DATA621_Home_Work_1

Dilip Ganesan

6/15/2018

Money Ball Training Data Set.

1.DATA EXPLORATION.(Exploratory Data Analysis EDA)

As first set in our EDA, let us load the train data and do a summary statistics on the loaded dataset.

```
# Let us load the train.csv data
train = read.csv("moneyball-training-data.csv")
test = read.csv("moneyball-evaluation-data.csv")

summary(train)
```

```
##      INDEX         TARGET_WINS     TEAM_BATTING_H TEAM_BATTING_2B
##  Min.   :   1.0   Min.   :  0.00   Min.   : 891   Min.   : 69.0
##  1st Qu.: 630.8   1st Qu.: 71.00   1st Qu.:1383   1st Qu.:208.0
##  Median :1270.5   Median : 82.00   Median :1454   Median :238.0
##  Mean   :1268.5   Mean   : 80.79   Mean   :1469   Mean   :241.2
##  3rd Qu.:1915.5   3rd Qu.: 92.00   3rd Qu.:1537   3rd Qu.:273.0
##  Max.   :2535.0   Max.   :146.00   Max.   :2554   Max.   :458.0
##
##  TEAM_BATTING_3B  TEAM_BATTING_HR  TEAM_BATTING_BB TEAM_BATTING_SO
##  Min.   :  0.00   Min.   :  0.00   Min.   :  0.0   Min.   :   0.0
##  1st Qu.: 34.00   1st Qu.: 42.00   1st Qu.:451.0   1st Qu.: 548.0
##  Median : 47.00   Median :102.00   Median :512.0   Median : 750.0
##  Mean   : 55.25   Mean   : 99.61   Mean   :501.6   Mean   : 735.6
##  3rd Qu.: 72.00   3rd Qu.:147.00   3rd Qu.:580.0   3rd Qu.: 930.0
##  Max.   :223.00   Max.   :264.00   Max.   :878.0   Max.   :1399.0
##                                                    NA's   :102
##  TEAM_BASERUN_SB TEAM_BASERUN_CS TEAM_BATTING_HBP TEAM_PITCHING_H
##  Min.   :  0.0   Min.   :  0.0   Min.   :29.00    Min.   : 1137
##  1st Qu.: 66.0   1st Qu.: 38.0   1st Qu.:50.50    1st Qu.: 1419
##  Median :101.0   Median : 49.0   Median :58.00    Median : 1518
##  Mean   :124.8   Mean   : 52.8   Mean   :59.36    Mean   : 1779
##  3rd Qu.:156.0   3rd Qu.: 62.0   3rd Qu.:67.00    3rd Qu.: 1682
##  Max.   :697.0   Max.   :201.0   Max.   :95.00    Max.   :30132
##  NA's   :131     NA's   :772     NA's   :2085
##  TEAM_PITCHING_HR TEAM_PITCHING_BB TEAM_PITCHING_SO  TEAM_FIELDING_E
##  Min.   :  0.0    Min.   :   0.0   Min.   :    0.0   Min.   :  65.0
##  1st Qu.: 50.0    1st Qu.: 476.0   1st Qu.:  615.0   1st Qu.: 127.0
##  Median :107.0    Median : 536.5   Median :  813.5   Median : 159.0
##  Mean   :105.7    Mean   : 553.0   Mean   :  817.7   Mean   : 246.5
##  3rd Qu.:150.0    3rd Qu.: 611.0   3rd Qu.:  968.0   3rd Qu.: 249.2
##  Max.   :343.0    Max.   :3645.0   Max.   :19278.0   Max.   :1898.0
##                                    NA's   :102
##  TEAM_FIELDING_DP
##  Min.   : 52.0
##  1st Qu.:131.0
```

```
##  Median :149.0
##  Mean   :146.4
##  3rd Qu.:164.0
##  Max.   :228.0
##  NA's   :286
```

Our Training data set contains 2276 Rows and 16 Varaibles. Out of these 16 Variables, TARGET_WINS is the dependent variable and rest are Independent Variables. All Variables are continous Variables.

