DATA 621 HW 1

Group 2 - Tilon Bobb, Jian Quan Chen, Shamecca Marshall

2024-02-22

Introduction

In this assignment, we are given a baseball training and evaluation dataset, which contains approximately 2200 records. The data spans from 1871 to 2006, with each row representing a baseball team's performance from that year. The statistics were all adjusted to reflect a 162 game season. Our objective is to construct a multiple linear regression model of the training data to predict the number of wins for a team.

1. Data Exploration

Glimpse of the data

There are 2,276 rows and 17 columns in the dataset. The response variable is `TARGET_WINS` and the remaining 15 variables, with the exception of the `INDEX` column, are predictor variables.

```
## Rows: 2,276
## Columns: 17
## $ INDEX            <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 11, 12, 13, 15, 16, 17, 18, 1~
## $ TARGET_WINS      <dbl> 39, 70, 86, 70, 82, 75, 80, 85, 86, 76, 78, 68, 72, 7~
## $ TEAM_BATTING_H   <dbl> 1445, 1339, 1377, 1387, 1297, 1279, 1244, 1273, 1391,~
## $ TEAM_BATTING_2B  <dbl> 194, 219, 232, 209, 186, 200, 179, 171, 197, 213, 179~
## $ TEAM_BATTING_3B  <dbl> 39, 22, 35, 38, 27, 36, 54, 37, 40, 18, 27, 31, 41, 2~
## $ TEAM_BATTING_HR  <dbl> 13, 190, 137, 96, 102, 92, 122, 115, 114, 96, 82, 95,~
## $ TEAM_BATTING_BB  <dbl> 143, 685, 602, 451, 472, 443, 525, 456, 447, 441, 374~
## $ TEAM_BATTING_SO  <dbl> 842, 1075, 917, 922, 920, 973, 1062, 1027, 922, 827, ~
## $ TEAM_BASERUN_SB  <dbl> NA, 37, 46, 43, 49, 107, 80, 40, 69, 72, 60, 119, 221~
## $ TEAM_BASERUN_CS  <dbl> NA, 28, 27, 30, 39, 59, 54, 36, 27, 34, 39, 79, 109, ~
## $ TEAM_BATTING_HBP <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N~
## $ TEAM_PITCHING_H  <dbl> 9364, 1347, 1377, 1396, 1297, 1279, 1244, 1281, 1391,~
## $ TEAM_PITCHING_HR <dbl> 84, 191, 137, 97, 102, 92, 122, 116, 114, 96, 86, 95,~
## $ TEAM_PITCHING_BB <dbl> 927, 689, 602, 454, 472, 443, 525, 459, 447, 441, 391~
## $ TEAM_PITCHING_SO <dbl> 5456, 1082, 917, 928, 920, 973, 1062, 1033, 922, 827,~
## $ TEAM_FIELDING_E  <dbl> 1011, 193, 175, 164, 138, 123, 136, 112, 127, 131, 11~
## $ TEAM_FIELDING_DP <dbl> NA, 155, 153, 156, 168, 149, 186, 136, 169, 159, 141,~
```

