DATA 621: Homework 1 (Baseball Regression)

Shoshana Farber, Josh Forster, Glen Davis, Andrew Bowen, Charles Ugiagbe

October 1, 2023

Setup:

First, let's read in the provided dataset.

Data Exploration:

```
## The dataset consists of 2276 observations of 17 variables.
```

The variables and their definitions can be seen below:

Variable	Definition
INDEX	Identification variable
TARGET_WINS	Number of wins
TEAM_BATTING_H	Base hits by batters (1B, 2B, 3B, HR)
TEAM_BATTING_2B	Doubles by batters (2B)
TEAM_BATTING_3B	Triples by batters (3B)
TEAM_BATTING_HR	Homeruns by batters (4B)
TEAM_BATTING_BB	Walks by batters
TEAM_BATTING_HBP	Batters hit by pitch (get a free base)
TEAM_BATTING_SO	Strikeouts by batters
TEAM_BASERUN_SB	Stolen bases
TEAM_BASRUN_CS	Caught stealing
TEAM_FIELDING_E	Errors
TEAM_FIELDING_DP	Double plays
TEAM_PITCHING_BB	Walks allowed
TEAM_PITCHING_H	Hits allowed
TEAM_PITCHING_HR	Homeruns allowed
TEAM_PITCHING_SO	Strikeouts by pitchers

INDEX is an identifying feature and should not be included in the linear regression model.

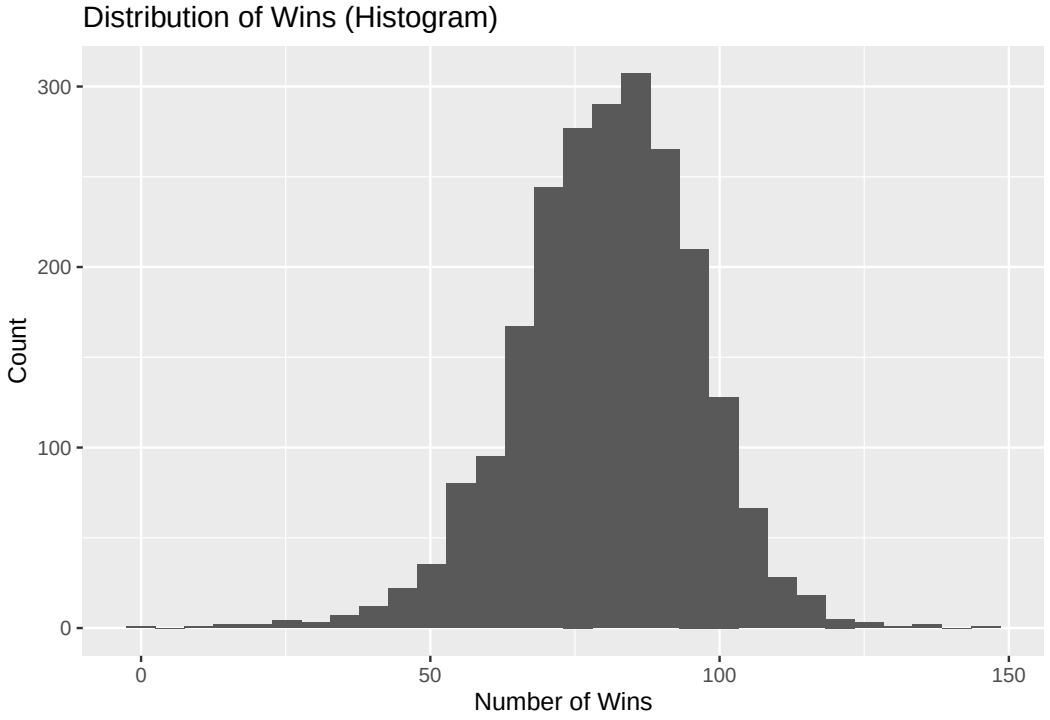
Next, let's print out some summary statistics. We're primarily interested in the TARGET_WINS variable, so we'll look at that first.

```
## The mean number of wins in a season is 80.79.
```

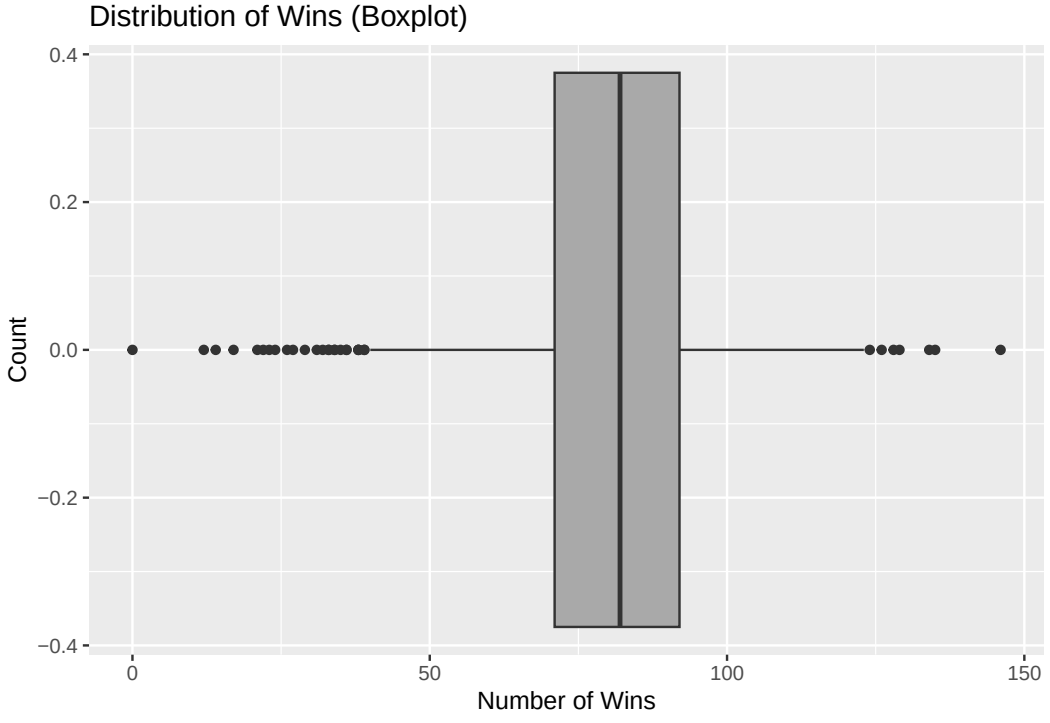
```
## The median number of wins in a season is 82.
```

```
## The standard deviation for number of wins in a season is 15.75.
```

Let's also make a histogram of the `TARGET_WINS` variable. This should give us a sense of the distribution of wins for teams/seasons in our population.



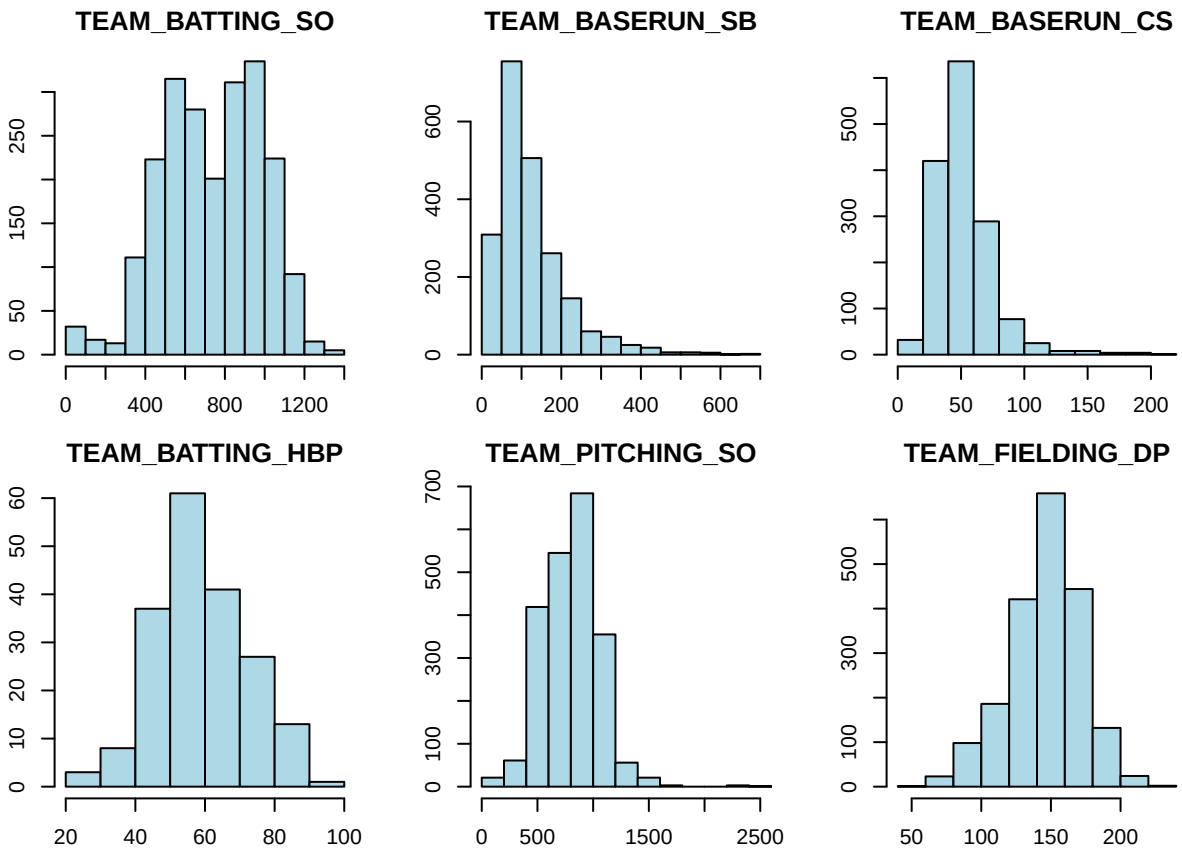
Overall, the number of wins in a season for a given baseball team looks fairly normally distributed. Looking at a boxplot helps to highlight the outliers.



