DATA621 Homework1 (Moneyball)

Jonathan Hernandez

June 5, 2018

This assignment focuses looking at the moneyball training data set and examining which variables are the best predictors for predicting team wins. I will look at statistics such as mean median and standard deviation, visual plots and examining any outliers if any. I will use the concept of multivariate regression to create models that best predict the team wins. Statistics like adjusted R^2, residual plots and p-values will be considered when picking the best model and predictors.

Data Preparation

Lets look and see what are some of the missing values and try replacing them with the mean for each column.

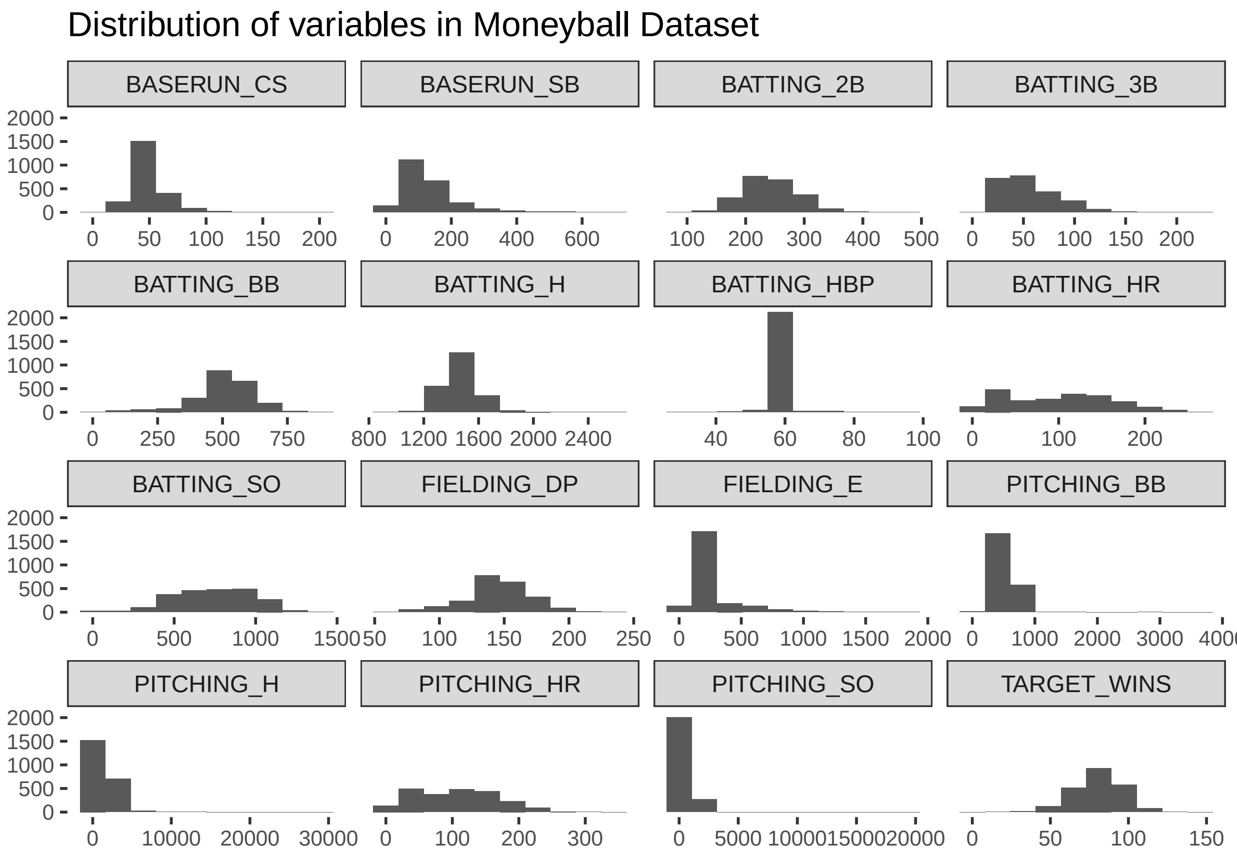
```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

- Missing values replacement, I decided to replace the NA's with the mean for each column. It is straightforward and a easy estimator to use.
- Another thing is to clean up for better readability are the variable names. I'll be removing the "TEAM_" and index as the Index is not needed and we know the dataset has data based on a particular team. See new names below