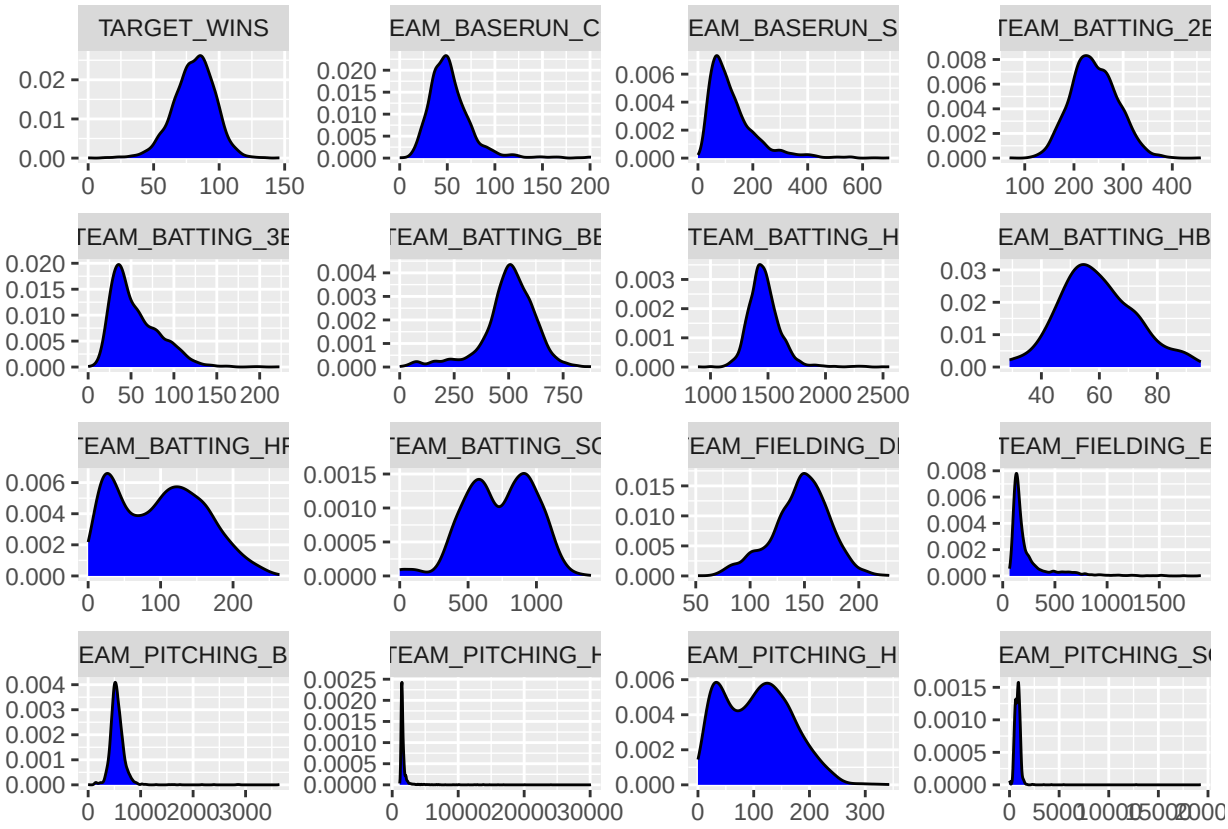
Summary table

We can see from the summary, that the mean of `TARGET_WINS` is 80.79, which is about half the games in a baseball season. For the most part, most variables have 2276 values but there are some, `TEAM_BATTING_HBP` in particular, have less, suggesting that there is missing data.

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

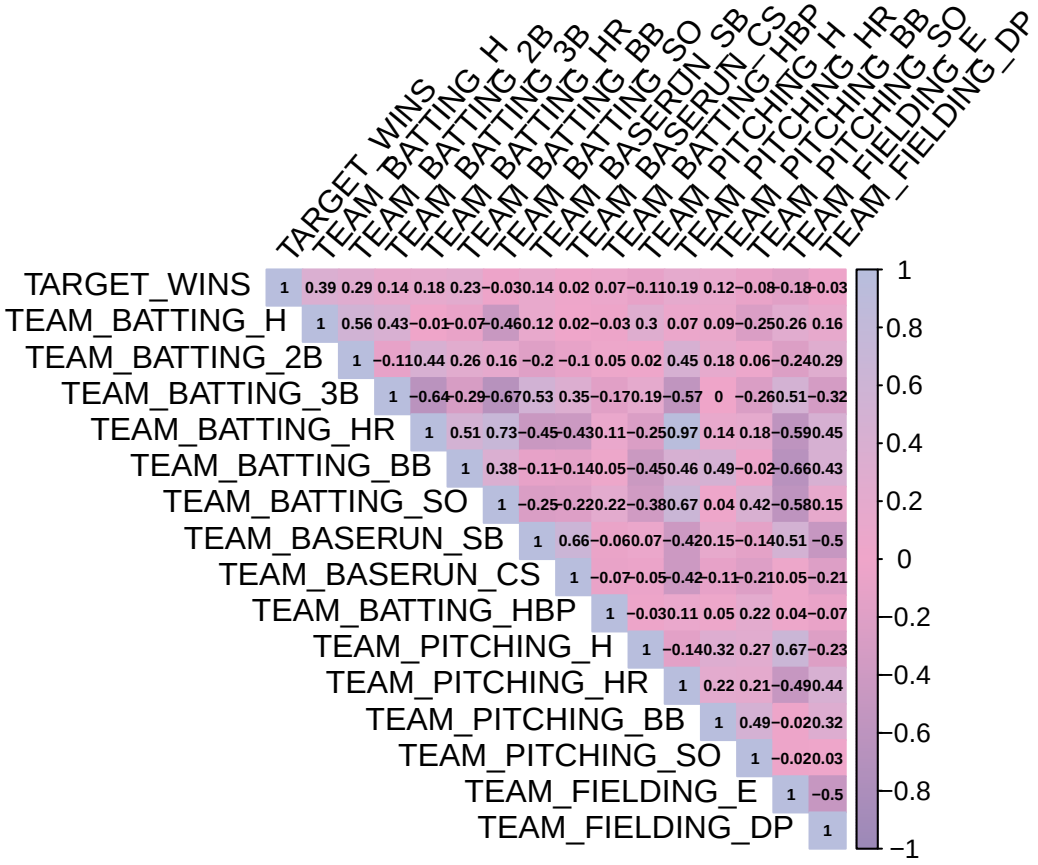
Distribution of variables

The `TARGET_WINS`, `TEAM_BATTING_2B`, `TEAM_BATTING_HBP`, and `TEAM_FIELDING_DP` variables show a normal distribution. The `TEAM_BATTING_HR`, `TEAM_BATTING_SO`, and `TEAM_PITCHING_HR` variables show a bimodal distribution.



