Name: Sachid Deshmukh

Date: 09-07-2019

Assignment: Data 621 HW-1

Contents

Data Exploration

Let's load the training dataset and take a preview

```
##   INDEX TARGET_WINS TEAM_BATTING_H TEAM_BATTING_2B TEAM_BATTING_3B
## 1     1          39           1445             194              39
## 2     2          70           1339             219              22
## 3     3          86           1377             232              35
## 4     4          70           1387             209              38
## 5     5          82           1297             186              27
## 6     6          75           1279             200              36
##   TEAM_BATTING_HR TEAM_BATTING_BB TEAM_BATTING_SO TEAM_BASERUN_SB
## 1              13             143             842              NA
## 2             190             685            1075              37
## 3             137             602             917              46
## 4              96             451             922              43
## 5             102             472             920              49
## 6              92             443             973             107
##   TEAM_BASERUN_CS TEAM_BATTING_HBP TEAM_PITCHING_H TEAM_PITCHING_HR
## 1              NA               NA            9364               84
## 2              28               NA            1347              191
## 3              27               NA            1377              137
## 4              30               NA            1396               97
## 5              39               NA            1297              102
## 6              59               NA            1279               92
##   TEAM_PITCHING_BB TEAM_PITCHING_SO TEAM_FIELDING_E TEAM_FIELDING_DP
## 1              927             5456            1011               NA
## 2              689             1082             193              155
## 3              602              917             175              153
## 4              454              928             164              156
## 5              472              920             138              168
## 6              443              973             123              149
```

Train Data has 2276 observations and 17 variables.

Following columns have missing values.

TEAM_BATTING_SO : Strikeouts by batters

TEAM_BASERUN_SB : Stolen bases

TEAM_BASERUN_CS : Caught stealing

TEAM_BATTING_HBP : Batters hit by pitch (get a free base)

TEAM_PITCHING_SO : Strikeouts by batters

TEAM_FIELDING_DP : Double Plays

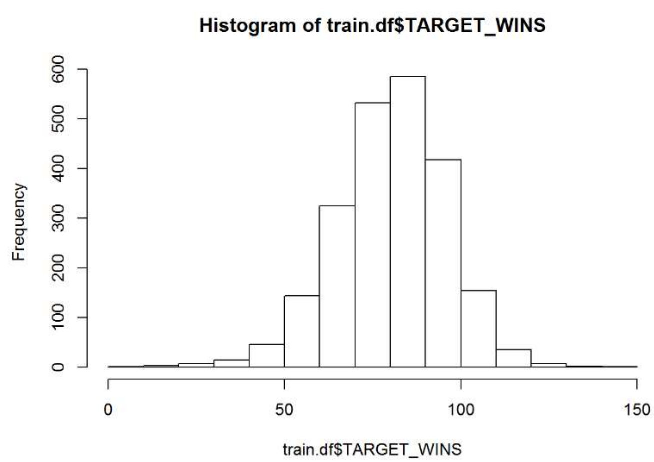
Let's look at the target variable which is TARGET_WINS