Let's take a look at the summary statistics for all the variables:

```
##   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_BATTING_HBP TEAM_PITCHING_H TEAM_PITCHING_HR
##  Min.   :  0.0   Min.   :29.00    Min.   : 1137   Min.   :  0.0
##  1st Qu.: 38.0   1st Qu.:50.50    1st Qu.: 1419   1st Qu.: 50.0
##  Median : 49.0   Median :58.00    Median : 1518   Median :107.0
##  Mean   : 52.8   Mean   :59.36    Mean   : 1779   Mean   :105.7
##  3rd Qu.: 62.0   3rd Qu.:67.00    3rd Qu.: 1682   3rd Qu.:150.0
##  Max.   :201.0   Max.   :95.00    Max.   :30132   Max.   :343.0
##  NA's   :772     NA's   :2085
##  TEAM_PITCHING_BB TEAM_PITCHING_SO  TEAM_FIELDING_E  TEAM_FIELDING_DP
##  Min.   :   0.0   Min.   :    0.0   Min.   :  65.0   Min.   : 52.0
##  1st Qu.: 476.0   1st Qu.:  615.0   1st Qu.: 127.0   1st Qu.:131.0
##  Median : 536.5   Median :  813.5   Median : 159.0   Median :149.0
##  Mean   : 553.0   Mean   :  817.7   Mean   : 246.5   Mean   :146.4
##  3rd Qu.: 611.0   3rd Qu.:  968.0   3rd Qu.: 249.2   3rd Qu.:164.0
##  Max.   :3645.0   Max.   :19278.0   Max.   :1898.0   Max.   :228.0
##                   NA's   :102                        NA's   :286
```

We can see quite a few NA values for `TEAM_BATTING_SO`, `TEAM_BASERUN_SB`, `TEAM_BASERUN_CS`, `TEAM_BATTING_HBP`, `TEAM_PITCHING_SO`, and `TEAM_FIELDING_DP`. Let's take a look at the distributions of these variables to see how to impute these missing values.



TEAM_BASERUN_SB and TEAM_BASERUN_CS seem to be skewed to the right so we should probably impute the missing values using the median value for these variables. TEAM_BATTING_HBP, TEAM_PITCHING_SO, and TEAM_FIELDING_DP seem basically normally distributed so we can use the mean here, although TEAM_BATTING_HBP has 2,085 NA values out of 2,276 observations so it may make sense to leave this variable out of our model entirely. TEAM_BATTING_SO is bimodally distributed, so we have decided to use KNN imputation, which does not rely on the shape of the distribution, for this variable.

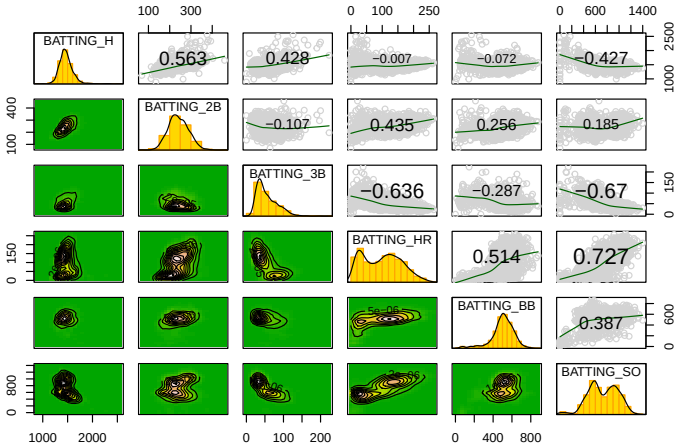
Let's look at raw correlations between our other included variables and a team's win total for a season:

```
##                         [,1]
## TARGET_WINS       1.00000000
## TEAM_BATTING_H    0.38876752
## TEAM_BATTING_2B   0.28910365
## TEAM_BATTING_3B   0.14260841
## TEAM_BATTING_HR   0.17615320
## TEAM_BATTING_BB   0.23255986
## TEAM_BATTING_SO  -0.03606403
## TEAM_BASERUN_SB   0.12361087
## TEAM_BASERUN_CS   0.01595982
## TEAM_PITCHING_H  -0.10993705
## TEAM_PITCHING_HR  0.18901373
## TEAM_PITCHING_BB  0.12417454
## TEAM_PITCHING_SO -0.07578725
## TEAM_FIELDING_E  -0.17648476
## TEAM_FIELDING_DP -0.02884126
```

None of the independent variables seem to have such high correlation with TARGET_WINS. TEAM_BATTING_H is

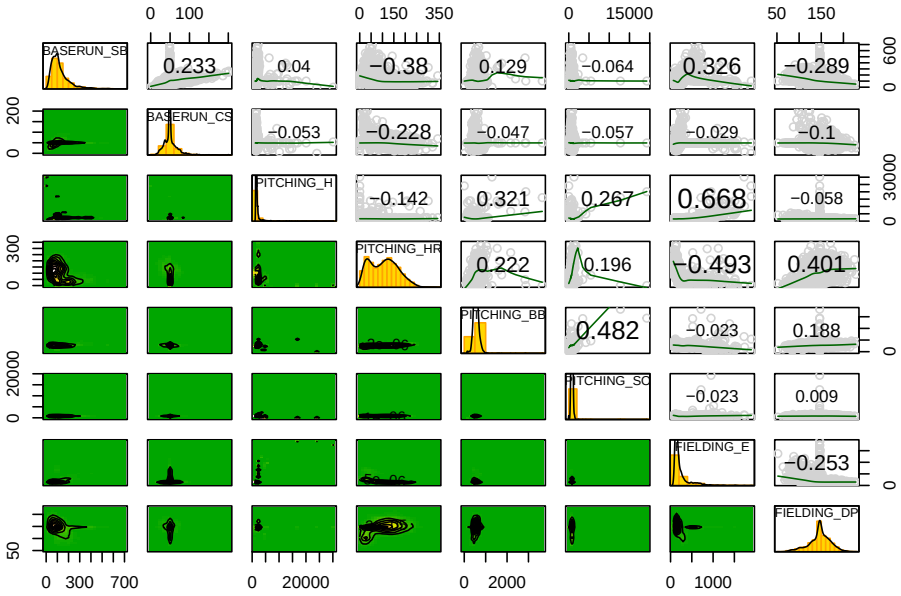
most highly correlated, with a correlation of 0.39. TEAM_BATTING_H, TEAM_BATTING_2B, TEAM_BATTING_3B, TEAM_BATTING_HR, TEAM_BATTING_BB, TEAM_BASERUN_SB, TEAM_BASERUN_CS, TEAM_PITCHING_HR, and TEAM_PITCHING_BB are all positively correlated with TARGET_WINS while TEAM_BATTING_SO, TEAM_PITCHING_H, TEAM_PITCHING_SO, TEAM_FIELDING_E, and TEAM_FIELDING_DP are negatively correlated.

Let's review relationships between batting independent variables.