6 Variables has NAs in them, constituting a total of 3478 Missing Values in the entire data set.

The Variables for whom which has NA values are

TEAM_BATTING_SO

TEAM_BASERUN_SB

TEAM_BASERUN_CS

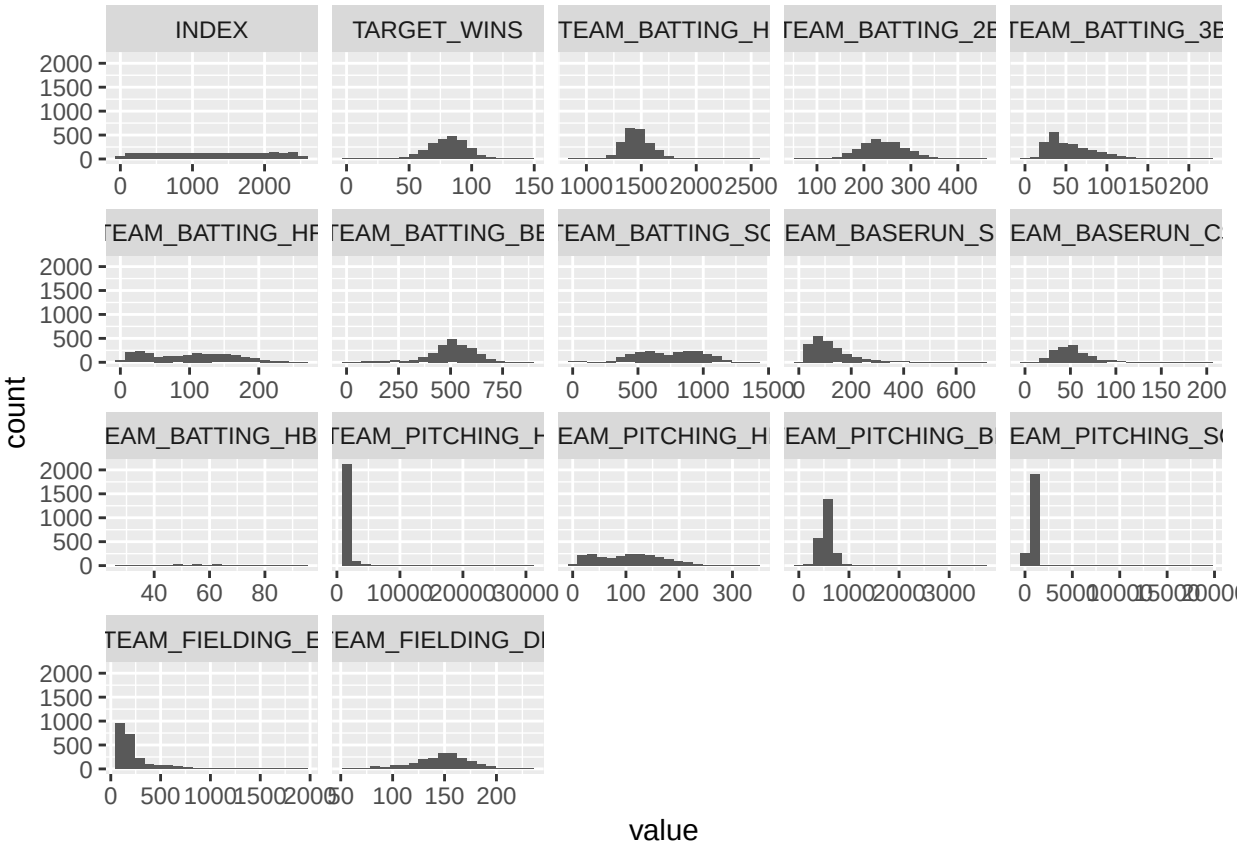
TEAM_BATTING_HBP

TEAM_PITCHING_SO

TEAM_FIELDING_DP

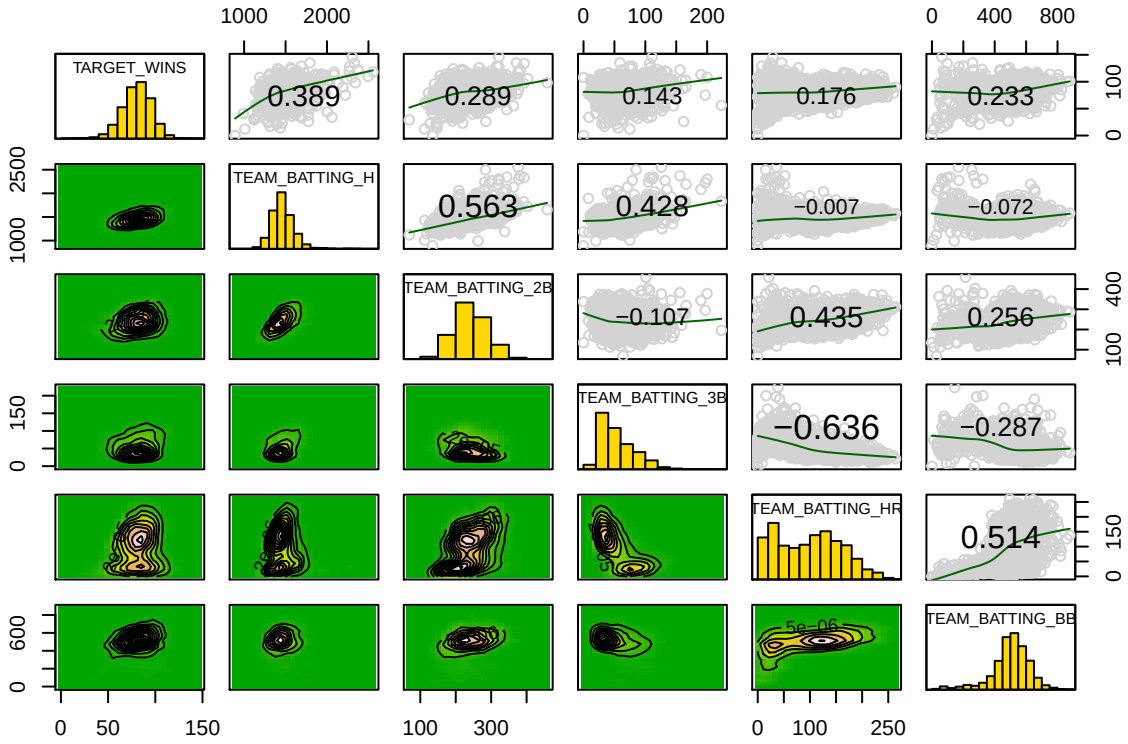
Below we will plot the Histogram for all our Variables.

```
ggtrain = reshape2::melt(train)
ggplot(ggtrain, aes(value)) + facet_wrap(~variable, scales = "free_x") + geom_histogram(bins = 20)
```



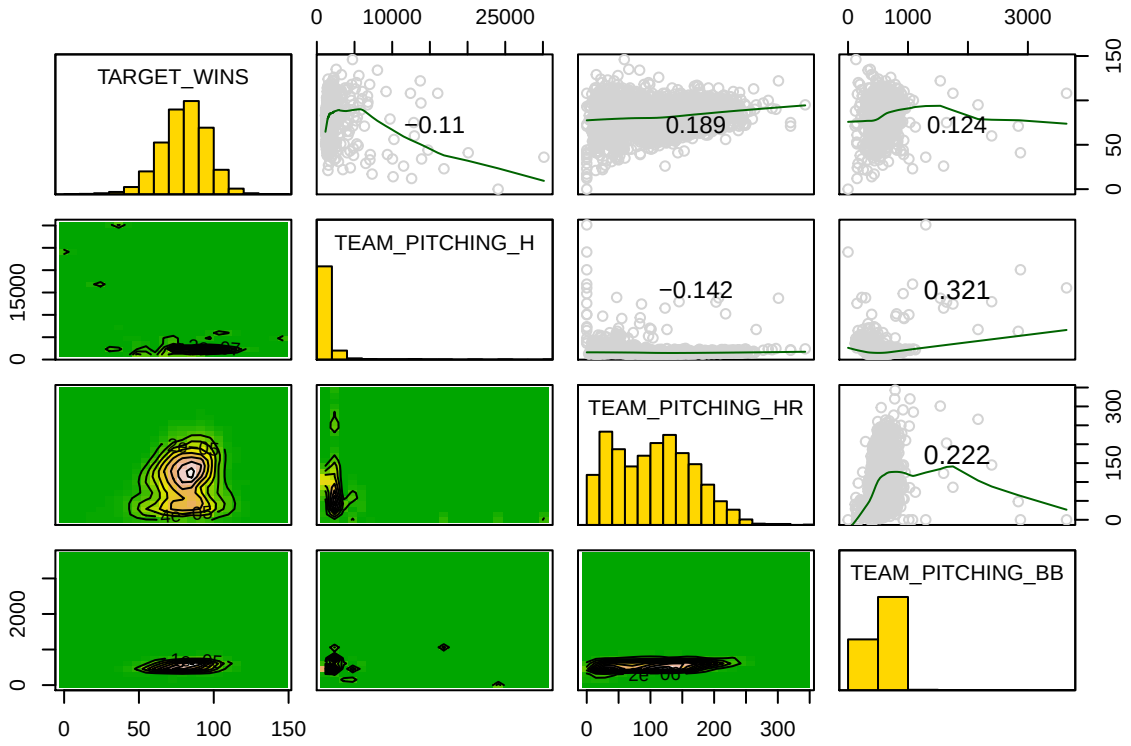
We will see the linearity between Dependent Variable and Independent Variables. For us the dependent variable is TARGET_WIN and independent variable are rest. For KDEPAIRS, excluded variables which contained NA values

```
battingdata = train[c("TARGET_WINS", "TEAM_BATTING_H", "TEAM_BATTING_2B", "TEAM_BATTING_3B",
    "TEAM_BATTING_HR", "TEAM_BATTING_BB")]
pitchingdata = train[c("TARGET_WINS", "TEAM_PITCHING_H", "TEAM_PITCHING_HR",
    "TEAM_PITCHING_BB")]
fieldingdata = train[c("TARGET_WINS", "TEAM_FIELDING_E")]

kdepairs(battingdata)
```