1] Summary

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   71.00   82.00   80.79   92.00  146.00
```

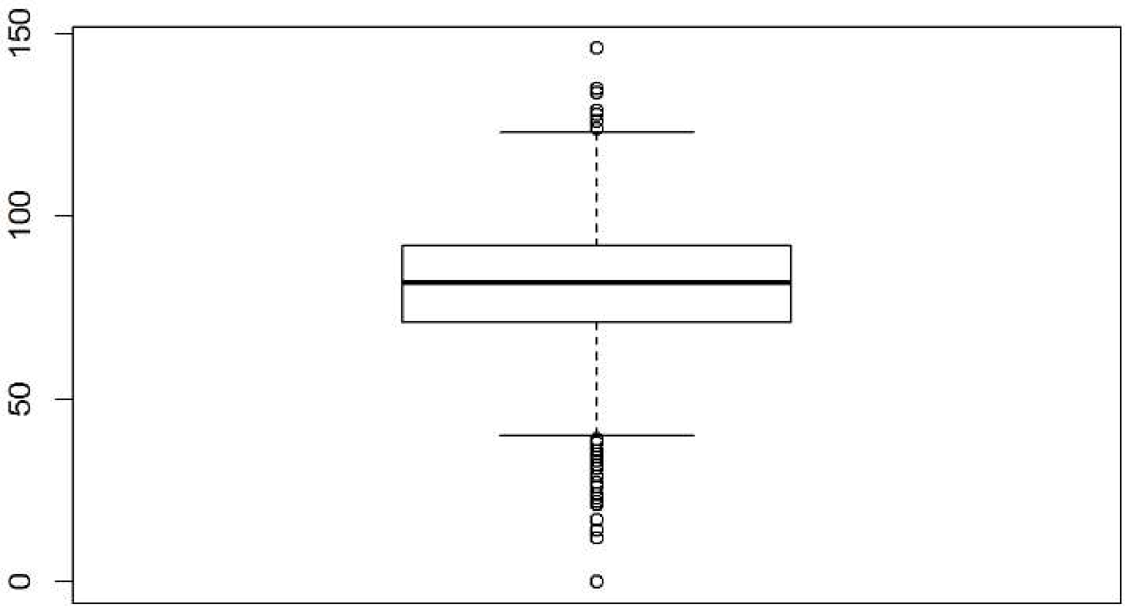
As we can see here Target_Wins has a range from 0 to 146. Median and Mean values are close to each other indicating there is no skew in the data

2] Distribution



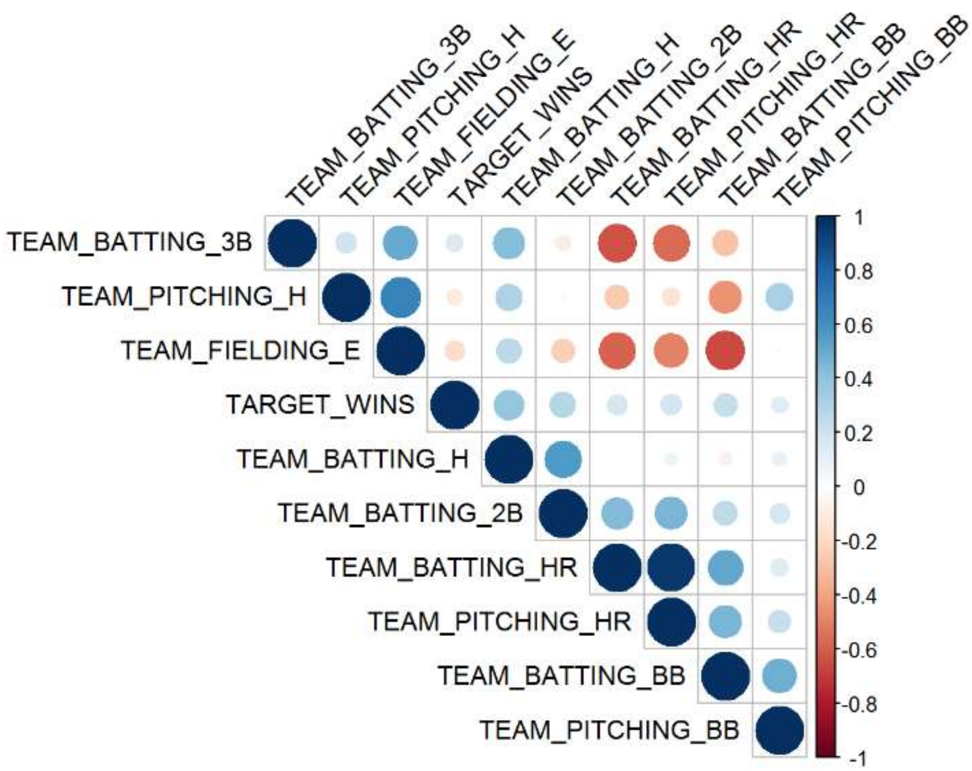
Target variable is normally distributed with mean as 80.79. Histogram above shows that new skew in the data.

3] Box Plot



Box plot for Target_Wins variable is very symmetric. This proves our point of no data skew. There seems to be two outliers in the box plot. We will keep then in data for now

Let's look at correlation plot of train dataframe