```
##  [1] "TARGET_WINS" "BATTING_H"   "BATTING_2B"  "BATTING_3B"  "BATTING_HR"
##  [6] "BATTING_BB"  "BATTING_SO"  "BASERUN_SB"  "BASERUN_CS"  "BATTING_HBP"
## [11] "PITCHING_H"  "PITCHING_HR" "PITCHING_BB" "PITCHING_SO" "FIELDING_E"
## [16] "FIELDING_DP"
```

Data Exploration

- With the data cleaned to our liking, lets do some sample plots to better understand how the data behave. For example, what kind of distrubution do the data follow? What are the mean, median and standard deviation, can we find some correlation coefficients between the Wins and other variables? Let's see.



- Calculating the mean, median and standard deviation which will understand the average, middle and variability we have in our data.

```
## TARGET_WINS   BATTING_H  BATTING_2B  BATTING_3B  BATTING_HR  BATTING_BB
##    80.79086  1469.26977   241.24692    55.25000    99.61204   501.55888
##  BATTING_SO  BASERUN_SB  BASERUN_CS BATTING_HBP  PITCHING_H PITCHING_HR
##   735.60534   124.76177    52.80386    59.35602  1779.21046   105.69859
## PITCHING_BB PITCHING_SO  FIELDING_E FIELDING_DP
##   553.00791   817.73045   246.48067   146.38794
```

```
## TARGET_WINS   BATTING_H  BATTING_2B  BATTING_3B  BATTING_HR  BATTING_BB
##    82.00000  1454.00000   238.00000    47.00000   102.00000   512.00000
##  BATTING_SO  BASERUN_SB  BASERUN_CS BATTING_HBP  PITCHING_H PITCHING_HR
##   735.60534   106.00000    52.80386    59.35602  1518.00000   107.00000
## PITCHING_BB PITCHING_SO  FIELDING_E FIELDING_DP
##   536.50000   817.73045   159.00000   146.38794
```

```
## TARGET_WINS   BATTING_H  BATTING_2B  BATTING_3B  BATTING_HR  BATTING_BB
##   15.752152  144.591195   46.801415   27.938557   60.546872  122.670862
##  BATTING_SO  BASERUN_SB  BASERUN_CS BATTING_HBP  PITCHING_H PITCHING_HR
##  242.891168   85.226079   18.659130    3.747397 1406.842930   61.298747
## PITCHING_BB PITCHING_SO  FIELDING_E FIELDING_DP
##  166.357362  540.544021  227.770972   24.522522
```

- We see that some of the variables are normally distributed, others left or right-skewed. This could also give us an idea of what our trained linear models will predict for a team's wins.

Building a Model - Multiple Linear Regression

- Let us now start building linear models. I will

use the idea of backward selection and forward selection using the p-value for judgement.By eliminating variables with high p-values we are removing variables where we would not fail to reject the null hypothesis. This should also give us a moderate R^2_{adj} value. Also I will try using the adjusted R^2 value as a statistic using also the selection techinques to grab two more models.
Let's start will all variables and see how valid is this model it is of the form

$$
\begin{aligned}
\widehat{team\_wins} = \beta_0 + \beta_1 * \widehat{basehits} + \beta_2 * \widehat{doubles} + \beta_3 * \widehat{triples} + \\
\beta_4 * \widehat{homeruns} + \beta_5 * \widehat{walks} + \beta_6 * \widehat{hitbypitch} + \beta_7 * \widehat{strikeouts} + \\
\beta_8 * \widehat{stolenbases} + \beta_9 * \widehat{caughtstealing} + \beta_{10} * \widehat{errors} + \beta_{11} * \widehat{doubleplays} + \\
+\beta_{12} * \widehat{walksallow} + \beta_{13} * \widehat{hitsallow} + \beta_{14} * \widehat{homerunsallow} + \\
\beta_{15} * \widehat{strikeouts\_pitch}
\end{aligned}
$$