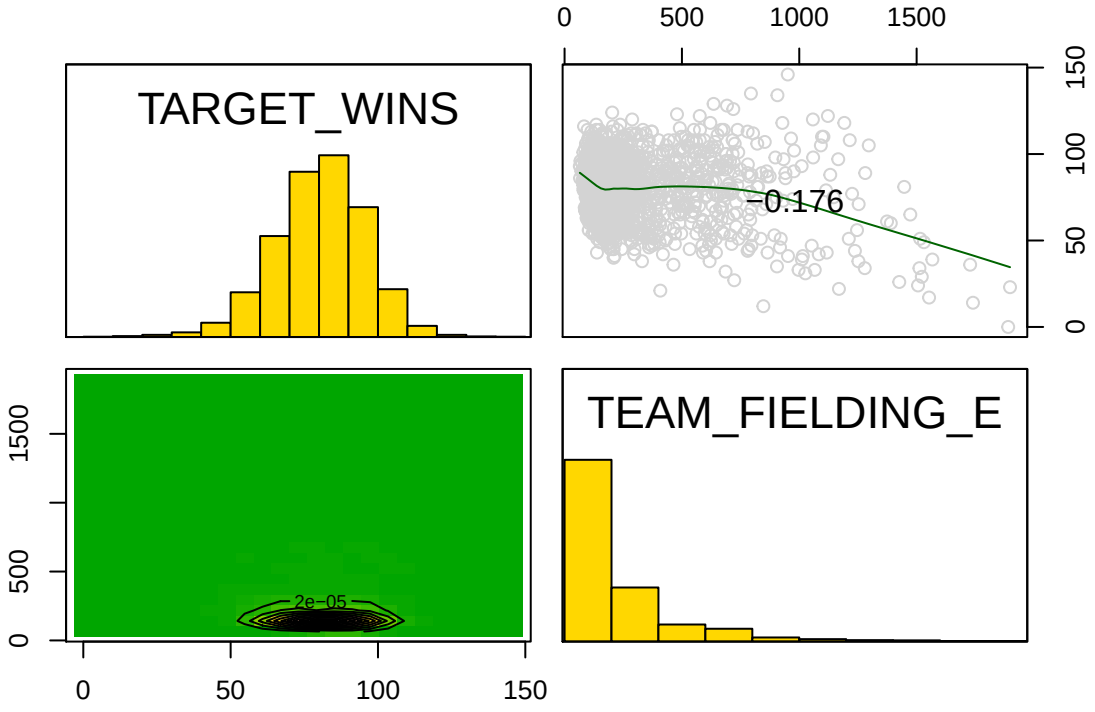
KDEPAIRS for Pitching Data Variables and TARGET_WINS

```
kdepairs(pitchingdata)
```



KDEPAIRS for Fielding Data Variables and TARGET_WINS

```
kdepairs(fieldingdata)
```



2.DATA PREPARATION

Removal of NAs

For NA values in the variables, we will fill the values with Mean value. In case of TEAM_BATTING_HBP almost 80% of the data is NAs. So we will drop it for our modelling.

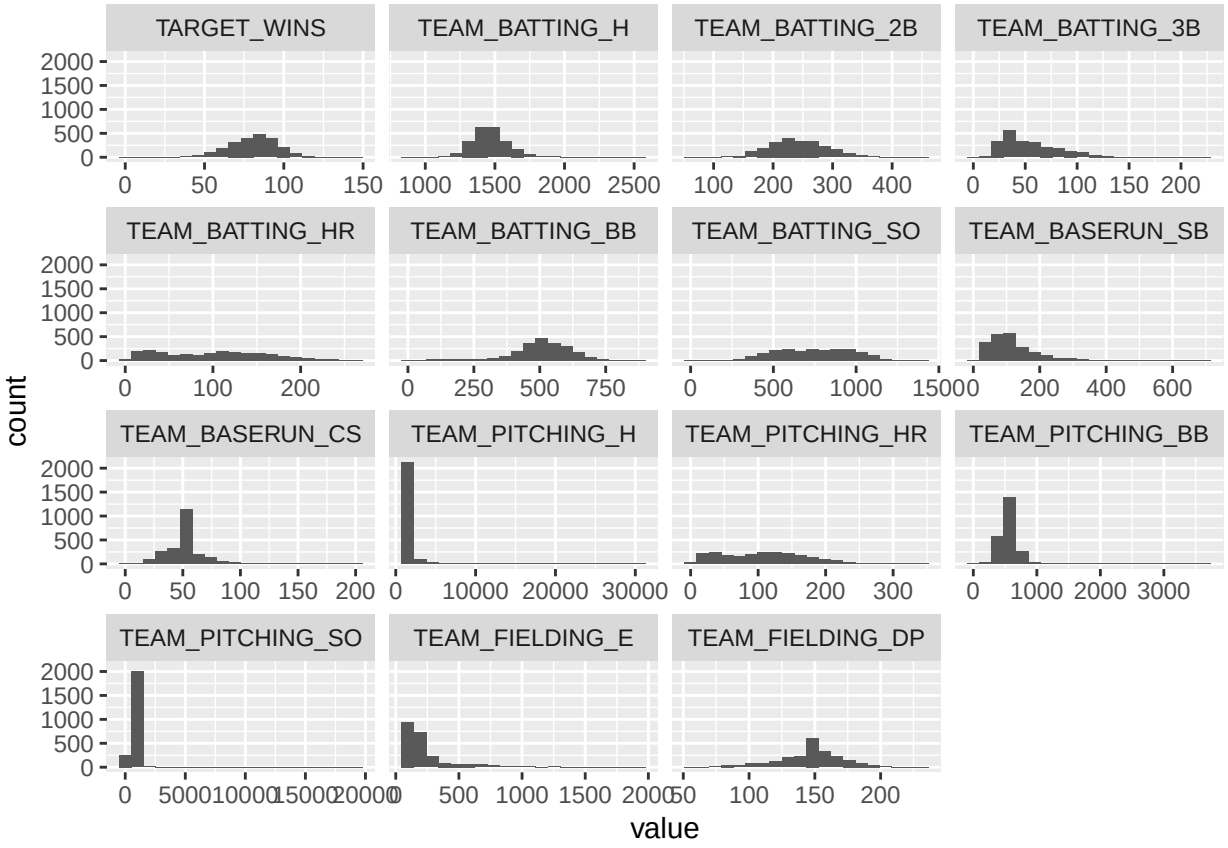
```
mean_CS = round(mean(train$TEAM_BASERUN_CS, na.rm = T))
mean_SB = round(mean(train$TEAM_BASERUN_SB, na.rm = T))
mean_DP = round(mean(train$TEAM_FIELDING_DP, na.rm = T))
mean_BAT_SO = round(mean(train$TEAM_BATTING_SO, na.rm = T))
mean_PIT_SO = round(mean(train$TEAM_PITCHING_SO, na.rm = T))

train$TEAM_BASERUN_CS[is.na(train$TEAM_BASERUN_CS)] = mean_CS
train$TEAM_BASERUN_SB[is.na(train$TEAM_BASERUN_SB)] = mean_SB
train$TEAM_FIELDING_DP[is.na(train$TEAM_FIELDING_DP)] = mean_DP
train$TEAM_BATTING_SO[is.na(train$TEAM_BATTING_SO)] = mean_BAT_SO
train$TEAM_PITCHING_SO[is.na(train$TEAM_PITCHING_SO)] = mean_PIT_SO

summary(train)
```

```
##      INDEX         TARGET_WINS     TEAM_BATTING_H TEAM_BATTING_2B
##  Min.   :   1.0   Min.   :  0.00   Min.   : 891   Min.   : 69.0
##  1st Qu.: 630.8   1st Qu.: 71.00   1st Qu.:1383   1st Qu.:208.0
##  Median :1270.5   Median : 82.00   Median :1454   Median :238.0
##  Mean   :1268.5   Mean   : 80.79   Mean   :1469   Mean   :241.2
##  3rd Qu.:1915.5   3rd Qu.: 92.00   3rd Qu.:1537   3rd Qu.:273.0
##  Max.   :2535.0   Max.   :146.00   Max.   :2554   Max.   :458.0
##
##  TEAM_BATTING_3B  TEAM_BATTING_HR  TEAM_BATTING_BB TEAM_BATTING_SO
##  Min.   :  0.00   Min.   :  0.00   Min.   :  0.0   Min.   :   0.0
##  1st Qu.: 34.00   1st Qu.: 42.00   1st Qu.:451.0   1st Qu.: 556.8
##  Median : 47.00   Median :102.00   Median :512.0   Median : 736.0
##  Mean   : 55.25   Mean   : 99.61   Mean   :501.6   Mean   : 735.6
##  3rd Qu.: 72.00   3rd Qu.:147.00   3rd Qu.:580.0   3rd Qu.: 925.0
##  Max.   :223.00   Max.   :264.00   Max.   :878.0   Max.   :1399.0
##
##  TEAM_BASERUN_SB TEAM_BASERUN_CS  TEAM_BATTING_HBP TEAM_PITCHING_H
##  Min.   :  0.0   Min.   :  0.00   Min.   :29.00    Min.   : 1137
##  1st Qu.: 67.0   1st Qu.: 44.00   1st Qu.:50.50    1st Qu.: 1419
##  Median :106.0   Median : 53.00   Median :58.00    Median : 1518
##  Mean   :124.8   Mean   : 52.87   Mean   :59.36    Mean   : 1779
##  3rd Qu.:151.0   3rd Qu.: 54.25   3rd Qu.:67.00    3rd Qu.: 1682
##  Max.   :697.0   Max.   :201.00   Max.   :95.00    Max.   :30132
##                                   NA's   :2085
##  TEAM_PITCHING_HR TEAM_PITCHING_BB TEAM_PITCHING_SO  TEAM_FIELDING_E
##  Min.   :  0.0    Min.   :   0.0   Min.   :    0.0   Min.   :  65.0
##  1st Qu.: 50.0    1st Qu.: 476.0   1st Qu.:  626.0   1st Qu.: 127.0
##  Median :107.0    Median : 536.5   Median :  818.0   Median : 159.0
##  Mean   :105.7    Mean   : 553.0   Mean   :  817.7   Mean   : 246.5
##  3rd Qu.:150.0    3rd Qu.: 611.0   3rd Qu.:  957.0   3rd Qu.: 249.2
##  Max.   :343.0    Max.   :3645.0   Max.   :19278.0   Max.   :1898.0
##
##  TEAM_FIELDING_DP
##  Min.   : 52.0
##  1st Qu.:134.0
##  Median :146.0
##  Mean   :146.3
##  3rd Qu.:161.2
```

```
##  Max.   :228.0
##
```