Above correlation plot shows that Target_Wins is highly correlated with following variables

Positive correlation

1] Team_Batting_H

2] Team_Batting_2B

3] Team_Batting_BB

4] Team_Batting_HR

5] Team_Pitching_HR

6]Team_Pitching_BB

Negative Correlation

1] Team_Fielding_E

2] Team_Pitching_H

Data Preparation

We will do following data clean up and data transformations task on the raw data

1] Drop columns with null values from training data frame

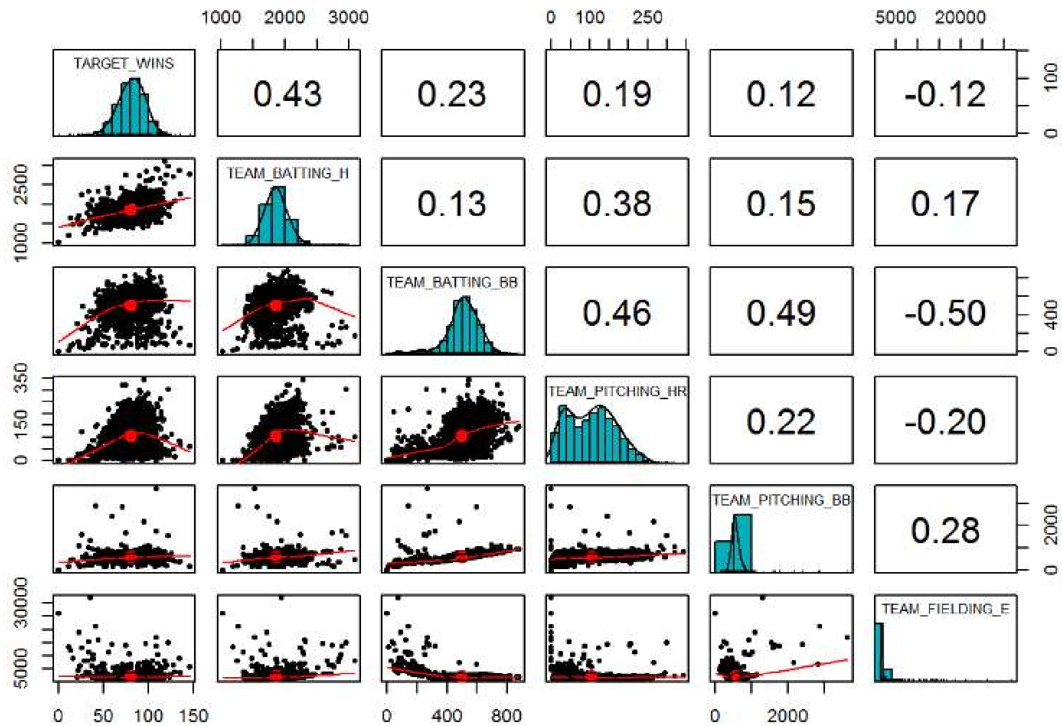
2] Combine Team_Batting_H, Team_Batting_2B and Team_Batting_3B variables into one by taking sum

3] Combine Team_Fielding_E and Team_Pitching_E variables into one by taking sum

Look at summary of resultant data frame