Most of the batting variables appear to be somewhat approximately normal although there are some cases of right skew. Overall, there aren't any very strong correlations between these statistics at least from a preliminary visual inspection. From the distributions of these variables, we can see some that require transforming to normalize them before we use them in our linear model.

Let's review relationships between other independent variables.



There isn't very strong correlation between the other independent variables similar to the batting statistics although there are more examples of skewed data with these inputs. Once again, we can see that we will need to transform some of these variables.

Modeling

First, let's create a basic model with the untransformed variables:

```
##
## Call:
## lm(formula = TARGET_WINS ~ ., data = train_imputed)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -50.258  -8.612   0.151   8.427  59.016
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       2.656e+01  5.234e+00   5.074 4.20e-07 ***
## TEAM_BATTING_H    4.709e-02  3.699e-03  12.729  < 2e-16 ***
## TEAM_BATTING_2B  -1.788e-02  9.206e-03  -1.942 0.052261 .
## TEAM_BATTING_3B   6.136e-02  1.678e-02   3.657 0.000261 ***
## TEAM_BATTING_HR   5.750e-02  2.748e-02   2.092 0.036530 *
## TEAM_BATTING_BB   1.086e-02  5.815e-03   1.867 0.062006 .
## TEAM_BATTING_SO  -1.142e-02  2.578e-03  -4.428 9.95e-06 ***
## TEAM_BASERUN_SB   2.579e-02  4.317e-03   5.975 2.67e-09 ***
## TEAM_BASERUN_CS  -7.135e-03  1.577e-02  -0.453 0.650942
## TEAM_PITCHING_H  -8.982e-04  3.673e-04  -2.445 0.014545 *
## TEAM_PITCHING_HR  1.615e-02  2.431e-02   0.664 0.506619
## TEAM_PITCHING_BB -3.272e-05  4.123e-03  -0.008 0.993667
## TEAM_PITCHING_SO  3.205e-03  9.132e-04   3.509 0.000458 ***
## TEAM_FIELDING_E  -1.961e-02  2.448e-03  -8.009 1.83e-15 ***
## TEAM_FIELDING_DP -1.201e-01  1.293e-02  -9.286  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.05 on 2261 degrees of freedom
## Multiple R-squared:  0.3181, Adjusted R-squared:  0.3139
## F-statistic: 75.34 on 14 and 2261 DF,  p-value: < 2.2e-16
```

We can see that the R^2 value of a model that includes all the variables is not particularly high.

We can remove some variables that are not significant using backward step-wise elimination.

```
## Start:  AIC=11707.33
## TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_2B + TEAM_BATTING_3B +
##     TEAM_BATTING_HR + TEAM_BATTING_BB + TEAM_BATTING_SO + TEAM_BASERUN_SB +
##     TEAM_BASERUN_CS + TEAM_PITCHING_H + TEAM_PITCHING_HR + TEAM_PITCHING_BB +
##     TEAM_PITCHING_SO + TEAM_FIELDING_E + TEAM_FIELDING_DP
##
##                     Df Sum of Sq    RSS   AIC
## - TEAM_PITCHING_BB  1       0.0 384928 11705
## - TEAM_BASERUN_CS   1      34.9 384963 11706
```

```
## - TEAM_PITCHING_HR  1      75.1 385003 11706
## <none>                          384928 11707
## - TEAM_BATTING_BB   1     593.5 385522 11709
## - TEAM_BATTING_2B   1     642.1 385570 11709
## - TEAM_BATTING_HR   1     745.2 385673 11710
## - TEAM_PITCHING_H   1    1018.1 385946 11711
## - TEAM_PITCHING_SO  1    2096.5 387025 11718
## - TEAM_BATTING_3B   1    2277.2 387205 11719
## - TEAM_BATTING_SO   1    3338.6 388267 11725
## - TEAM_BASERUN_SB   1    6077.5 391006 11741
## - TEAM_FIELDING_E   1   10921.2 395849 11769
## - TEAM_FIELDING_DP  1   14681.7 399610 11790
## - TEAM_BATTING_H    1   27586.8 412515 11863
##
## Step:  AIC=11705.33
## 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
##
##                     Df Sum of Sq    RSS   AIC
## - TEAM_BASERUN_CS   1      34.9 384963 11704
## - TEAM_PITCHING_HR  1      99.3 385027 11704
## <none>                          384928 11705
## - TEAM_BATTING_2B   1     642.3 385570 11707
## - TEAM_BATTING_HR   1     968.4 385896 11709
## - TEAM_PITCHING_H   1    1276.3 386204 11711
## - TEAM_BATTING_BB   1    1746.1 386674 11714
## - TEAM_BATTING_3B   1    2277.2 387205 11717
## - TEAM_BATTING_SO   1    3544.8 388473 11724
## - TEAM_PITCHING_SO  1    3862.5 388791 11726
## - TEAM_BASERUN_SB   1    6195.0 391123 11740
## - TEAM_FIELDING_E   1   10946.8 395875 11767
## - TEAM_FIELDING_DP  1   14708.4 399636 11789
## - TEAM_BATTING_H    1   27645.2 412573 11861
##
## Step:  AIC=11703.54
## 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_HR + TEAM_PITCHING_SO + TEAM_FIELDING_E +
##     TEAM_FIELDING_DP
##
##                     Df Sum of Sq    RSS   AIC
## - TEAM_PITCHING_HR  1      97.1 385060 11702
## <none>                          384963 11704
## - TEAM_BATTING_2B   1     651.5 385614 11705
## - TEAM_BATTING_HR   1    1015.3 385978 11708
## - TEAM_PITCHING_H   1    1303.5 386266 11709
## - TEAM_BATTING_BB   1    1803.9 386767 11712
## - TEAM_BATTING_3B   1    2297.2 387260 11715
## - TEAM_BATTING_SO   1    3587.9 388551 11723
## - TEAM_PITCHING_SO  1    3896.5 388859 11724
## - TEAM_BASERUN_SB   1    6259.5 391222 11738
## - TEAM_FIELDING_E   1   11210.3 396173 11767
```

```
## - TEAM_FIELDING_DP  1   14762.5 399725 11787
## - TEAM_BATTING_H    1   27610.4 412573 11859
##
## Step:  AIC=11702.11
## 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
##
##                     Df Sum of Sq    RSS   AIC
## <none>                           385060 11702
## - TEAM_BATTING_2B   1     667.2 385727 11704
## - TEAM_PITCHING_H   1    1212.6 386273 11707
## - TEAM_BATTING_BB   1    1797.8 386858 11711
## - TEAM_BATTING_3B   1    2478.1 387538 11715
## - TEAM_BATTING_SO   1    3558.0 388618 11721
## - TEAM_PITCHING_SO  1    3962.8 389023 11723
## - TEAM_BASERUN_SB   1    6261.8 391322 11737
## - TEAM_BATTING_HR   1    9954.1 395014 11758
## - TEAM_FIELDING_E   1   11117.2 396177 11765
## - TEAM_FIELDING_DP  1   14730.1 399790 11786
## - TEAM_BATTING_H    1   28214.8 413275 11861


##
## 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 = train_imputed)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -50.199  -8.546   0.137   8.403  59.078
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)      25.5796287  5.0908125   5.025 5.44e-07 ***
## TEAM_BATTING_H    0.0473059  0.0036728  12.880  < 2e-16 ***
## TEAM_BATTING_2B  -0.0182067  0.0091925  -1.981 0.047757 *
## TEAM_BATTING_3B   0.0633618  0.0165995   3.817 0.000139 ***
## TEAM_BATTING_HR   0.0752492  0.0098362   7.650 2.95e-14 ***
## TEAM_BATTING_BB   0.0109367  0.0033639   3.251 0.001166 **
## TEAM_BATTING_SO  -0.0114164  0.0024960  -4.574 5.05e-06 ***
## TEAM_BASERUN_SB   0.0254070  0.0041873   6.068 1.52e-09 ***
## TEAM_PITCHING_H  -0.0008567  0.0003209  -2.670 0.007636 **
## TEAM_PITCHING_SO  0.0032349  0.0006702   4.827 1.48e-06 ***
## TEAM_FIELDING_E  -0.0192360  0.0023793  -8.085 1.00e-15 ***
## TEAM_FIELDING_DP -0.1200857  0.0129037  -9.306  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.04 on 2264 degrees of freedom
## Multiple R-squared:  0.3179, Adjusted R-squared:  0.3146
## F-statistic: 95.91 on 11 and 2264 DF,  p-value: < 2.2e-16
```

The R^2 for this model is not much imporoved. Let's check for multicollinearity between variables.

