Homework 1: MoneyBall

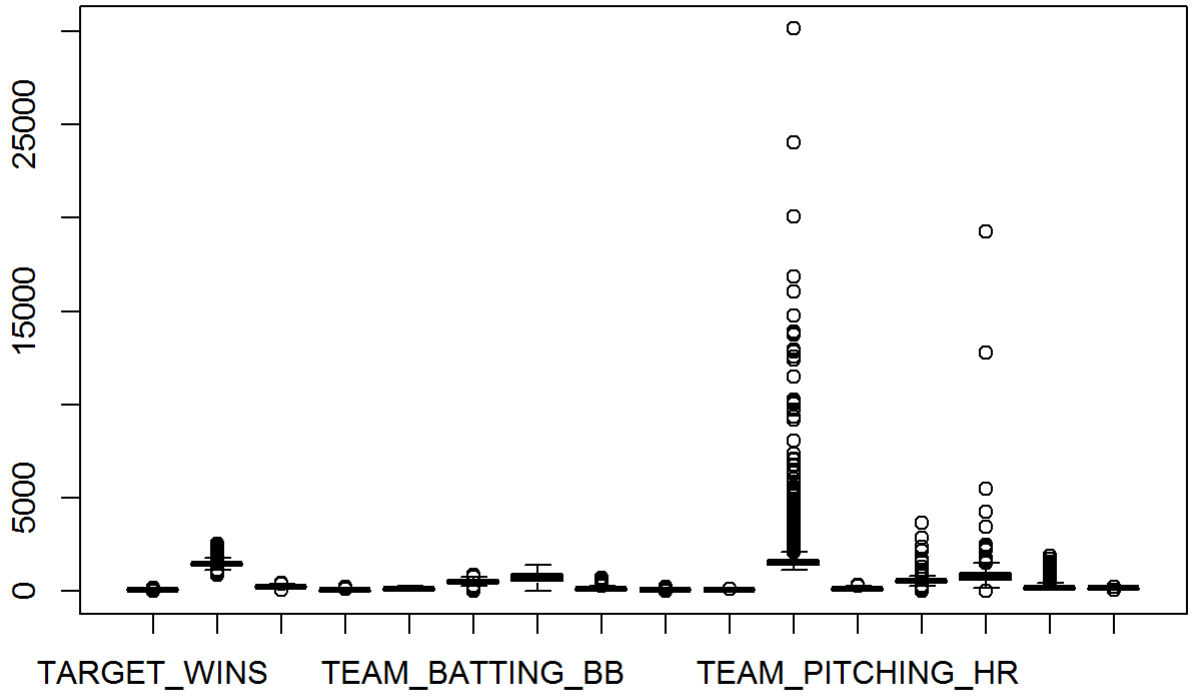
Omar Pineda, Jeffrey Littlejohn, Sergio Ortega Cruz, Chester Poon, Simon Ustoyev

September 25, 2019

1. Data Exploration

Our dataset includes 2,276 observations, meaning performances for professional baseball teams between the years 1871-2006. Initially, we had 15 variables that we could use to model/predict TARGET_WINS, the number of wins a team will have. The variable TEAM_BATTING_HBP only has values for 191 of our observations, and TEAM_BASERUN_CS values were missing for 772 observations. Some other variables were missing a neglible number of values. A boxplot of the values for our variables revealed outliers in our TEAM_PITCHING_H and TEAM_PITCHING_SO variables.

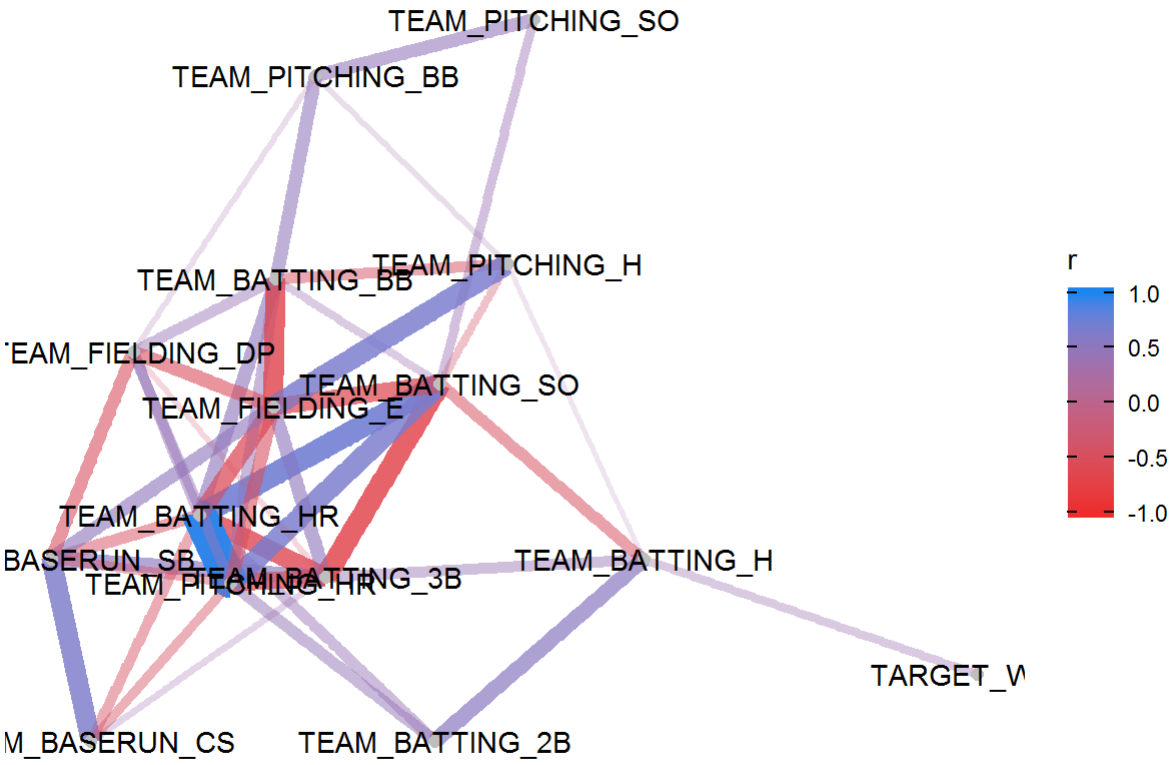
```
##                  vars    n    mean      sd median trimmed    mad  min
## TARGET_WINS         1 2276   80.79   15.75   82.0   81.31  14.83    0
## TEAM_BATTING_H      2 2276 1469.27  144.59 1454.0 1459.04 114.16  891
## TEAM_BATTING_2B     3 2276  241.25   46.80  238.0  240.40  47.44   69
## TEAM_BATTING_3B     4 2276   55.25   27.94   47.0   52.18  23.72    0
## TEAM_BATTING_HR     5 2276   99.61   60.55  102.0   97.39  78.58    0
## TEAM_BATTING_BB     6 2276  501.56  122.67  512.0  512.18  94.89    0
## TEAM_BATTING_SO     7 2174  735.61  248.53  750.0  742.31 284.66    0
## TEAM_BASERUN_SB     8 2145  124.76   87.79  101.0  110.81  60.79    0
## TEAM_BASERUN_CS     9 1504   52.80   22.96   49.0   50.36  17.79    0
## TEAM_BATTING_HBP   10  191   59.36   12.97   58.0   58.86  11.86   29
## TEAM_PITCHING_H    11 2276 1779.21 1406.84 1518.0 1555.90 174.95 1137
## TEAM_PITCHING_HR   12 2276  105.70   61.30  107.0  103.16  74.13    0
## TEAM_PITCHING_BB   13 2276  553.01  166.36  536.5  542.62  98.59    0
## TEAM_PITCHING_SO   14 2174  817.73  553.09  813.5  796.93 257.23    0
## TEAM_FIELDING_E    15 2276  246.48  227.77  159.0  193.44  62.27   65
## TEAM_FIELDING_DP   16 1990  146.39   26.23  149.0  147.58  23.72   52
##                    max range  skew kurtosis    se
## TARGET_WINS        146   146 -0.40     1.03  0.33
## TEAM_BATTING_H    2554  1663  1.57     7.28  3.03
## TEAM_BATTING_2B    458   389  0.22     0.01  0.98
## TEAM_BATTING_3B    223   223  1.11     1.50  0.59
## TEAM_BATTING_HR    264   264  0.19    -0.96  1.27
## TEAM_BATTING_BB    878   878 -1.03     2.18  2.57
## TEAM_BATTING_SO   1399  1399 -0.30    -0.32  5.33
## TEAM_BASERUN_SB    697   697  1.97     5.49  1.90
## TEAM_BASERUN_CS    201   201  1.98     7.62  0.59
## TEAM_BATTING_HBP    95    66  0.32    -0.11  0.94
## TEAM_PITCHING_H  30132 28995 10.33   141.84 29.49
## TEAM_PITCHING_HR   343   343  0.29    -0.60  1.28
## TEAM_PITCHING_BB  3645  3645  6.74    96.97  3.49
## TEAM_PITCHING_SO 19278 19278 22.17   671.19 11.86
## TEAM_FIELDING_E   1898  1833  2.99    10.97  4.77
## TEAM_FIELDING_DP   228   176 -0.39     0.18  0.59
```