```
##   TARGET_WINS     TEAM_BATTING_H TEAM_BATTING_BB TEAM_PITCHING_HR
##  Min.   :  0.00   Min.   :1026   Min.   :  0.0   Min.   :  0.0
##  1st Qu.: 71.00   1st Qu.:1739   1st Qu.:451.0   1st Qu.: 50.0
##  Median : 82.00   Median :1862   Median :512.0   Median :107.0
##  Mean   : 80.79   Mean   :1865   Mean   :501.6   Mean   :105.7
##  3rd Qu.: 92.00   3rd Qu.:1978   3rd Qu.:580.0   3rd Qu.:150.0
##  Max.   :146.00   Max.   :3092   Max.   :878.0   Max.   :343.0
##  TEAM_PITCHING_BB TEAM_FIELDING_E
##  Min.   :   0.0   Min.   : 1276
##  1st Qu.: 476.0   1st Qu.: 1566
##  Median : 536.5   Median : 1679
##  Mean   : 553.0   Mean   : 2026
##  3rd Qu.: 611.0   3rd Qu.: 1922
##  Max.   :3645.0   Max.   :31860
```

Look at pair plot (fig 1.0) of resultant data frame



From the pair plot (fig 1.0) above we can see that following variables are not normally distributed.

1] Team_Pitching_HR

2] Team_Pitching_BB

3] Team_Fielding_E

Linear regression model works better if the input variables are normally distributed. Let's take log transformation of those variables and plot the pair plot (fig 1.0) after transformation

