621 Assignment1

Raghu

Sep 15, 2018

Contents

Introduction

In this assignment, I will explore, analyze and model a data set containing approximately 2200 records. 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.

The objective is to build a multiple linear regression model on the training data to predict the number of wins for the team. I can only use the variables given to me (or variables that I derive from the variables provided).

1. Data Exploration

```
## Parsed with column specification:
## cols(
##   INDEX = col_integer(),
##   TARGET_WINS = col_integer(),
##   TEAM_BATTING_H = col_integer(),
##   TEAM_BATTING_2B = col_integer(),
##   TEAM_BATTING_3B = col_integer(),
##   TEAM_BATTING_HR = col_integer(),
##   TEAM_BATTING_BB = col_integer(),
##   TEAM_BATTING_SO = col_integer(),
##   TEAM_BASERUN_SB = col_integer(),
##   TEAM_BASERUN_CS = col_integer(),
##   TEAM_BATTING_HBP = col_integer(),
##   TEAM_PITCHING_H = col_integer(),
##   TEAM_PITCHING_HR = col_integer(),
##   TEAM_PITCHING_BB = col_integer(),
##   TEAM_PITCHING_SO = col_integer(),
##   TEAM_FIELDING_E = col_integer(),
##   TEAM_FIELDING_DP = col_integer()
## )
```

```
## Parsed with column specification:
## cols(
##   INDEX = col_integer(),
##   TEAM_BATTING_H = col_integer(),
##   TEAM_BATTING_2B = col_integer(),
##   TEAM_BATTING_3B = col_integer(),
##   TEAM_BATTING_HR = col_integer(),
##   TEAM_BATTING_BB = col_integer(),
##   TEAM_BATTING_SO = col_integer(),
##   TEAM_BASERUN_SB = col_integer(),
##   TEAM_BASERUN_CS = col_integer(),
##   TEAM_BATTING_HBP = col_integer(),
##   TEAM_PITCHING_H = col_integer(),
##   TEAM_PITCHING_HR = col_integer(),
##   TEAM_PITCHING_BB = col_integer(),
##   TEAM_PITCHING_SO = col_integer(),
##   TEAM_FIELDING_E = col_integer(),
##   TEAM_FIELDING_DP = col_integer()
## )
```

Print first 5 rows of the training data set.

```
## # A tibble: 5 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
## # ... 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>
```

Columns in the data set after removing TEAM_

```
##  [1] "INDEX"       "TARGET_WINS" "BATTING_H"   "BATTING_2B"  "BATTING_3B"
##  [6] "BATTING_HR"  "BATTING_BB"  "BATTING_SO"  "BASERUN_SB"  "BASERUN_CS"
## [11] "BATTING_HBP" "PITCHING_H"  "PITCHING_HR" "PITCHING_BB" "PITCHING_SO"
## [16] "FIELDING_E"  "FIELDING_DP"
```

```
## [1] 2276   17
```

Summary

Of the 17 columns, INDEX is simply an index value used for sorting while TARGET_WINS represents the response variable we are to use within our regression models. The remaining 15 elements are all potential predictor variables for our linear models. A summary table for the data set is provided below. All variables are numbers and none of them are categorical. TARGET_WINS is not existing in the test data set which need to be added and predicted.

