTING_2B    -0.010047   0.007205 -1.395   0.163    
## TEAM_BASERUN_CS     0.093695   0.014230  6.584 6.33e-11 ***  
## TEAM_BATTING_BB     0.047501   0.003567 13.317 < 2e-16 ***  
## ---                
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 10.37 on 1476 degrees of freedom
## Multiple R-squared:  0.3367, Adjusted R-squared:  0.3327 
## F-statistic: 83.25 on 9 and 1476 DF,  p-value: < 2.2e-16

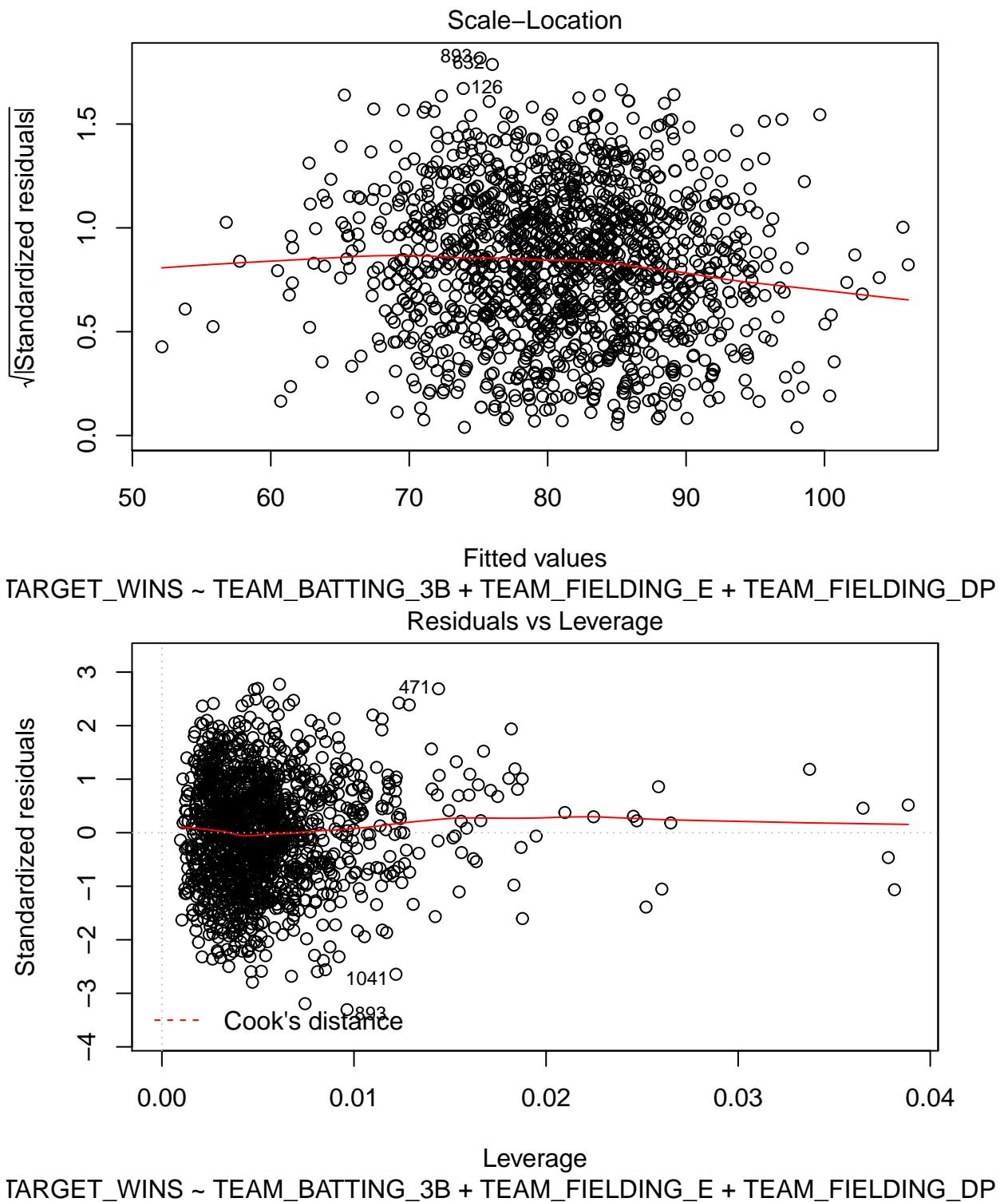
## Start:  AIC=6961.08
## TARGET_WINS ~ Pitch_H_Outlier + BathR_Filter + TEAM_BATTING_3B +
##           TEAM_FIELDING_E + TEAM_FIELDING_DP + TEAM_BATTING_HR + TEAM_BATTING_2B +
##           TEAM_BASERUN_CS + TEAM_BATTING_BB
## 
##                               Df Sum of Sq   RSS   AIC
## - Pitch_H_Outlier     1    113.2 158822 6960.1
## - BathR_Filter        1    193.1 158902 6960.9
## - TEAM_BATTING_2B     1    209.1 158918 6961.0
## <none>                  158709 6961.1
## - TEAM_FIELDING_DP   1    3283.2 161992 6989.5
## - TEAM_BASERUN_CS    1    4661.9 163371 7002.1
## - TEAM_BATTING_HR    1    8541.2 167250 7037.0
## - TEAM_FIELDING_E    1    17917.2 176626 7118.0
## - TEAM_BATTING_BB    1    19068.9 177778 7127.7
## - TEAM_BATTING_3B    1    28596.5 187306 7205.3
## 
## Step:  AIC=6960.14
## TARGET_WINS ~ BathR_Filter + TEAM_BATTING_3B + TEAM_FIELDING_E +
##           TEAM_FIELDING_DP + TEAM_BATTING_HR + TEAM_BATTING_2B + TEAM_BASERUN_CS +
##           TEAM_BATTING_BB
## 
##                               Df Sum of Sq   RSS   AIC
## - BathR_Filter        1    185.1 159007 6959.9
## <none>                  158822 6960.1
## - TEAM_BATTING_2B     1    228.9 159051 6960.3
## + Pitch_H_Outlier     1    113.2 158709 6961.1
## - TEAM_FIELDING_DP   1    3290.3 162113 6988.6
## - TEAM_BASERUN_CS    1    4575.9 163398 7000.4
## - TEAM_BATTING_HR    1    8609.8 167432 7036.6
## - TEAM_FIELDING_E    1    17808.0 176630 7116.1
## - TEAM_BATTING_BB    1    19079.2 177901 7126.7
## - TEAM_BATTING_3B    1    28724.1 187546 7205.2
## 
```

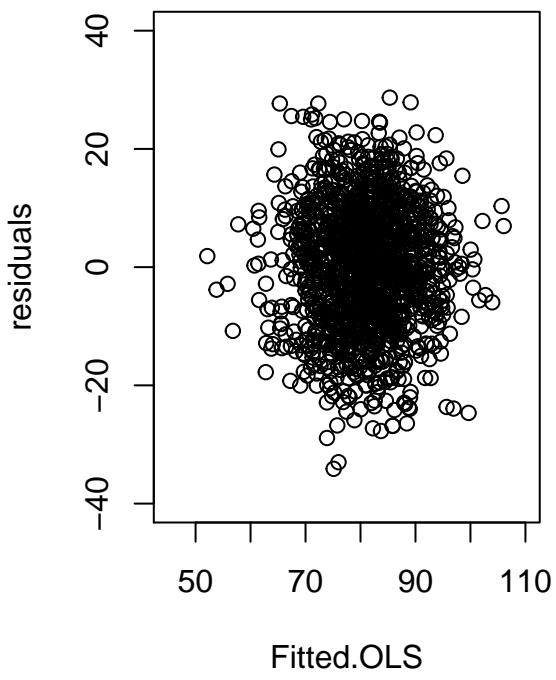
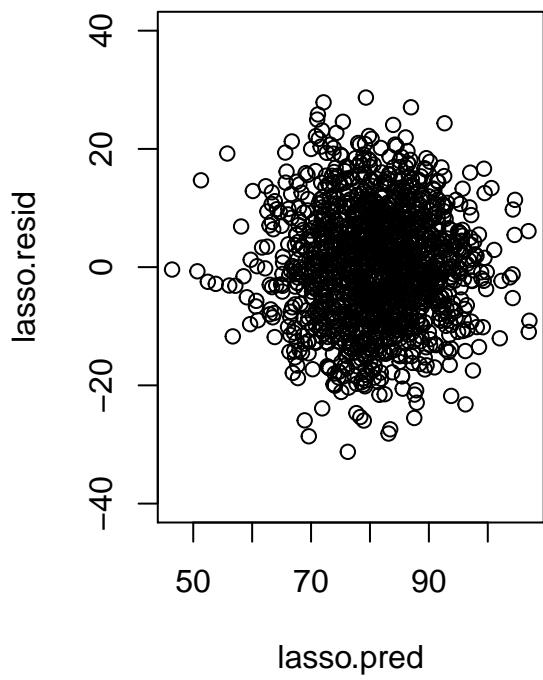
```