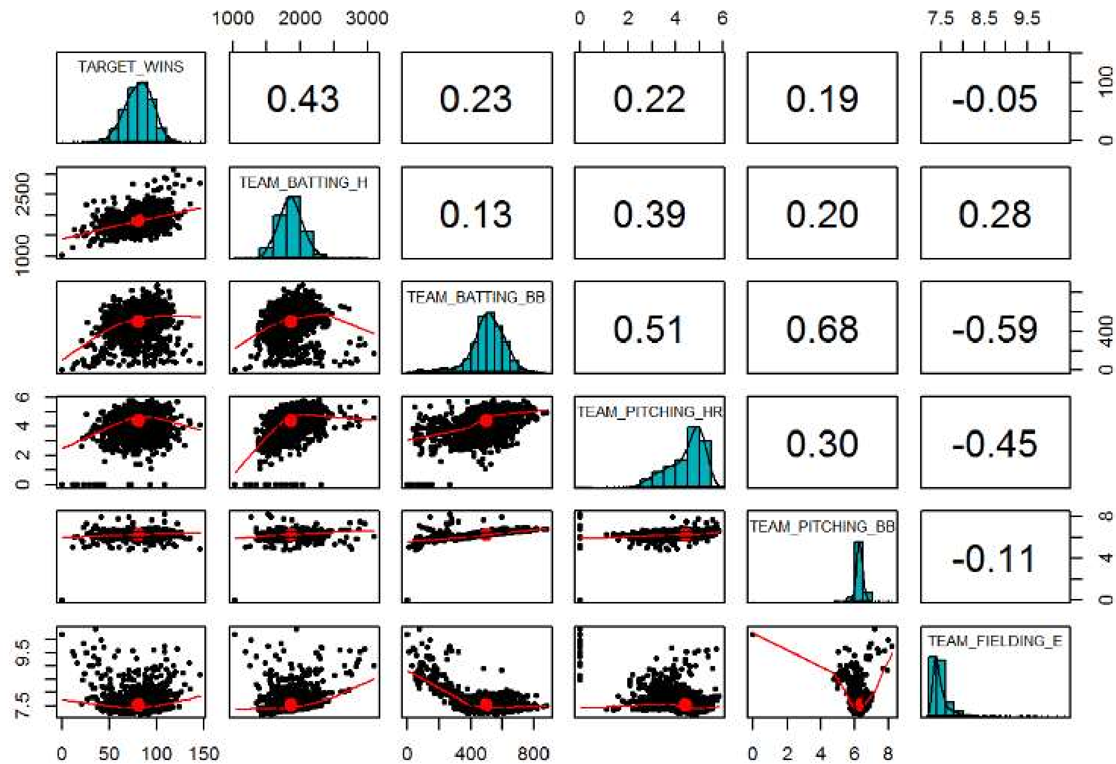


Figure 1.0

Pair plot after variable transformation looks much better. We can see that all the variables have approximately normal distribution

Build Models

1] Model-1 In the first model we will model Target_Wins against following variables

- Team_Batting_H
- Team_Batting_BB
- Team_Pitching_HR
- Team_Pitching_BB
- Team_Fielding_E

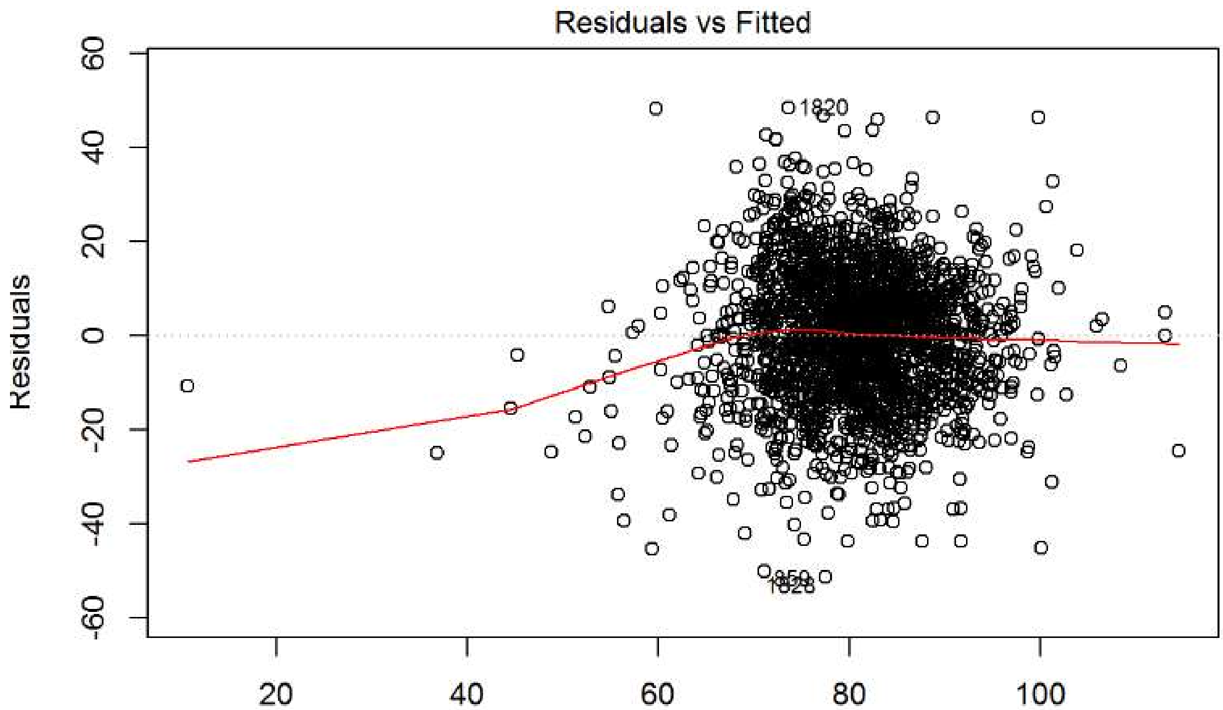
2] Print the model summary

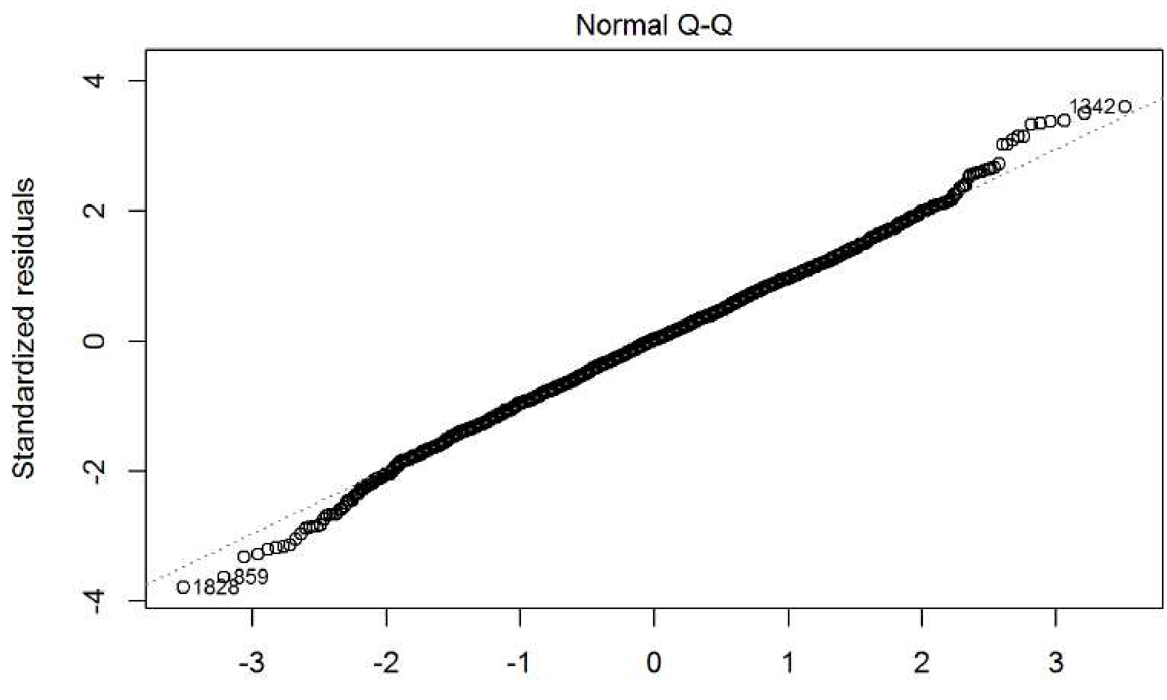
```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_BB +
##     TEAM_PITCHING_HR + TEAM_PITCHING_BB + TEAM_FIELDING_E, data = train.trans)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -51.452  -9.134   0.337   9.240  48.351
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      61.101138  10.855178   5.629 2.04e-08 ***
## TEAM_BATTING_H    0.041235   0.001990  20.720  < 2e-16 ***
## TEAM_BATTING_BB   0.011761   0.004849   2.426   0.0154 *
## TEAM_PITCHING_HR -2.451877   0.488779  -5.016 5.67e-07 ***
## TEAM_PITCHING_BB  2.600056   1.641425   1.584   