```
##
## Call:
## lm(formula = TARGET_WINS ~ ., data = moneyball)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -50.019  -8.640   0.148   8.354  58.658
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.095e+01  6.874e+00   3.048 0.002332 **
## BATTING_H    4.821e-02  3.687e-03  13.075  < 2e-16 ***
## BATTING_2B  -2.006e-02  9.152e-03  -2.192 0.028489 *
## BATTING_3B   6.057e-02  1.676e-02   3.614 0.000308 ***
## BATTING_HR   5.302e-02  2.743e-02   1.933 0.053347 .
## BATTING_BB   1.037e-02  5.818e-03   1.782 0.074945 .
## BATTING_SO  -9.408e-03  2.552e-03  -3.687 0.000232 ***
## BASERUN_SB   2.955e-02  4.462e-03   6.623  4.4e-11 ***
## BASERUN_CS  -1.182e-02  1.614e-02  -0.732 0.464219
## BATTING_HBP  6.982e-02  7.303e-02   0.956 0.339166
## PITCHING_H  -7.325e-04  3.677e-04  -1.993 0.046433 *
## PITCHING_HR  1.483e-02  2.432e-02   0.610 0.542126
## PITCHING_BB  7.764e-05  4.146e-03   0.019 0.985058
## PITCHING_SO  2.846e-03  9.188e-04   3.098 0.001972 **
## FIELDING_E  -2.118e-02  2.481e-03  -8.536  < 2e-16 ***
## FIELDING_DP -1.208e-01  1.302e-02  -9.274  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.04 on 2260 degrees of freedom
## Multiple R-squared:  0.3192, Adjusted R-squared:  0.3147
## F-statistic: 70.65 on 15 and 2260 DF,  p-value: < 2.2e-16
```

- Here we see the estimated beta values for each variable and p-value as well as the R^2 and R^2_{adj}.

- Lets look at the first model using backward elimination (looking at the variable with the highest p-value, removing it

repeat until all variables have a low p-value below 0.05). We go back to the model table and see we can eliminate the following variables in the order below:

- PITCHING_BB, PITCHING_HR, BASERUN_CS, BATTING_HBP

```
##
## Call:
## lm(formula = TARGET_WINS ~ BATTING_H + BATTING_2B + BATTING_3B +
##     BATTING_HR + BATTING_SO + BASERUN_SB + PITCHING_SO + FIELDING_E +
##     FIELDING_DP + BATTING_BB + PITCHING_H, data = moneyball)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -49.899  -8.568   0.091   8.397  58.651
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 23.6666983  5.2220414   4.532 6.14e-06 ***
## BATTING_H    0.0484570  0.0036621  13.232  < 2e-16 ***
## BATTING_2B  -0.0205123  0.0091358  -2.245 0.024847 *
## BATTING_3B   0.0624661  0.0165843   3.767 0.000170 ***
## BATTING_HR   0.0697785  0.0096266   7.249 5.75e-13 ***
## BATTING_SO  -0.0093019  0.0024571  -3.786 0.000157 ***
## BASERUN_SB   0.0287708  0.0042901   6.706 2.51e-11 ***
## PITCHING_SO  0.0028867  0.0006707   4.304 1.75e-05 ***
## FIELDING_E  -0.0205973  0.0024120  -8.540  < 2e-16 ***
## FIELDING_DP -0.1210083  0.0130082  -9.302  < 2e-16 ***
## BATTING_BB   0.0107446  0.0033489   3.208 0.001354 **
## PITCHING_H  -0.0006920  0.0003211  -2.155 0.031253 *
## ---
## 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:  0.3186, Adjusted R-squared:  0.3153
## F-statistic: 96.25 on 11 and 2264 DF,  p-value: < 2.2e-16
```

- Having this model has estimates of low p-values and even though the R^2 and

R^2_{adj} didn't change much our variables have estimated low p-values.

To me I am confident using this model to use as I feel that getting strikeouts, errors, and letting the opposing team get hits can affect how the team will win and having these values too high I think will not make a team have many wins; you need not only a good offense but a good defense as well. The 3 variables in my last sentence have negative estimated coefficents which makes sense; more strikeouts/errors/hits allowed, the less wins a team is estimated to have.

- Model #2: For my second model, I will manually pickout the variables I want to

include and will go off based on what I think contributes to a winning team.

The variables I will use are

- BATTING_H
- BATTING_2B
- BATTING_3B
- BATTING_HR
- BATTING_BB
- BATTING_HBP
- BASERUN_SB
- FIELDING_DP
- PITCHING_SO