We also created a correlation matrix and correlation network to assess which variables are most useful for predicting TARGET_WINS and to explore possible multicollinearity between variables. TEAM_BATTING_H is the variable most highly correlated with TARGET_WINS. We visualized this and more through a correlation network with variables positioned and clustered by their correlation to one another. Red edges indicate negative correlations while blue ones indicate positive correlations.

```
## # A tibble: 16 x 17
##    rowname TARGET_WINS TEAM_BATTING_H TEAM_BATTING_2B TEAM_BATTING_3B
##    <chr>         <dbl>          <dbl>           <dbl>           <dbl>
##  1 TARGET~     NA             0.389           0.289           0.143
##  2 TEAM_B~      0.389        NA               0.563           0.428
##  3 TEAM_B~      0.289         0.563          NA              -0.107
##  4 TEAM_B~      0.143         0.428          -0.107          NA
##  5 TEAM_B~      0.176        -0.00654         0.435          -0.636
##  6 TEAM_B~      0.233        -0.0725          0.256          -0.287
##  7 TEAM_B~     -0.0318       -0.464           0.163          -0.670
##  8 TEAM_B~      0.135         0.124          -0.200           0.534
##  9 TEAM_B~      0.0224        0.0167         -0.0998          0.349
## 10 TEAM_B~      0.0735       -0.0291          0.0461         -0.174
## 11 TEAM_P~     -0.110         0.303           0.0237          0.195
## 12 TEAM_P~      0.189         0.0729          0.455          -0.568
## 13 TEAM_P~      0.124         0.0942          0.178          -0.00222
## 14 TEAM_P~     -0.0784       -0.253           0.0648         -0.259
## 15 TEAM_F~     -0.176         0.265          -0.235           0.510
## 16 TEAM_F~     -0.0349        0.155           0.291          -0.323
## # ... with 12 more variables: TEAM_BATTING_HR <dbl>,
## #   TEAM_BATTING_BB <dbl>, TEAM_BATTING_SO <dbl>, TEAM_BASERUN_SB <dbl>,
## #   TEAM_BASERUN_CS <dbl>, TEAM_BATTING_HBP <dbl>, TEAM_PITCHING_H <dbl>,
## #   TEAM_PITCHING_HR <dbl>, TEAM_PITCHING_BB <dbl>,
## #   TEAM_PITCHING_SO <dbl>, TEAM_FIELDING_E <dbl>, TEAM_FIELDING_DP <dbl>
```

```
## # A tibble: 256 x 3
##    x           y                     r
##    <chr>       <chr>             <dbl>
##  1 TARGET_WINS TARGET_WINS       NA
##  2 TARGET_WINS TEAM_BATTING_H     0.389
##  3 TARGET_WINS TEAM_BATTING_2B    0.289
##  4 TARGET_WINS TEAM_BATTING_3B    0.143
##  5 TARGET_WINS TEAM_BATTING_HR    0.176
##  6 TARGET_WINS TEAM_BATTING_BB    0.233
##  7 TARGET_WINS TEAM_BATTING_SO   -0.0318
##  8 TARGET_WINS TEAM_BASERUN_SB    0.135
##  9 TARGET_WINS TEAM_BASERUN_CS    0.0224
## 10 TARGET_WINS TEAM_BATTING_HBP   0.0735
## # ... with 246 more rows
```



2. Data Preparation

We transformed the data by first removing the INDEX variable since it was just an identification variable. We also removed TEAM_PITCHING_H and TEAM_PITCHING_SO since they had several outlier values based on our exploratory boxplots. TEAM_BATTING_HBP only had values for 191 (8.4%) of our performance observations, so we excluded it as well. We considered filling in missing values for the TEAM_BASERUN_CS variable with its mean value since we had values for 1504 (67%) of the observations, but decided to exclude it entirely since it had a very weak correlation (0.02) with TARGET_WINS. TEAM_FIELDING_DP also had several missing values but was weakly correlated with TARGET_WINS, so we removed it. We were thus left with 11 explanatory variables to predict TARGET_WINS.

TEAM_BATTING_SO and TEAM_BASERUN_SB had a few missing values and were both somewhat correlated with TARGET_WINS, so we imputed them with the average value for each respective variable.

3. Build Models

We built 3 different models to predict TARGET_WINS.