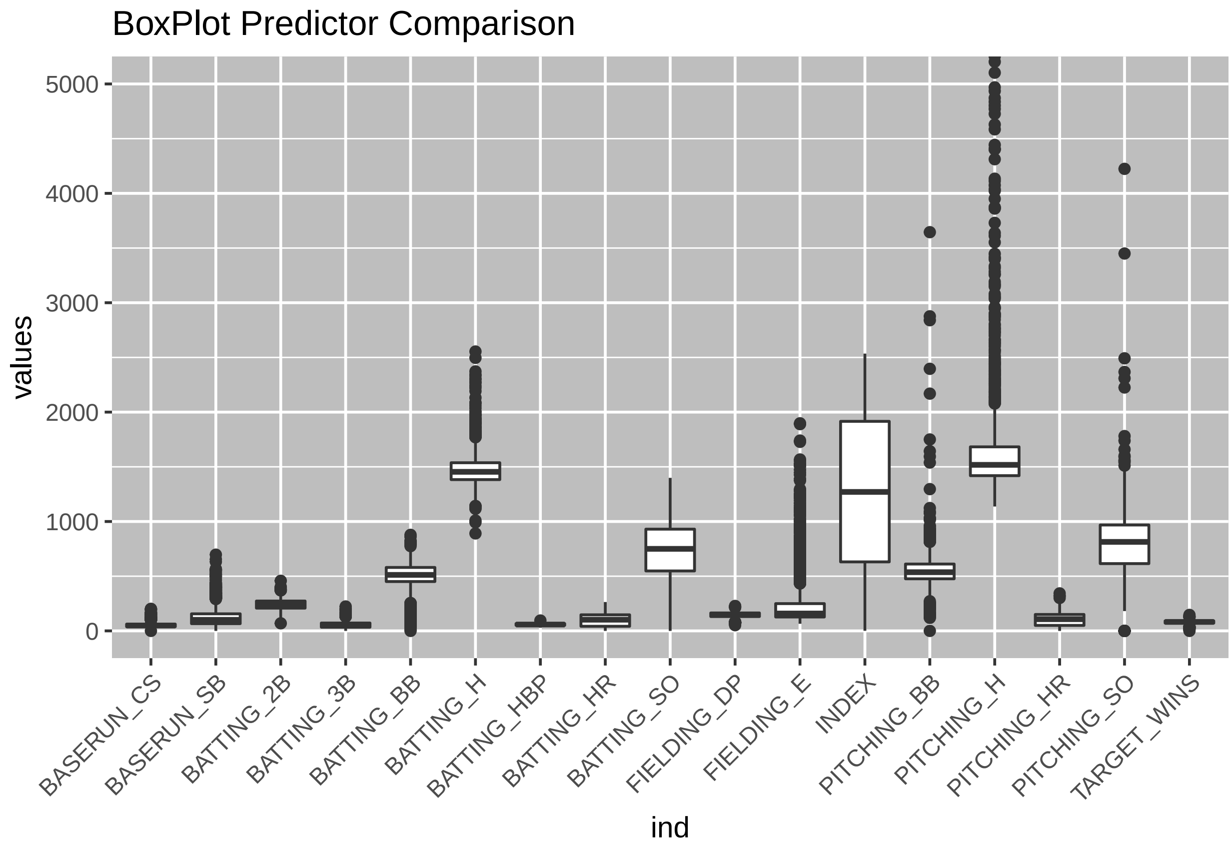
Descriptive statistics

	vars	n	mean	sd	median	trimmed	mad	min	max	range
INDEX	1	2276	1268.46353	736.34904	1270.5	1268.56970	952.5705	1	2535	2534
TARGET_WINS	2	2276	80.79086	15.75215	82.0	81.31229	14.8260	0	146	146
BATTING_H	3	2276	1469.26977	144.59120	1454.0	1459.04116	114.1602	891	2554	1663
BATTING_2B	4	2276	241.24692	46.80141	238.0	240.39627	47.4432	69	458	389
BATTING_3B	5	2276	55.25000	27.93856	47.0	52.17563	23.7216	0	223	223
BATTING_HR	6	2276	99.61204	60.54687	102.0	97.38529	78.5778	0	264	264
BATTING_BB	7	2276	501.55888	122.67086	512.0	512.18331	94.8864	0	878	878
BATTING_SO	8	2174	735.60534	248.52642	750.0	742.31322	284.6592	0	1399	1399
BASERUN_SB	9	2145	124.76177	87.79117	101.0	110.81188	60.7866	0	697	697
BASERUN_CS	10	1504	52.80386	22.95634	49.0	50.35963	17.7912	0	201	201
BATTING_HBP	11	191	59.35602	12.96712	58.0	58.86275	11.8608	29	95	66
PITCHING_H	12	2276	1779.21046	1406.84293	1518.0	1555.89517	174.9468	1137	30132	28995
PITCHING_HR	13	2276	105.69859	61.29875	107.0	103.15697	74.1300	0	343	343
PITCHING_BB	14	2276	553.00791	166.35736	536.5	542.62459	98.5929	0	3645	3645
PITCHING_SO	15	2174	817.73045	553.08503	813.5	796.93391	257.2311	0	19278	19278
FIELDING_E	16	2276	246.48067	227.77097	159.0	193.43798	62.2692	65	1898	1833
FIELDING_DP	17	1990	146.38794	26.22639	149.0	147.57789	23.7216	52	228	176

Summary of data

```
##      INDEX         TARGET_WINS       BATTING_H      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  
##                                                                  
##    BATTING_3B       BATTING_HR       BATTING_BB      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     
##    BASERUN_SB      BASERUN_CS     BATTING_HBP      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                   
##   PITCHING_HR     PITCHING_BB      PITCHING_SO        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                       
##   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
```

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

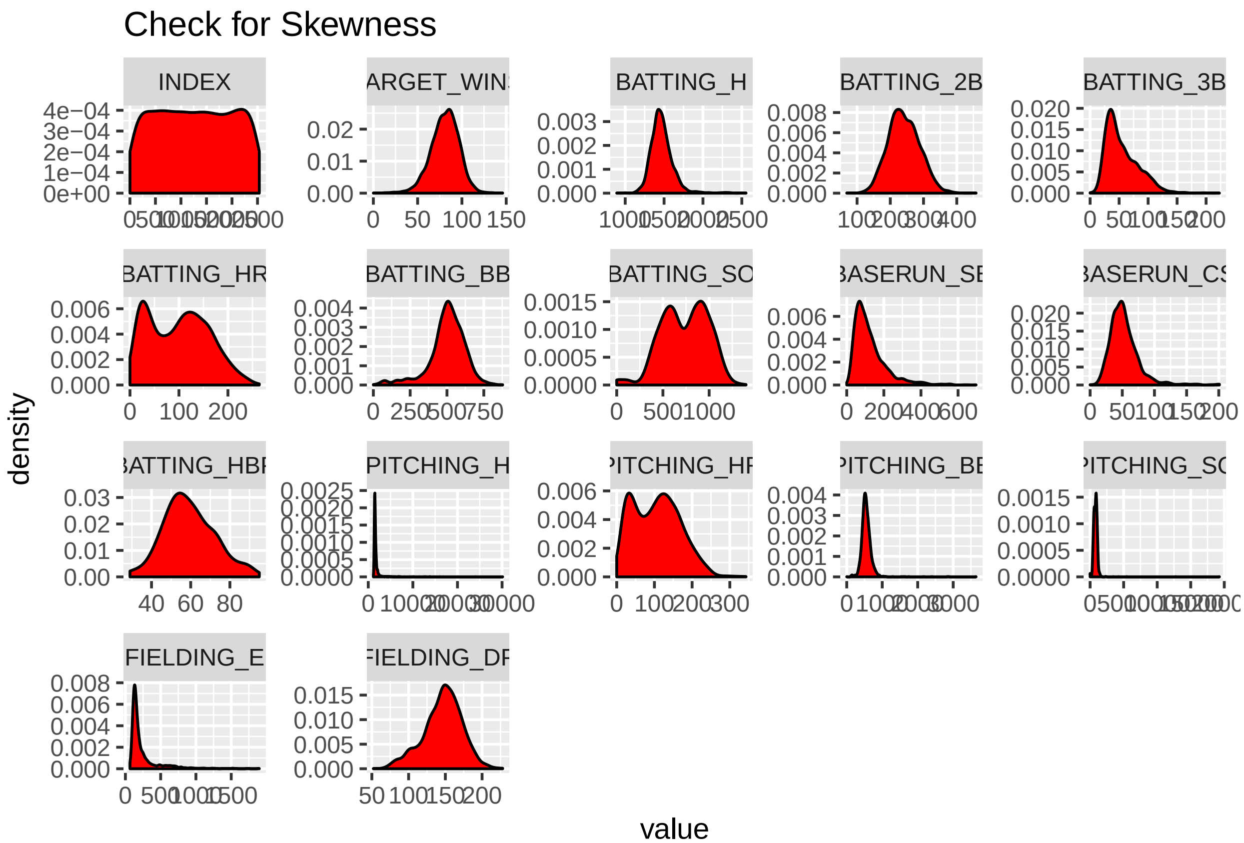


the boxplots of all the variables in the data set give an idea of how the data is spread.

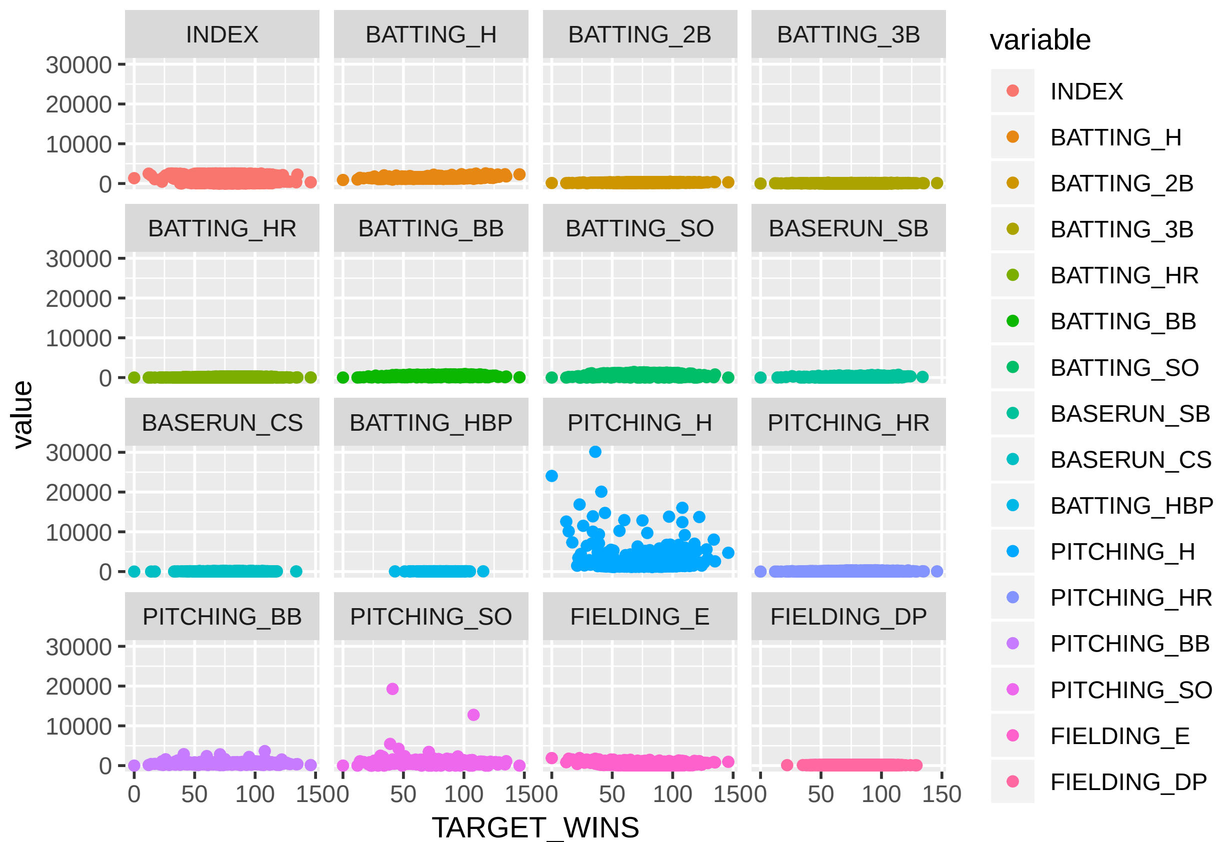
Checking for Skewness

```
## No id variables; using all as measure variables
```