```
##
## Call:
## lm(formula = TARGET_WINS ~ BATTING_H + BATTING_2B + BATTING_3B +
##     BATTING_HR + BATTING_BB + BATTING_HBP + BASERUN_SB + FIELDING_DP +
##     PITCHING_SO, data = moneyball)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -64.796  -8.336   0.190   8.720  52.217
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.9747708  5.8877925   1.354 0.175725
## BATTING_H    0.0422318  0.0031507  13.404  < 2e-16 ***
## BATTING_2B  -0.0070966  0.0090177  -0.787 0.431386
## BATTING_3B   0.0663954  0.0166550   3.987 6.92e-05 ***
## BATTING_HR   0.0648627  0.0077572   8.362  < 2e-16 ***
## BATTING_BB   0.0316772  0.0028994  10.926  < 2e-16 ***
## BATTING_HBP  0.0477227  0.0750927   0.636 0.525155
## BASERUN_SB   0.0154454  0.0040562   3.808 0.000144 ***
## FIELDING_DP -0.1278949  0.0131943  -9.693  < 2e-16 ***
## PITCHING_SO  0.0005177  0.0005695   0.909 0.363474
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.42 on 2266 degrees of freedom
## Multiple R-squared:  0.2775, Adjusted R-squared:  0.2747
## F-statistic: 96.73 on 9 and 2266 DF,  p-value: < 2.2e-16
```

- Looking at this second model, the R^2 and R^2_{adj} decreased and looking

at the coefficents, it indicates that hitting more doubles decreases wins slowly and getting more double plays decreases wins as well. Even using backward elimination will make all coefficents more positive, but will leave out variables regarding defense which I don't feel confident as I believe the best model and to account for in real-life baseball is to have both offense and defense.

- For my 3rd and final model. I will use even less variables that not only constitute to getting many wins but such that the sign of the coefficent of each variable makes sense (negative coefficent for batting strikeouts, postive coefficent for homeruns etc.)

- BATTING_H
- BATTING_HR
- BASERUN_SB
- BATTING_SO
- FIELDING_DP
- PITCHING_BB
- PITCHING_H
- PITCHING_HR
- PITCHING_SO