Model 1:

Our first model initially included all available variables to model TARGET_WINS and it produced an adjusted R^2 value of 0.286, meaning that our predictors explain about 30% of the variance in TARGET_WINS. We found that some of the predictors were not significant, so we returned to our correlation matrix to look for signs of collinearity in these variables (TEAM_PITCHING_HR, TEAM_BATTING_SO, TEAM_BATTING_BB). TEAM_PITCHING_BB was also not significant but we kept it because it's p-vaue was approximate to our significance level p=0.09>0.05.

```
##
## Call:
## lm(formula = TARGET_WINS ~ ., data = td2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -56.374  -9.030   0.009   8.455  58.635
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)      11.0054863  5.1590309   2.133   0.0330 *
## TEAM_BATTING_H    0.0454696  0.0037152  12.239  < 2e-16 ***
## TEAM_BATTING_2B  -0.0202961  0.0092971  -2.183   0.0291 *
## TEAM_BATTING_3B   0.0771264  0.0167533   4.604 4.38e-06 ***
## TEAM_BATTING_HR   0.0556791  0.0266665   2.088   0.0369 *
## TEAM_BATTING_BB  -0.0004925  0.0043696  -0.113   0.9103
## TEAM_BATTING_SO  -0.0024215  0.0022881  -1.058   0.2900
## TEAM_BASERUN_SB   0.0355961  0.0042933   8.291  < 2e-16 ***
## TEAM_PITCHING_HR -0.0064176  0.0235127  -0.273   0.7849
## TEAM_PITCHING_BB  0.0042140  0.0024776   1.701   0.0891 .
## TEAM_FIELDING_E  -0.0243333  0.0022101 -11.010  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.31 on 2265 degrees of freedom
## Multiple R-squared:  0.2889, Adjusted R-squared:  0.2857
## F-statistic: 92.01 on 10 and 2265 DF,  p-value: < 2.2e-16
```

```
## # A tibble: 11 x 12
##    rowname TARGET_WINS TEAM_BATTING_H TEAM_BATTING_2B TEAM_BATTING_3B
##    <chr>         <dbl>          <dbl>           <dbl>           <dbl>
##  1 TARGET~     NA             0.389            0.289          0.143
##  2 TEAM_B~      0.389        NA                0.563          0.428
##  3 TEAM_B~      0.289         0.563           NA             -0.107
##  4 TEAM_B~      0.143         0.428           -0.107         NA
##  5 TEAM_B~      0.176        -0.00654          0.435         -0.636
##  6 TEAM_B~      0.233        -0.0725           0.256         -0.287
##  7 TEAM_B~     -0.0307       -0.451            0.155         -0.657
##  8 TEAM_B~      0.123         0.114           -0.190          0.501
##  9 TEAM_P~      0.189         0.0729           0.455         -0.568
## 10 TEAM_P~      0.124         0.0942           0.178         -0.00222
## 11 TEAM_F~     -0.176         0.265           -0.235          0.510
## # ... with 7 more variables: TEAM_BATTING_HR <dbl>, TEAM_BATTING_BB <dbl>,
## #   TEAM_BATTING_SO <dbl>, TEAM_BASERUN_SB <dbl>, TEAM_PITCHING_HR <dbl>,
## #   TEAM_PITCHING_BB <dbl>, TEAM_FIELDING_E <dbl>
```

a. TEAM_PITCHING_HR has a correlation coefficient of 0.96 with TEAM_BATTING_HR, and out of the two we chose to keep TEAM_PITCHING_HR since it correlates more strongly with TARGET_WINS.
b. TEAM_BATTING_SO is strongly correlated with TEAM_PITCHING_HR but the former is less correlated with TARGET_WINS so we remove it from our model.
c. TEAM_BATTING_BB is strongly correlated with TEAM_FIELDING_E but it correlates more with TARGET_WINS so we remove TEAM_FIELDING_E.

After making these changes, our adjusted R^2 value becomes 0.246, and all predictors are significant except for TEAM_BATTING_2B, so we decided to remove it. Our final version of model 1 uses 7 variables with all of them being significant to predict TARGET_WINS. This model has an adjusted R^2 value of 0.246.

```
##
## Call:
## lm(formula = TARGET_WINS ~ . - TEAM_BATTING_HR - TEAM_BATTING_SO -
##     TEAM_FIELDING_E, data = td2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -61.138  -8.836   0.511   8.984  80.047
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       2.6055077  3.4744276   0.750 0.453387
## TEAM_BATTING_H    0.0373056  0.0030850  12.093  < 2e-16 ***
## TEAM_BATTING_2B   0.0009266  0.0089373   0.104 0.917436
## TEAM_BATTING_3B   0.0582851  0.0164526   3.543 0.000404 ***
## TEAM_BATTING_BB   0.0335788  0.0030366  11.058  < 2e-16 ***
## TEAM_BASERUN_SB   0.0248728  0.0040479   6.145 9.44e-10 ***
## TEAM_PITCHING_HR  0.0450042  0.0070636   6.371 2.26e-10 ***
## TEAM_PITCHING_BB -0.0086295  0.0020380  -4.234 2.38e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.68 on 2268 degrees of freedom
## Multiple R-squared:  0.2478, Adjusted R-squared:  0.2455
## F-statistic: 106.7 on 7 and 2268 DF,  p-value: < 2.2e-16
```

```
##
## Call:
## lm(formula = TARGET_WINS ~ . - TEAM_BATTING_HR - TEAM_BATTING_SO -
##     TEAM_FIELDING_E - TEAM_BATTING_2B, data = td2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -61.109  -8.824   0.512   8.959  80.193
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       2.522559   3.380315   0.746 0.455594
## TEAM_BATTING_H    0.037500   0.002449  15.314  < 2e-16 ***
## TEAM_BATTING_3B   0.057989   0.016199   3.580 0.000351 ***
## TEAM_BATTING_BB   0.033630   0.002996  11.227  < 2e-16 ***
## TEAM_BASERUN_SB   0.024833   0.004029   6.164 8.37e-10 ***
## TEAM_PITCHING_HR  0.045145   0.006931   6.514 9.00e-11 ***
## TEAM_PITCHING_BB -0.008627   0.002037  -4.234 2.39e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.68 on 2269 degrees of freedom
## Multiple R-squared:  0.2478, Adjusted R-squared:  0.2458
## F-statistic: 124.6 on 6 and 2269 DF,  p-value: < 2.2e-16
```

All predictors in this model influence wins as initially assumed except for TEAM_PITCHING_HR (homeruns allowed) which positively impacts wins when it was predicted that it would have a negative impact. We permit this in the model as its coefficient is 0.05 which is not substantially positive. The most impactful predictor to a team's number of wins is TEAM_BATTING_3B (triples by batters) which makes sense since players that make it to the third base after batting are very likely to score a point for their team since they would only have to run one more base.