## Step: AIC=6959.87
## TARGET_WINS ~ TEAM_BATTING_3B + TEAM_FIELDING_E + TEAM_FIELDING_DP +
##           TEAM_BATTING_HR + TEAM_BATTING_2B + TEAM_BASERUN_CS + TEAM_BATTING_BB
##
##          Df Sum of Sq   RSS   AIC
## <none>             159007 6959.9
## + BathHR_Filter     1     185.1 158822 6960.1
## - TEAM_BATTING_2B   1     295.3 159303 6960.6
## + Pitch_H_Outlier   1     105.2 158902 6960.9
## - TEAM_FIELDING_DP 1     3253.3 162261 6988.0
## - TEAM_BASERUN_CS   1     4395.6 163403 6998.4
## - TEAM_BATTING_HR   1     10413.5 169421 7052.1
## - TEAM_BATTING_BB   1     18980.1 177987 7125.4
## - TEAM_FIELDING_E   1     19384.3 178392 7128.8
## - TEAM_BATTING_3B   1     28554.7 187562 7203.3
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_3B + TEAM_FIELDING_E +
##      TEAM_FIELDING_DP + TEAM_BATTING_HR + TEAM_BATTING_2B + TEAM_BASERUN_CS +
##      TEAM_BATTING_BB, data = cordata[, c("TARGET_WINS", Candidate_Lasso)])
##
## Residuals:
##    Min      1Q  Median      3Q      Max
## -34.118  -7.411   0.201   7.152  28.674
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 59.812925  3.627653 16.488 < 2e-16 ***
## TEAM_BATTING_3B 0.335779  0.020610 16.292 < 2e-16 ***
## TEAM_FIELDING_E -0.135922  0.010126 -13.423 < 2e-16 ***
## TEAM_FIELDING_DP -0.075655  0.013758 -5.499 4.49e-08 ***
## TEAM_BATTING_HR  0.080295  0.008161  9.838 < 2e-16 ***
## TEAM_BATTING_2B -0.011802  0.007124 -1.657  0.0978 .