Correlation of variables

From this visual, wins seem to be most linearly correlated with `TEAM_BATTING_H` (0.39), `TEAM_BATTING_2B` (0.29), `TEAM_BATTING_BB` (0.23), `TEAM_PITCHING_HR` (0.19), and `TEAM_BATTING_HR` (0.18).



2. Data Preparation

Checking for missing values within the dataset by creating flags for every column

```
## [1] "TARGET_WINS missing values? FALSE"
## [1] "TEAM_BATTING_H missing values? FALSE"
## [1] "TEAM_BATTING_2B missing values? FALSE"
## [1] "TEAM_BATTING_3B missing values? FALSE"
## [1] "TEAM_BATTING_HR missing values? FALSE"
## [1] "TEAM_BATTING_BB missing values? FALSE"
## [1] "TEAM_BATTING_SO missing values? TRUE"
## [1] "TEAM_BASERUN_SB missing values? TRUE"
## [1] "TEAM_BASERUN_CS missing values? TRUE"
## [1] "TEAM_BATTING_HBP missing values? TRUE"
## [1] "TEAM_PITCHING_H missing values? FALSE"
## [1] "TEAM_PITCHING_HR missing values? FALSE"
## [1] "TEAM_PITCHING_BB missing values? FALSE"
## [1] "TEAM_PITCHING_SO missing values? TRUE"
## [1] "TEAM_FIELDING_E missing values? FALSE"
## [1] "TEAM_FIELDING_DP missing values? TRUE"
```

```
##      TARGET_WINS   TEAM_BATTING_H  TEAM_BATTING_2B  TEAM_BATTING_3B
##                0                0                0                0
##  TEAM_BATTING_HR  TEAM_BATTING_BB  TEAM_BATTING_SO  TEAM_BASERUN_SB
```

```
##                0                0              102              131
##  TEAM_BASERUN_CS TEAM_BATTING_HBP  TEAM_PITCHING_H TEAM_PITCHING_HR
##              772             2085                0                0
## TEAM_PITCHING_BB TEAM_PITCHING_SO  TEAM_FIELDING_E TEAM_FIELDING_DP
##                0              102                0              286
```

Since the TEAM_BATTING_HBP variable was missing 2000 values, we will just remove this column from the dataset. The other columns that had missing values (TEAM_BATTING_SO, TEAM_BASERUN_SB, TEAM_BASERUN_CS, TEAM_PITCHING_SO, and TEAM_FIELDING_DP) will be replaced with the median value of that variable.

```
##   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.: 556.8   1st Qu.: 67.0
##  Median :102.00   Median :512.0   Median : 750.0   Median :101.0
##  Mean   : 99.61   Mean   :501.6   Mean   : 736.3   Mean   :123.4
##  3rd Qu.:147.00   3rd Qu.:580.0   3rd Qu.: 925.0   3rd Qu.:151.0
##  Max.   :264.00   Max.   :878.0   Max.   :1399.0   Max.   :697.0
##  TEAM_BASERUN_CS  TEAM_PITCHING_H TEAM_PITCHING_HR TEAM_PITCHING_BB
##  Min.   :  0.00   Min.   : 1137   Min.   :  0.0    Min.   :   0.0
##  1st Qu.: 44.00   1st Qu.: 1419   1st Qu.: 50.0    1st Qu.: 476.0
##  Median : 49.00   Median : 1518   Median :107.0    Median : 536.5
##  Mean   : 51.51   Mean   : 1779   Mean   :105.7    Mean   : 553.0
##  3rd Qu.: 54.25   3rd Qu.: 1682   3rd Qu.:150.0    3rd Qu.: 611.0
##  Max.   :201.00   Max.   :30132   Max.   :343.0    Max.   :3645.0
##  TEAM_PITCHING_SO  TEAM_FIELDING_E  TEAM_FIELDING_DP
##  Min.   :    0.0   Min.   :  65.0   Min.   : 52.0
##  1st Qu.:  626.0   1st Qu.: 127.0   1st Qu.:134.0
##  Median :  813.5   Median : 159.0   Median :149.0
##  Mean   :  817.5   Mean   : 246.5   Mean   :146.7
##  3rd Qu.:  957.0   3rd Qu.: 249.2   3rd Qu.:161.2
##  Max.   :19278.0   Max.   :1898.0   Max.   :228.0
```

Checking for any missing values within the dataset.

```
## [1] "Count of total missing values  "
```

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

Now there is no more missing values within the dataset.