Model 2:

For our second model, we use the same variables as those in our first model and implement a square root tranformation on TARGET_WINS. This model's adjusted R^2 increases to 0.253. We then removed 132 influential points that we identified using Cook's Distances, and our resulting model's adjusted R^2 value increased to 0.3.

```
##
## Call:
## lm(formula = sqrt(TARGET_WINS) ~ . - TEAM_BATTING_HR - TEAM_BATTING_SO -
##     TEAM_FIELDING_E - TEAM_BATTING_2B, data = td2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.2341 -0.4699  0.0555  0.5167  4.9068
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       4.3162687  0.1988613  21.705  < 2e-16 ***
## TEAM_BATTING_H    0.0021525  0.0001441  14.942  < 2e-16 ***
## TEAM_BATTING_3B   0.0034289  0.0009530   3.598 0.000327 ***
## TEAM_BATTING_BB   0.0022635  0.0001762  12.844  < 2e-16 ***
## TEAM_BASERUN_SB   0.0014456  0.0002370   6.099 1.25e-09 ***
## TEAM_PITCHING_HR  0.0027134  0.0004077   6.655 3.54e-11 ***
## TEAM_PITCHING_BB -0.0005979  0.0001199  -4.988 6.55e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8048 on 2269 degrees of freedom
## Multiple R-squared:  0.255,  Adjusted R-squared:  0.253
## F-statistic: 129.4 on 6 and 2269 DF,  p-value: < 2.2e-16
```

```
##
## Call:
## lm(formula = sqrt(TARGET_WINS) ~ . - TEAM_BATTING_HR - TEAM_BATTING_SO -
##     TEAM_FIELDING_E - TEAM_BATTING_2B, data = td3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.11156 -0.45238  0.04012  0.46772  1.73413
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       4.6239394  0.1790765  25.821  < 2e-16 ***
## TEAM_BATTING_H    0.0018507  0.0001349  13.723  < 2e-16 ***
## TEAM_BATTING_3B   0.0056626  0.0009005   6.288 3.89e-10 ***
## TEAM_BATTING_BB   0.0023257  0.0002106  11.043  < 2e-16 ***
## TEAM_BASERUN_SB   0.0017245  0.0002038   8.463  < 2e-16 ***
## TEAM_PITCHING_HR  0.0032889  0.0003609   9.114  < 2e-16 ***
## TEAM_PITCHING_BB -0.0007515  0.0001843  -4.078 4.71e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6466 on 2137 degrees of freedom
## Multiple R-squared:  0.3014, Adjusted R-squared:  0.2995
## F-statistic: 153.7 on 6 and 2137 DF,  p-value: < 2.2e-16
```

The coefficients for this model tell the same story as those in our first model, but in this model, the predictors explain the variance in our wins better.

Model 3:

We used forward selection to build our last model, which starts with no predictors in the model, iteratively adds the most contributive predictors, and stops when the improvement is no longer statistically significant. It has 8 variables and an adjusted R^2 of 0.286. All predictors are significant.

In this model, all explanatory variables behave as expected except for TEAM_BATTING_2B and TEAM_PITCHING_BB, but both of these coefficients are close enough to 0 that we can dismiss their change in sign. TEAM_BATTIG_3B has the largest influence on TARGET_WINS, similar to what we found in model 1 which makes sense as we previously discussed. It is followed by TEAM_BATTING_HR (homeruns allowed) and this follows our logic as homeruns by batters would naturally contribute to the number of wins for a team.

```
## Start:  AIC=12550.76
## TARGET_WINS ~ 1
##
##                    Df Sum of Sq    RSS   AIC
## + TEAM_BATTING_H    1     85318 479178 12180
## + TEAM_BATTING_2B   1     47181 517315 12354
## + TEAM_BATTING_BB   1     30530 533966 12426
## + TEAM_PITCHING_HR  1     20167 544329 12470
## + TEAM_FIELDING_E   1     17582 546914 12481
## + TEAM_BATTING_HR   1     17516 546980 12481
## + TEAM_BATTING_3B   1     11480 553016 12506
## + TEAM_PITCHING_BB  1      8704 555792 12517
## + TEAM_BASERUN_SB   1      8536 555960 12518
## + TEAM_BATTING_SO   1       531 563965 12551
## <none>                          564496 12551
##
## Step:  AIC=12179.81
## TARGET_WINS ~ TEAM_BATTING_H
##
##                    Df Sum of Sq    RSS   AIC
## + TEAM_FIELDING_E   1     47417 431762 11945
## + TEAM_BATTING_BB   1     38578 440601 11991
## + TEAM_BATTING_HR   1     18027 461152 12094
## + TEAM_BATTING_SO   1     14792 464387 12110
## + TEAM_PITCHING_HR  1     14654 464524 12111
## + TEAM_PITCHING_BB  1      4366 474812 12161
## + TEAM_BATTING_2B   1      4082 475097 12162
## + TEAM_BASERUN_SB   1      3537 475641 12165
## <none>                          479178 12180
## + TEAM_BATTING_3B   1       387 478791 12180
##
## Step:  AIC=11944.65
## TARGET_WINS ~ TEAM_BATTING_H + TEAM_FIELDING_E
##
##                    Df Sum of Sq    RSS   AIC
## + TEAM_BASERUN_SB   1   21256.9 410505 11832
## + TEAM_BATTING_3B   1    7944.9 423817 11904
## + TEAM_BATTING_BB   1    4858.4 426903 11921
## + TEAM_PITCHING_BB  1    3058.7 428703 11930
## + TEAM_BATTING_2B   1    2199.7 429562 11935
## <none>                          431762 11945
## + TEAM_PITCHING_HR  1      35.3 431727 11946
## + TEAM_BATTING_SO   1      25.7 431736 11946
## + TEAM_BATTING_HR   1       6.5 431755 11947
##
## Step:  AIC=11831.75
## TARGET_WINS ~ TEAM_BATTING_H + TEAM_FIELDING_E + TEAM_BASERUN_SB
##
##                    Df Sum of Sq    RSS   AIC
## + TEAM_PITCHING_HR  1   2376.20 408129 11820
## + TEAM_BATTING_HR   1   2200.62 408304 11822
## + TEAM_BATTING_BB   1   1405.84 409099 11826
## + TEAM_PITCHING_BB  1   1287.10 409218 11827
```

```
## + TEAM_BATTING_3B   1   1021.19 409484 11828
## + TEAM_BATTING_2B   1    506.16 409999 11831
## <none>                          410505 11832
## + TEAM_BATTING_SO   1     55.23 410450 11833
##
## Step:  AIC=11820.54
## TARGET_WINS ~ TEAM_BATTING_H + TEAM_FIELDING_E + TEAM_BASERUN_SB +
##     TEAM_PITCHING_HR
##
##                     Df Sum of Sq    RSS   AIC
## + TEAM_BATTING_3B   1    4340.4 403788 11798
## + TEAM_BATTING_2B   1    1508.1 406621 11814
## + TEAM_BATTING_SO   1    1187.7 406941 11816
## + TEAM_BATTING_BB   1     657.6 407471 11819
## + TEAM_PITCHING_BB  1     543.4 407585 11820
## <none>                          408129 11820
## + TEAM_BATTING_HR   1       0.0 408129 11822
##
## Step:  AIC=11798.2
## TARGET_WINS ~ TEAM_BATTING_H + TEAM_FIELDING_E + TEAM_BASERUN_SB +
##     TEAM_PITCHING_HR + TEAM_BATTING_3B
##
##                     Df Sum of Sq    RSS   AIC
## + TEAM_BATTING_2B   1    862.30 402926 11795
## + TEAM_BATTING_BB   1    452.02 403336 11798
## + TEAM_BATTING_SO   1    444.96 403343 11798
## + TEAM_PITCHING_BB  1    380.80 403408 11798
## <none>                          403788 11798
## + TEAM_BATTING_HR   1    312.33 403476 11798
##
## Step:  AIC=11795.33
## TARGET_WINS ~ TEAM_BATTING_H + TEAM_FIELDING_E + TEAM_BASERUN_SB +
##     TEAM_PITCHING_HR + TEAM_BATTING_3B + TEAM_BATTING_2B
##
##                     Df Sum of Sq    RSS   AIC
## + TEAM_PITCHING_BB  1    522.10 402404 11794
## + TEAM_BATTING_BB   1    503.01 402423 11794
## + TEAM_BATTING_HR   1    372.77 402553 11795
## <none>                          402926 11795
## + TEAM_BATTING_SO   1    193.50 402733 11796
##
## Step:  AIC=11794.38
## TARGET_WINS ~ TEAM_BATTING_H + TEAM_FIELDING_E + TEAM_BASERUN_SB +
##     TEAM_PITCHING_HR + TEAM_BATTING_3B + TEAM_BATTING_2B + TEAM_PITCHING_BB
##
##                    Df Sum of Sq    RSS   AIC
## + TEAM_BATTING_HR  1    776.19 401628 11792
## <none>                          402404 11794
## + TEAM_BATTING_BB  1    125.84 402278 11796
## + TEAM_BATTING_SO  1     98.42 402306 11796
##
## Step:  AIC=11791.99
## TARGET_WINS ~ TEAM_BATTING_H + TEAM_FIELDING_E + TEAM_BASERUN_SB +
##     TEAM_PITCHING_HR + TEAM_BATTING_3B + TEAM_BATTING_2B + TEAM_PITCHING_BB +
```

```
##     TEAM_BATTING_HR
##
##                   Df Sum of Sq    RSS   AIC
## <none>                         401628 11792
## + TEAM_BATTING_SO  1   197.156 401431 11793
## + TEAM_BATTING_BB  1     0.906 401627 11794
```

```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_H + TEAM_FIELDING_E +
##     TEAM_BASERUN_SB + TEAM_PITCHING_HR + TEAM_BATTING_3B + TEAM_BATTING_2B +
##     TEAM_PITCHING_BB + TEAM_BATTING_HR, data = td2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -56.275  -9.027  -0.019   8.463  57.804
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       7.021546   3.305444   2.124   0.0338 *
## TEAM_BATTING_H    0.047593   0.003127  15.219  < 2e-16 ***
## TEAM_FIELDING_E  -0.023948   0.001743 -13.742  < 2e-16 ***
## TEAM_BASERUN_SB   0.034056   0.003957   8.606  < 2e-16 ***
## TEAM_PITCHING_HR -0.007420   0.022272  -0.333   0.7390
## TEAM_BATTING_3B   0.079204   0.016507   4.798  1.7e-06 ***
## TEAM_BATTING_2B  -0.022935   0.008945  -2.564   0.0104 *
## TEAM_PITCHING_BB  0.004297   0.001880   2.286   0.0224 *
## TEAM_BATTING_HR   0.050807   0.024273   2.093   0.0364 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.31 on 2267 degrees of freedom
## Multiple R-squared:  0.2885, Adjusted R-squared:  0.286
## F-statistic: 114.9 on 8 and 2267 DF,  p-value: < 2.2e-16
```