## TEAM_BASERUN_CS  0.089381  0.013983  6.392 2.19e-10 ***
## TEAM_BATTING_BB  0.047367  0.003566 13.282 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.37 on 1478 degrees of freedom
## Multiple R-squared:  0.3355, Adjusted R-squared:  0.3323
## F-statistic: 106.6 on 7 and 1478 DF,  p-value: < 2.2e-16

```



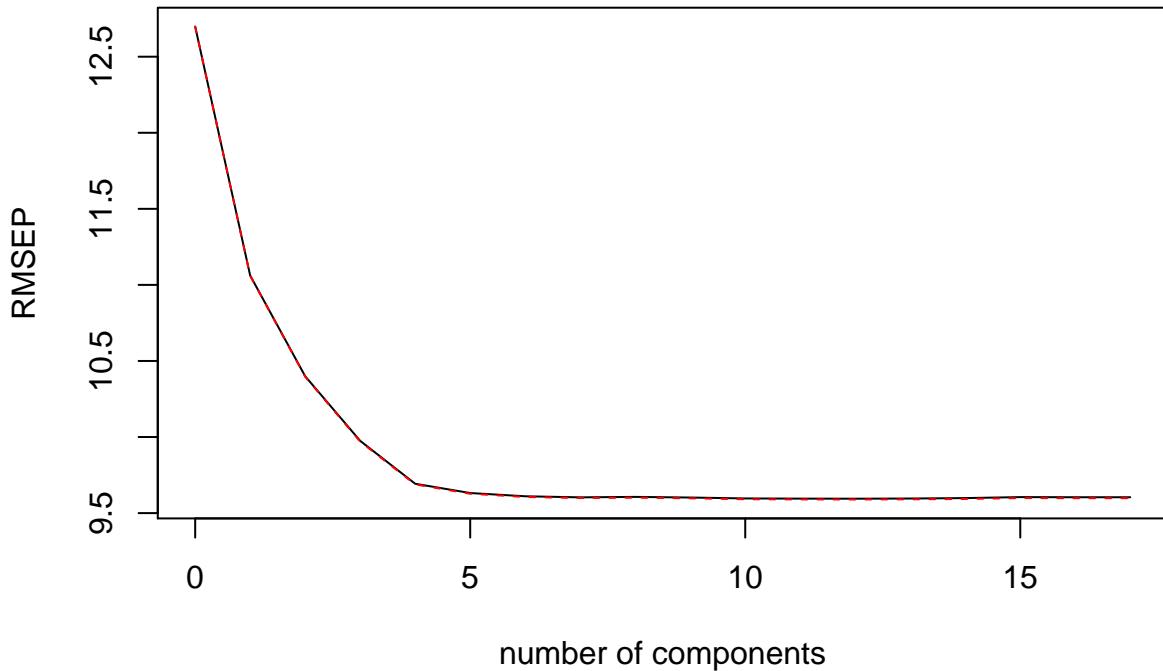




3.3 Partial Least Squares (PLS) Regression

How to add indicator variables to the model?

TARGET_WINS

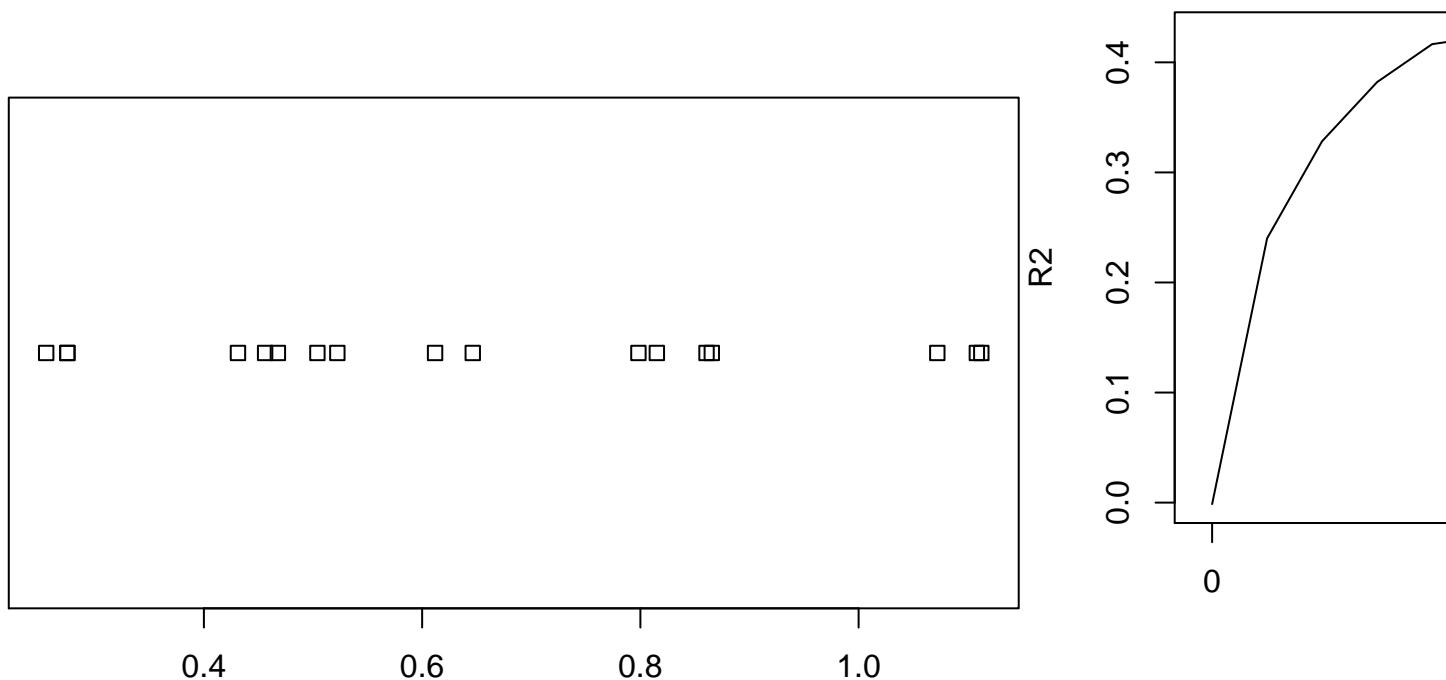


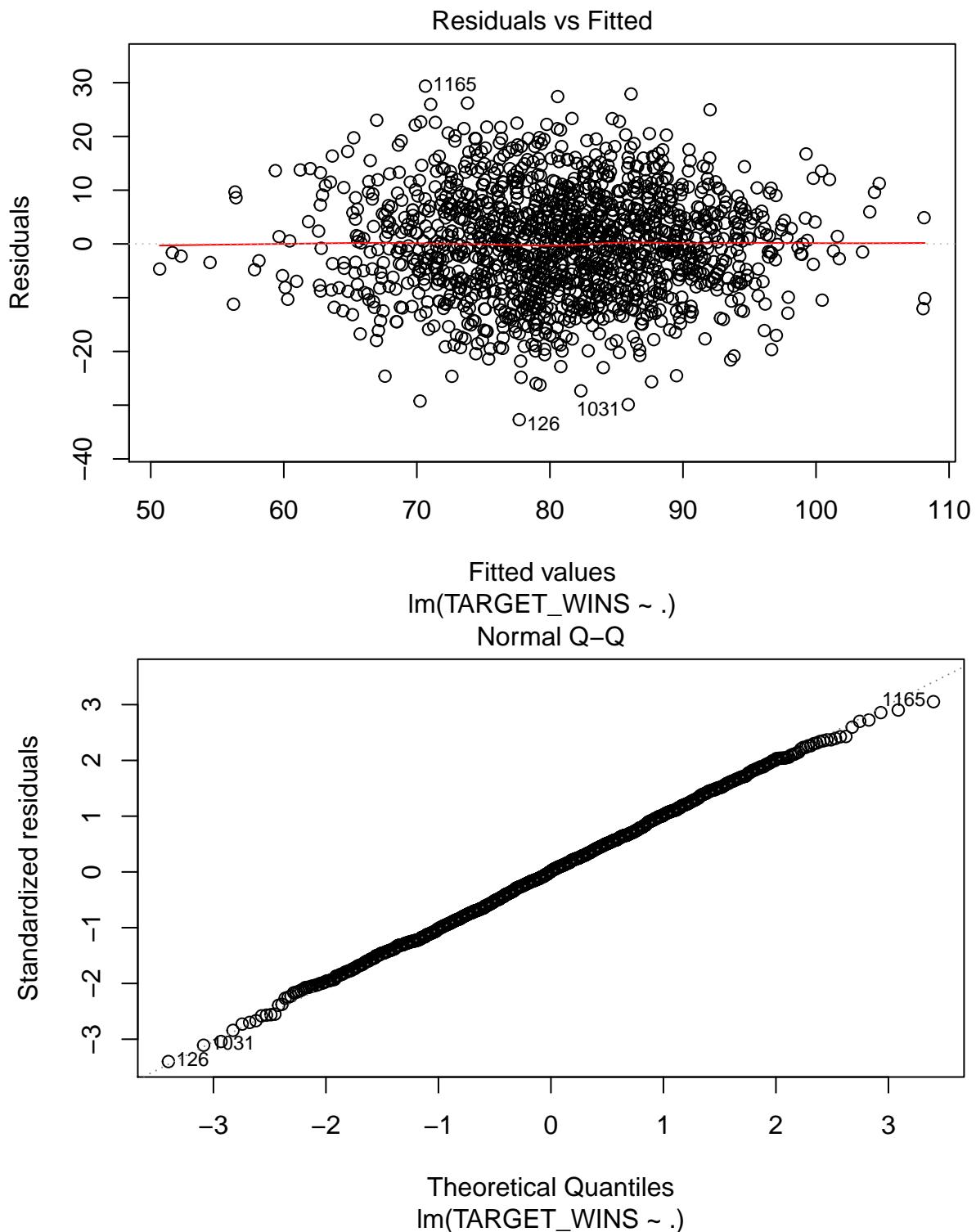
```
## Data:      X dimension: 1486 17
##   Y dimension: 1486 1
## Fit method: kernelpls
```

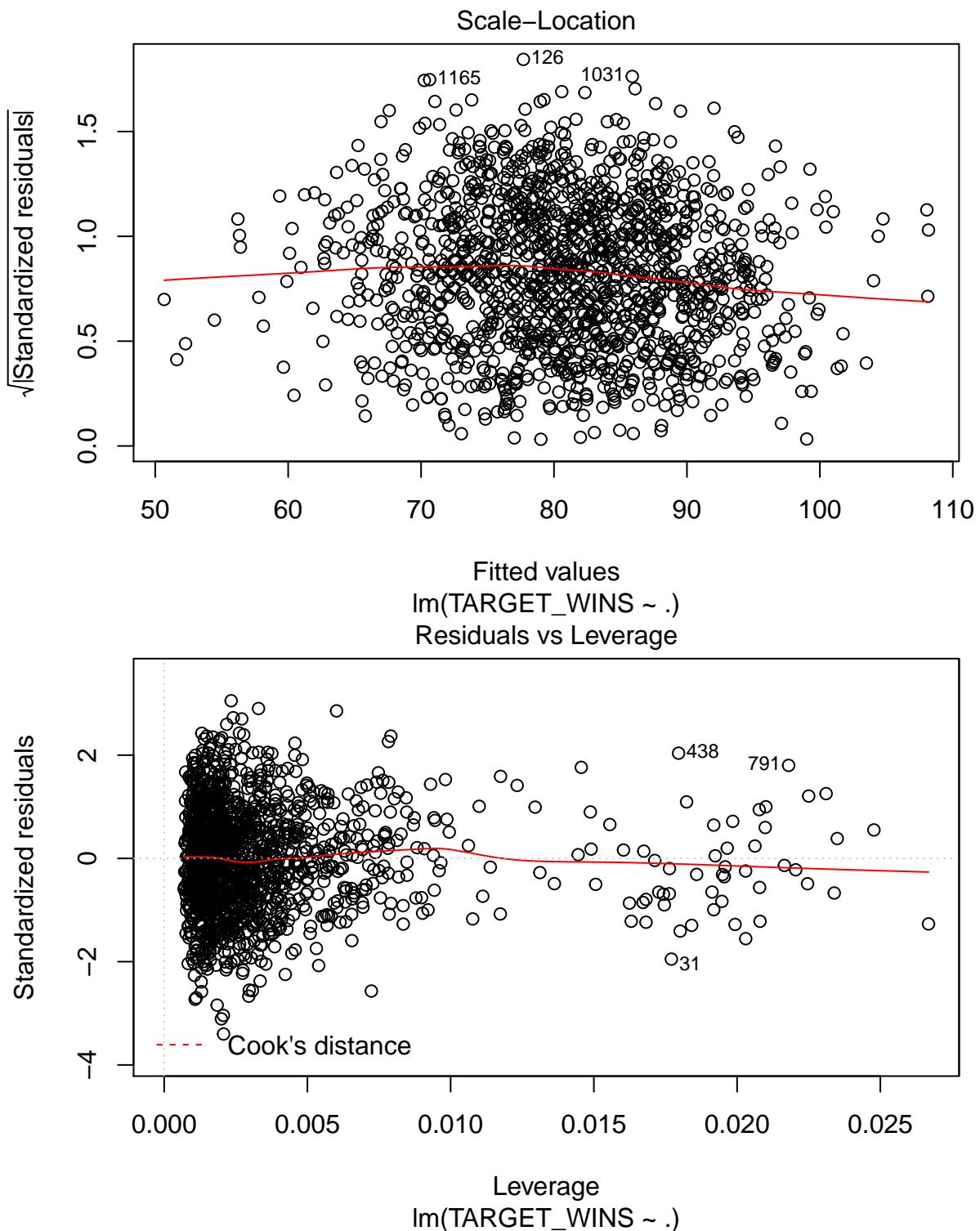
```