3. Build Models

Model 1

For this model, we are going to use the three variables that were most linearly correlated to target wins: TEAM_BATTING_H (0.39), TEAM_BATTING_2B (0.29), TEAM_BATTING_BB (0.23).

```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_2B +
##     TEAM_BATTING_BB, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -59.753  -8.719   0.529   9.176  48.847
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -1.753191   3.384614  -0.518    0.605
## TEAM_BATTING_H   0.045218   0.002538  17.819   <2e-16 ***
## TEAM_BATTING_2B -0.004206   0.008088  -0.520    0.603
## TEAM_BATTING_BB  0.034136   0.002557  13.348   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.92 on 2272 degrees of freedom
## Multiple R-squared:  0.2196, Adjusted R-squared:  0.2185
## F-statistic: 213.1 on 3 and 2272 DF,  p-value: < 2.2e-16
```

The coefficient estimate for `TEAM_BATTING_H` is 0.045218, which means that each hit by a batter will increase the `TARGET_WINS` by 0.045. In addition, the coefficient estimate for `TEAM_BATTING_BB` is 0.034136, which indicates that every walk by a batter increases the `TARGET_WINS` by 0.034. Both `TEAM_BATTING_H` and `TEAM_BATTING_BB` have a p-value less than 0.05 so they are statistically significant in predicting the number of wins. On the other hand, `TEAM_BATTING_2B` has a coefficient of -0.04206 with a p-value of 0.603. The negative coefficient suggests that the more doubles a team has, the lower amount of wins. However, since the p-value is so high, we cannot say there is a relationship between the number of doubles and the number of wins.

The residual standard error of 13.92 indicates that the typical difference between the observed and predicted number of wins is 14 wins. This error is relatively large considering the average wins is 80.79. Also, the Multiple R-squared value of 0.2196 indicates that about 22% of the variability in wins can be explained by this multiple linear regression model. Lastly, the p-value of the model is <2.2e-16, signifies that this model is statistically significant in predicting the number of wins.

In summary, while this model is statistically significant in predicting the number of wins, we would not use it as the R-squared is low at 0.22 and the standard error is relatively high. Although it makes sense that the increasing the number of hits (`TEAM_BATTING_H`) and walks(`TEAM_BATTING_BB`) increases the amount of wins, it is odd that increasing the amount of doubles decreases the total wins.

Model 2

This multiple linear regression model omits the intercept within the model because if the intercept in a regression model predicting baseball team wins is negative, it suggests that even when all independent variables are set to zero, the model predicts a negative number of wins. This negative prediction essentially indicates that the team is expected to have more losses than wins, which is not realistic or meaningful in the context of baseball.

```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_BB +
##     TEAM_BATTING_2B + 0, data = df)
```

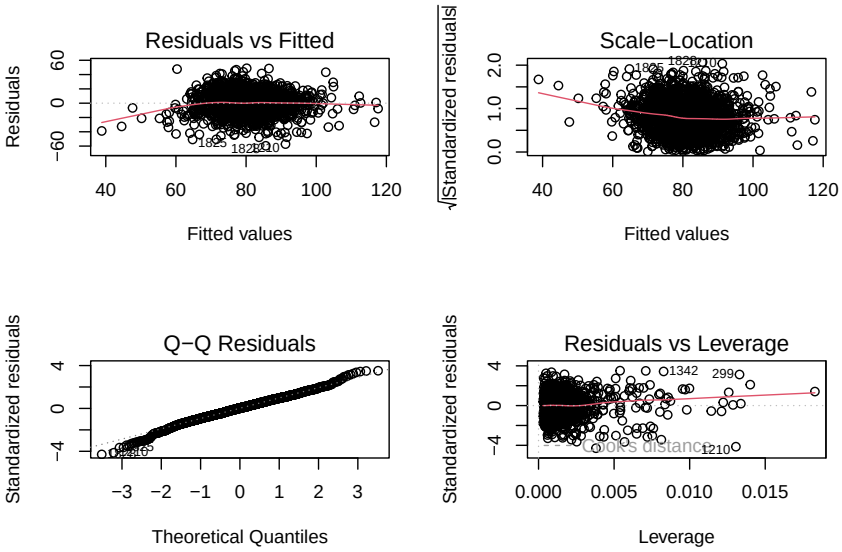
```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -59.621  -8.766   0.515   9.242  48.825
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## TEAM_BATTING_H   0.044092   0.001309  33.680   <2e-16 ***
## TEAM_BATTING_BB  0.033513   0.002257  14.848   <2e-16 ***
## TEAM_BATTING_2B -0.003267   0.007881  -0.415    0.679
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.92 on 2273 degrees of freedom
## Multiple R-squared:  0.9714, Adjusted R-squared:  0.9714
## F-statistic: 2.576e+04 on 3 and 2273 DF,  p-value: < 2.2e-16
```

The coefficients obtained from our multiple linear regression model shed light on the relationship between specific baseball metrics and the number of wins. For instance, the coefficient for "TEAM_BATTING_H" (Base Hits by batters) is approximately 0.044, indicating that for each additional base hit, we expect around 0.044 more wins, holding other variables constant. Similarly, the coefficient for "TEAM_BATTING_BB" (Walks allowed) is approximately 0.034, suggesting that each additional walk allowed by the pitching team is associated with around 0.034 more wins.

However, the coefficient for "TEAM_BATTING_2B" (Doubles by batters) is approximately -0.003, which is statistically insignificant (p-value = 0.679). This suggests that the number of doubles by batters may not have a significant effect on wins. This finding may appear counterintuitive, as one might expect teams with more doubles to win more games.The residual standard error of 13.92 indicates that the typical difference between the observed and predicted number of wins is 14 wins. This error is relatively large considering the average wins is 80.79.