Reviewing the variance inflation factors:

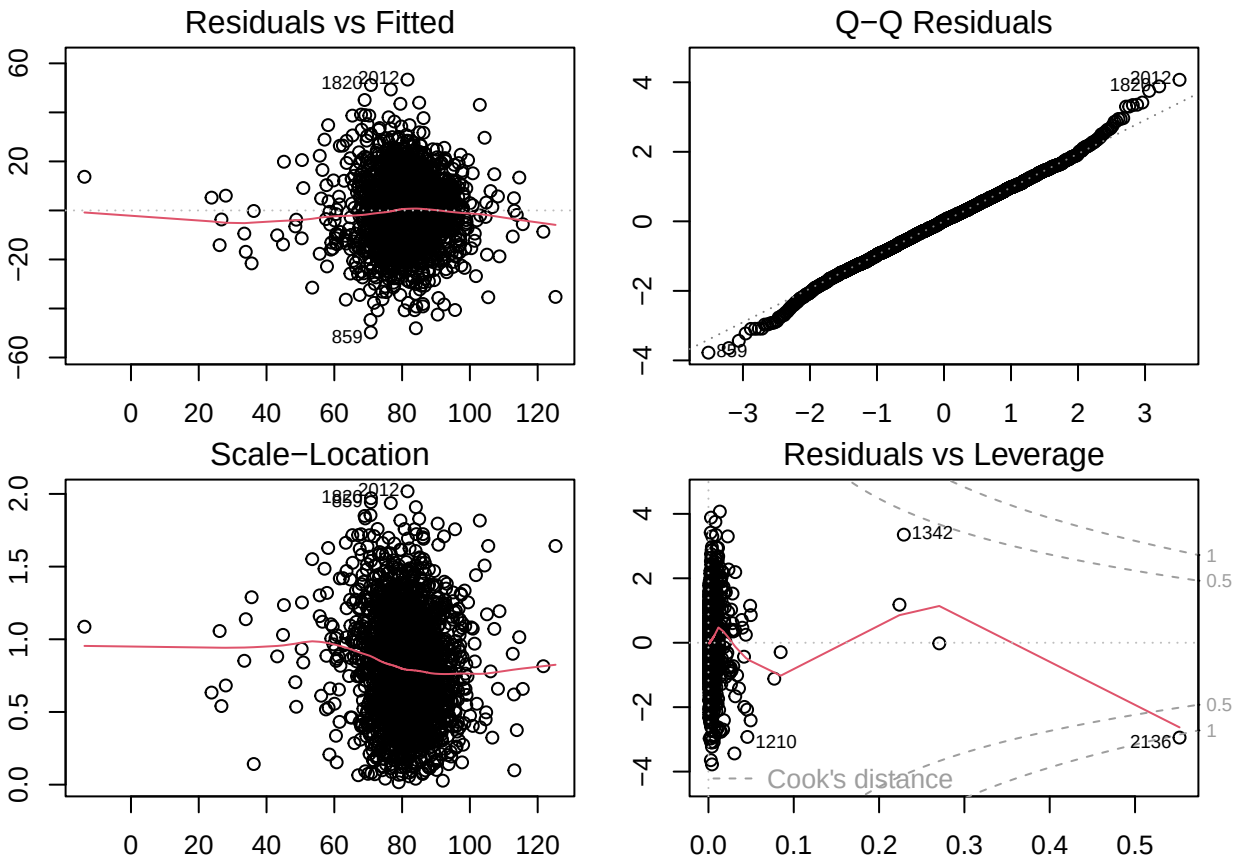
```
##   TEAM_BATTING_H  TEAM_BATTING_2B  TEAM_BATTING_3B  TEAM_BATTING_HR
##         3.772400         2.475799         2.876904         4.744241
##  TEAM_BATTING_BB  TEAM_BATTING_SO  TEAM_BASERUN_SB  TEAM_PITCHING_H
##         2.277646         5.004546         1.710668         2.725389
## TEAM_PITCHING_SO  TEAM_FIELDING_E TEAM_FIELDING_DP
##         1.755393         3.928359         1.339334
```

None of the variance inflation factors are more than 5, but `TEAM_BATTING_HR` and `TEAM_BATTING_SO` come close, with vifs of more than 4.5. We can remove these.

```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_2B +
##     TEAM_BATTING_3B + TEAM_BATTING_BB + TEAM_BASERUN_SB + TEAM_PITCHING_H +
##     TEAM_PITCHING_SO + TEAM_FIELDING_E + TEAM_FIELDING_DP, data = train_imputed)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -49.767  -8.556   0.284   8.688  53.395
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       5.9369813  3.8445024   1.544   0.1227
## TEAM_BATTING_H    0.0588501  0.0032944  17.864  < 2e-16 ***
## TEAM_BATTING_2B  -0.0200136  0.0089860  -2.227   0.0260 *
## TEAM_BATTING_3B   0.0245436  0.0144736   1.696   0.0901 .
## TEAM_BATTING_BB   0.0182085  0.0032668   5.574 2.79e-08 ***
## TEAM_BASERUN_SB   0.0177614  0.0040714   4.362 1.34e-05 ***
## TEAM_PITCHING_H  -0.0006363  0.0003150  -2.020   0.0435 *
## TEAM_PITCHING_SO  0.0026345  0.0006164   4.274 2.00e-05 ***
## TEAM_FIELDING_E  -0.0204470  0.0023866  -8.567  < 2e-16 ***
## TEAM_FIELDING_DP -0.1055226  0.0129131  -8.172 5.00e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.2 on 2266 degrees of freedom
## Multiple R-squared:  0.3002, Adjusted R-squared:  0.2974
## F-statistic:   108 on 9 and 2266 DF,  p-value: < 2.2e-16
```

```
##   TEAM_BATTING_H  TEAM_BATTING_2B  TEAM_BATTING_3B  TEAM_BATTING_BB
##         2.961141         2.308168         2.133915         2.095827
##  TEAM_BASERUN_SB  TEAM_PITCHING_H TEAM_PITCHING_SO  TEAM_FIELDING_E
##         1.577937         2.563523         1.448590         3.856361
## TEAM_FIELDING_DP
##         1.308612
```

We can make some plots to help test our assumptions of our basic model using the `plot` function on our model variable:



We can see from the diagnostic plots that there is definitely something up with this model. Let's try transforming some of the variables to see if we can come up with a better model.

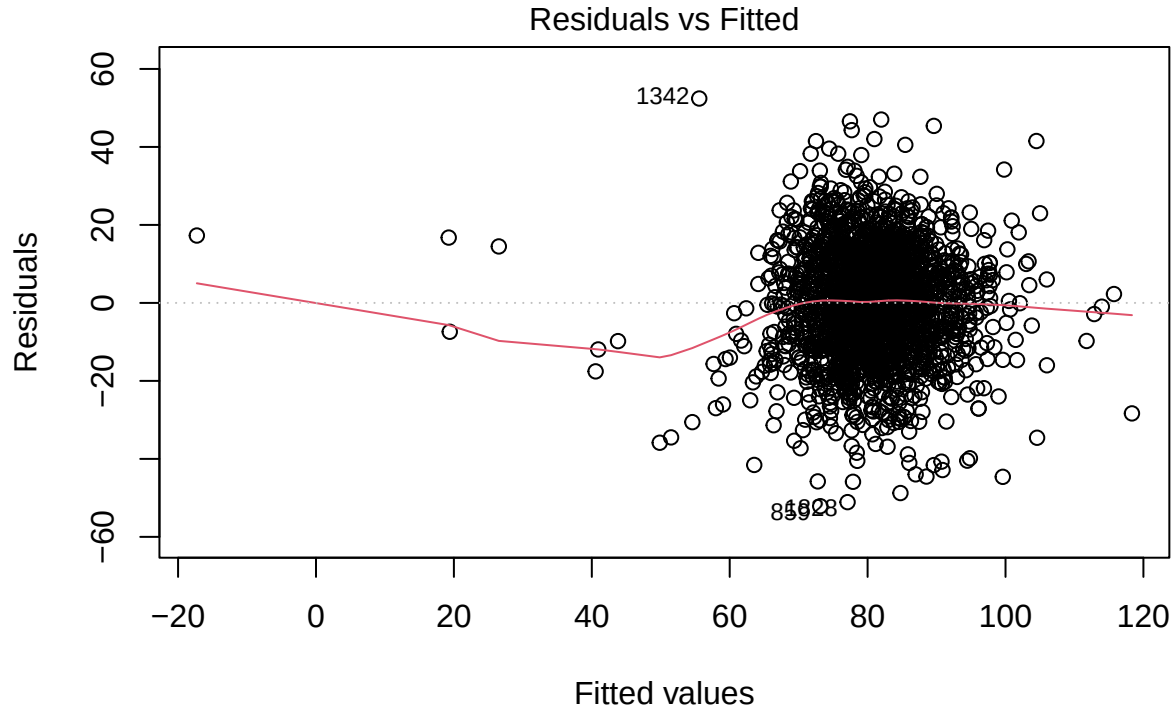
Now we can make a model with inputs that we know from baseball.