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




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



The plot on the other hand provides visulatization of each of the independent variables to determine the skewness. scatterplot displays TARGET_WINS vs each of the predictor variables. we can see some outliers on PITCHING_SO and PITCHING_H has strong variations.

Checking for NAs

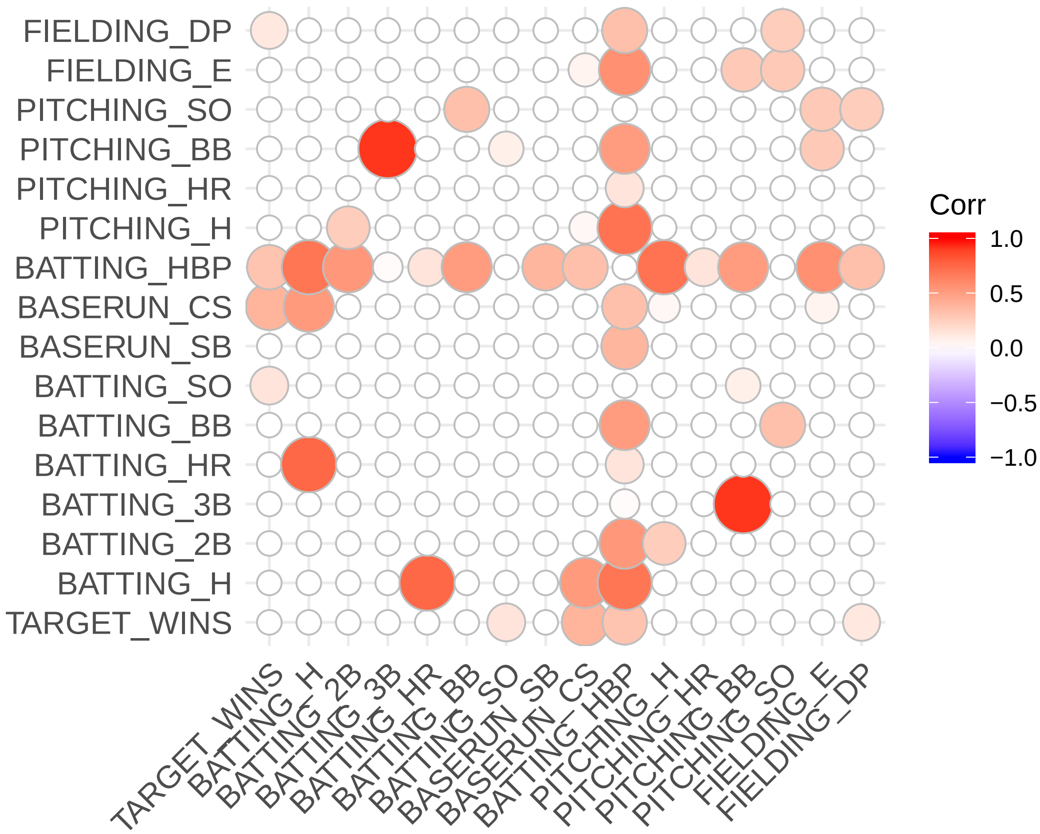
```
##       INDEX TARGET_WINS   BATTING_H  BATTING_2B  BATTING_3B  BATTING_HR
##           0           0           0           0           0           0
##  BATTING_BB  BATTING_SO  BASERUN_SB  BASERUN_CS BATTING_HBP  PITCHING_H
##           0         102         131         772        2085           0
## PITCHING_HR PITCHING_BB PITCHING_SO  FIELDING_E FIELDING_DP
##           0           0         102           0         286
```

Based on the plots, several outliers and skewness is observed. BATTING_HBP has the highest NAs. BASERUN_CS is the second largest. FIELDING_DP is the 3rd largest.

Correlation Plot

Using the cor function across the data frame we notice some strong correlations. BATTING_H obviously has some colinearity with BATTING_2B, BATTING_3B and BATTING_HR as these values are a subset of hits. BATTING_3B and PITCHING_BB have strong correlation, as do PITCHING_HR and BATTING_HR. Since we are focusing on wins, the following table shows the correlation when the NA's are omitted: There are positive and negative correlation observed.

There are missing data, severe outliers, and collinearity observed based on the data exploration.