Despite this inconsistency, the overall model demonstrates a strong ability to explain win variance, with an adjusted R-squared value of 0.9714. This indicates that the model accounts for a significant portion of the variability in wins based on the included variables. Therefore, it may be advisable to retain the model for further analysis and refinement.

Residual Analysis - Model 2



Residuals vs Fitted: The residuals are clustered around 60-100, suggesting that the assumption of linearity is not met. Scale-location: The data is not randomly dispersed around the horizontal line so the assumption of homoscedasticity is not met. *Normal Q-Q: For the most part, the plot follows the normal line but there are some deviations at the tail.

Judging from the residual plots, this model might not be the best fit for predicting the response variable.

Model 3

Let's construct a multiple linear regression model with the response variable as TARGET_WINS and additional explanatory variables as TEAM_PITCHING_H, TEAM_PITCHING_HR, and TEAM_PITCHING_BB. This selection is based on their correlation coefficients with TARGET_WINS: TEAM_PITCHING_H (-0.10993705), TEAM_PITCHING_HR (0.18901373), and TEAM_PITCHING_BB (0.124174536). Despite TEAM_PITCHING_H having a negative correlation coefficient, indicating a potentially negative impact on wins, it's essential to consider its significance in the model along with the positive coefficients of TEAM_PITCHING_HR and TEAM_PITCHING_BB. By including these variables, we aim to capture the collective influence of pitching-related statistics on the number of wins in our dataset.

```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_BB +
##     TEAM_BATTING_2B + TEAM_PITCHING_H + TEAM_PITCHING_HR + TEAM_PITCHING_BB +
##     0, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -52.370  -8.915   0.355   9.095  55.659
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## TEAM_BATTING_H    0.0522497  0.0015908  32.844  < 2e-16 ***
## TEAM_BATTING_BB   0.0107415  0.0040568   2.648 0.008158 **
## TEAM_BATTING_2B  -0.0180974  0.0085538  -2.116 0.034476 *
```

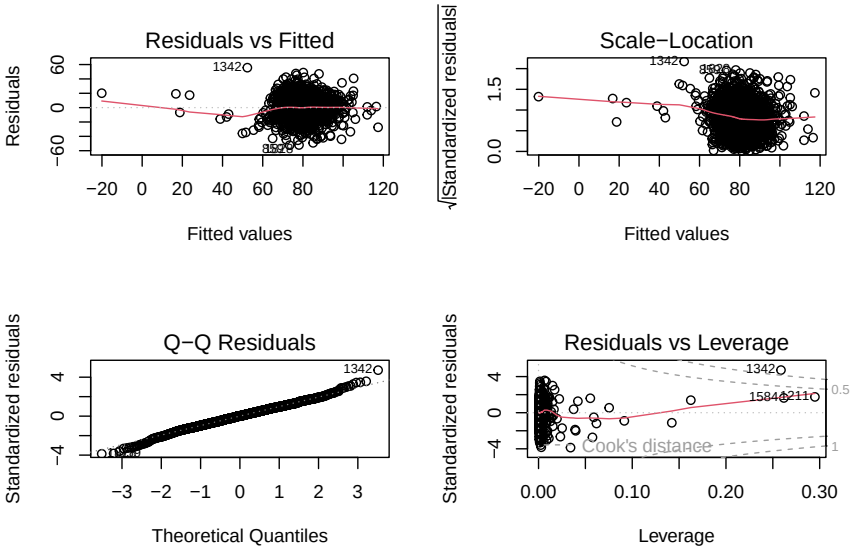
```
## TEAM_PITCHING_H  -0.0026699  0.0003321  -8.040 1.43e-15 ***
## TEAM_PITCHING_HR  0.0212424  0.0058310   3.643 0.000276 ***
## TEAM_PITCHING_BB  0.0099604  0.0027667   3.600 0.000325 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.69 on 2270 degrees of freedom
## Multiple R-squared:  0.9724, Adjusted R-squared:  0.9723
## F-statistic: 1.333e+04 on 6 and 2270 DF,  p-value: < 2.2e-16
```

The coefficients obtained from our multiple linear regression model shed light on the relationship between specific baseball metrics and the number of wins. For instance, the coefficient for "TEAM_BATTING_H" (Base Hits by batters) suggests that each additional base hit is associated with around 0.044 more wins, holding other variables constant. Similarly, the coefficient for "TEAM_BATTING_BB" (Walks allowed) indicates that each additional walk allowed by the pitching team correlates with around 0.034 more wins.

However, the coefficient for "TEAM_BATTING_2B" (Doubles by batters) is statistically insignificant (p-value = 0.679), suggesting that the number of doubles by batters may not significantly impact wins. This finding may seem counterintuitive, as one might expect teams with more doubles to win more games.

Despite this inconsistency, the overall model demonstrates a strong ability to explain win variance, with an adjusted R-squared value of 0.9714. Therefore, it may be advisable to retain the model for further analysis and refinement.