- Total hits (`TEAM_BATTING_H`)
- Total walks gained (`TEAM_BATTING_BB`)
- Total hits allowed (`TEAM_PITCHING_H`)
- Total walks allowed (`TEAM_PITCHING_BB`)

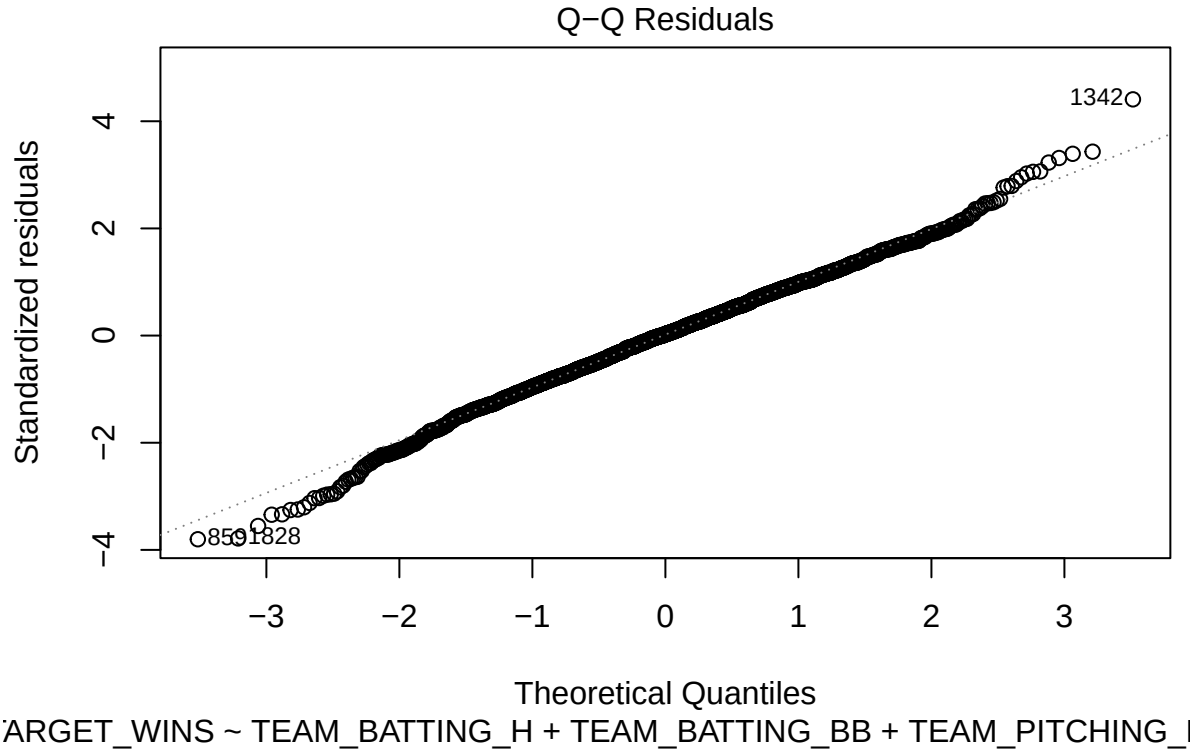
We chose these variables based on our understanding that good teams generally tend to get on base more frequently (positive predictor variables `TEAM_BATTING_HITS` and `TEAM_BATTING_BB`) while allowing *fewer* runners on base (negative predictor variables `TEAM_PITCHING_H` and `TEAM_PITCHING_BB`).

```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_BB +
##     TEAM_PITCHING_H + TEAM_PITCHING_BB, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -52.133  -8.860   0.379   9.373  52.416
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -0.3518000  3.2552864  -0.108 0.913949
```

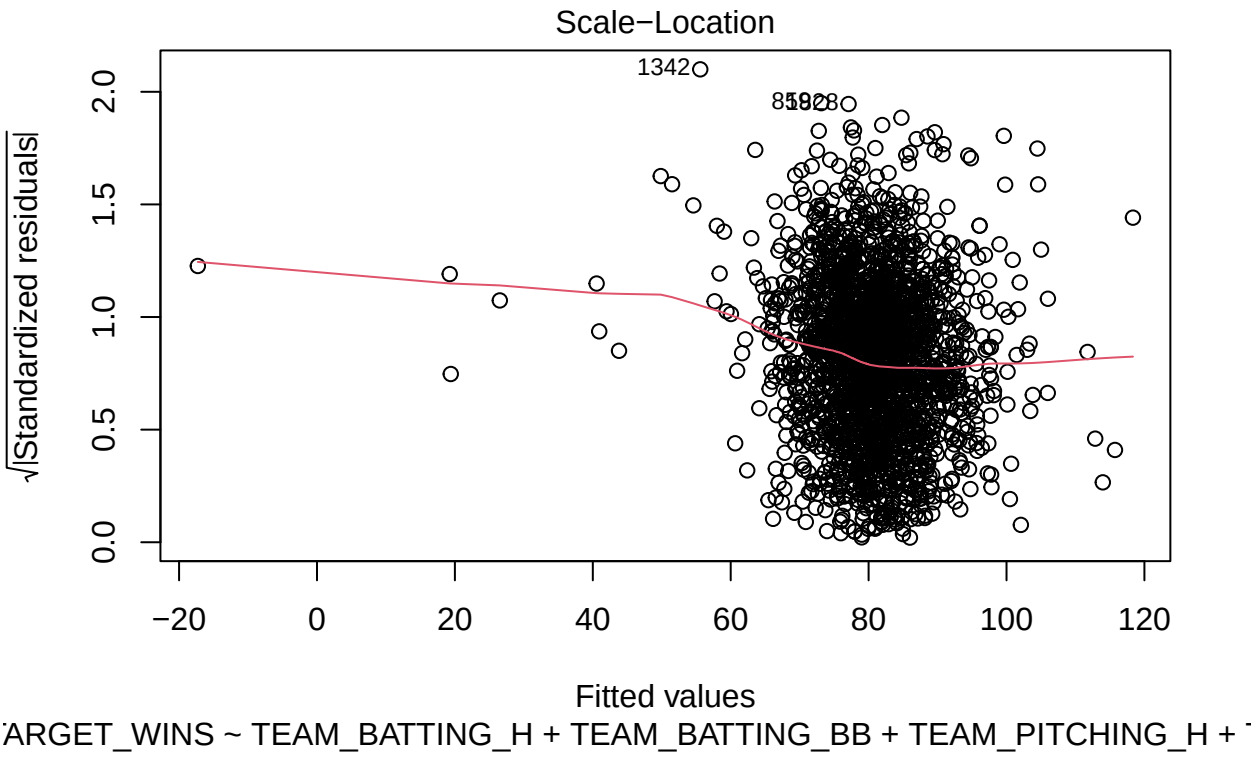
```
## TEAM_BATTING_H    0.0497667  0.0021032  23.663  < 2e-16 ***
## TEAM_BATTING_BB   0.0148499  0.0039923   3.720 0.000204 ***
## TEAM_PITCHING_H  -0.0025469  0.0003317  -7.679 2.36e-14 ***
## TEAM_PITCHING_BB  0.0092317  0.0027681   3.335 0.000867 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.73 on 2271 degrees of freedom
## Multiple R-squared:  0.2416, Adjusted R-squared:  0.2403
## F-statistic: 180.9 on 4 and 2271 DF,  p-value: < 2.2e-16
```