```
##
## Call:
## lm(formula = TARGET_WINS ~ BATTING_H + BATTING_HR + BASERUN_SB +
##     BATTING_SO + FIELDING_DP + PITCHING_BB + PITCHING_H + PITCHING_HR +
##     PITCHING_SO, data = moneyball)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -53.980  -8.701   0.215   8.872  48.184
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 20.7062930  4.8596270   4.261 2.12e-05 ***
## BATTING_H    0.0477693  0.0026452  18.059  < 2e-16 ***
## BATTING_HR   0.1074833  0.0233597   4.601 4.43e-06 ***
## BASERUN_SB   0.0214236  0.0039743   5.391 7.75e-08 ***
## BATTING_SO  -0.0073478  0.0024585  -2.989  0.00283 **
## FIELDING_DP -0.1221450  0.0132027  -9.251  < 2e-16 ***
## PITCHING_BB  0.0142372  0.0022559   6.311 3.32e-10 ***
## PITCHING_H  -0.0031066  0.0002676 -11.608  < 2e-16 ***
## PITCHING_HR -0.0332853  0.0218716  -1.522  0.12819
## PITCHING_SO  0.0011948  0.0007703   1.551  0.12102
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.32 on 2266 degrees of freedom
## Multiple R-squared:  0.288,  Adjusted R-squared:  0.2852
## F-statistic: 101.9 on 9 and 2266 DF,  p-value: < 2.2e-16
```

- This model I feel doesn't make as much sense as it is saying that more double plays a team gets the less wins and in reality, getting double plays is great defensive work and strategy. Also it shows that allowing more walks increases the wins by a small amount which I disagree as allowing walks can help the opposing team make comebacks and more likley to get an RBI and win.
- Out of the 3 models although they are counter-intitutive in some way, a model

will have to be selected.

Selected Model

- Each of the 3 models I went off based on p-value as ones with a high p-value

I can reject the null hypothesis (H_0: p = 0) and favor the alternative (H_A: p != 0) and eliminate that variable.

- Out of the 3 models, I will select the first one that was based on backward elimination.

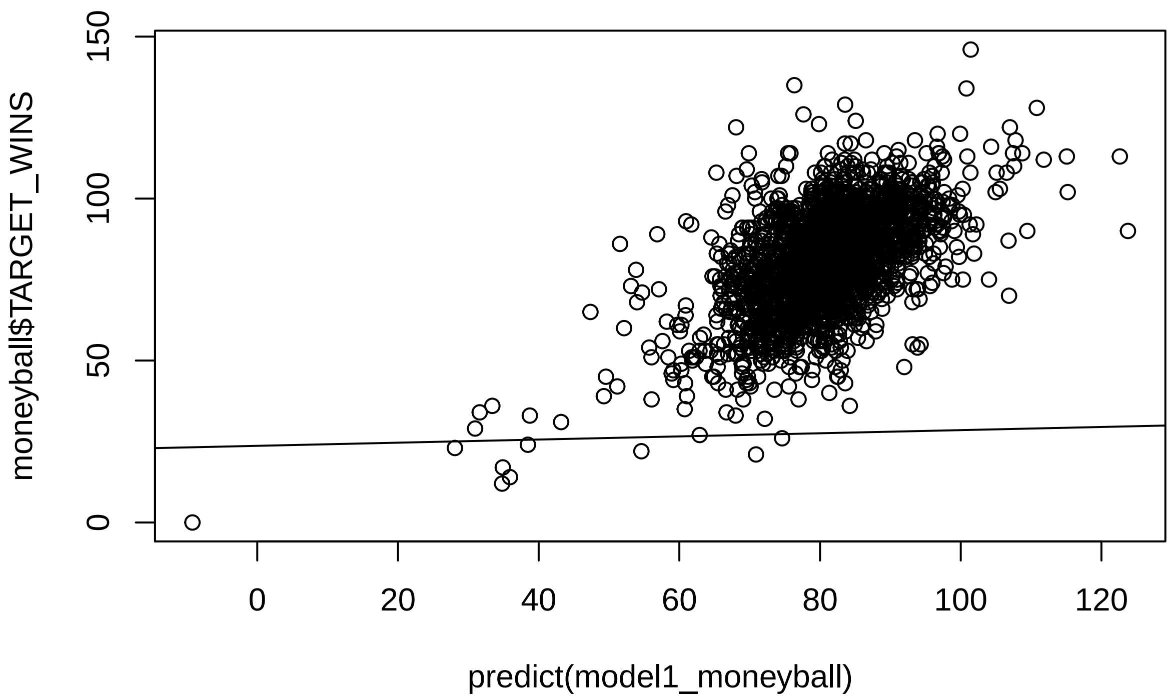
I am going by this as after using the p-values, I want to use other statistics like R^2 and R^2_{adj} and the first model has the highest R^2_{adj} = 0.3186.

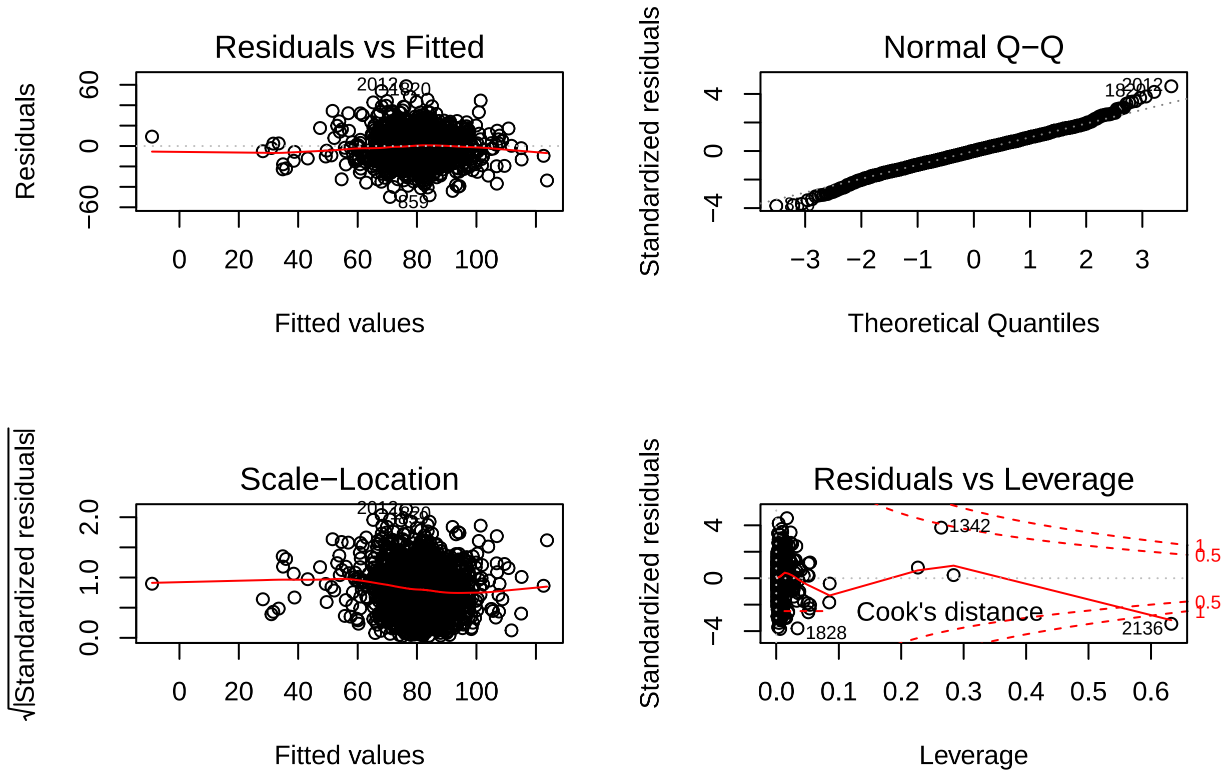
My first model has predictors that take into account both batting and pitching and reasonable signs in the coefficients.

- The model selected is the equation below (rounded to the nearest thousandth)

$$\begin{aligned}\widehat{team\_wins} = 23.6667 + 0.0485 * \widehat{basehits} - 0.0205 * \widehat{doubles} + 0.0625 * \widehat{triples} + \\ 0.0698 * \widehat{homeruns} + 0.0107 * \widehat{walks} - 0.0093 \widehat{strikeouts} + \\ 0.0288 * \widehat{stolenbases} - 0.0206 * \widehat{errors} - 0.1210 * \widehat{doubleplays} \\ -0.0007 * \widehat{hitsallow}\end{aligned}$$