But for TEAM_BATTING_HBP rest of the field does not contain NA values. We would like to see the Histogram again after filling NAs will substitutes.

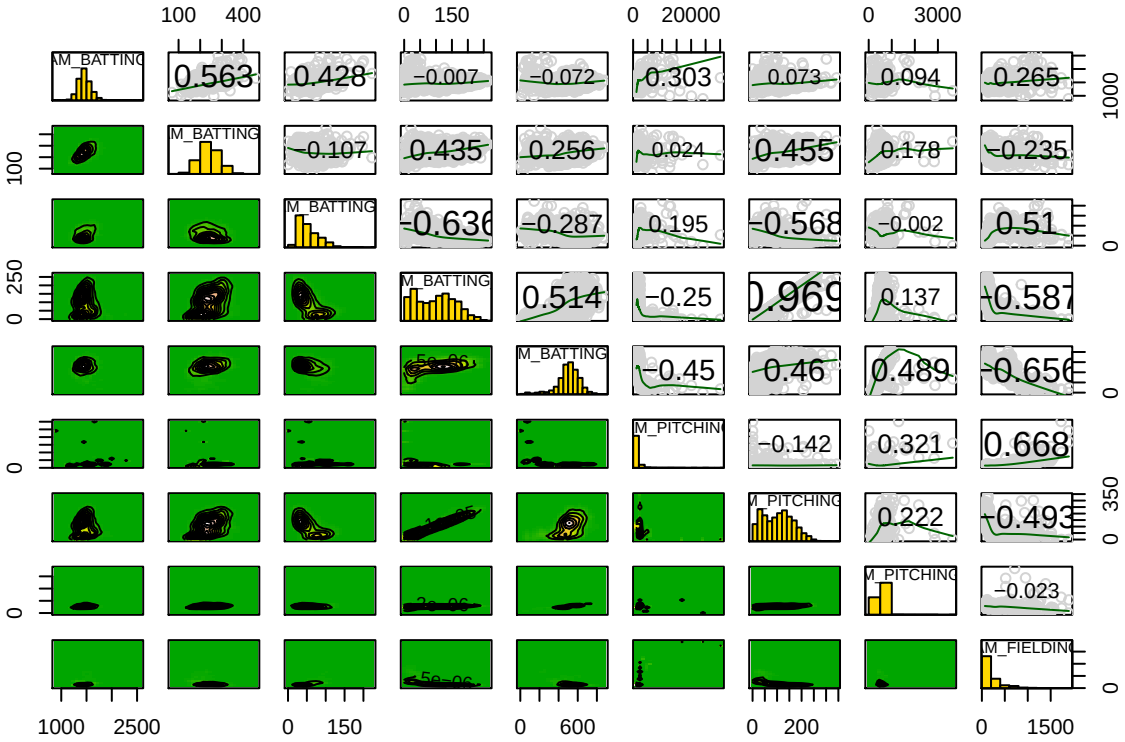
```
newtrain = subset(train, select = -c(INDEX, TEAM_BATTING_HBP))
ggtrain = reshape2::melt(newtrain)
ggplot(ggtrain, aes(value)) + facet_wrap(~variable, scales = "free_x") + geom_histogram(bins = 20)
```



Correlation between Independent Variables.