4. Select Models

We will select our best multiple linear regression model based on its adjusted R^2 value, mean squared error, F-statistic and residual plots as shown below.

First, we summarize our three models.

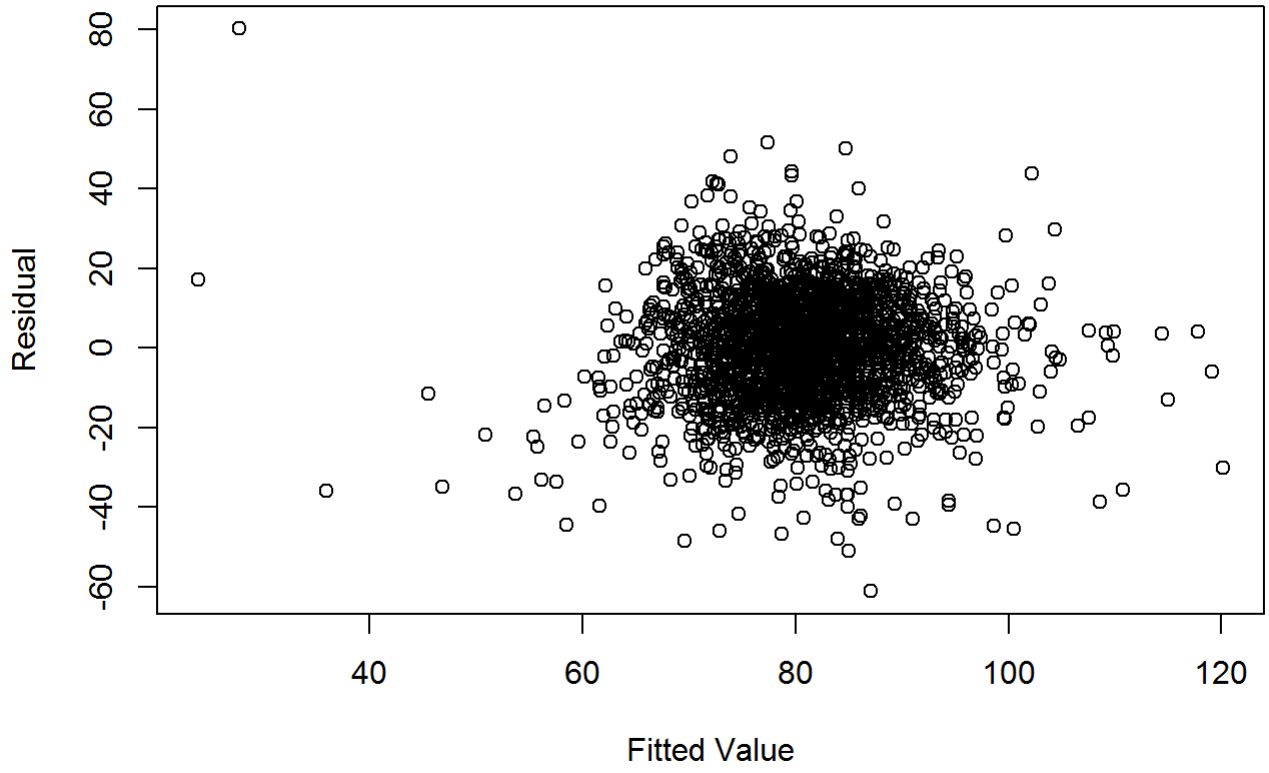
```
##
## Call:
## lm(formula = TARGET_WINS ~ . - TEAM_BATTING_HR - TEAM_BATTING_SO -
##     TEAM_FIELDING_E - TEAM_BATTING_2B, data = td2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -61.109  -8.824   0.512   8.959  80.193
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       2.522559   3.380315   0.746 0.455594
## TEAM_BATTING_H    0.037500   0.002449  15.314  < 2e-16 ***
## TEAM_BATTING_3B   0.057989   0.016199   3.580 0.000351 ***
## TEAM_BATTING_BB   0.033630   0.002996  11.227  < 2e-16 ***
## TEAM_BASERUN_SB   0.024833   0.004029   6.164 8.37e-10 ***
## TEAM_PITCHING_HR  0.045145   0.006931   6.514 9.00e-11 ***
## TEAM_PITCHING_BB -0.008627   0.002037  -4.234 2.39e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.68 on 2269 degrees of freedom
## Multiple R-squared:  0.2478, Adjusted R-squared:  0.2458
## F-statistic: 124.6 on 6 and 2269 DF,  p-value: < 2.2e-16
```