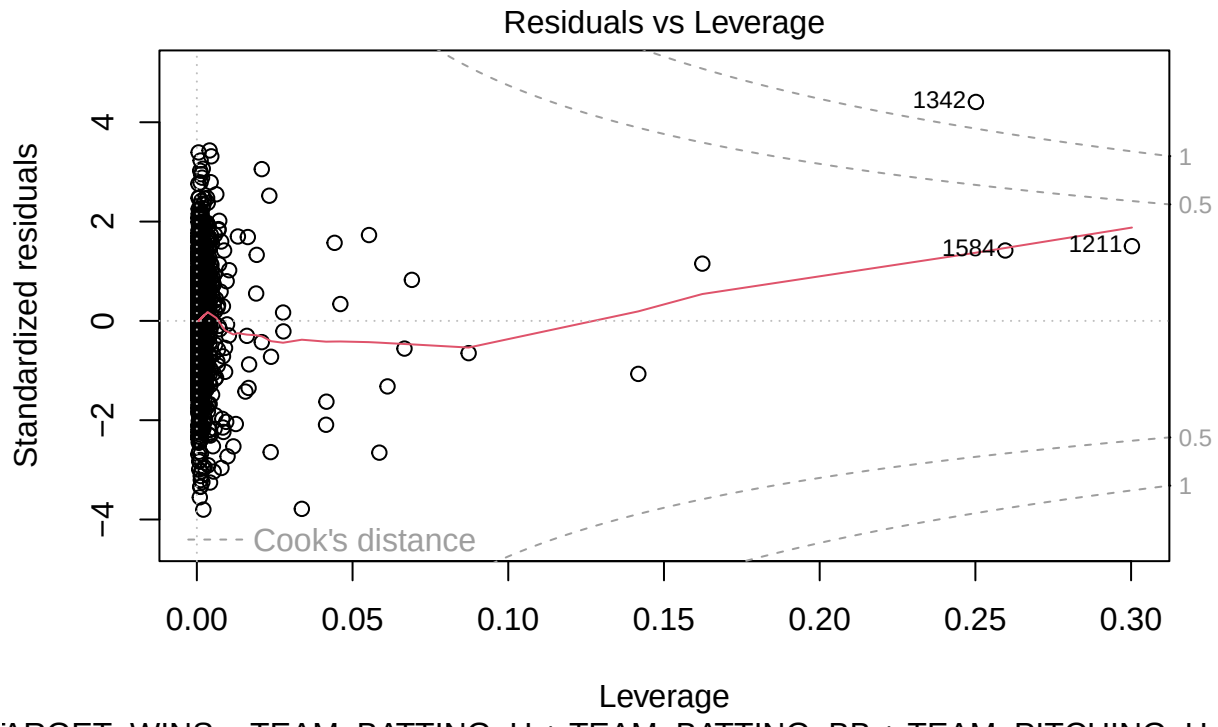


'ARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_BB + TEAM_PITCHING_H + T



'ARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_BB + TEAM_PITCHING_H + T

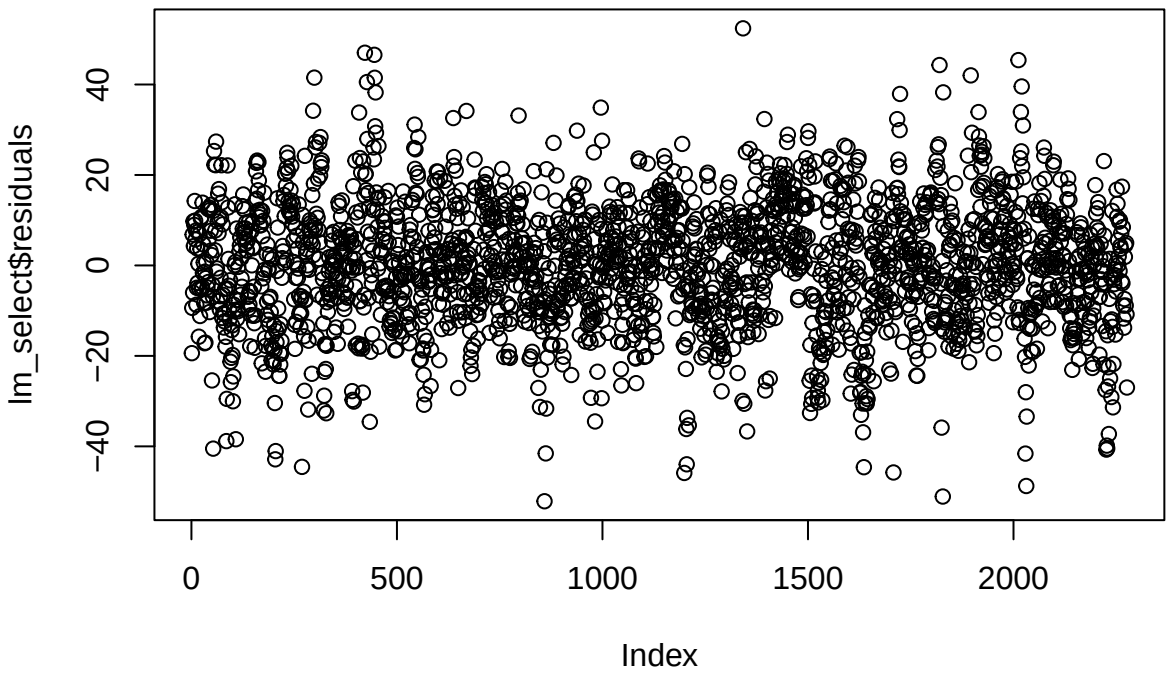

Scale−Location
1342
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
−20
0
20
40
60
80
100
120
Fitted values
ARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_BB + TEAM_PITCHING_H +



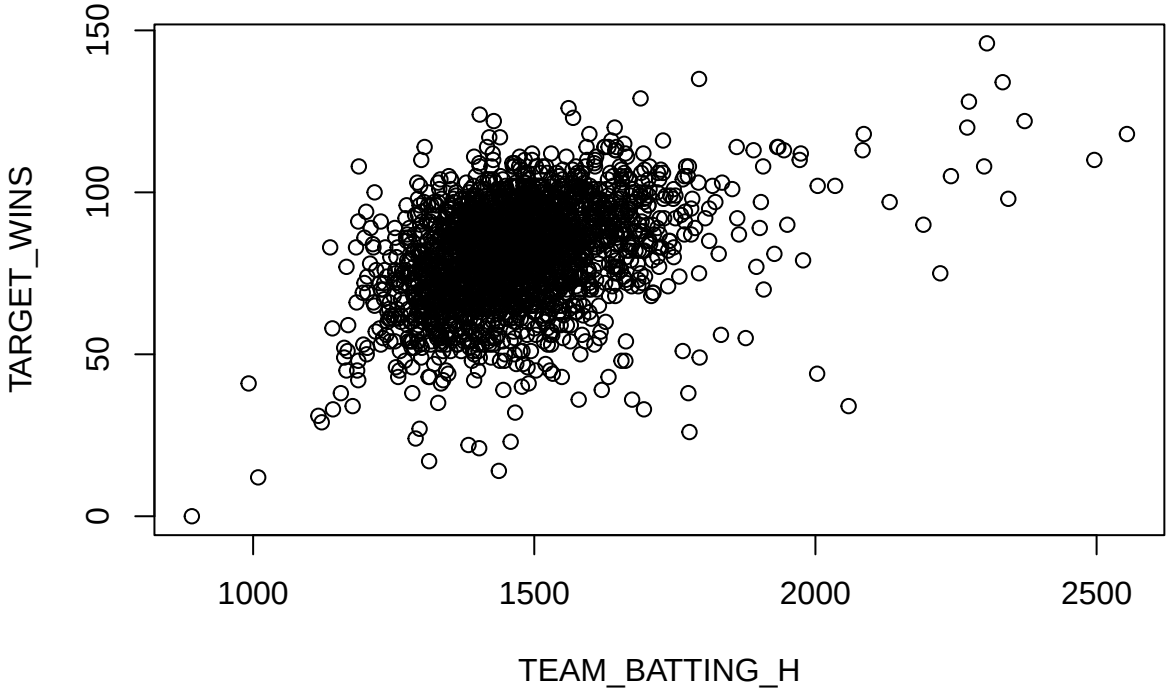
'ARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_BB + TEAM_PITCHING_H + T