## Number of components considered: 17
##
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##      (Intercept) 1 comps 2 comps 3 comps 4 comps 5 comps 6 comps
## CV          12.7    11.06   10.4    9.975   9.693   9.632   9.611
## adjCV       12.7    11.06   10.4    9.970   9.688   9.627   9.606
##      7 comps 8 comps 9 comps 10 comps 11 comps 12 comps 13 comps
## CV          9.604   9.607   9.602   9.596   9.595   9.595   9.596
## adjCV       9.599   9.601   9.597   9.591   9.590   9.589   9.591
##      14 comps 15 comps 16 comps 17 comps
## CV          9.599   9.605   9.604   9.604
## adjCV       9.593   9.599   9.598   9.598
##
## TRAINING: % variance explained
##      1 comps 2 comps 3 comps 4 comps 5 comps 6 comps 7 comps
## X           21.27   40.91   56.24   65.64   75.42   80.57   87.18
## TARGET_WINS 24.72   33.68   39.26   42.66   43.42   43.71   43.79
##      8 comps 9 comps 10 comps 11 comps 12 comps 13 comps
## X           89.66   92.85   94.64   96.31   98.85   99.84
## TARGET_WINS 43.87   43.94   43.99   44.00   44.01   44.02
##      14 comps 15 comps 16 comps 17 comps
## X           99.98   99.99   100.00  100.09
## TARGET_WINS 44.08   44.09   44.09   44.09

```