0.1133
## TEAM_FIELDING_E  -9.122132   1.562982  -5.836 6.10e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.82 on 2270 degrees of freedom
## Multiple R-squared:  0.2316, Adjusted R-squared:  0.2299
## F-statistic: 136.9 on 5 and 2270 DF,  p-value: < 2.2e-16
```

Please note that all the variable coefficients are statistically significant except Team_PitchingBB. F statistic is significant indicating one or more variables are useful in predicting Target_Wins variable.

R square value is 0.23 which indicates the model effectiveness

3] Plot the residuals





From the above residual plot we can see that there is some non-linearity between independent variable and dependent variable. We can certainly enhance this model by adding polinimial tearms

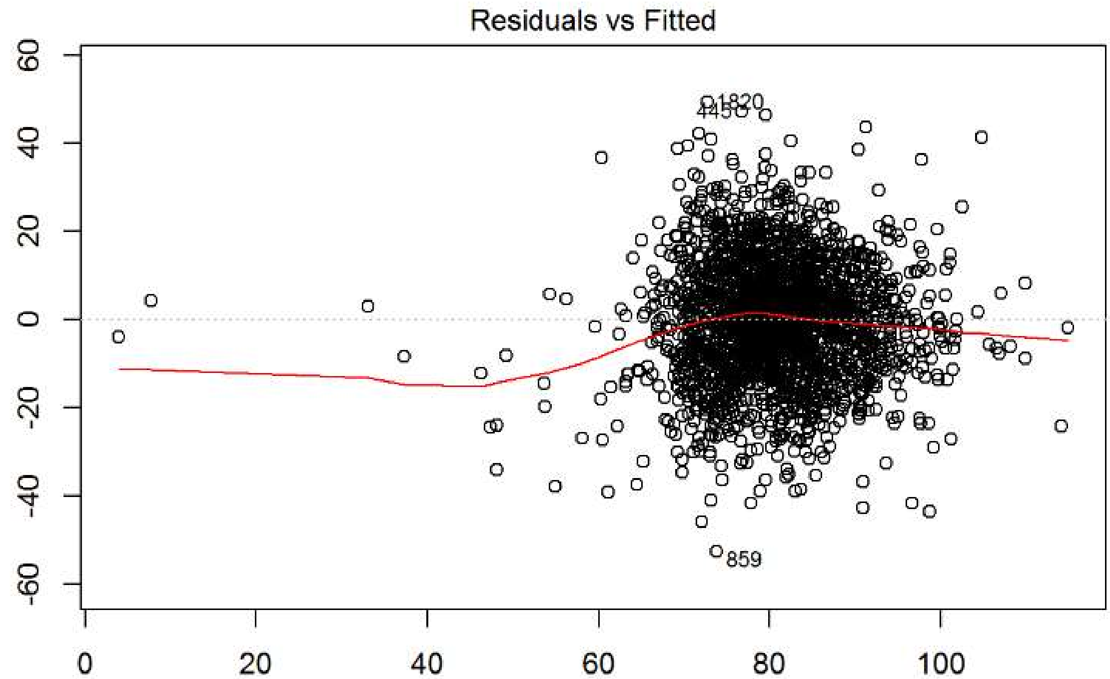
4] Model -2 In the second model from pair plot above (fig 1.0) we can see that there is a non linear relationship between Target_Wins and Team_Batting_BB variable. Let's add polynomial term for Team_Batting_BB variable in the model

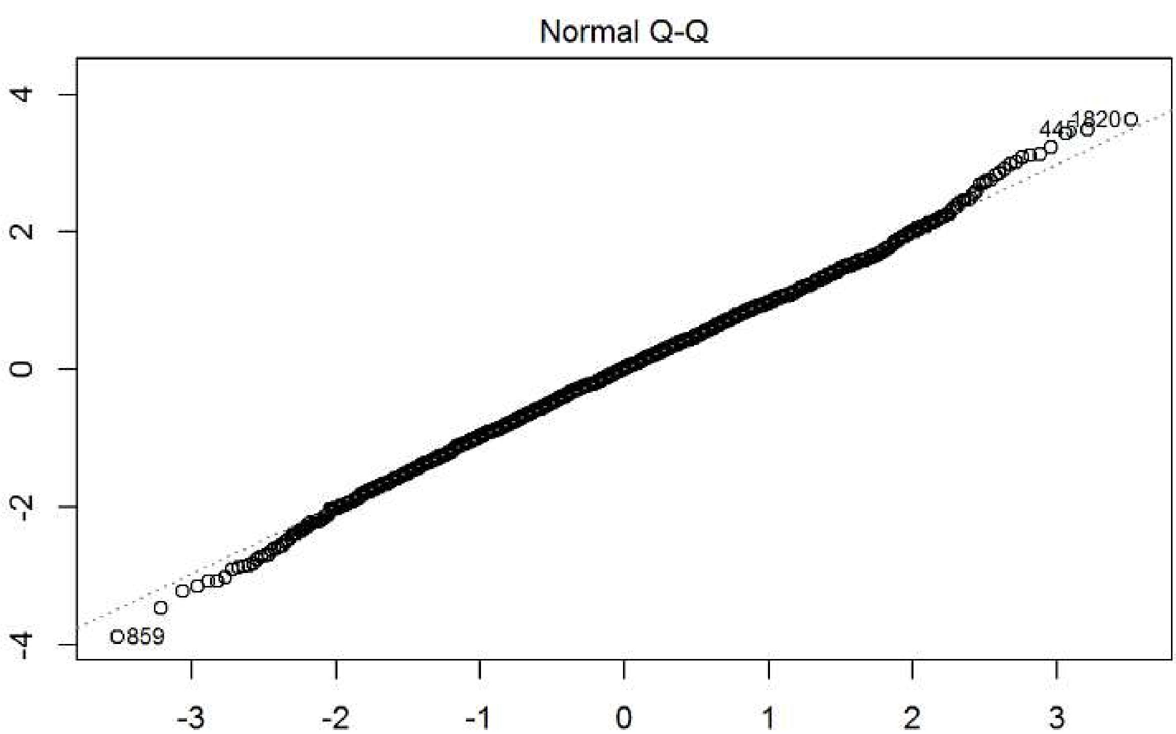
5] Print model summary