2. Data Preparation

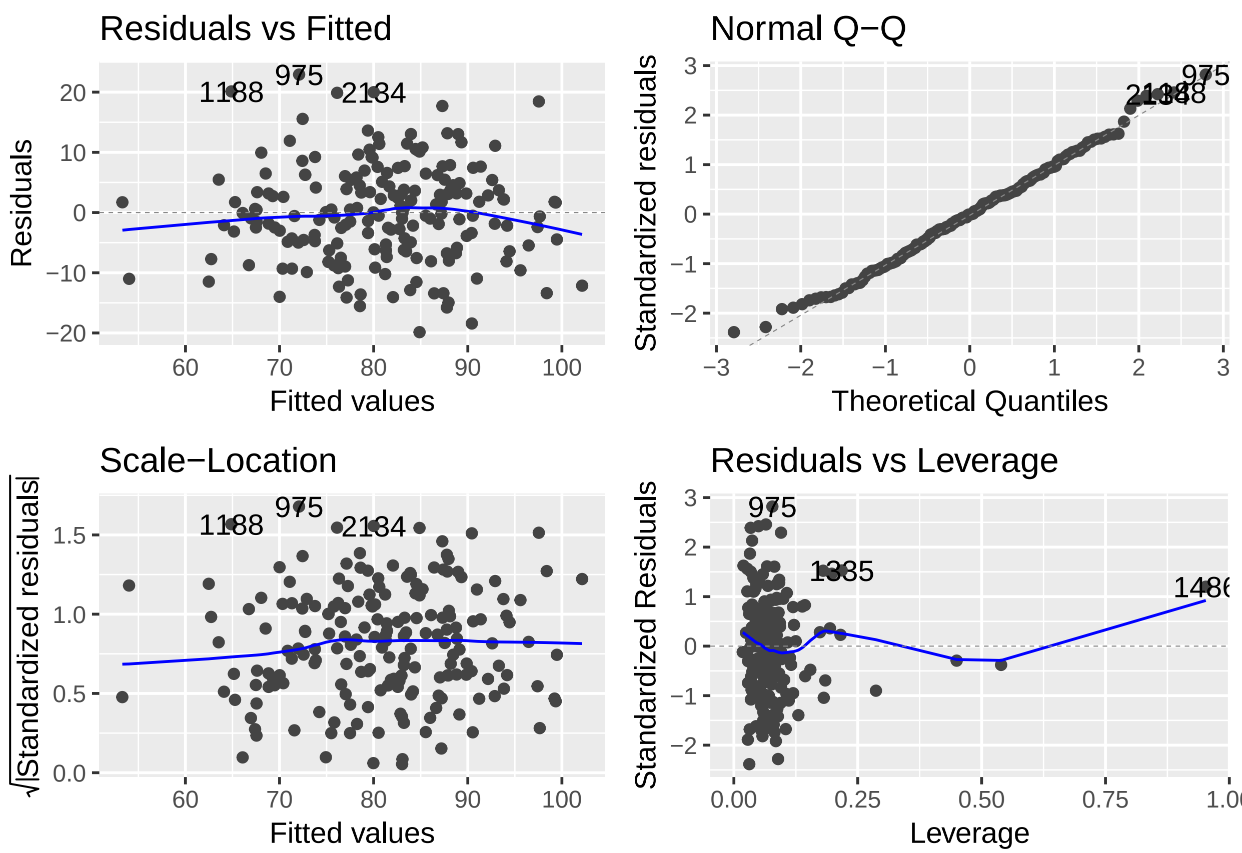
As we can see from summary statistics and plots, we have a number of missing values. The first step is to take care of missing values. We'll use Median imputation for CS, SB, and DP. Since HBP has the maximum missing values, we will remove that entirely. Interestingly, Pitching and Batting SO are missing in the same observations(see section Checking for NAs). I see that no problem with the residuals before transformation. qq is linear.

I will also create one new variable: BATTING_1B = BATTING_H - BATTING_HR - BATTING_3B -BATTING_2B Once its created, will remove BATTING_H from the model.

Linear Model before transformation.

By looking at the linear model before tranformation, FIELDING_E and FIELDING_DP are only significant out of all the variables. R-squared is less. need to compare it after the transformation and with other modals to see how significantly the modal can be improved.

```
##
## Call:
## lm(formula = TARGET_WINS ~ ., data = mb_tr_data)
##
## 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 **
## BATTING_H    1.91348    2.76139   0.693  0.48927
## BATTING_2B   0.02639    0.03029   0.871  0.38484
## BATTING_3B  -0.10118    0.07751  -1.305  0.19348
## BATTING_HR  -4.84371   10.50851  -0.461  0.64542
## BATTING_BB  -4.45969    3.63624  -1.226  0.22167
## BATTING_SO   0.34196    2.59876   0.132  0.89546
## BASERUN_SB   0.03304    0.02867   1.152  0.25071
## BASERUN_CS  -0.01104    0.07143  -0.155  0.87730
## BATTING_HBP  0.08247    0.04960   1.663  0.09815 .
## PITCHING_H  -1.89096    2.76095  -0.685  0.49432
## PITCHING_HR  4.93043   10.50664   0.469  0.63946
## PITCHING_BB  4.51089    3.63372   1.241  0.21612
## PITCHING_SO -0.37364    2.59705  -0.144  0.88577
## FIELDING_E  -0.17204    0.04140  -4.155 5.08e-05 ***
## 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
```

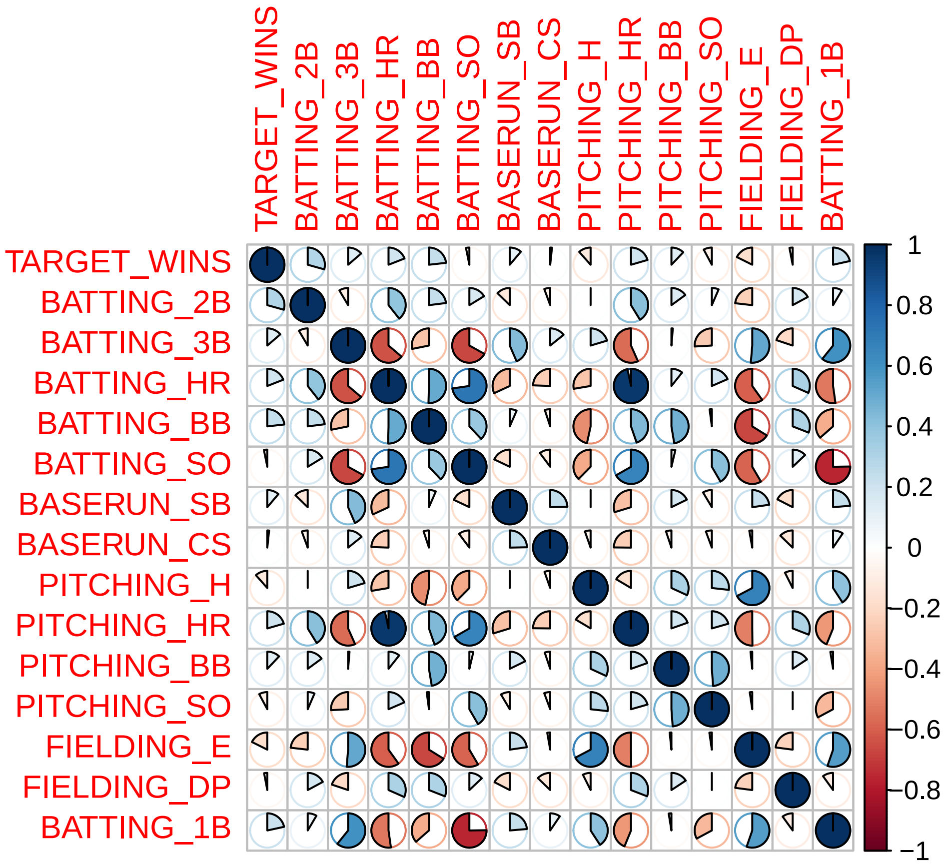

Residuals vs Fitted
Residuals
Fitted values
1188
975
2184
Normal Q−Q
Standardized residuals
Theoretical Quantiles
975
Scale−Location
√|Standardized residuals|
Fitted values
1188
975
2184
Residuals vs Leverage
Standardized Residuals
Leverage
975
1385