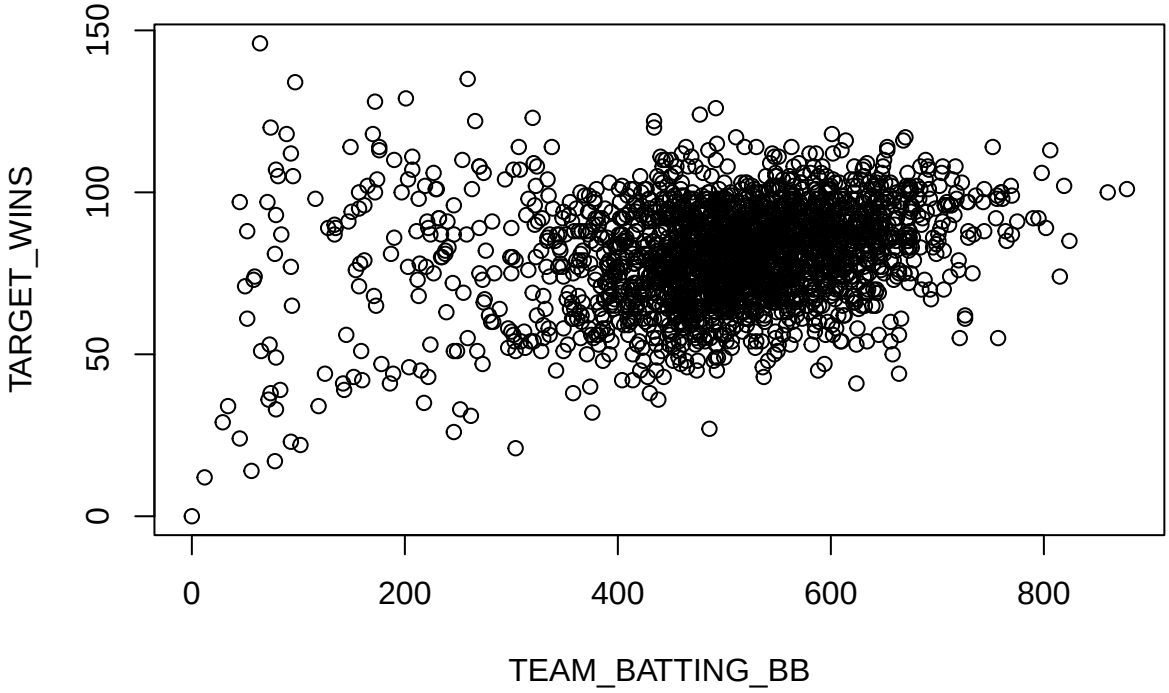
It's interesting to note that with selected variables (walks and hits gained/allowed per team) that our adjusted R^2 actually went *down*, indicating the amount of variability in `TARGET_WINS` explained by our more selective walks/hits model is *less* than the model including all variables.

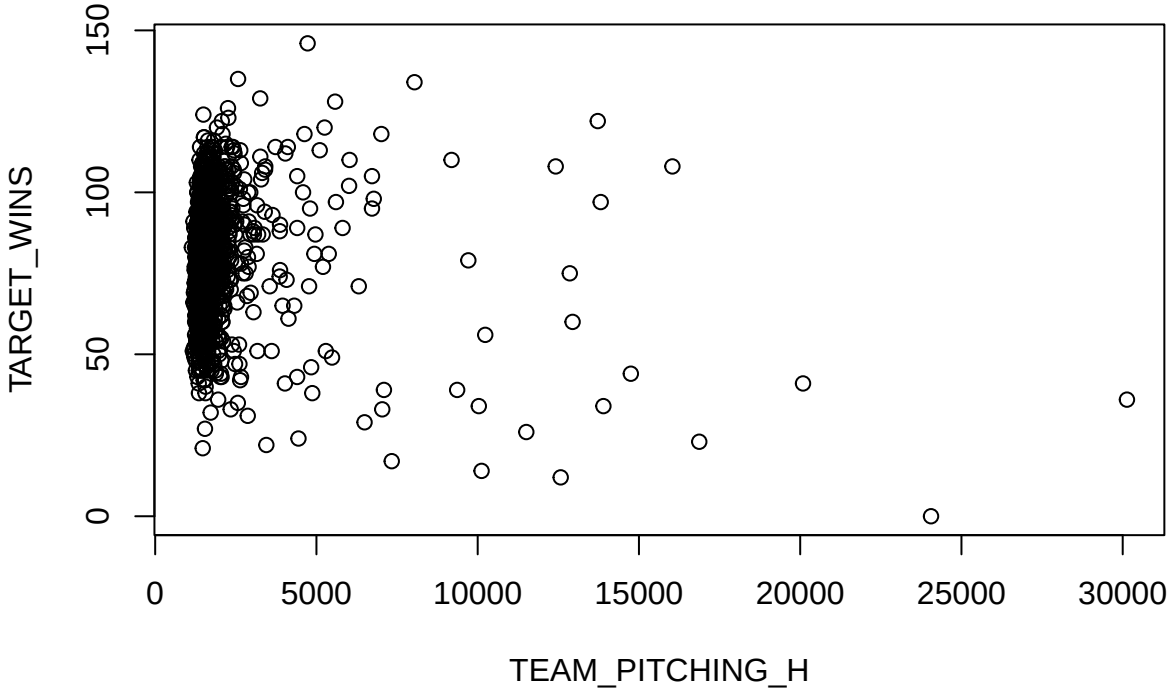
Looking at our residual plot above, there seems to be a clustering of residuals along the x-axis at $X \approx 80$. This shows a pattern in our residuals.

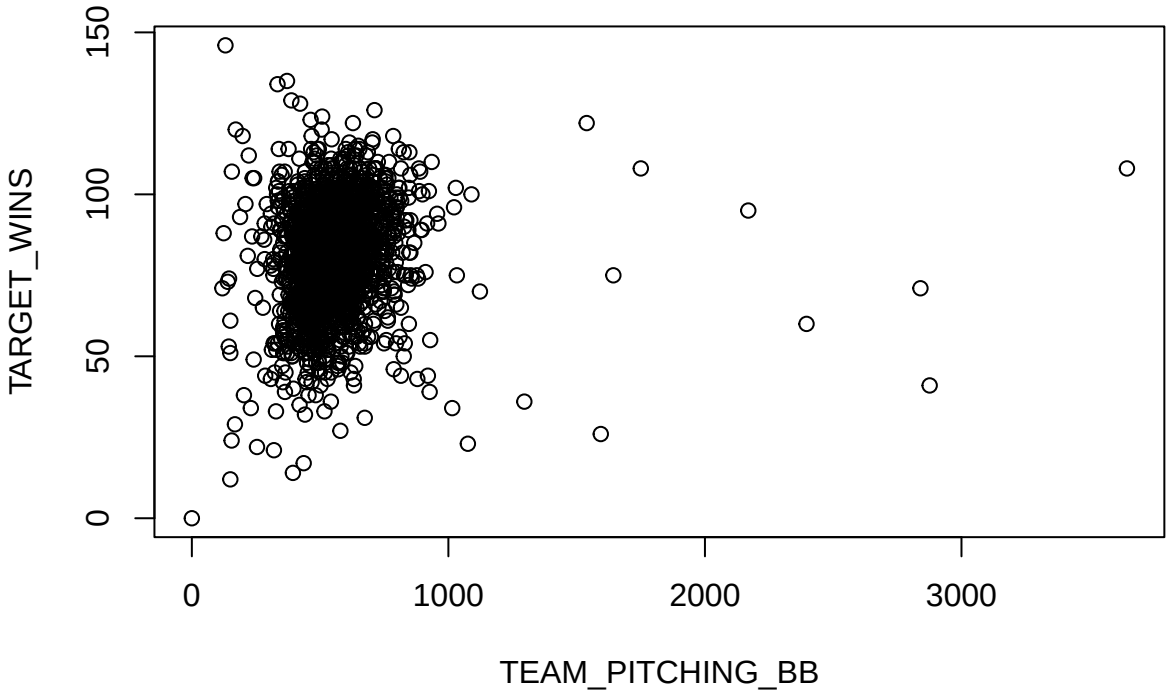


Let's plot our response variable (*Total Wins*) versus each of our predictor variables to get a sense of linear relationships.