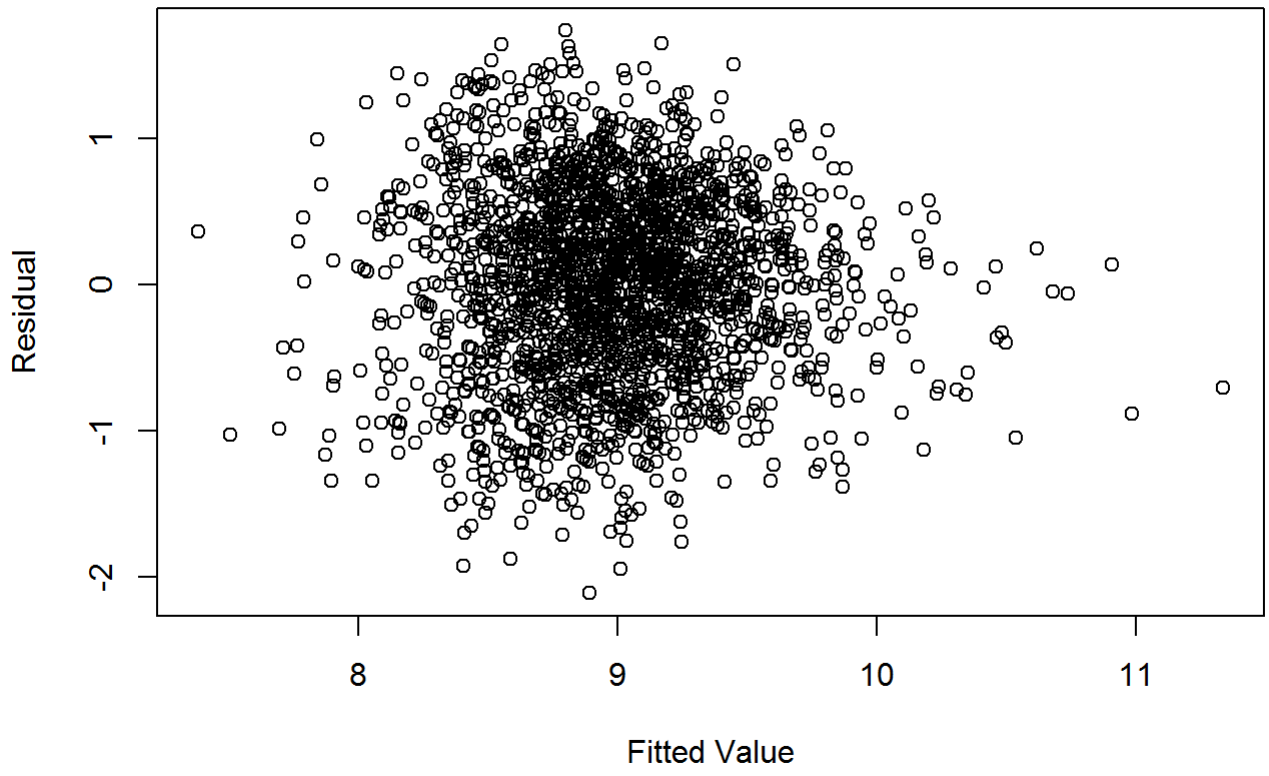
```
##
## Call:
## lm(formula = sqrt(TARGET_WINS) ~ . - TEAM_BATTING_HR - TEAM_BATTING_SO -
##     TEAM_FIELDING_E - TEAM_BATTING_2B, data = td3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.11156 -0.45238  0.04012  0.46772  1.73413
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       4.6239394  0.1790765  25.821  < 2e-16 ***
## TEAM_BATTING_H    0.0018507  0.0001349  13.723  < 2e-16 ***
## TEAM_BATTING_3B   0.0056626  0.0009005   6.288 3.89e-10 ***
## TEAM_BATTING_BB   0.0023257  0.0002106  11.043  < 2e-16 ***
## TEAM_BASERUN_SB   0.0017245  0.0002038   8.463  < 2e-16 ***
## TEAM_PITCHING_HR  0.0032889  0.0003609   9.114  < 2e-16 ***
## TEAM_PITCHING_BB -0.0007515  0.0001843  -4.078 4.71e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6466 on 2137 degrees of freedom
## Multiple R-squared:  0.3014, Adjusted R-squared:  0.2995
## F-statistic: 153.7 on 6 and 2137 DF,  p-value: < 2.2e-16
```