Residual Analysis - Model 3



Residuals vs Fitted: The residuals are clustered around 60-100, suggesting that the assumption of linearity is not met. Scale-location: The previous plot showed a more uniform distribution of residuals across fitted values, while the current plot exhibits a dip in residuals around the range of 60 to 100 fitted values, indicating potential heteroscedasticity or a distinct pattern of variability in that range. *Normal Q-Q: For the most part, the plot follows the normal line but there are some deviations at the tail.* Residuals vs Leverage: We see a concentration of points toward the left end of the x-axis in the plot which suggests the presence of influential data points with high leverage, indicating they have a significant impact on the regression model's coefficients.

Model 4

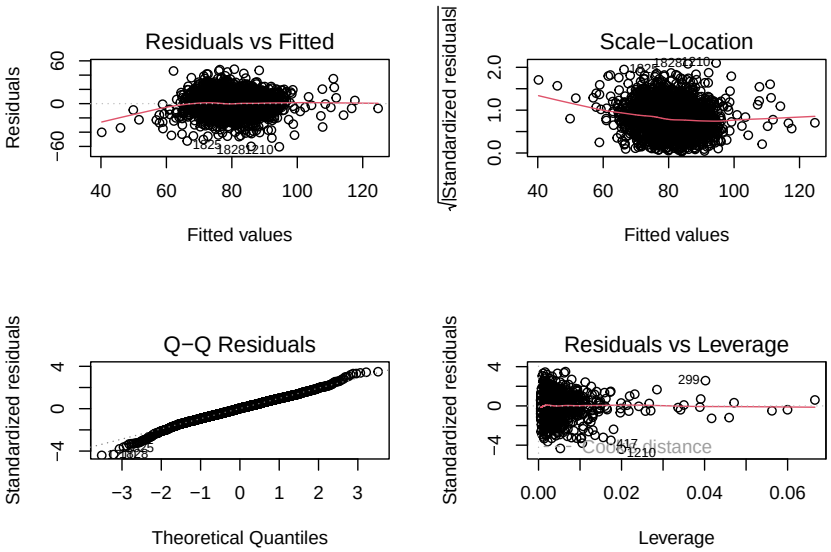
For this model, we are going to create an interaction term by taking the product of TEAM_BATTING_H,TEAM_BATTING_HR, and TEAM_BATTING_BB. These were most linearly correlated with TARGET_WINS and all of these variables have a positive impact on the number of wins. An interaction term will show that the multiplicative effects of these variables on predicting the TARGET_WINS might be better than the sum of them individually. Doubles hit by batters was not included as it had a high p-value in the first model, suggesting it might not be a significant predictor, and it is slightly correlated with hits by batter.

```
##
## Call:
## lm(formula = TARGET_WINS ~ (TEAM_BATTING_H * TEAM_BATTING_BB *
##     TEAM_BATTING_HR) + 0, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -60.510  -8.755   0.406   9.190  48.240
##
## Coefficients:
##                                                  Estimate Std. Error t value
## TEAM_BATTING_H                                  4.516e-02  1.117e-03  40.423
## TEAM_BATTING_BB                                 5.706e-02  1.324e-02   4.310
## TEAM_BATTING_HR                                -5.191e-01  1.956e-01  -2.654
## TEAM_BATTING_H:TEAM_BATTING_BB                 -2.142e-05  9.034e-06  -2.371
## TEAM_BATTING_H:TEAM_BATTING_HR                  3.305e-04  1.332e-04   2.481
## TEAM_BATTING_BB:TEAM_BATTING_HR                 8.261e-04  3.401e-04   2.429
## TEAM_BATTING_H:TEAM_BATTING_BB:TEAM_BATTING_HR -4.995e-07  2.304e-07  -2.169
##                                                Pr(>|t|)
## TEAM_BATTING_H                                  < 2e-16 ***
## TEAM_BATTING_BB                                 1.7e-05 ***
## TEAM_BATTING_HR                                 0.00802 **
## TEAM_BATTING_H:TEAM_BATTING_BB                  0.01782 *
## TEAM_BATTING_H:TEAM_BATTING_HR                  0.01316 *
## TEAM_BATTING_BB:TEAM_BATTING_HR                 0.01523 *
## TEAM_BATTING_H:TEAM_BATTING_BB:TEAM_BATTING_HR  0.03021 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.86 on 2269 degrees of freedom
## Multiple R-squared:  0.9717, Adjusted R-squared:  0.9716
## F-statistic: 1.114e+04 on 7 and 2269 DF,  p-value: < 2.2e-16
```

These results of the coefficient estimates are a bit surprising as I was expecting all of them to be positive since they all have a positive impact on wins. The coefficients for TEAM_BATTING_H and TEAM_BATTING_BB are positive, indicating that an increase in these variables is associated with an increase in TARGET_WINS. However, the main effect of TEAM_BATTING_HR is negative, suggesting that an increase in home runs is associated with a decrease in wins. This is counterintuitive as one would expect the more home runs a team hit, the more wins they have. Also, the coefficient for the interaction term TEAM_BATTING_H:TEAM_BATTING_BB:TEAM_BATTING_HR is -4.995e-07, indicating that for each combined base hit, walk, and home run, th model predicts a decrease in the amount of wins by 4.995e-07. Again, this does not seem to align with the fact that these individually have a positive impact on wins. All of the p-values are low indicating statistical significance. However, coefficients themselves are small so it debatable how relative this model is in predicting wins.