```
## Warning in abline(model1_moneyball): only using the first two of 12
## regression coefficients
```

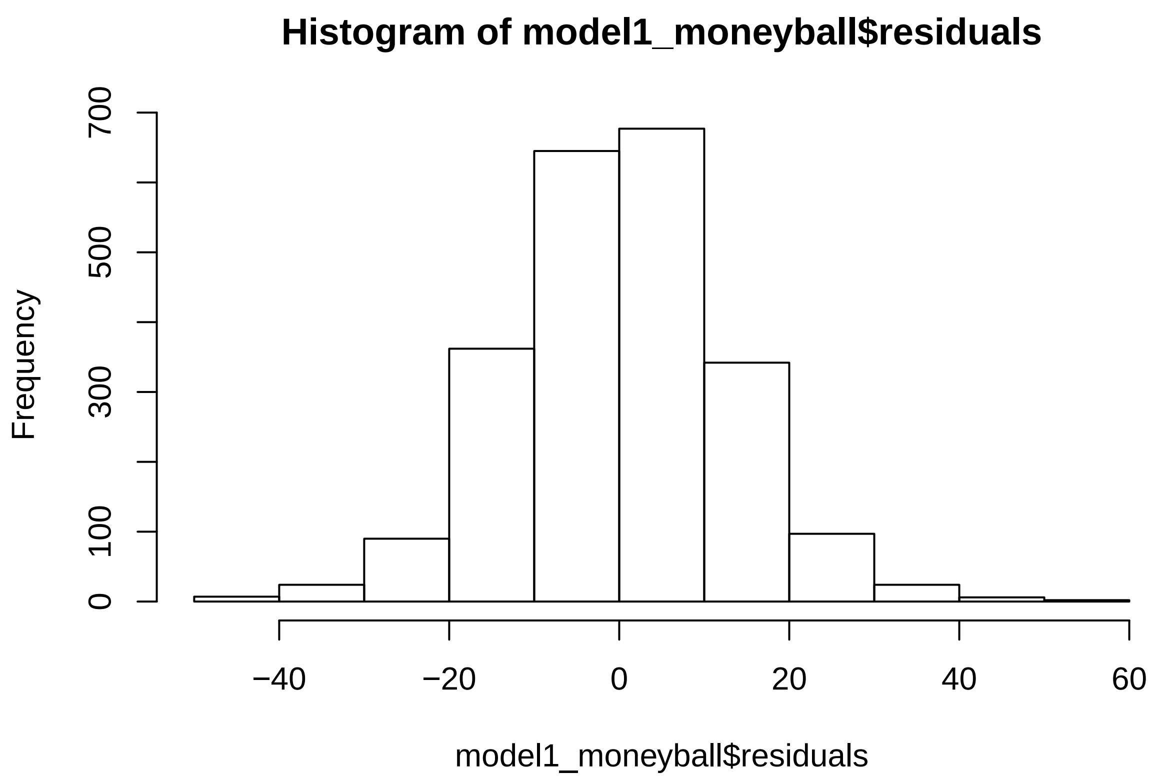




The F-score, R^2 and RMSE (Root Mean Squared Error) below:

```
##        value        numdf        dendf
##   96.2482041   11.0000000 2264.0000000    0.3153221   168.9936146
```

- Histogram of residuals and normal probability plot and residuals vs fitted values



- While the model does not fit the data that great, based on what we have is something

I will use as our model for predicting wins for a particular team. It is the model with p-values very low which is what I am going after as well as the best R^2 value.

- Making the predictions for the evaluation data set:

```
##         1         2         3         4         5         6         7 
##  63.67770  65.35447  75.05018  86.17206        NA        NA        NA 
##         8         9        10        11        12        13        14 
##  77.54572  70.99625  74.03253  69.66460  82.56473  82.05377  82.23139 
##        15        16        17        18        19        20        21 
##  84.75502  77.49560  74.71202  78.50497        NA  91.43791  81.38244 
##        22        23        24        25        26        27        28 
##  83.67658  81.18187  72.33135  81.89889  86.73784        NA  75.55260 
##        29        30        31        32        33        34        35 
##  84.09007  75.74837  90.63349  85.41542  82.27772  84.56967  80.69899 
##        36        37        38        39        40        41        42 
##  86.94142  76.00760  90.63189  85.53883  92.75279  82.58938  90.43394 
##        43        44        45        46        47        48        49 
##        NA        NA        NA        NA        NA  77.65547  70.27300 
##        50        51        52        53        54        55        56 
##  79.88623  76.76609  84.81803  78.01878  73.94062  75.83861  78.70755 
##        57        58        59        60        61        62        63 
##  93.11501  75.77370        NA        NA  87.14992  74.02042  87.97735 
##        64        65        66        67        68        69        70 
##  85.38424  83.29261        NA  77.81491  83.07243        NA  89.08372 
##        71        72        73        74        75        76        77 
##  86.44896  69.52511  77.25101  89.34574  82.04331  86.14696  81.61310 
##        78        79        80        81        82        83        84 
##  83.37783        NA        NA  84.54738  88.84505  97.59703  74.84373 
##        85        86        87        88        89        90        91 
##  85.76233  79.89919  82.45598  83.43497  87.48358  89.72995        NA 
##        92        93        94        95        96        97        98 
##        NA  75.63397        NA        NA        NA  88.20156 104.13348 
##        99       100       101       102       103       104       105 
##  86.98811  86.87349  79.97916  74.49943  84.00583  84.10366  79.88969 
##       106       107       108       109       110       111       112 
##        NA        NA  77.33018  86.52839        NA  83.49493  83.93099 
##       113       114       115       116       117       118       119 
##  93.09762  91.17246  80.94242  78.03609  85.45630  80.35841  74.98994 
##       120       121       122       123       124       125       126 
##        NA        NA        NA        NA        NA  69.55849  88.46056 
##       127       128       129       130       131       132       133 
##  92.29957  77.77801  93.44789  92.43404  86.52143  78.59637  79.86725 
##       134       135       136       137       138       139       140 
##  85.82971  86.96762        NA  73.83568  77.41470  84.90232  80.48769 
##       141       142       143       144       145       146       147 
##  67.71421        NA  90.54578  74.21675  71.56898  72.18585  77.88629 
##       148       149       150       151       152       153       154 
##  78.57218  78.63333  82.74550  82.26186  80.07694        NA  71.11913 
##       155       156       157       158       159       160       161 
##  77.00077  70.66566  88.93767        NA  95.94722        NA 105.08918 
##       162       163       164       165       166       167       168 
## 107.09781  94.12777 104.32163  98.28625  89.24743  81.63477  80.59103 
```