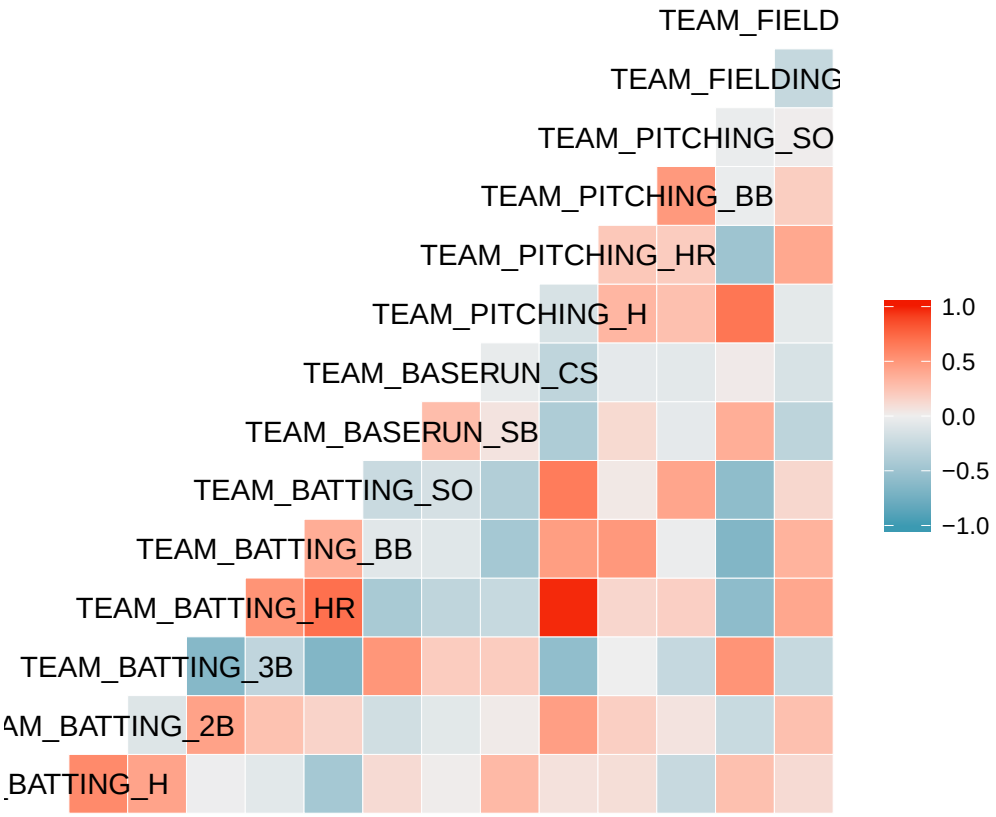
We will see the linearity between Independent Variables, excluding Dependent Variable TARGET_WINS and other variables which has NA Values.

```
IndData = train[c("TEAM_BATTING_H", "TEAM_BATTING_2B", "TEAM_BATTING_3B", "TEAM_BATTING_HR",
    "TEAM_BATTING_BB", "TEAM_PITCHING_H", "TEAM_PITCHING_HR", "TEAM_PITCHING_BB",
    "TEAM_FIELDING_E")]

kdepairs(IndData)
```



From the above plot we can see the following strong correlation betweem some of Independent Variables. We will perform a Corr Plot between the Independent variables, which will give us more insight This is need to check Multicollinearity between independent variables.

```
multicol = subset(train, select = -c(INDEX, TARGET_WINS, TEAM_BATTING_HBP))
ggcorr(multicol)
```



From the above analysis we can see TEAM_BATTING_HR and TEAM_PITCHIBG_H has a high correlation, Similarly those variables which has correlation coefficient > 0.5, we might have to drop. But for Initial analysis we will take all into consideration and will decide based on P-Value which one to reject when we do the model.

3. BUILD MODELS

Model 1 : For first we will create a model with TARGET_WINS and all other independent variables we are interested with.

```
## Raw data, all variables included in the model
train = subset(train, select = -c(INDEX))
model1 = lm(TARGET_WINS ~ ., data = train)
summary(model1)
```

```
##
## Call:
## lm(formula = TARGET_WINS ~ ., data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.8708  -5.6564  -0.0599   5.2545  22.9274
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      60.28826   19.67842   3.064  0.00253 **
## TEAM_BATTING_H    1.91348    2.76139   0.693  0.48927
```

```
## TEAM_BATTING_2B   0.02639    0.03029   0.871  0.38484
## TEAM_BATTING_3B  -0.10118    0.07751  -1.305  0.19348
## TEAM_BATTING_HR  -4.84371   10.50851  -0.461  0.64542
## TEAM_BATTING_BB  -4.45969    3.63624  -1.226  0.22167
## TEAM_BATTING_SO   0.34196    2.59876   0.132  0.89546
## TEAM_BASERUN_SB   0.03304    0.02867   1.152  0.25071
## TEAM_BASERUN_CS  -0.01104    0.07143  -0.155  0.87730
## TEAM_BATTING_HBP  0.08247    0.04960   1.663  0.09815 .
## TEAM_PITCHING_H  -1.89096    2.76095  -0.685  0.49432
## TEAM_PITCHING_HR  4.93043   10.50664   0.469  0.63946
## TEAM_PITCHING_BB  4.51089    3.63372   1.241  0.21612
## TEAM_PITCHING_SO -0.37364    2.59705  -0.144  0.88577
## TEAM_FIELDING_E  -0.17204    0.04140  -4.155 5.08e-05 ***
## TEAM_FIELDING_DP -0.10819    0.03654  -2.961  0.00349 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.467 on 175 degrees of freedom
##   (2085 observations deleted due to missingness)
## Multiple R-squared:  0.5501, Adjusted R-squared:  0.5116
## F-statistic: 14.27 on 15 and 175 DF,  p-value: < 2.2e-16
```

From the first Model we can see that most of the independent variables have a higher p-value but for TEAM_FIELDING_DP and TEAM_FIELDING_E. We are not taking TEAM_BATTING_HBP this in to consideration, because we will dropping this from our model because of NA values. The R^2 and Adjusted R^2 are at 55% and 50% respectively. This model defenitely needs some tunning.

For Model 2 : We would like to do some back ward elimination and see whether there is some scope of improvement.

We will drop the following from train data

TEAM_BATTING_HBP(Because of NA values), TEAM_BATTING_HR(Because of strong correlation with other independent variable)

```
train2 = subset(train, select = -c(TEAM_BATTING_HBP))

model2 = lm(TARGET_WINS ~ ., data = train2)
summary(model2)
```

```
##
## Call:
## lm(formula = TARGET_WINS ~ ., data = train2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -50.027  -8.575   0.137   8.342  58.611
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       2.507e+01  5.399e+00   4.643 3.63e-06 ***
## TEAM_BATTING_H    4.824e-02  3.687e-03  13.084  < 2e-16 ***
## TEAM_BATTING_2B  -2.004e-02  9.151e-03  -2.190 0.028604 *
## TEAM_BATTING_3B   6.040e-02  1.676e-02   3.604 0.000320 ***
## TEAM_BATTING_HR   5.298e-02  2.743e-02   1.931 0.053548 .
```

```
## TEAM_BATTING_BB   1.041e-02  5.818e-03   1.789 0.073671 .
## TEAM_BATTING_SO  -9.351e-03  2.550e-03  -3.667 0.000251 ***
## TEAM_BASERUN_SB   2.946e-02  4.464e-03   6.600 5.12e-11 ***
## TEAM_BASERUN_CS  -1.173e-02  1.616e-02  -0.726 0.467830
## TEAM_PITCHING_H  -7.315e-04  3.676e-04  -1.990 0.046736 *
## TEAM_PITCHING_HR  1.481e-02  2.432e-02   0.609 0.542633
## TEAM_PITCHING_BB  8.066e-05  4.145e-03   0.019 0.984477
## TEAM_PITCHING_SO  2.841e-03  9.187e-04   3.092 0.002009 **
## TEAM_FIELDING_E  -2.118e-02  2.480e-03  -8.542  < 2e-16 ***
## TEAM_FIELDING_DP -1.212e-01  1.304e-02  -9.298  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.04 on 2261 degrees of freedom
## Multiple R-squared:  0.319,  Adjusted R-squared:  0.3148
## F-statistic: 75.65 on 14 and 2261 DF,  p-value: < 2.2e-16
```

For Model 3 : From Training Data Set 2, we will remove the following variables, which has relatively higher p-values.