```
##  

## Call:  

## lm(formula = TARGET_WINS ~ ., data = pls.df)  

##  

## Residuals:  

##      Min       1Q   Median       3Q      Max  

##
```

```

## -32.697 -6.743 0.106 6.336 29.356
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 80.9966    0.2497 324.38 <2e-16 ***
## Comp.1      3.7604    0.1488  25.27 <2e-16 ***
## Comp.2      2.5600    0.1683  15.21 <2e-16 ***
## Comp.3      2.2174    0.1848  12.00 <2e-16 ***
## Comp.4      2.0707    0.2210   9.37 <2e-16 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.625 on 1481 degrees of freedom
## Multiple R-squared: 0.4266, Adjusted R-squared: 0.425
## F-statistic: 275.4 on 4 and 1481 DF, p-value: < 2.2e-16

## Analysis of Variance Table
##
## Response: TARGET_WINS
##             Df Sum Sq Mean Sq F value    Pr(>F)
## Comp.1      1 59149  59149 638.425 < 2.2e-16 ***
## Comp.2      1 21439  21439 231.398 < 2.2e-16 ***
## Comp.3      1 13343  13343 144.013 < 2.2e-16 ***
## Comp.4      1  8133   8133  87.789 < 2.2e-16 ***
## Residuals 1481 137212       93
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

3.4 Random forest

```

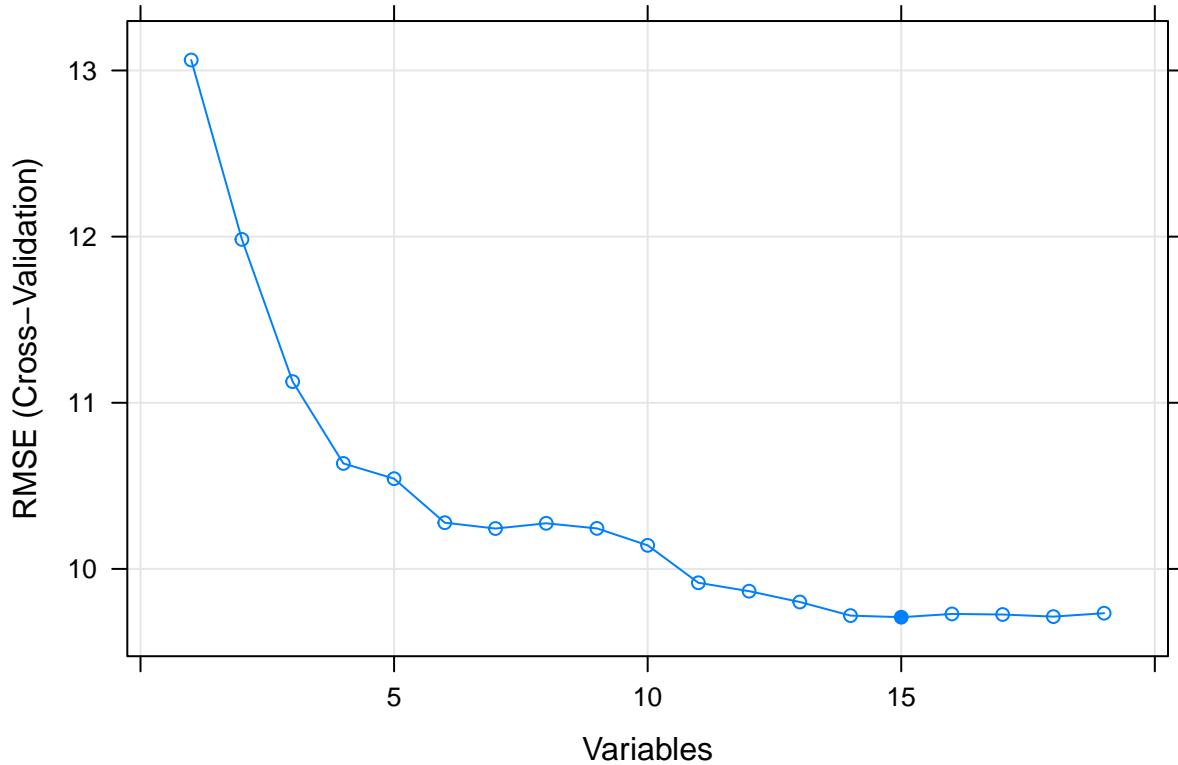
##
## Recursive feature selection
##
## Outer resampling method: Cross-Validated (10 fold)
##
## Resampling performance over subset size:
##
##    Variables    RMSE Rsquared     MAE RMSESD RsquaredSD  MAESD Selected
##          1 13.063  0.03622 10.504 0.4755    0.03789 0.3272
##          2 11.983  0.15098  9.550 0.6078    0.06224 0.5551
##          3 11.127  0.24115  8.993 0.6374    0.06828 0.4756
##          4 10.635  0.30226  8.543 0.4792    0.06385 0.4134
##          5 10.543  0.31626  8.428 0.4910    0.06267 0.3900
##          6 10.278  0.34885  8.256 0.6150    0.07218 0.5269
##          7 10.243  0.35276  8.250 0.5664    0.06724 0.5030
##          8 10.275  0.34896  8.277 0.5838    0.06906 0.5046
##          9 10.244  0.35360  8.240 0.6396    0.07355 0.5536
##         10 10.142  0.36601  8.168 0.6268    0.08073 0.5352
##         11  9.916  0.39694  7.985 0.5436    0.07557 0.4650
##         12  9.866  0.40371  7.936 0.5834    0.07837 0.4895
##         13  9.801  0.41406  7.885 0.5578    0.07594 0.4493
##         14  9.719  0.42714  7.832 0.5543    0.07639 0.4633
##         15  9.709  0.42704  7.839 0.5483    0.07639 0.4599 *
##         16  9.729  0.42517  7.844 0.5564    0.07734 0.4737