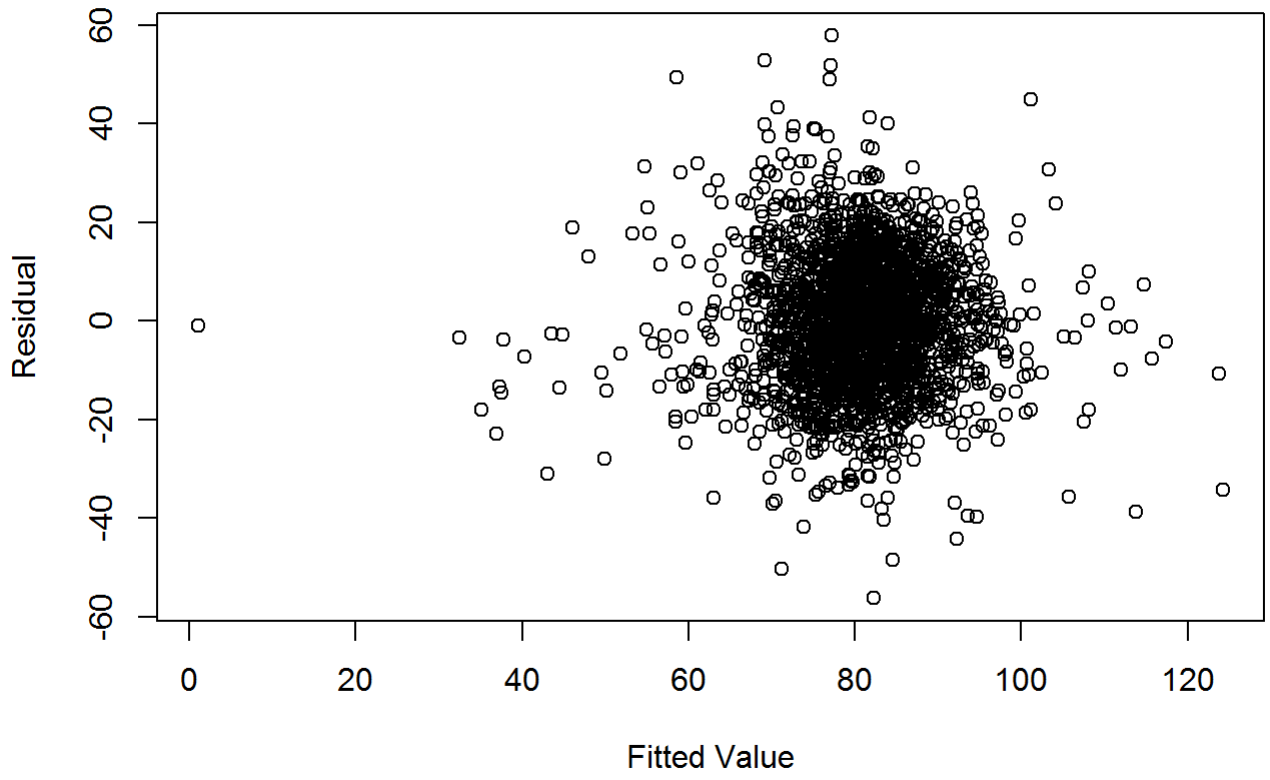
```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_H + TEAM_FIELDING_E +
##     TEAM_BASERUN_SB + TEAM_PITCHING_HR + TEAM_BATTING_3B + TEAM_BATTING_2B +
##     TEAM_PITCHING_BB + TEAM_BATTING_HR, data = td2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -56.275  -9.027  -0.019   8.463  57.804
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       7.021546   3.305444   2.124   0.0338 *
## TEAM_BATTING_H    0.047593   0.003127  15.219  < 2e-16 ***
## TEAM_FIELDING_E  -0.023948   0.001743 -13.742  < 2e-16 ***
## TEAM_BASERUN_SB   0.034056   0.003957   8.606  < 2e-16 ***
## TEAM_PITCHING_HR -0.007420   0.022272  -0.333   0.7390
## TEAM_BATTING_3B   0.079204   0.016507   4.798  1.7e-06 ***
## TEAM_BATTING_2B  -0.022935   0.008945  -2.564   0.0104 *
## TEAM_PITCHING_BB  0.004297   0.001880   2.286   0.0224 *
## TEAM_BATTING_HR   0.050807   0.024273   2.093   0.0364 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.31 on 2267 degrees of freedom
## Multiple R-squared:  0.2885, Adjusted R-squared:  0.286
## F-statistic: 114.9 on 8 and 2267 DF,  p-value: < 2.2e-16
```

Next, we assess our models based on some statistics. Model 3 has the lowest MSE and only explains 1% less of the variance in the predicted wins than model 2, the model with the highest R^2 value. The p-values associated with the F-statistic for all models are statistically significant. Model 3 has the lowest F-statistic but it is not much lower than that of the other models. The residuals for all three models appear to be normally distributed. We thus chose model 3, our forward selection model, as our best method of modeling/predicting the number of wins that a team would have.

```
##     model adjR2Values mseValues fStatValues
## 1 Model 1      0.2458  186.5593       124.6
## 2 Model 2      0.2995 5399.1891       153.7
## 3 Model 3      0.2860  176.4621       114.9
```

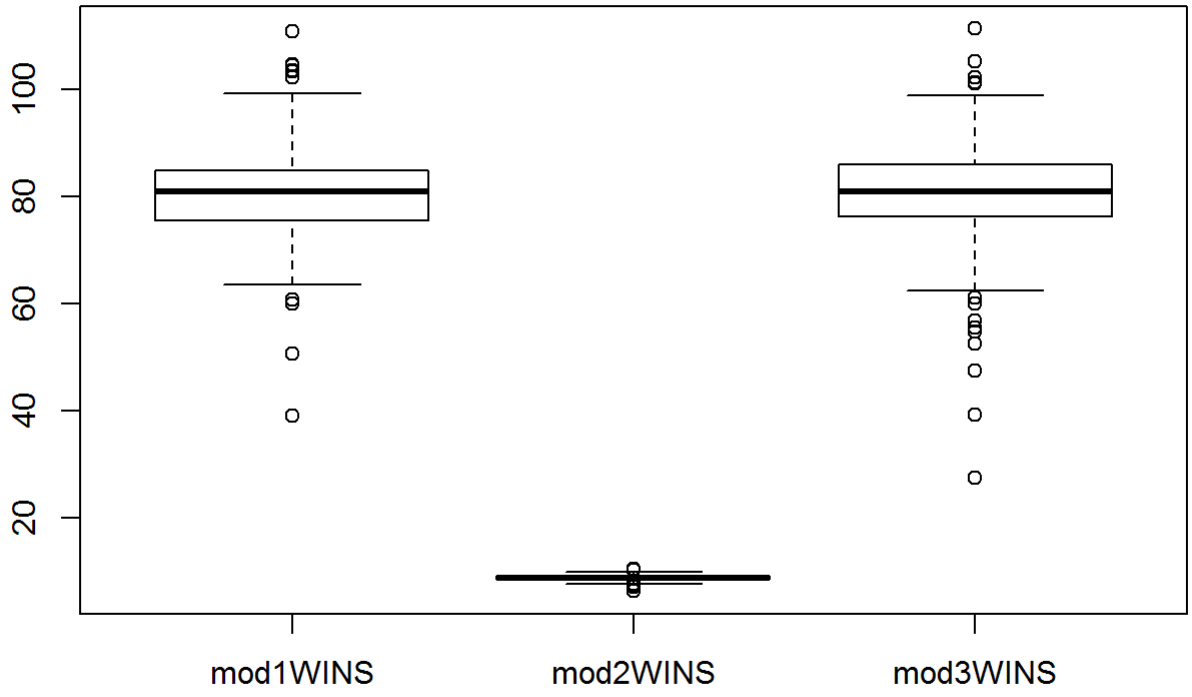

Residual
80
60
40
20
0
-20
-40
-60
40
60
80
100
120
Fitted Value


Residual
1
0
-1
-2
8
9
10
11
Fitted Value



Finally, we also predicted wins for the performances in our evaluation dataset. We prepared the evaluation dataset in a similar way to how we prepared the training dataset, and then we predicted wins using all 3 of our models and saved those predictions in new columns. A preview of this table is shown below. Model 2 predicted substantially fewer wins than model 1 and model 3 did, as demonstrated through a boxplot. This is another reason why we chose model 3 over model 2.

```
##   INDEX TEAM_BATTING_H TEAM_BATTING_2B TEAM_BATTING_3B TEAM_BATTING_HR
## 1     9           1209             170              33              83
## 2    10           1221             151              29              88
## 3    14           1395             183              29              93
## 4    47           1539             309              29             159
## 5    60           1445             203              68               5
## 6    63           1431             236              53              10
##   TEAM_BATTING_BB TEAM_BATTING_SO TEAM_BASERUN_SB TEAM_PITCHING_HR
## 1             447            1080         62.0000               83
## 2             516             929         54.0000               88
## 3             509             816         59.0000               93
## 4             486             914        148.0000              159
## 5              95             416        123.7033               14
## 6             215             377        123.7033               20
##   TEAM_PITCHING_BB TEAM_FIELDING_E mod1WINS mod2WINS mod3WINS
## 1              447             140 66.23689 8.131795 67.55702
## 2              516             135 68.20723 8.242618 68.60779
## 3              509             156 74.90709 8.578680 76.00931
## 4              486             124 84.92170 9.179514 86.53497
## 5              257             616 65.33507 7.970363 67.23924
## 6              420             572 66.84056 8.035833 66.59160
```



The predicted wins for our evaluation dataset are attached in a .csv file.