TARGET_WINS
0
50
100
150
1000
1500
2000
2500
TEAM_BATTING_H


TARGET_WINS
0
50
100
150
0
200
400
600
800
TEAM_BATTING_BB


TARGET_WINS
0
50
100
150
0
5000
10000
15000
20000
25000
30000
TEAM_PITCHING_H



Model Evaluation:

We'll need to read in our evaluation data, which is hosted on GitHub for reproducability.

```
##        1        2        3        4        5        6        7        8
##  63.23602  65.24971  74.91579  85.85282       NA       NA       NA       NA
##        9       10       11       12       13       14       15       16
##       NA       NA       NA  83.24926  82.85256  83.18613  85.00536  77.70143
##       17       18       19       20       21       22       23       24
##  74.46524  78.45655       NA  91.80627  81.76728  84.13052  81.70410  73.00311
##       25       26       27       28       29       30       31       32
##  81.84349  86.66850       NA  75.85111  84.33410  76.05059  91.10386  85.74112
##       33       34       35       36       37       38       39       40
##  82.70785  85.25632  80.82549  87.26430  76.24457  90.68992  86.60551  92.95695
##       41       42       43       44       45       46       47       48
##  83.19694  90.60615       NA       NA       NA       NA       NA  76.97539
##       49       50       51       52       53       54       55       56
##       NA  79.50032       NA       NA  77.76400  74.16505  75.52137  78.52167
##       57       58       59       60       61       62       63       64
##       NA       NA       NA       NA       NA  74.03376  87.76037  85.45513
##       65       66       67       68       69       70       71       72
##  82.91220       NA       NA       NA       NA       NA       NA       NA
##       73       74       75       76       77       78       79       80
##  77.94366  90.08195  81.62173  85.63471  81.15208  83.00628       NA       NA
##       81       82       83       84       85       86       87       88
```

```
##  85.39660  89.07483  98.03472  75.17305  86.09336  80.53230  82.37584  83.10393
##        89        90        91        92        93        94        95        96
##  87.29484  89.35988        NA        NA        NA        NA        NA        NA
##        97        98        99       100       101       102       103       104
##  86.96746 103.77723  86.94610  86.77244  80.19659  75.20188  84.47138  84.74274
##       105       106       107       108       109       110       111       112
##  79.42854        NA        NA  77.08059  86.58201        NA  83.35091  83.55853
##       113       114       115       116       117       118       119       120
##  92.68678  90.95458  80.63537  77.76309  85.16487  80.04574  74.92265        NA
##       121       122       123       124       125       126       127       128
##        NA        NA        NA        NA        NA  88.66168  92.72379        NA
##       129       130       131       132       133       134       135       136
##        NA        NA        NA        NA  79.34480  85.33466  86.61533        NA
##       137       138       139       140       141       142       143       144
##  73.47470  77.51922  83.74592  79.78498        NA        NA  90.79688  74.41690
##       145       146       147       148       149       150       151       152
##  71.91754  72.68748  77.90210  78.52320  78.75304  82.96470  82.15067  79.51136
##       153       154       155       156       157       158       159       160
##        NA  71.06134  76.61200  69.96087  88.75493        NA        NA        NA
##       161       162       163       164       165       166       167       168
## 105.21687 107.03951  94.58070 104.81734  98.92249  90.15923  81.97034  81.09179
##       169       170       171       172       173       174       175       176
##  72.66909  80.29999        NA        NA  81.31745  94.23157  84.74321  73.87729
##       177       178       179       180       181       182       183       184
##  77.71037  71.54637  75.10342  79.10381  83.64392  88.15604  84.42011  85.42939
##       185       186       187       188       189       190       191       192
##        NA        NA        NA        NA        NA        NA        NA        NA
##       193       194       195       196       197       198       199       200
##  77.28902        NA        NA        NA        NA  84.89172  80.43223  84.98273
##       201       202       203       204       205       206       207       208
##  76.72658  79.90165  73.82205  88.74155  79.88163  82.84956  77.86634  77.23620
##       209       210       211       212       213       214       215       216
##        NA        NA        NA        NA        NA  66.02570  69.64692  84.40673
##       217       218       219       220       221       222       223       224
##  79.54112  90.95266  77.09233  78.09602  78.27364  73.38589  81.35709  73.27934
##       225       226       227       228       229       230       231       232
##        NA  74.70968  81.28400  79.14518  81.36531        NA        NA  92.30749
##       233       234       235       236       237       238       239       240
##        NA  89.06365  80.49515  75.16815  82.51824  77.07291        NA        NA
##       241       242       243       244       245       246       247       248
##        NA        NA        NA  80.63526  60.67502  86.87628  80.93372  84.82034
##       249       250       251       252       253       254       255       256
##  73.10597  82.51426  81.29360        NA        NA        NA  69.32660  76.62059
##       257       258       259
##  81.02967  81.93077  77.65060
```

Appendix: Report Code

Below is the code for this report to generate the models and charts above.

```
knitr::opts_chunk$set(echo = FALSE, warning = FALSE, message = FALSE)
library(glue)
```

```r
library(tidyverse)
library(car)
library(ResourceSelection)
library(VIM)
df <- read.csv("https://raw.githubusercontent.com/andrewbowen19/businessAnalyticsDataMiningDATA621/main/
df <- data.frame(df)
dim = dim(df)

print(glue("The dataset consists of {dim[1]} observations of {dim[2]} variables."))
train <- subset(df, select=-c(INDEX))
mean_wins <- mean(train$TARGET_WINS)
median_wins <- median(train$TARGET_WINS)
sd_wins <- sd(train$TARGET_WINS)

# Print summary stats
print(glue("The mean number of wins in a season is {round(mean_wins,2)}."))
print(glue("The median number of wins in a season is {median_wins}."))
print(glue("The standard deviation for number of wins in a season is {round(sd_wins,2)}."))
ggplot(train, aes(x=TARGET_WINS)) +
  geom_histogram() +
  labs(title = "Distribution of Wins (Histogram)", x = "Number of Wins", y = "Count")
ggplot(train, aes(x=TARGET_WINS)) +
  geom_boxplot(fill="darkgrey") +
  labs(title = "Distribution of Wins (Boxplot)", x = "Number of Wins", y = "Count")
summary(train)
par(mfrow=c(2,3))
par(mai=c(.3,.3,.3,.3))

variables <- c("TEAM_BATTING_SO", "TEAM_BASERUN_SB", "TEAM_BASERUN_CS", "TEAM_BATTING_HBP", "TEAM_PITCH

for (i in 1:(length(variables))) {
  if (variables[i] == "TEAM_PITCHING_SO"){
    hist(train[[variables[i]]], main = variables[i], col = "lightblue", breaks = 100, xlim = c(0,2500))
  }
  else {
    hist(train[[variables[i]]], main = variables[i], col = "lightblue")
  }
}
train_imputed <- train |>
  mutate(TEAM_BASERUN_SB = replace(TEAM_BASERUN_SB, is.na(TEAM_BASERUN_SB), median(TEAM_BASERUN_SB, na.
         TEAM_BASERUN_CS = replace(TEAM_BASERUN_CS, is.na(TEAM_BASERUN_CS), median(TEAM_BASERUN_CS, na.
         TEAM_PITCHING_SO = replace(TEAM_PITCHING_SO, is.na(TEAM_PITCHING_SO), mean(TEAM_PITCHING_SO, na
         TEAM_FIELDING_DP = replace(TEAM_FIELDING_DP, is.na(TEAM_FIELDING_DP), mean(TEAM_FIELDING_DP, na
  select(-TEAM_BATTING_HBP)

train_imputed <- train_imputed |>
    VIM::kNN(variable = "TEAM_BATTING_SO", k = 15, numFun = weighted.mean,
             weightDist = TRUE) |>
    select(-TEAM_BATTING_SO_imp)

cor(train_imputed, df$TARGET_WINS)
train_cleaned <- train_imputed |> rename_all(~stringr::str_replace(.,"^TEAM_",""))
subset_batting <- train_cleaned |> select(contains('batting'))
```

```r
kdepairs(subset_batting)
subset_pitching <- train_cleaned |> select(!contains('batting'), -TARGET_WINS)
kdepairs(subset_pitching)
lm_all <- lm(TARGET_WINS~., train_imputed)
summary(lm_all)
lm_all_reduced <- step(lm_all, direction="backward")
summary(lm_all_reduced)
vif(lm_all_reduced)
lm_all_reduced <- update(lm_all_reduced, .~. - TEAM_BATTING_HR - TEAM_BATTING_SO)
summary(lm_all_reduced)
vif(lm_all_reduced)
par(mfrow=c(2,2))
par(mai=c(.3,.3,.3,.3))
plot(lm_all_reduced)
# Create model with select inputs (walks and hits allowed/gained)
lm_select <- lm(TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_BB + TEAM_PITCHING_H + TEAM_PITCHING_BB, tra

summary(lm_select)
plot(lm_select)

# Plot selective model residuals
plot(lm_select$residuals)

# Plot our response variable for each predictor variable to get a sense of
plot(TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_BB + TEAM_PITCHING_H + TEAM_PITCHING_BB, data=train)

eval_data_url <- "https://raw.githubusercontent.com/andrewbowen19/businessAnalyticsDataMiningDATA621/ma

test <- read.csv(eval_data_url)
predict(lm_all, test)
```