```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_H + poly(TEAM_BATTING_BB,
##     2) + TEAM_PITCHING_HR + poly(TEAM_PITCHING_BB, 2) + poly(TEAM_FIELDING_E,
##     2), data = train.trans)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -52.787  -8.938   0.192   9.165  49.336
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                2.848e+00  3.764e+00   0.757    0.449
## TEAM_BATTING_H             4.945e-02  2.586e-03  19.122  < 2e-16 ***
## poly(TEAM_BATTING_BB, 2)1  -3.781e+02  6.711e+01  -5.634 1.98e-08 ***
## poly(TEAM_BATTING_BB, 2)2   2.057e+02  2.629e+01   7.826 7.65e-15 ***
## TEAM_PITCHING_HR          -3.246e+00  5.333e-01  -6.088 1.34e-09 ***
## poly(TEAM_PITCHING_BB, 2)1  2.847e+02  4.062e+01   7.009 3.16e-12 ***
## poly(TEAM_PITCHING_BB, 2)2  1.759e+02  2.884e+01   6.100 1.24e-09 ***
## poly(TEAM_FIELDING_E, 2)1  -5.730e+02  6.423e+01  -8.920  < 2e-16 ***
## poly(TEAM_FIELDING_E, 2)2  -1.043e+02  1.606e+01  -6.496 1.01e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.58 on 2267 degrees of freedom
## Multiple R-squared:  0.259,  Adjusted R-squared:  0.2563
## F-statistic: 99.02 on 8 and 2267 DF,  p-value: < 2.2e-16
```

We can see from the above model summary that all the coefficient are statistically significant. Adding polinomial term to our model has increased the R square value to 0.259 which is improvement over model-1

6] Plot the residuals





We can see that residuals are looking better now.

7] Model-3 Pair plot above (fig 1.0) shows one more variable which has non-linear relationship with Target_Wins. Variable Team_Fielding_E. This is important variable which is negatively correlated with Target_Wins. Adding polynomial term for this variable may further improve the model effectiveness

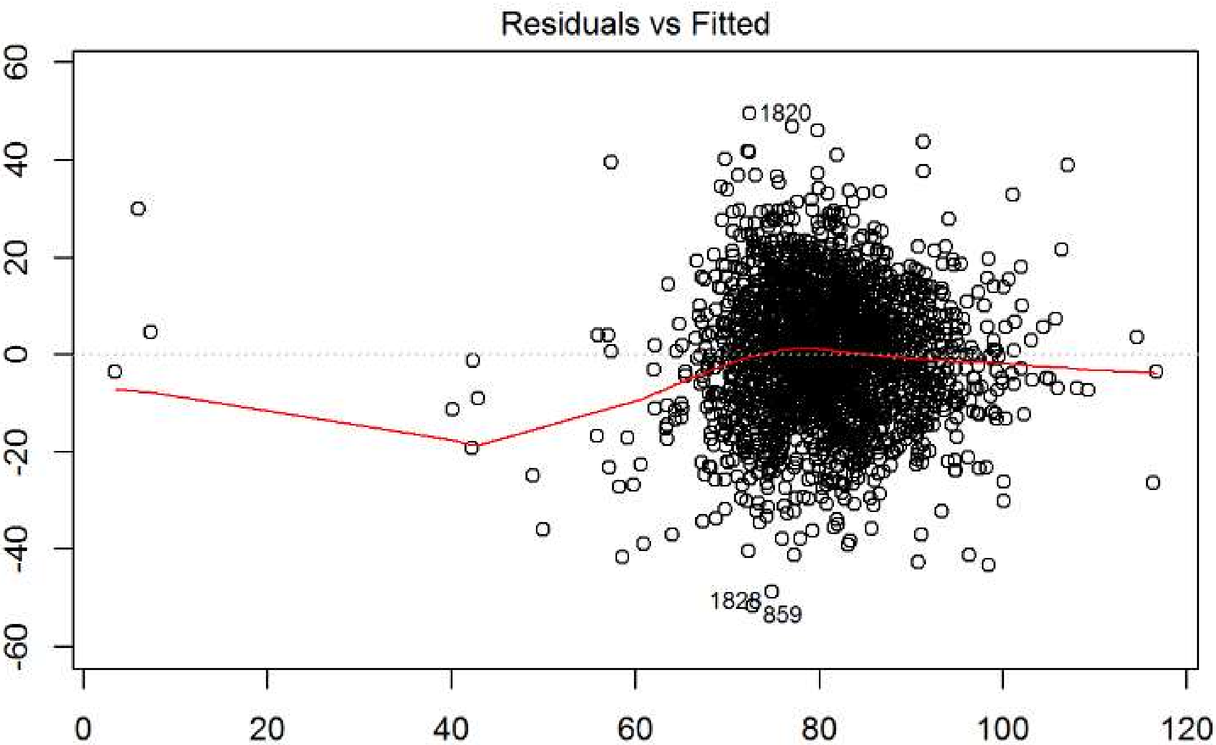
Let's add polynomial term for Team_Fielding_E variable

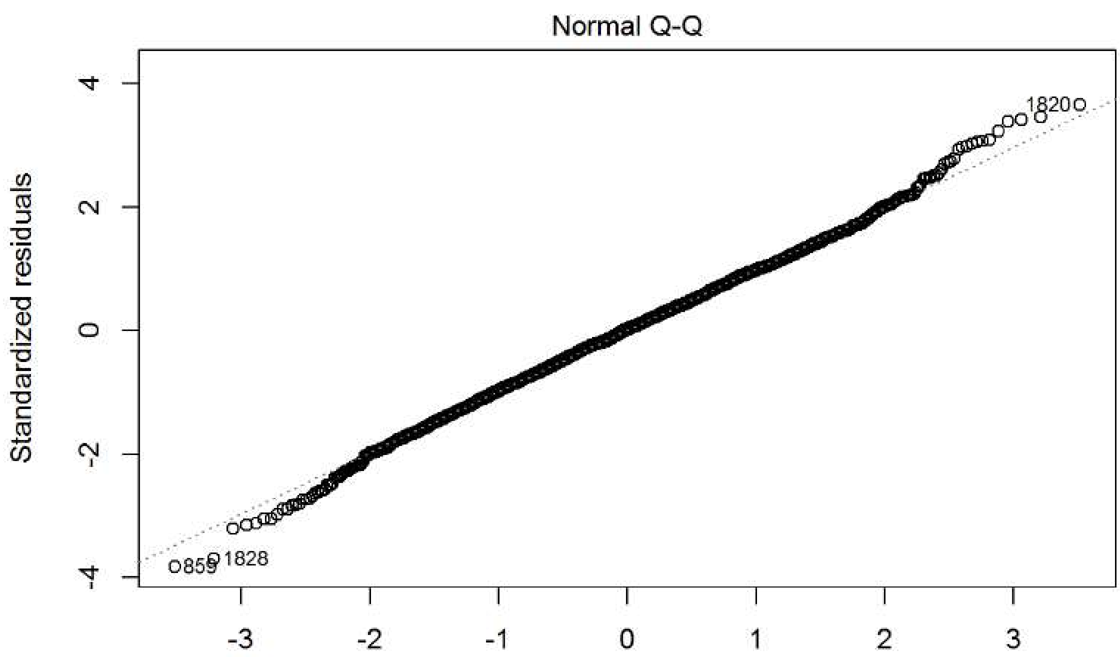
8] Print model summary