TEAM_PITCHING_BB(Highest p-value) TEAM_BASERUN_CS(High P-value) TEAM_PITCHING_HR(Strong correlation with another variable)

```
train3 = subset(train2, select = -c(TEAM_PITCHING_BB, TEAM_PITCHING_HR, TEAM_BASERUN_CS))

model3 = lm(TARGET_WINS ~ ., data = train3)
summary(model3)
```

```
##
## Call:
## lm(formula = TARGET_WINS ~ ., data = train3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -49.933  -8.565   0.093   8.392  58.634
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      23.7228255  5.2236488   4.541 5.88e-06 ***
## TEAM_BATTING_H    0.0484514  0.0036619  13.231  < 2e-16 ***
## TEAM_BATTING_2B  -0.0204939  0.0091353  -2.243 0.024969 *
## TEAM_BATTING_3B   0.0623760  0.0165840   3.761 0.000173 ***
## TEAM_BATTING_HR   0.0697597  0.0096249   7.248 5.78e-13 ***
## TEAM_BATTING_BB   0.0107278  0.0033484   3.204 0.001375 **
## TEAM_BATTING_SO  -0.0093020  0.0024567  -3.786 0.000157 ***
## TEAM_BASERUN_SB   0.0287487  0.0042919   6.698 2.65e-11 ***
## TEAM_PITCHING_H  -0.0006894  0.0003211  -2.147 0.031893 *
## TEAM_PITCHING_SO  0.0028828  0.0006707   4.298 1.79e-05 ***
## TEAM_FIELDING_E  -0.0206650  0.0024120  -8.568  < 2e-16 ***
## TEAM_FIELDING_DP -0.1211806  0.0130256  -9.303  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.03 on 2264 degrees of freedom
```

```
## Multiple R-squared:  0.3187, Adjusted R-squared:  0.3154
## F-statistic: 96.27 on 11 and 2264 DF,  p-value: < 2.2e-16
```