Detect multicollinearity

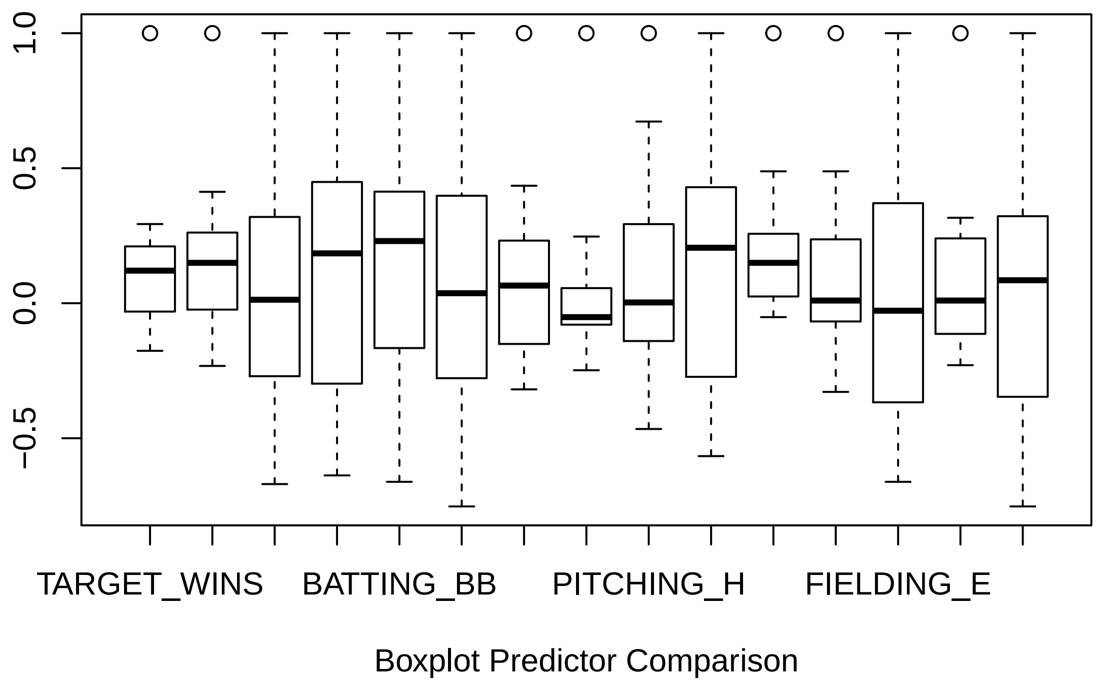
```
##   BATTING_H  BATTING_2B  BATTING_3B  BATTING_HR  BATTING_BB  BATTING_SO
##   117182.38        1.69        1.30   307480.44   196285.34   194175.22
##  BASERUN_SB  BASERUN_CS BATTING_HBP  PITCHING_H PITCHING_HR PITCHING_BB
##        1.95        1.91        1.10   116041.71   306962.39   196403.93
## PITCHING_SO  FIELDING_E FIELDING_DP
##   194631.56        1.26        1.10
```

we can see that 8 of the variables has high values due to collinearity.

Correlation Plot after transformation.



Box Plot after transformation.



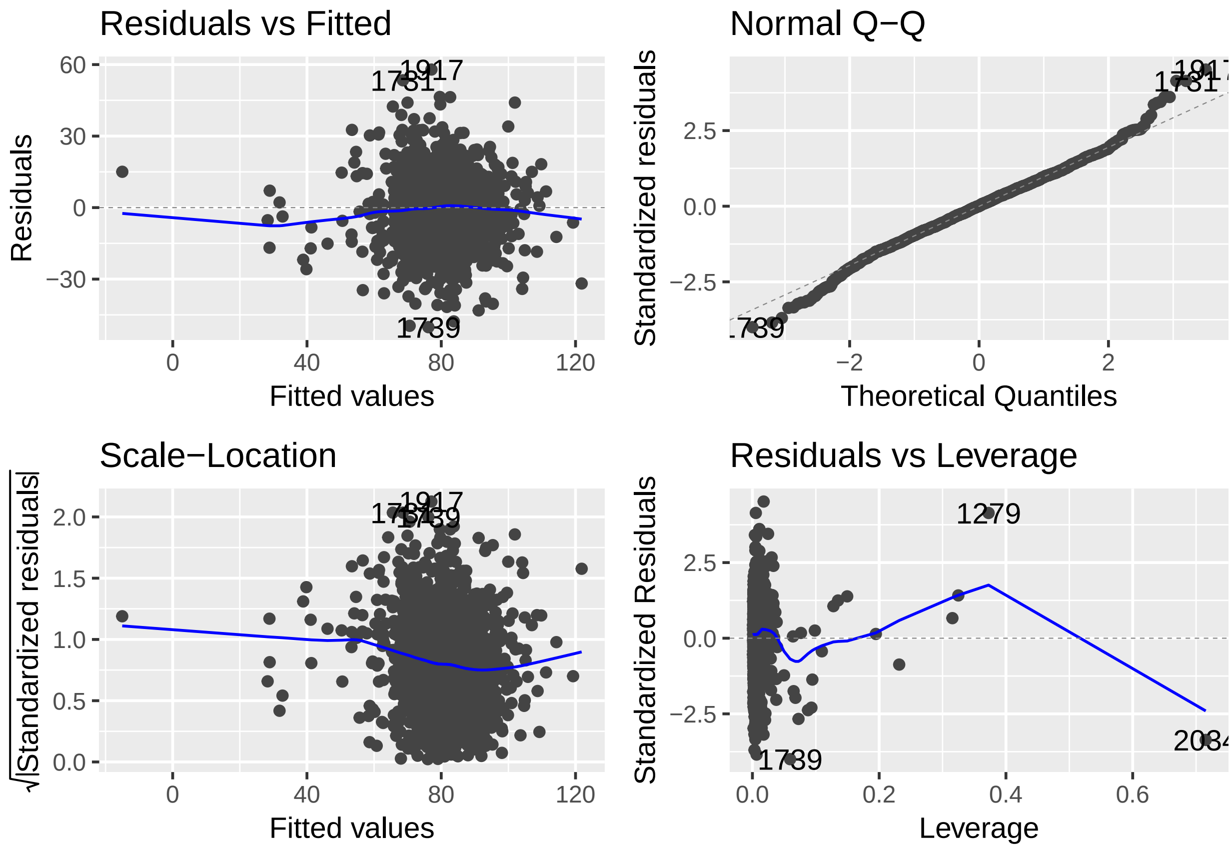
After transformation, box plot appears to be much better and normally distributed with outliers removed. correlation looks better after removing some of the correlations.