```

```

##      17  9.725  0.42635  7.842 0.5596   0.07618 0.4648
##      18  9.713  0.42696  7.836 0.5596   0.07786 0.4703
##      19  9.733  0.42373  7.844 0.5374   0.07304 0.4623
##
## The top 5 variables (out of 15):
##     TEAM_FIELDING_E, TEAM_BATTING_H, TEAM_BATTING_BB, TEAM_PITCHING_SO, TEAM_BATTING_SO
## [1] "TEAM_FIELDING_E"    "TEAM_BATTING_H"      "TEAM_BATTING_BB"
## [4] "TEAM_PITCHING_SO"   "TEAM_BATTING_SO"    "TEAM_BATTING_HR"
## [7] "TEAM_PITCHING_HR"   "TEAM_PITCHING_H"   "TEAM_PITCHING_BB"
## [10] "TEAM_BASERUN_SB"   "TEAM_BATTING_3B"   "TEAM_BATTING_2B"
## [13] "TEAM_FIELDING_DP"   "TEAM_BASERUN_CS"  "BathR_Filter"

```



```

##
## Call:
## lm(formula = form, data = cordata)
##
## Residuals:
##    Min      1Q  Median      3Q     Max
## -39.096 -7.124 -0.104  7.334 35.923
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 33.905310  5.444283  6.228 6.15e-10 ***
## TEAM_FIELDING_E -0.117979  0.008919 -13.228 < 2e-16 ***
## TEAM_BATTING_H   0.036384  0.002865  12.699 < 2e-16 ***
## TEAM_BATTING_BB   0.038722  0.003543  10.929 < 2e-16 ***
## TEAM_PITCHING_SO -0.011319  0.001702  -6.649 4.13e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

## 
## Residual standard error: 10.65 on 1481 degrees of freedom
## Multiple R-squared:  0.2977, Adjusted R-squared:  0.2959
## F-statistic:   157 on 4 and 1481 DF,  p-value: < 2.2e-16

```

4. Imputing and Data Prep for testing

5. Model Selection

Type	Formula	Adj.R.Squared
Simple	TARGET_WINS = 30.52 + 0.03 * TEAM_BATTING_H - 0.05 * TEAM_BATTING_2B + 0.08 * TEAM_BATTING_3B + 0.11 * TEAM_BATTING_HR + 0.04 * TEAM_BATTING_BB - 0.02 * TEAM_BATTING_SO + 0.07 * TEAM_BASERUN_SB - 0.01 * TEAM_BASERUN_CS - 0.09 * TEAM_FIELDING_DP - 0.87 * Pitch_H_Outlier - 5.4 * BatHR_Filter	0.3456
Simple revised	TARGET_WINS = 34.12 + 0.01 * TEAM_BATTING_H + 0.1 * TEAM_BATTING_3B + 0.12 * TEAM_BATTING_HR + 0.04 * TEAM_BATTING_BB - 0.02 * TEAM_BATTING_SO + 0.07 * TEAM_BASERUN_SB - 5.49 * BatHR_Filter	0.3178
Stepwise	TARGET_WINS = 34.12 + 0.01 * TEAM_BATTING_H + 0.1 * TEAM_BATTING_3B + 0.12 * TEAM_BATTING_HR + 0.04 * TEAM_BATTING_BB - 0.02 * TEAM_BATTING_SO + 0.07 * TEAM_BASERUN_SB - 5.49 * BatHR_Filter	0.3178
Prin.Comp.Regression	TARGET_WINS = 81 - 1.84 * Comp.2 - 2.48 * Comp.5 + 1.77 * Comp.3 + 2.07 * Comp.4 - 0.82 * Comp.1 - 0.57 * Comp.6	0.2325
Lasso Predictors	TARGET_WINS = 59.43 - 1.53 * Pitch_H_Outlier - 1.72 * BatHR_Filter + 0.34 * TEAM_BATTING_3B - 0.13 *	0.3327