Like the previous models, the R-squared is high at 0.9717 so 97% of the variance in the response model can be attributed to this model. In addition, all of the p-values for these variables are low, indicating that they are statistically significant predictors in this model. But, the magnitude of these coefficients is quite small, which suggests the practical significance of these predictors in this model. Overall, while this model has strong statistical significance, the coefficients do not match with our conception of baseball so it will need further refinement.

Residual Analysis - Model 4



These residual plots are similar to the previous models. The plots suggest some issues with the assumptions of homoscedasticity and linearity. Therefore a linear regression model may not be the best fit for the data.

4. Select Models

To evaluate the performance of the models, let's look at the R-squared, Mean Squared Error, and Root Mean Squared Error.

```
##
##  Model 1
## Adjusted R-squared:  0.218542337673064
## MSE:  193.562551032368
## RMSE:  13.9126759120008
##
##  Model 2
## Adjusted R-squared:  0.97138957456795
## MSE:  193.58540982541
## RMSE:  13.9134973973264
##
##  Model 3
## Adjusted R-squared:  0.972330323973487
## MSE:  186.972959428232
```

```
## RMSE:  13.6738055942094
##
##  Model 4
## Adjusted R-squared:  0.971633024820375
## MSE:  191.600392909153
## RMSE:  13.8419793710709
```

Based on all of our multiple linear regression models, model 3 seems to have performed best. In comparison to the other models, model 3 has the highest R-squared value at 0.9724, the lowest MSE at 186.97, and the lowest RMSE at 13.67. In addition, the p-value for the coefficient estimates and for the f-statistic were all statistically significant. Also, we did take into account whether the models made practical sense. We concluded that all the models seem to have one or more variables that seem counterinituitive and they all have residual plots that suggest a linear model might not be the best fit for the data. Thus, we decided to go with the model that the highest R-squared and lowest RMSE, which was model 3.

Predicting the evaluation data set We will use our trained model to predict the number of wins in the evaluation dataset. Here is a summary of the evaluation dataset:

```
##  TEAM_BATTING_H TEAM_BATTING_2B TEAM_BATTING_3B  TEAM_BATTING_HR
##  Min.   : 819   Min.   : 44.0   Min.   : 14.00   Min.   :  0.00
##  1st Qu.:1387   1st Qu.:210.0   1st Qu.: 35.00   1st Qu.: 44.50
##  Median :1455   Median :239.0   Median : 52.00   Median :101.00
##  Mean   :1469   Mean   :241.3   Mean   : 55.91   Mean   : 95.63
##  3rd Qu.:1548   3rd Qu.:278.5   3rd Qu.: 72.00   3rd Qu.:135.50
##  Max.   :2170   Max.   :376.0   Max.   :155.00   Max.   :242.00
##  TEAM_BATTING_BB TEAM_BATTING_SO  TEAM_BASERUN_SB TEAM_BASERUN_CS
##  Min.   : 15.0   Min.   :   0.0   Min.   :  0.0   Min.   :  0.00
##  1st Qu.:436.5   1st Qu.: 565.0   1st Qu.: 60.5   1st Qu.: 44.00
##  Median :509.0   Median : 686.0   Median : 92.0   Median : 49.50
##  Mean   :499.0   Mean   : 707.7   Mean   :122.1   Mean   : 51.37
##  3rd Qu.:565.5   3rd Qu.: 904.5   3rd Qu.:149.0   3rd Qu.: 56.00
##  Max.   :792.0   Max.   :1268.0   Max.   :580.0   Max.   :154.00
##  TEAM_PITCHING_H TEAM_PITCHING_HR TEAM_PITCHING_BB TEAM_PITCHING_SO
##  Min.   : 1155   Min.   :  0.0    Min.   : 136.0   Min.   :   0.0
##  1st Qu.: 1426   1st Qu.: 52.0    1st Qu.: 471.0   1st Qu.: 622.5
##  Median : 1515   Median :104.0    Median : 526.0   Median : 745.0
##  Mean   : 1813   Mean   :102.1    Mean   : 552.4   Mean   : 795.9
##  3rd Qu.: 1681   3rd Qu.:142.5    3rd Qu.: 606.5   3rd Qu.: 927.5
##  Max.   :22768   Max.   :336.0    Max.   :2008.0   Max.   :9963.0
##  TEAM_FIELDING_E  TEAM_FIELDING_DP
##  Min.   :  73.0   Min.   : 69.0
##  1st Qu.: 131.0   1st Qu.:134.5
##  Median : 163.0   Median :148.0
##  Mean   : 249.7   Mean   :146.3
##  3rd Qu.: 252.0   3rd Qu.:160.5
##  Max.   :1568.0   Max.   :204.0
```

```
## [1] "Model 3: Evaluation data"
```

```
## [1] "Mean Squared Error:  330.38129718844"
```

```
## [1] "Root Mean Squared Error:  18.1763939544795"
```

When model 3 was applied to the evaluation data, the RSME increased from 13.67 to 18.17, which suggests that the model's predictions are not as accurate on the evaluation data as they were on the training data. The high R-squared value of the model on the training data is an indication that there is possible overfitting. This means that the model is good at predicting data from the training dataset but not unseen data in the evaluation set. In conclusion, more work needs to be done so that the model performs well on both seen and unseen data.