3. Build Models

Model1: Full Modal

In this Modal, I will not exclude any explanatory variables and evaluate the metrics.

```
##
## Call:
## lm(formula = TARGET_WINS ~ ., data = train2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -50.158  -8.529   0.078   8.415  57.927
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 20.2501836  5.4778605   3.697 0.000224 ***
## BATTING_2B   0.0242978  0.0074226   3.273 0.001079 **
## BATTING_3B   0.1199234  0.0161301   7.435 1.50e-13 ***
## BATTING_HR   0.1156126  0.0274740   4.208 2.68e-05 ***
## BATTING_BB   0.0092709  0.0058324   1.590 0.112085
## BATTING_SO  -0.0081864  0.0025811  -3.172 0.001537 **
## BASERUN_SB   0.0125347  0.0041367   3.030 0.002474 **
## BASERUN_CS   0.0088469  0.0158275   0.559 0.576246
## PITCHING_H  -0.0012881  0.0003630  -3.549 0.000395 ***
## PITCHING_HR  0.0065942  0.0241671   0.273 0.784987
## PITCHING_BB  0.0034332  0.0041187   0.834 0.404619
## PITCHING_SO  0.0027197  0.0009149   2.973 0.002986 **
## FIELDING_E  -0.0150214  0.0024185  -6.211 6.30e-10 ***
## FIELDING_DP -0.1136115  0.0135656  -8.375  < 2e-16 ***
## BATTING_1B   0.0499267  0.0037337  13.372  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.94 on 2159 degrees of freedom
## Multiple R-squared:  0.3144, Adjusted R-squared:   0.31
## F-statistic: 70.72 on 14 and 2159 DF,  p-value: < 2.2e-16
```



Summary:

Full modal includes all explanatory variables. our goal is to assess whether full modal is the best modal. If not, we want to identify a smaller modal that is preferable. Residuals plot is more dense after transformation of data. R-squared has reduced a lot after transformation. no change in the p-value. 7 of the variables are highly significant but the t-value is high. so, removing some of the variables is an option to check if it improves the modal.

Detect multicollinearity

```
##  BATTING_2B  BATTING_3B  BATTING_HR  BATTING_BB  BATTING_SO  BASERUN_SB
##        1.49        2.66       34.53        6.72        5.34        1.66
##  BASERUN_CS  PITCHING_H PITCHING_HR PITCHING_BB PITCHING_SO  FIELDING_E
##        1.20        3.53       27.16        6.21        3.32        4.11
## FIELDING_DP  BATTING_1B
##        1.21        3.10
```

we can see that multicollinearity has significantly reduced after transformation.

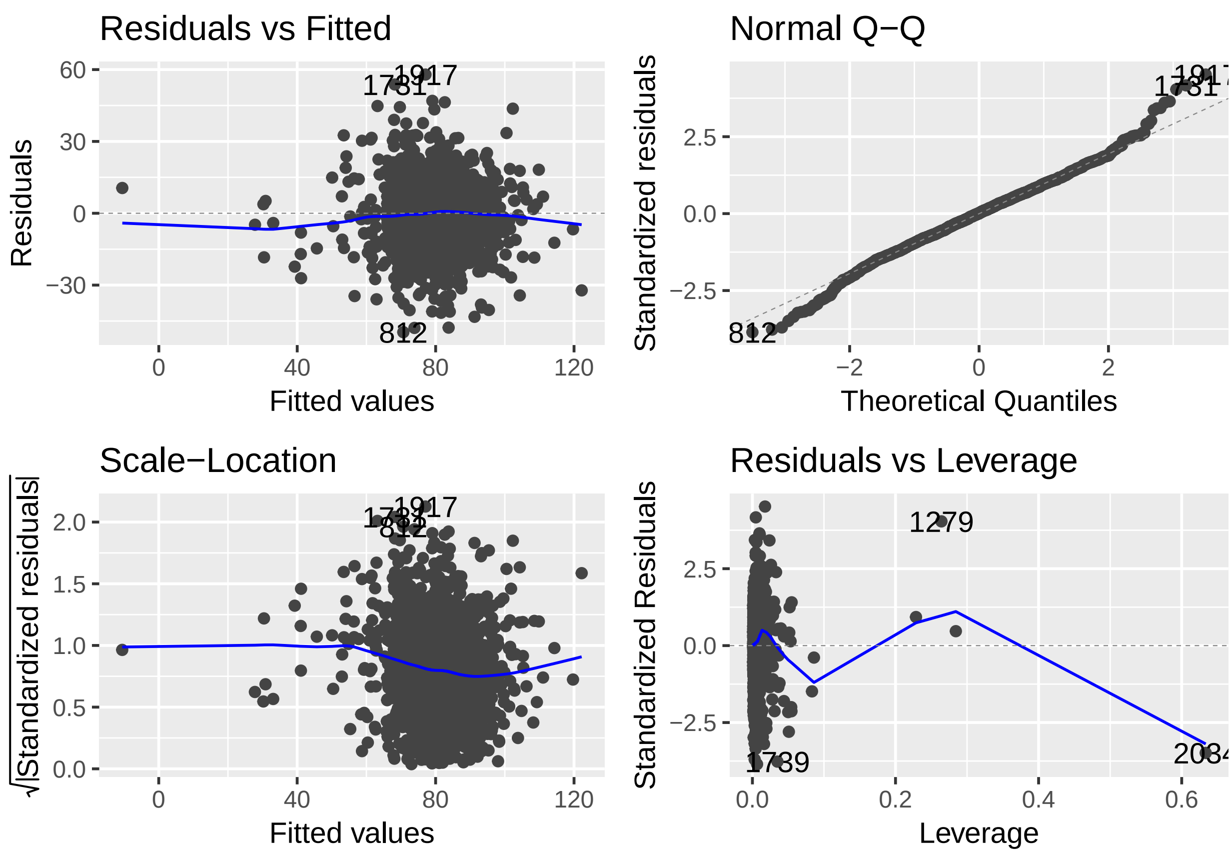
Model2: Stepwise Regression

In stepwise, i have chosen direction as both that includes forward and backward selection.

The backward-elimination strategy starts with the model that includes all potential predictor variables. Variables are eliminated one-at-a-time from the model until only variables with statistically significant p-values remain.

The forward-selection strategy is the reverse of the backward-elimination technique. Instead of eliminating variables one-at-a-time, we add variables one-at-a-time until we cannot find any variables the present strong evidence of thier importance in the model.

```
##
## Call:
## lm(formula = TARGET_WINS ~ BATTING_2B + BATTING_3B + BATTING_HR +
##     BATTING_BB + BATTING_SO + BASERUN_SB + PITCHING_H + PITCHING_SO +
##     FIELDING_E + FIELDING_DP + BATTING_1B, data = train2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -49.659  -8.483   0.154   8.429  57.936
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 19.9215015  5.3090383   3.752 0.000180 ***
## BATTING_2B   0.0244468  0.0074130   3.298 0.000990 ***
## BATTING_3B   0.1218529  0.0158688   7.679 2.42e-14 ***
## BATTING_HR   0.1212428  0.0089437  13.556  < 2e-16 ***
## BATTING_BB   0.0130070  0.0034238   3.799 0.000149 ***
## BATTING_SO  -0.0086054  0.0024958  -3.448 0.000576 ***
## BASERUN_SB   0.0134911  0.0039993   3.373 0.000756 ***
## PITCHING_H  -0.0010911  0.0003174  -3.437 0.000599 ***
## PITCHING_SO  0.0032599  0.0006660   4.895 1.06e-06 ***
## FIELDING_E  -0.0150792  0.0023486  -6.421 1.66e-10 ***
## FIELDING_DP -0.1137020  0.0135532  -8.389  < 2e-16 ***
## BATTING_1B   0.0501514  0.0037064  13.531  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.93 on 2162 degrees of freedom
## Multiple R-squared:  0.3139, Adjusted R-squared:  0.3104
## F-statistic: 89.92 on 11 and 2162 DF,  p-value: < 2.2e-16
```



Summary:

I dont see any difference between stepwise and full model interms of residual plots but F-stat has increased significantly from 70.72 to 89.92. R-squared and p-value is almost the same. there are only 11 coefficients displayed in stepwise model but all are statistically significant.