For Model 4, We are going to do a Step for our first training model and see what are the variables that are being dropped.

```
model1 = lm(TARGET_WINS ~ ., data = train)
selectedMod = step(model1)
```

```
## Start:  AIC=831.31
## 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_BATTING_HBP + 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_BATTING_SO    1      1.24 12547 829.33
## - TEAM_PITCHING_SO   1      1.48 12547 829.33
## - TEAM_BASERUN_CS    1      1.71 12548 829.34
## - TEAM_BATTING_HR    1     15.23 12561 829.54
## - TEAM_PITCHING_HR   1     15.79 12562 829.55
## - TEAM_PITCHING_H    1     33.63 12580 829.82
## - TEAM_BATTING_H     1     34.42 12580 829.83
## - TEAM_BATTING_2B    1     54.41 12600 830.14
## - TEAM_BASERUN_SB    1     95.22 12641 830.76
## - TEAM_BATTING_BB    1    107.84 12654 830.95
## - TEAM_PITCHING_BB   1    110.48 12656 830.99
## - TEAM_BATTING_3B    1    122.16 12668 831.16
## <none>                           12546 831.31
## - TEAM_BATTING_HBP   1    198.21 12744 832.31
## - TEAM_FIELDING_DP   1    628.49 13174 838.65
## - TEAM_FIELDING_E    1   1237.79 13784 847.28
##
## Step:  AIC=829.33
## TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_2B + TEAM_BATTING_3B +
##     TEAM_BATTING_HR + TEAM_BATTING_BB + TEAM_BASERUN_SB + TEAM_BASERUN_CS +
##     TEAM_BATTING_HBP + 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_BASERUN_CS    1      1.59 12549 827.35
## - TEAM_BATTING_HR    1     15.82 12563 827.57
## - TEAM_PITCHING_HR   1     16.39 12564 827.58
## - TEAM_BATTING_2B    1     53.47 12601 828.14
## - TEAM_PITCHING_H    1     88.45 12636 828.67
## - TEAM_BATTING_H     1     90.30 12637 828.70
## - TEAM_BASERUN_SB    1     94.19 12641 828.76
## - TEAM_BATTING_BB    1    107.95 12655 828.97
## - TEAM_PITCHING_BB   1    110.60 12658 829.01
## - TEAM_BATTING_3B    1    122.20 12669 829.18
## <none>                           12547 829.33
## - TEAM_BATTING_HBP   1    197.11 12744 830.31
## - TEAM_FIELDING_DP   1    630.68 13178 836.70
```

```
## - TEAM_FIELDING_E   1   1240.80 13788 845.34
## - TEAM_PITCHING_SO  1   1312.89 13860 846.34
##
## Step:  AIC=827.35
## TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_2B + TEAM_BATTING_3B +
##     TEAM_BATTING_HR + TEAM_BATTING_BB + TEAM_BASERUN_SB + TEAM_BATTING_HBP +
##     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_BATTING_HR   1     16.06 12565 825.60
## - TEAM_PITCHING_HR  1     16.64 12565 825.61
## - TEAM_BATTING_2B   1     53.05 12602 826.16
## - TEAM_PITCHING_H   1     90.24 12639 826.72
## - TEAM_BATTING_H    1     92.13 12641 826.75
## - TEAM_BATTING_BB   1    110.31 12659 827.03
## - TEAM_PITCHING_BB  1    113.00 12662 827.07
## - TEAM_BASERUN_SB   1    123.42 12672 827.22
## - TEAM_BATTING_3B   1    129.33 12678 827.31
## <none>                          12549 827.35
## - TEAM_BATTING_HBP  1    197.23 12746 828.33
## - TEAM_FIELDING_DP  1    635.62 13184 834.79
## - TEAM_PITCHING_SO  1   1311.88 13861 844.35
## - TEAM_FIELDING_E   1   1322.05 13871 844.49
##
## Step:  AIC=825.6
## TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_2B + TEAM_BATTING_3B +
##     TEAM_BATTING_BB + TEAM_BASERUN_SB + TEAM_BATTING_HBP + 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_BATTING_2B   1     55.48 12620 824.44
## - TEAM_PITCHING_H   1     89.26 12654 824.95
## - TEAM_BATTING_H    1     91.97 12657 824.99
## - TEAM_BATTING_BB   1    104.58 12669 825.18
## - TEAM_PITCHING_BB  1    107.19 12672 825.22
## <none>                          12565 825.60
## - TEAM_BATTING_3B   1    137.48 12702 825.68
## - TEAM_BASERUN_SB   1    146.90 12712 825.82
## - TEAM_BATTING_HBP  1    200.36 12765 826.62
## - TEAM_FIELDING_DP  1    628.95 13194 832.93
## - TEAM_PITCHING_HR  1    853.54 13418 836.15
## - TEAM_PITCHING_SO  1   1316.68 13882 842.63
## - TEAM_FIELDING_E   1   1333.15 13898 842.86
##
## Step:  AIC=824.44
## TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_3B + TEAM_BATTING_BB +
##     TEAM_BASERUN_SB + TEAM_BATTING_HBP + 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_H   1     84.47 12705 823.71
## - TEAM_BATTING_H    1     87.79 12708 823.76
```

```
## - TEAM_BATTING_BB   1     98.92 12719 823.93
## - TEAM_PITCHING_BB  1    101.48 12722 823.97
## - TEAM_BASERUN_SB   1    109.27 12730 824.09
## <none>                          12620 824.44
## - TEAM_BATTING_3B   1    147.01 12767 824.65
## - TEAM_BATTING_HBP  1    204.39 12825 825.51
## - TEAM_FIELDING_DP  1    649.12 13269 832.02
## - TEAM_PITCHING_HR  1    812.92 13433 834.36
## - TEAM_PITCHING_SO  1   1262.90 13883 840.66
## - TEAM_FIELDING_E   1   1379.34 14000 842.25
##
## Step:  AIC=823.71
## TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_3B + TEAM_BATTING_BB +
##     TEAM_BASERUN_SB + TEAM_BATTING_HBP + TEAM_PITCHING_HR + TEAM_PITCHING_BB +
##     TEAM_PITCHING_SO + TEAM_FIELDING_E + TEAM_FIELDING_DP
##
##                    Df Sum of Sq   RSS    AIC
## - TEAM_BATTING_BB   1     32.85 12738 822.21
## - TEAM_PITCHING_BB  1     43.42 12748 822.37
## - TEAM_BASERUN_SB   1    105.16 12810 823.29
## <none>                          12705 823.71
## - TEAM_BATTING_3B   1    153.13 12858 824.00
## - TEAM_BATTING_HBP  1    183.82 12888 824.46
## - TEAM_BATTING_H    1    504.11 13209 829.15
## - TEAM_FIELDING_DP  1    602.80 13308 830.57
## - TEAM_PITCHING_HR  1    850.25 13555 834.09
## - TEAM_PITCHING_SO  1   1259.72 13964 839.77
## - TEAM_FIELDING_E   1   1419.39 14124 841.94
##
## Step:  AIC=822.21
## TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_3B + TEAM_BASERUN_SB +
##     TEAM_BATTING_HBP + TEAM_PITCHING_HR + TEAM_PITCHING_BB +
##     TEAM_PITCHING_SO + TEAM_FIELDING_E + TEAM_FIELDING_DP
##
##                    Df Sum of Sq   RSS    AIC
## - TEAM_BASERUN_SB   1    109.99 12848 821.85
## <none>                          12738 822.21
## - TEAM_BATTING_3B   1    156.45 12894 822.54
## - TEAM_BATTING_HBP  1    186.58 12924 822.98
## - TEAM_BATTING_H    1    485.67 13223 827.35
## - TEAM_FIELDING_DP  1    623.19 13361 829.33
## - TEAM_PITCHING_HR  1    843.83 13581 832.46
## - TEAM_PITCHING_SO  1   1267.25 14005 838.32
## - TEAM_FIELDING_E   1   1395.02 14133 840.06
## - TEAM_PITCHING_BB  1   2364.81 15102 852.73
##
## Step:  AIC=821.85
## TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_3B + TEAM_BATTING_HBP +
##     TEAM_PITCHING_HR + TEAM_PITCHING_BB + TEAM_PITCHING_SO +
##     TEAM_FIELDING_E + TEAM_FIELDING_DP
##
##                    Df Sum of Sq   RSS    AIC
## - TEAM_BATTING_3B   1    133.47 12981 821.82
## <none>                          12848 821.85
```

```
## - TEAM_BATTING_HBP  1    177.11 13025 822.46
## - TEAM_BATTING_H    1    566.11 13414 828.09
## - TEAM_FIELDING_DP  1    737.46 13585 830.51
## - TEAM_PITCHING_HR  1    756.49 13604 830.78
## - TEAM_PITCHING_SO  1   1257.91 14106 837.69
## - TEAM_FIELDING_E   1   1330.40 14178 838.67
## - TEAM_PITCHING_BB  1   2371.12 15219 852.20
##
## Step:  AIC=821.82
## TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_HBP + TEAM_PITCHING_HR +
##     TEAM_PITCHING_BB + TEAM_PITCHING_SO + TEAM_FIELDING_E + TEAM_FIELDING_DP
##
##                     Df Sum of Sq   RSS    AIC
## <none>                           12981 821.82
## - TEAM_BATTING_HBP  1    228.70 13210 823.16
## - TEAM_BATTING_H    1    449.87 13431 826.33
## - TEAM_FIELDING_DP  1    813.17 13794 831.43
## - TEAM_PITCHING_HR  1    990.20 13971 833.86
## - TEAM_PITCHING_SO  1   1316.56 14298 838.27
## - TEAM_FIELDING_E   1   1334.60 14316 838.52
## - TEAM_PITCHING_BB  1   2583.00 15564 854.49
```

```
summary(selectedMod)
```

```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_BATTING_H + TEAM_BATTING_HBP +
##     TEAM_PITCHING_HR + TEAM_PITCHING_BB + TEAM_PITCHING_SO +
##     TEAM_FIELDING_E + TEAM_FIELDING_DP, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -20.2248  -5.6294  -0.0212   5.0439  21.3065
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      60.95454   19.10292   3.191 0.001670 **
## TEAM_BATTING_H    0.02541    0.01009   2.518 0.012648 *
## TEAM_BATTING_HBP  0.08712    0.04852   1.796 0.074211 .
## TEAM_PITCHING_HR  0.08945    0.02394   3.736 0.000249 ***
## TEAM_PITCHING_BB  0.05672    0.00940   6.034 8.66e-09 ***
## TEAM_PITCHING_SO -0.03136    0.00728  -4.308 2.68e-05 ***
## TEAM_FIELDING_E  -0.17218    0.03970  -4.338 2.38e-05 ***
## TEAM_FIELDING_DP -0.11904    0.03516  -3.386 0.000869 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.422 on 183 degrees of freedom
##   (2085 observations deleted due to missingness)
## Multiple R-squared:  0.5345, Adjusted R-squared:  0.5167
## F-statistic: 30.02 on 7 and 183 DF,  p-value: < 2.2e-16
```