```
##        169        170        171        172        173        174        175
##   72.97232   80.17040         NA   88.70342   80.80428   94.06246   84.10359
##        176        177        178        179        180        181        182
##   73.39619   77.19145   71.11003   74.50024   79.27393   84.94249   88.55583
##        183        184        185        186        187        188        189
##   84.75119   85.13085         NA         NA         NA         NA         NA
##        190        191        192        193        194        195        196
##         NA         NA         NA   77.36400   77.60733   80.58574   68.65654
##        197        198        199        200        201        202        203
##   79.11843   84.34044   79.84527   84.98266   76.95921   80.21936   74.36836
##        204        205        206        207        208        209        210
##   88.26599   80.29793   83.38095   77.84044   77.61706         NA         NA
##        211        212        213        214        215        216        217
##         NA         NA   82.59648   65.61955   68.96501   84.16841   79.65177
##        218        219        220        221        222        223        224
##   92.21710   77.40237   78.51097   78.48769   74.02368   81.44050   73.50218
##        225        226        227        228        229        230        231
##         NA   74.85029   81.65572   79.72811   81.61661         NA         NA
##        232        233        234        235        236        237        238
##   92.76442   78.45783   88.98250   80.57789   75.48454   83.32365   77.39272
##        239        240        241        242        243        244        245
##         NA   73.04670   89.75591   85.75585   83.13279   80.86066   61.20716
##        246        247        248        249        250        251        252
##   86.72279   81.04924   85.12161   72.73651   83.01376   81.30727         NA
##        253        254        255        256        257        258        259
##         NA         NA   69.50712   76.63582   81.78858   82.40477   77.67525
```

- Appendix of R code

```r
library(dplyr)
library(ggplot2)
library(tidyr)
moneyball <- read.csv("moneyball-training-data.csv")
means <- apply(moneyball,2,mean, na.rm = TRUE)

# replace NA's with the mean
for (i in 2:ncol(moneyball)){
  moneyball[is.na(moneyball[, i]), i] <- means[i]
}
moneyball <- moneyball %>% select(-c("INDEX"))

# remove "TEAM_" from each column
colnames(moneyball) <- gsub("TEAM_", "", colnames(moneyball))
names(moneyball)
# sample plots use ggplot2 and geom_histogram
histograms <- moneyball %>% gather()
ggplot(gather(moneyball), aes(value)) +
  geom_histogram(bins = 10) +
  facet_wrap(~key, scales = 'free_x') +
  xlab(element_blank()) +
  ylab(element_blank()) +
  theme_bw() +
  theme(panel.grid = element_blank(), panel.border = element_blank(),
        axis.title = element_blank(), axis.text =element_text(size = 8)) +
  ggtitle("Distribution of variables in Moneyball Dataset")
```

```r
apply(moneyball, 2, mean)
apply(moneyball, 2, median)
apply(moneyball, 2, sd)
# build a model containing all the variables to predict team wins
default_lm_moneyball <- lm(TARGET_WINS ~ ., data = moneyball)
summary(default_lm_moneyball)
model1_moneyball <- lm(TARGET_WINS ~ BATTING_H + BATTING_2B + BATTING_3B +
                         BATTING_HR + BATTING_SO + BASERUN_SB +
                         PITCHING_SO + FIELDING_E + FIELDING_DP +
                         BATTING_BB + PITCHING_H,
                       data = moneyball)
summary(model1_moneyball)
model2_moneyball <- lm(TARGET_WINS ~ BATTING_H + BATTING_2B + BATTING_3B +
                         BATTING_HR + BATTING_BB + BATTING_HBP + BASERUN_SB +
                         FIELDING_DP + PITCHING_SO, data = moneyball)
summary(model2_moneyball)
model3_moneyball <- lm(TARGET_WINS ~ BATTING_H  + BATTING_HR +
                         BASERUN_SB + BATTING_SO + FIELDING_DP + PITCHING_BB +
                         PITCHING_H + PITCHING_HR + PITCHING_SO,
                       data = moneyball)
summary(model3_moneyball)
# fitting the values in our model with the evaluation data
plot(predict(model1_moneyball), moneyball$TARGET_WINS)
abline(model1_moneyball)
par(mfrow=c(2,2))
plot(model1_moneyball)
c(summary(model1_moneyball)$fstatistic, summary(model1_moneyball)$adj.r.squared,
  mean(summary(model1_moneyball)$residuals^2))
hist(model1_moneyball$residuals)
eval_moneyball <- read.csv("moneyball-evaluation-data.csv")

# remove "TEAM_" from each column
colnames(eval_moneyball) <- gsub("TEAM_", "", colnames(eval_moneyball))
predict(model1_moneyball,eval_moneyball)
```