```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_H + poly(TEAM_BATTING_BB,
##     2) + TEAM_PITCHING_HR + poly(TEAM_PITCHING_BB, 2) + poly(TEAM_FIELDING_E,
##     3), data = train.trans)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -51.800  -8.957   0.291   9.101  49.593
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                2.836e+00  3.756e+00   0.755  0.45028
## TEAM_BATTING_H             5.083e-02  2.614e-03  19.443  < 2e-16 ***
## poly(TEAM_BATTING_BB, 2)1 -3.519e+02  6.744e+01  -5.218 1.97e-07 ***
## poly(TEAM_BATTING_BB, 2)2  1.826e+02  2.716e+01   6.722 2.26e-11 ***
## TEAM_PITCHING_HR          -3.827e+00  5.607e-01  -6.826 1.12e-11 ***
## poly(TEAM_PITCHING_BB, 2)1  2.781e+02  4.058e+01   6.855 9.19e-12 ***
## poly(TEAM_PITCHING_BB, 2)2  1.952e+02  2.937e+01   6.646 3.77e-11 ***
## poly(TEAM_FIELDING_E, 3)1 -5.658e+02  6.413e+01  -8.823  < 2e-16 ***
## poly(TEAM_FIELDING_E, 3)2 -1.051e+02  1.602e+01  -6.561 6.61e-11 ***
## poly(TEAM_FIELDING_E, 3)3 -5.528e+01  1.682e+01  -3.286  0.00103 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.55 on 2266 degrees of freedom
## Multiple R-squared:  0.2625, Adjusted R-squared:  0.2595
## F-statistic:  89.6 on 9 and 2266 DF,  p-value: < 2.2e-16
```

From the above model summary we can see that all the coefficient are statistically significant. This indicates that all the variables in the model are effective in predicting Target_Wins. Statistically significant F stats also proves this point. If we look at the R square it is further improved at 0.2625

9] Print Residual plot





Above residual plot for model-3 looks much better now

Select Models

For model selection we will use R square as model selection criteria. R square indicates the portion of variance explained by model from total available variance in the data set. Value of R square ranges from 0 to 1. 0 indicates poor model fitting and 1 indicates good fit of the model.

In general model with high R square value indicates better model fit and can be used for model selection.

In the three models above we can see that model-3 has the highest R square 0.2625 so we will select model-3 as our winning model