4 . SELECT MODELS

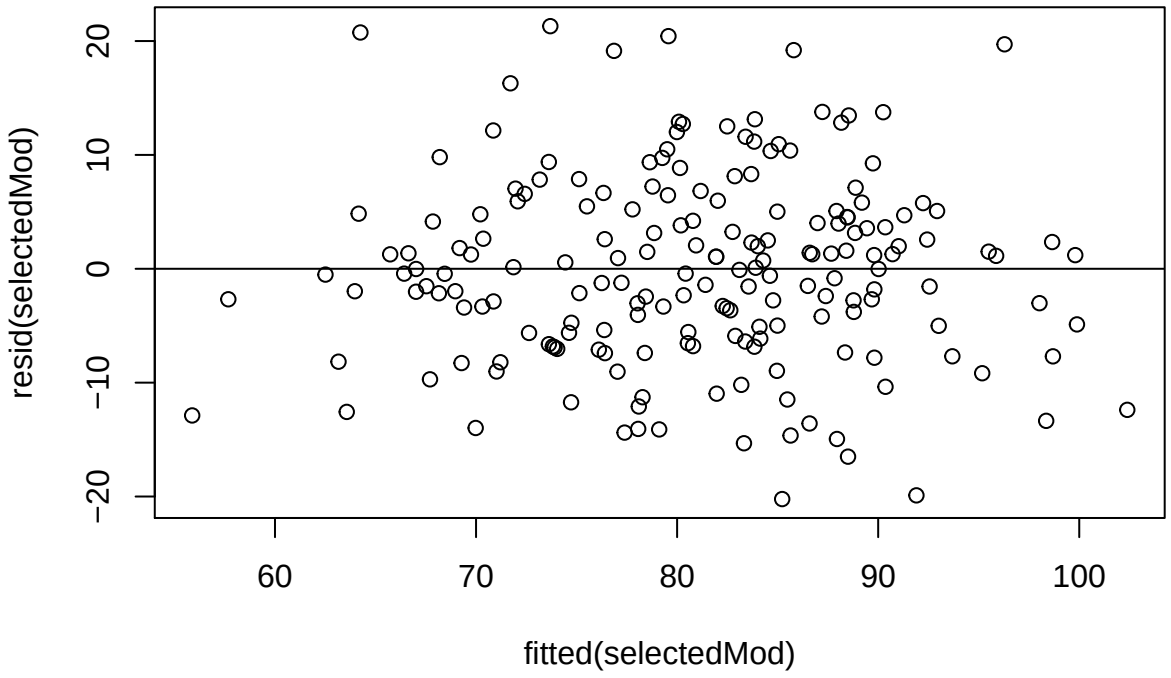
Now we have four models to select our model. We will see all the important statistics of the models and compare.

```
results = NULL
modellist = list(m1 = model1, m2 = model2, m3 = model3, m4 = selectedMod)

for (i in names(modellist)) {
    s = summary(modellist[[i]])
    name = i
    mse <- mean(s$residuals^2)
    r2 <- s$r.squared
    f <- s$fstatistic[1]
    k <- s$fstatistic[2]
    n <- s$fstatistic[3]
    results = rbind(results, data.frame(name = name, rsquared = r2, mse = mse,
        f = f, k = k, n = n))
}
rownames(results) = NULL
results
```

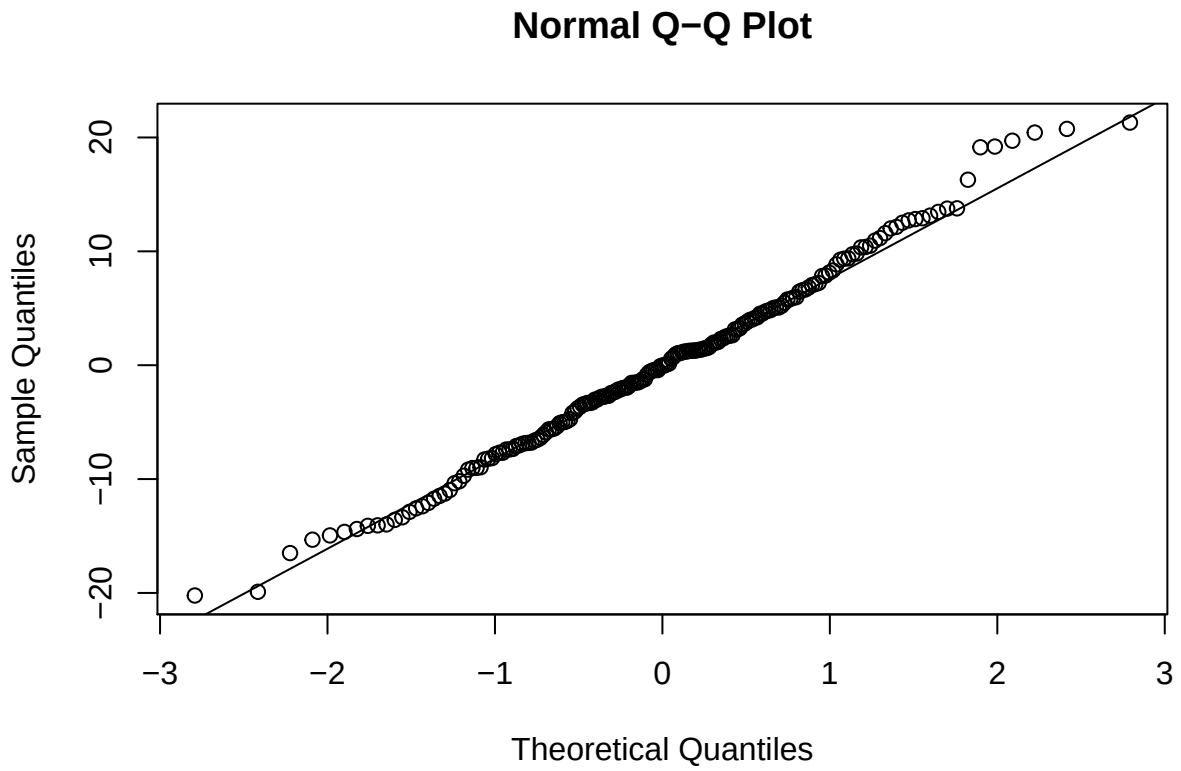
```
##   name  rsquared       mse        f  k    n
## 1   m1 0.5501165  65.68529 14.26597 15  175
## 2   m2 0.3189830 168.90672 75.64533 14 2261
## 3   m3 0.3186760 168.98287 96.26742 11 2264
## 4   m4 0.5345121  67.96361 30.01941  7  183
```

Out of 4 models, we will go with Model 4(M4) as our model and we see how the QQ Plots of Residuals looks like.

```
plot(fitted(selectedMod), resid(selectedMod))
abline(h = 0)
```



```
qqnorm(selectedMod$residuals)
qqline(selectedMod$residuals)
```



Residuals are almost normally distributed except for the at the top of the QQline where points are little displaced from the line.

```
# We will predict test data using our selected model
outprediction = predict(selectedMod, newdata = test, type = "response")
outprediction[is.na(outprediction)] = mean(train$TARGET_WINS)
```

Since most of the values from our prediction are NA values, we have replaced those with mean of training data set.

References:

http://r-statistics.co/Linear-Regression.html

http://r-statistics.co/Model-Selection-in-R.html