Model3: Principal Component Regression (PCR)

I will next evaluate the PCR model.

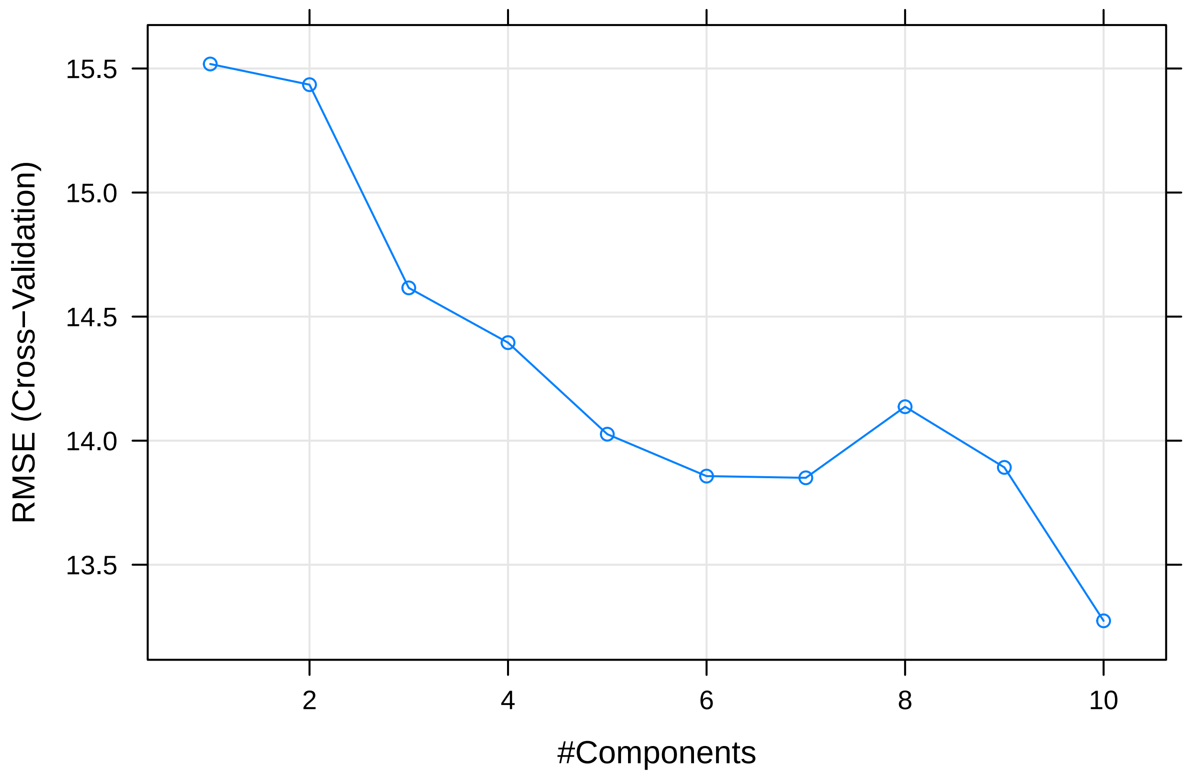
Model summary

```
## Data:    X dimension: 2174 14
##  Y dimension: 2174 1
## Fit method: svdpc
## Number of components considered: 10
## TRAINING: % variance explained
##           1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
## X         36.3648  49.9689    61.34    71.48    78.33    84.24    88.82
## .outcome   0.7241   0.7482    11.68    15.96    20.21    23.38    23.74
##           8 comps  9 comps  10 comps
## X           92.62    95.06     97.03
## .outcome    23.90    26.24     30.96
```

Model Results

ncomp	RMSE	Rsquared	MAE	RMSESD	RsquaredSD	MAESD
1	15.51810	0.0139617	12.12328	0.5950761	0.0113722	0.3561924
2	15.43475	0.0193717	12.02163	0.6886180	0.0268824	0.4909598
3	14.61578	0.1244875	11.37933	0.6487658	0.0455459	0.4555423
4	14.39503	0.1567343	11.16959	0.6984108	0.0603163	0.4113462
5	14.02608	0.1971479	10.98336	0.6221318	0.0701730	0.4129652
6	13.85708	0.2283595	10.73883	0.9383585	0.0825138	0.4091514
7	13.85022	0.2290321	10.73300	0.9384417	0.0843107	0.4087471
8	14.13678	0.2278775	10.71863	1.8028506	0.0894212	0.4455576
9	13.89227	0.2484456	10.54218	1.6960355	0.0793514	0.3952689
10	13.27385	0.2910398	10.26281	0.9392047	0.0855977	0.3054222

Model Plot



	ncomp
10	10

Summary:

Caret uses cross-validation to automatically identify the optimal number of principal components (ncomp) to be incorporated in the model.

Here, we'll test 10 different values of the tuning parameter ncomp. This is specified using the option tuneLength. The optimal number of principal components is selected so that the cross-validation error (RMSE) is minimized. RMSE is ranging from 13.0 to 15.6

Model4: Partial Least Squares Regression (PLS)

Partial least squares regression extends multiple linear regression without imposing the restrictions employed by discriminant analysis, principal components regression, and canonical correlation.

Partial least squares regression can be used as an exploratory analysis tool to select suitable predictor variables and to identify outliers before classical linear regression.

Principal components regression and partial least squares regression differ in the methods used in extracting factor scores.