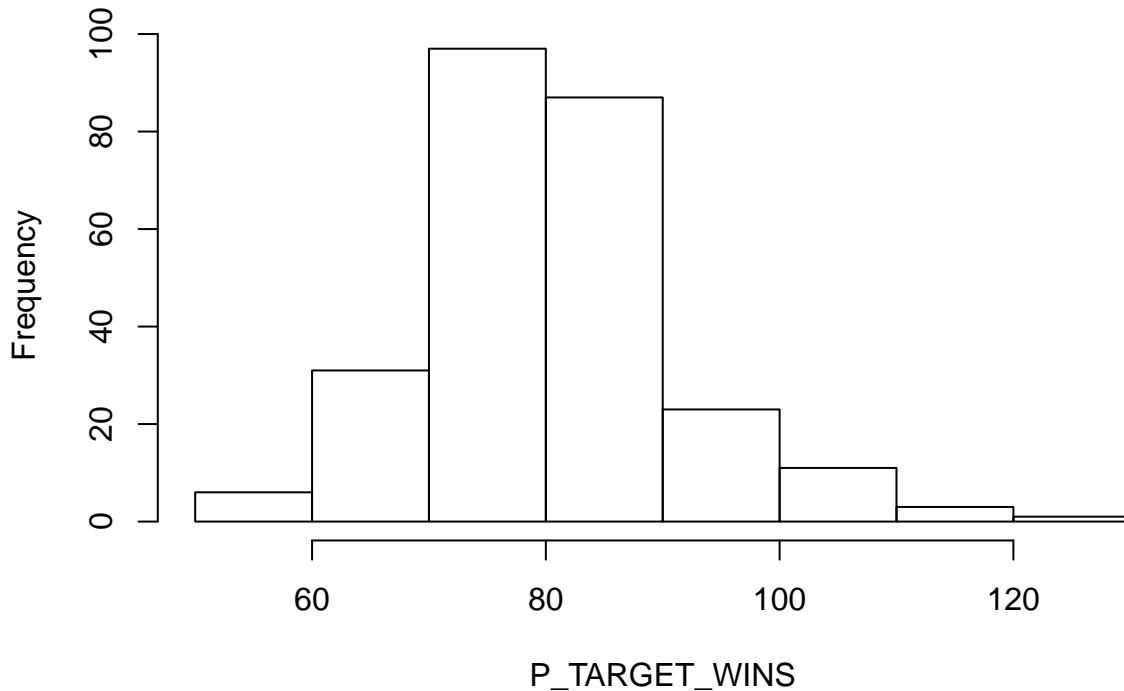
```

##                                     TEAM_FIELDING_E - 0.08 *
##                                     TEAM_FIELDING_DP + 0.08 *
##                                     TEAM_BATTING_HR - 0.01 *
##                                     TEAM_BATTING_2B - 0.09 *
##                                     TEAM_BASERUN_CS + 0.05 *
##                                     TEAM_BATTING_BB
##
##      Partial Least Squares      TARGET_WINS = 81 + 3.76 *
##                                         Comp.1 + 2.56 * Comp.2 + 2.22
##                                         * Comp.3 + 2.07 * Comp.4
##
##      Random forest predictors   TARGET_WINS = 33.91 - 0.12 *
##                                         TEAM_FIELDING_E + 0.04 *
##                                         TEAM_BATTING_H + 0.04 *
##                                         TEAM_BATTING_BB - 0.01 *
##                                         TEAM_PITCHING_SO
## -----
##
## Table: Table continues below
##
##
## -----
##      AIC      MAE
## -----
##    11153    10.58
##
##    11211    10.71
##
##    11211    10.71
##
##    11385    11.32
##
##    11180    18.53
##
##    10954    14.52
##
##    11255    18.73
## -----

```

6. Prediction

Histogram of P_TARGET_WINS



APPENDIX

A.1 Data dictionary

VARIABLE NAME	DEFINITION	THEORETICAL EFFECT
INDEX	Identification Variable (do not use)	None
TARGET_WINS	NA	NA
TEAM_BATTING_H	Base Hits by batters (1B,2B,3B,HR)	Positive Impact on Wins
TEAM_BATTING_2B	Doubles by batters (2B)	Positive Impact on Wins
TEAM_BATTING_3B	Triples by batters (3B)	Positive Impact on Wins
TEAM_BATTING_HR	Homeruns by batters (4B)	Positive Impact on Wins
TEAM_BATTING_BB	Walks by batters	Positive Impact on Wins
TEAM_BATTING_HBP	Batters hit by pitch (get a free base)	Positive Impact on Wins
TEAM_BATTING_SO	Strikeouts by batters	Negative Impact on Wins
TEAM_BASERUN_SB	Stolen bases	Positive Impact on Wins
TEAM_BASERUN_CS	Caught stealing	Negative Impact on Wins
TEAM_FIELDING_E	Errors	Negative Impact on Wins
TEAM_FIELDING_DP	Double Plays	Positive Impact on Wins
TEAM_PITCHING_BB	Walks allowed	Negative Impact on Wins
TEAM_PITCHING_H	Hits allowed	Negative Impact on Wins
TEAM_PITCHING_HR	Homeruns allowed	Negative Impact on Wins
TEAM_PITCHING_SO	Strikeouts by pitchers	Positive Impact on Wins