Appendix

```
library(psych)
library(corrr)
library(tidyr)
library(dplyr)
library(igraph)
library(ggraph)
library(readxl)
library(caTools)
library(Metrics)
library(MASS)

td <- read.csv('moneyball-training-data.csv')
ed <- read.csv('moneyball-evaluation-data.csv')
#1. Data Exploration

td1 <- td[,2:17] #removes index variable from training dataset

#Summary statistics for variables
describe(td1)

#Boxplot of TARGET_WINS by each variable in order to see outliers
boxplot(td1)
#Correlation matrix for variables
correlation <- correlate(td1)
correlation

#Correlation network for variables
tidy_cors <- td1 %>%
  correlate() %>%
  stretch()
tidy_cors

graph_cors <- tidy_cors %>%
  filter(abs(r) > .3) %>%
  graph_from_data_frame(directed = FALSE)

ggraph(graph_cors) +
  geom_edge_link(aes(edge_alpha = abs(r), edge_width = abs(r), color = r)) +
  guides(edge_alpha = "none", edge_width = "none") +
  scale_edge_colour_gradientn(limits = c(-1, 1), colors = c("firebrick2", "dodgerblue2")) +
  geom_node_point(color = "grey", size = 2) +
  geom_node_text(aes(label = name), repel = FALSE) +
  theme_graph()
#2. Data Preparation

td2 <- subset(td1, select=-c(TEAM_PITCHING_H,TEAM_PITCHING_SO, TEAM_BATTING_HBP, TEAM_BASERUN_C
S, TEAM_FIELDING_DP))
meanBattingSO <- mean(td2$TEAM_BATTING_SO, na.rm = TRUE)
td2$TEAM_BATTING_SO[which(is.na(td2$TEAM_BATTING_SO))] <- meanBattingSO
meanBaserunSB <- mean(td2$TEAM_BASERUN_SB, na.rm = TRUE)
td2$TEAM_BASERUN_SB[which(is.na(td2$TEAM_BASERUN_SB))] <- meanBaserunSB
#describe(td2)
#3. Build Models
```

```
#Model 1

mod1a <- lm(TARGET_WINS ~ ., data=td2)
summary(mod1a)

correlation2 <- correlate(td2)
correlation2
mod1b <- lm(TARGET_WINS ~ . -TEAM_BATTING_HR-TEAM_BATTING_SO-TEAM_FIELDING_E, data=td2)
summary(mod1b)

mod1c <- lm(TARGET_WINS ~ . -TEAM_BATTING_HR-TEAM_BATTING_SO-TEAM_FIELDING_E-TEAM_BATTING_2B, da
ta=td2)
summary(mod1c)

#plot(mod1c)
#Model 2

mod2a <- lm(sqrt(TARGET_WINS) ~ . -TEAM_BATTING_HR-TEAM_BATTING_SO-TEAM_FIELDING_E-TEAM_BATTING_
2B, data=td2)
summary(mod2a)

#identifying and removing influential points
sample_size = nrow(td2)
cooksd <- cooks.distance(mod2a)
influential <- as.numeric(names(cooksd)[(cooksd > (4/sample_size))])

#new model after removing influential points
td3 <- td2[-influential,]
mod2b <- lm(sqrt(TARGET_WINS) ~ . -TEAM_BATTING_HR-TEAM_BATTING_SO-TEAM_FIELDING_E-TEAM_BATTING_
2B, data=td3)
summary(mod2b)

#plot(mod2a)
#Model 3

mod3a1 <- lm(TARGET_WINS ~ ., data=td2)
mod3a2 <- lm(TARGET_WINS ~ 1, data=td2)
mod3a <- stepAIC(mod3a2, direction="forward", scope = list(upper=mod3a1, lower=mod3a2))
summary(mod3a)
#4. Select Models

#Summary of 3 models
summary(mod1c)
summary(mod2b)
summary(mod3a)
#Calculate MSE

trainMod1WINS <- predict(mod1c, td2[,-td2$TARGET_WINS])
trainMod2WINS <- predict(mod2b, td2[,-td2$TARGET_WINS])
trainMod3WINS <- predict(mod3a, td2[,-td2$TARGET_WINS])
mse1 <- mse(td2$TARGET_WINS, trainMod1WINS)
mse2 <- mse(td2$TARGET_WINS, trainMod2WINS)
mse3 <- mse(td2$TARGET_WINS, trainMod3WINS)
```

```
#Create a table comparing different metrics for the models
model <- c("Model 1", "Model 2", "Model 3")
adjR2Values <- c(0.2458,0.2995, 0.286)
mseValues <- c(mse1,mse2, mse3)
fStatValues <- c(124.6, 153.7, 114.9)

summ <- cbind.data.frame(model, adjR2Values, mseValues, fStatValues)
summ

#Residual plots of 3 models to assess normality
plot(fitted(mod1c), residuals(mod1c), xlab = "Fitted Value", ylab = "Residual")
plot(fitted(mod2b), residuals(mod2b), xlab = "Fitted Value", ylab = "Residual")
plot(fitted(mod3a), residuals(mod3a), xlab = "Fitted Value", ylab = "Residual")
#preparing the evaluation dataset, imputating missing values similar to what we did with the tra
ining dataset
#ed1 <- ed[,2:16] #removes index variable
ed2 <- subset(ed, select=-c(TEAM_PITCHING_H,TEAM_PITCHING_SO, TEAM_BATTING_HBP, TEAM_BASERUN_CS,
TEAM_FIELDING_DP))
meanBattingSOed <- mean(ed2$TEAM_BATTING_SO, na.rm = TRUE)
ed2$TEAM_BATTING_SO[which(is.na(ed2$TEAM_BATTING_SO))] <- meanBattingSOed
meanBaserunSBed <- mean(ed2$TEAM_BASERUN_SB, na.rm = TRUE)
ed2$TEAM_BASERUN_SB[which(is.na(ed2$TEAM_BASERUN_SB))] <- meanBaserunSBed
#describe(ed2)

#predicted TARGET_WINS values for our models
ed2$mod1WINS <- predict(mod1c, ed2[2:11])
ed2$mod2WINS <- predict(mod2b, ed2[2:11])
ed2$mod3WINS <- predict(mod3a, ed2[2:11])

head(ed2)
boxplot(ed2[,12:14])
#output evaluation dataset with model win predictions
write.csv(ed2, 'moneyball-evaluation-data-model-predictions.csv')
```