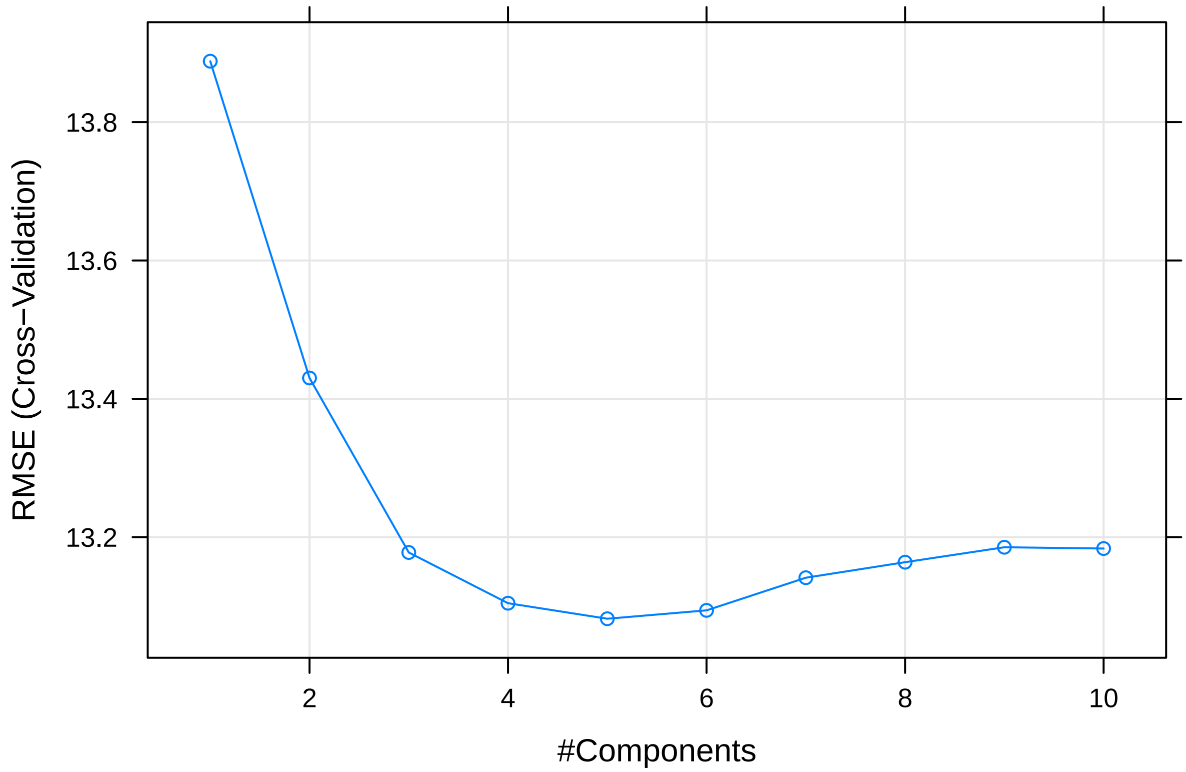
Model summary

```
## Data:    X dimension: 2174 14
##  Y dimension: 2174 1
## Fit method: oscorespls
## Number of components considered: 5
## TRAINING: % variance explained
##           1 comps  2 comps  3 comps  4 comps  5 comps
## X           18.17    45.93    52.51    57.63    66.62
## .outcome    21.31    26.50    29.87    31.10    31.24
```

Model Results

ncomp	RMSE	Rsquared	MAE	RMSESD	RsquaredSD	MAESD
1	13.88806	0.2077133	10.88043	0.8307983	0.0601437	0.5460388
2	13.43010	0.2614885	10.47829	1.0501008	0.0785657	0.6851920
3	13.17783	0.2904859	10.32191	0.9855724	0.0789673	0.5802327
4	13.10449	0.2978736	10.24221	0.9039254	0.0721503	0.5863951
5	13.08205	0.3022110	10.21435	0.9653799	0.0785120	0.6201722
6	13.09416	0.2994721	10.21922	0.9258214	0.0771781	0.6109036
7	13.14133	0.2942983	10.22765	0.9251068	0.0790378	0.6070205
8	13.16378	0.2920185	10.22800	0.9368037	0.0806933	0.6071607
9	13.18540	0.2901453	10.22586	0.9589980	0.0832928	0.6082483
10	13.18355	0.2905599	10.22028	0.9571808	0.0830977	0.6083738

Model Plot



	ncomp
5	5

Summary:

The optimal number of principal components included in the PLS model is 5 or sometimes 4. This captures 90% of the variation in the predictors and 75% of the variation in the outcome variable.

In our example, the cross-validation error RMSE obtained with the PLS model is lower than the RMSE obtained using the PCR method. RMSE is ranging from 13.05 to 13.89. So, the PLS model is the best model, for explaining our data, compared to the PCR model.

4. Select Model

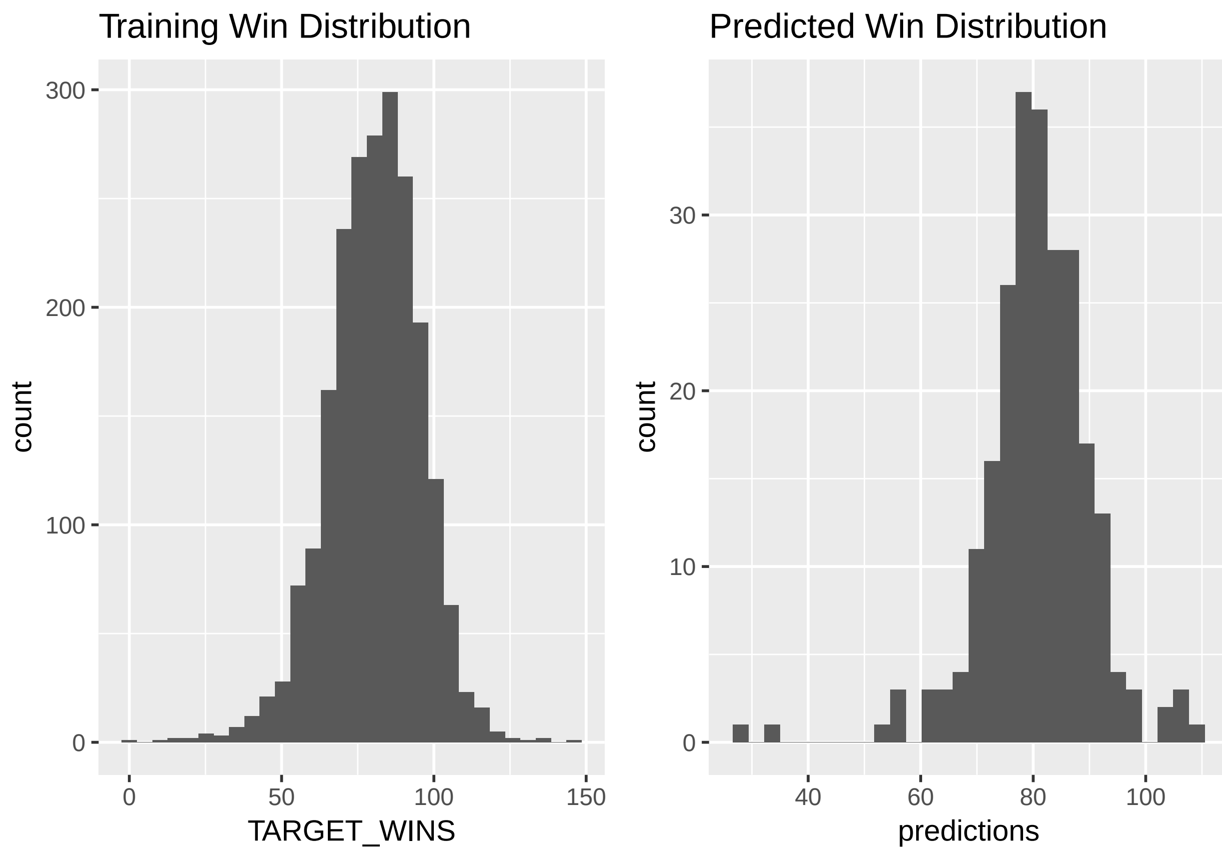
Conclusion:

By analyzing the R-squared, F-statistic and RMSE, PLS model seems to be good as the number of principal components is 5 and RMSE is lower than the other model. I will predict the test data using PLS model.

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   28.34   75.86   80.88   80.78   86.30  109.50
```

```
## Warning: Unknown or uninitialised column: 'medv'.
```

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



The plots shows training distribution and predicted distribution. comparing the 2 plots, distribution of the predicted values seems to be more aligned with the test distribution.