Appendix: Code for this assignment

```
knitr::opts_chunk$set(echo=FALSE,warning = FALSE, message = FALSE)
library(tidyverse)
library(psych)
library(corrplot)
df <- read_csv("https://raw.githubusercontent.com/LeJQC/DATA-621-Group-2/main/HW1/moneyball-training-da
glimpse(df)
# Setting index column to index
rownames(df) <- df$INDEX
df$INDEX <- NULL

# Print summary table
summary_table <- describe(df)

print(round(summary_table,2))
df_long <- df %>%
  pivot_longer(
    cols = everything(),
    names_to = "variable",
    values_to = "value"
  )

df_long %>%
  ggplot(aes(value)) +
  geom_density(fill = "blue") +
  facet_wrap(~variable, scales ="free", ncol = 4) +
  labs(x = element_blank(), y = element_blank())
df %>%
  cor(use = "pairwise.complete.obs") %>%
  corrplot(method = "color", type = "upper", tl.col = "black", diag = TRUE, number.cex = 0.5, addCoef.c
# Loop through columns
for (col_name in names(df)) {
  missing <- is.na(df[[col_name]])
  output <- paste(col_name,"missing values?",any(missing))
  print(output)
}
# Checking for any missing values
sapply(df, function(x) sum(is.na(x)))
df <- df %>% select(-TEAM_BATTING_HBP)

na_variables <- c("TEAM_BATTING_SO", "TEAM_BASERUN_SB", "TEAM_BASERUN_CS", "TEAM_PITCHING_SO", "TEAM_FI

for (col in na_variables) {
```

```r
  median_value <- median(df[[col]], na.rm = TRUE)
  df[[col]][is.na(df[[col]])] <- median_value
}

summary(df)
#which(is.na(df))

# Count total missing values
print("Count of total missing values  ")
sum(is.na(df))
model1 <- lm(TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_2B + TEAM_BATTING_BB, data = df)

summary(model1)
##Fit the multiple linear regression model
model2 <- lm(TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_BB + TEAM_BATTING_2B+0, data = df)

summary(model2)
layout(matrix(c(1,2,3,4),2,2))
plot(model2)
model3 <- lm(TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_BB + TEAM_BATTING_2B +  TEAM_PITCHING_H + TEAM_

summary(model3)
layout(matrix(c(1,2,3,4),2,2))
plot(model3)
model4 <- lm(TARGET_WINS ~ (TEAM_BATTING_H * TEAM_BATTING_BB* TEAM_BATTING_HR) + 0, data = df)

summary(model4)
layout(matrix(c(1,2,3,4),2,2))
plot(model4)
models <- list(model1, model2, model3,model4)
names(models) <- c("Model 1", "Model 2", "Model 3", "Model 4")

# Function to calculate RMSE
rmse <- function(actual, predicted) {
  sqrt(mean((actual - predicted)^2))
}

# Evaluate the models
for (name in names(models)) {
  model <- models[[name]]
  predictions <- predict(model, newdata = df)
  actual_values <- df$TARGET_WINS

  cat(paste("\n", name, "\n"))
  cat(paste("Adjusted R-squared: ", summary(model)$adj.r.squared, "\n"))
  cat(paste("MSE: ", mean((predictions - actual_values)^2), "\n"))
  cat(paste("RMSE: ", rmse(actual_values, predictions), "\n"))
}
train <- read_csv("https://raw.githubusercontent.com/LeJQC/DATA-621-Group-2/main/HW1/moneyball-evaluati

rownames(train) <- train$INDEX

train$INDEX <- NULL
```

```r
# Deleting HBP column as it has too much missing data
train <- train %>% select(-TEAM_BATTING_HBP)

na_variables <- c("TEAM_BATTING_SO", "TEAM_BASERUN_SB", "TEAM_BASERUN_CS", "TEAM_PITCHING_SO", "TEAM_FI

# Setting the missing values to the median
for (col in na_variables) {
  median_value <- median(train[[col]], na.rm = TRUE)
  train[[col]][is.na(train[[col]])] <- median_value
}

summary(train)
# Looking at the predicts based on our model
predictions <- predict(model3, newdata = train)
actual_values <- df$TARGET_WINS


mse <- mean((predictions - actual_values)^2)
rmse <- sqrt(mse)


print("Model 3: Evaluation data")
print(paste("Mean Squared Error: ", mse))
print(paste("Root Mean Squared Error: ", rmse))
```