Regression Model Assessment

Arindam

June 11th, 2016

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     891    1383    1454    1469    1537    2554
```

Model Assessment

```r
m1<-lm(TARGET_WINS~TEAM_FIELDING_E+TEAM_PITCHING_HR+TEAM_BATTING_BB+TEAM_BATTING_HR+TEAM_BATTING_2B+TEAM
summary(m1)
```

```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_FIELDING_E + TEAM_PITCHING_HR +
##     TEAM_BATTING_BB + TEAM_BATTING_HR + TEAM_BATTING_2B + TEAM_BATTING_H,
##     data = train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -52.697  -8.838  -0.030   8.850  58.613
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       2.204777   3.439149   0.641  0.52153
## TEAM_FIELDING_E  -0.017565   0.002027  -8.667  < 2e-16 ***
## TEAM_PITCHING_HR  0.021252   0.021168   1.004  0.31549
## TEAM_BATTING_BB   0.016398   0.003183   5.151 2.81e-07 ***
## TEAM_BATTING_HR  -0.018821   0.022911  -0.821  0.41147
## TEAM_BATTING_2B  -0.033308   0.009001  -3.700  0.00022 ***
## TEAM_BATTING_H    0.056052   0.002823  19.859  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.67 on 2269 degrees of freedom
## Multiple R-squared:  0.2488, Adjusted R-squared:  0.2468
## F-statistic: 125.3 on 6 and 2269 DF,  p-value: < 2.2e-16
```

```r
m2<-lm(TARGET_WINS~TEAM_FIELDING_E+TEAM_PITCHING_HR+TEAM_BATTING_BB+TEAM_BATTING_HR+TEAM_BATTING_2B,tra
```

Enhancing the model:

```r
# model selection using AIC- backward way model selection

step(lm(TARGET_WINS~TEAM_FIELDING_E+TEAM_PITCHING_HR+TEAM_BATTING_BB+TEAM_BATTING_HR+TEAM_BATTING_2B+TE
```

```
## Start:  AIC=11911.56
## TARGET_WINS ~ TEAM_FIELDING_E + TEAM_PITCHING_HR + TEAM_BATTING_BB +
##     TEAM_BATTING_HR + TEAM_BATTING_2B + TEAM_BATTING_H
##
##                    Df Sum of Sq    RSS   AIC
## - TEAM_BATTING_HR   1       126 424162 11910
## - TEAM_PITCHING_HR  1       188 424225 11911
## <none>                          424036 11912
## - TEAM_BATTING_2B   1      2559 426595 11923
## - TEAM_BATTING_BB   1      4959 428995 11936
## - TEAM_FIELDING_E   1     14037 438073 11984
## - TEAM_BATTING_H    1     73699 497735 12274
##
## Step:  AIC=11910.24
## TARGET_WINS ~ TEAM_FIELDING_E + TEAM_PITCHING_HR + TEAM_BATTING_BB +
##     TEAM_BATTING_2B + TEAM_BATTING_H
##
##                    Df Sum of Sq    RSS   AIC
## - TEAM_PITCHING_HR  1       110 424272 11909
## <none>                          424162 11910
## - TEAM_BATTING_2B   1      2695 426858 11923
## - TEAM_BATTING_BB   1      4885 429047 11934
## - TEAM_FIELDING_E   1     14834 438996 11986
## - TEAM_BATTING_H    1     77893 502055 12292
##
## Step:  AIC=11908.83
## TARGET_WINS ~ TEAM_FIELDING_E + TEAM_BATTING_BB + TEAM_BATTING_2B +
##     TEAM_BATTING_H
##
##                   Df Sum of Sq    RSS   AIC
## <none>                         424272 11909
## - TEAM_BATTING_2B  1      2631 426903 11921
## - TEAM_BATTING_BB  1      5290 429562 11935
## - TEAM_FIELDING_E  1     16276 440548 11992
## - TEAM_BATTING_H   1     77791 502063 12290
```

```
##
## Call:
## lm(formula = TARGET_WINS ~ TEAM_FIELDING_E + TEAM_BATTING_BB +
##     TEAM_BATTING_2B + TEAM_BATTING_H, data = train_data)
##
## Coefficients:
##     (Intercept)  TEAM_FIELDING_E  TEAM_BATTING_BB  TEAM_BATTING_2B
##         1.50215         -0.01734          0.01666         -0.03179
##  TEAM_BATTING_H
##         0.05641
```

```
#  Comparing models (partial F test) - This is used to evaluate if all the variables are important  or
anova(m1,m2)
```

```
## Analysis of Variance Table
##
## Model 1: TARGET_WINS ~ TEAM_FIELDING_E + TEAM_PITCHING_HR + TEAM_BATTING_BB +
```

```
##     TEAM_BATTING_HR + TEAM_BATTING_2B + TEAM_BATTING_H
## Model 2: TARGET_WINS ~ TEAM_FIELDING_E + TEAM_PITCHING_HR + TEAM_BATTING_BB +
##     TEAM_BATTING_HR + TEAM_BATTING_2B
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1   2269 424036
## 2   2270 497735 -1    -73699 394.36 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# confidence interval for coeffient
confint(m1)
```

```
##                        2.5 %      97.5 %
## (Intercept)      -4.53942766  8.94898237
## TEAM_FIELDING_E  -0.02153965 -0.01359060
## TEAM_PITCHING_HR -0.02025822  0.06276260
## TEAM_BATTING_BB   0.01015544  0.02264069
## TEAM_BATTING_HR  -0.06374926  0.02610817
## TEAM_BATTING_2B  -0.05095985 -0.01565627
## TEAM_BATTING_H    0.05051651  0.06158659
```

Model selection strategy:

1. Use R^2(adj) as it penalize bigger model and hence better than R^2.Select highest R^2(adjust)
2. AIC (model with lowest value is selected) or BIC (model with lowest value is selected) for model (there is one Mallow's Cp which is almost a linear function of AIC). These two are model comparison statistics no p value. Applicable to all types of regression model comparison

Note on AIC:

AIC is founded on information theory: it offers a relative estimate of the information lost when a given model is used to represent the process that generates the data. In doing so, it deals with the trade-off between the goodness of fit of the model and the complexity of the model.AIC does not provide a test of a model in the sense of testing a null hypothesis; i.e. AIC can tell nothing about the quality of the model in an absolute sense. If all the candidate models fit poorly, AIC will not give any warning of that.

```
# AIC/BIC vaalues for regression model

AIC(m1)
```

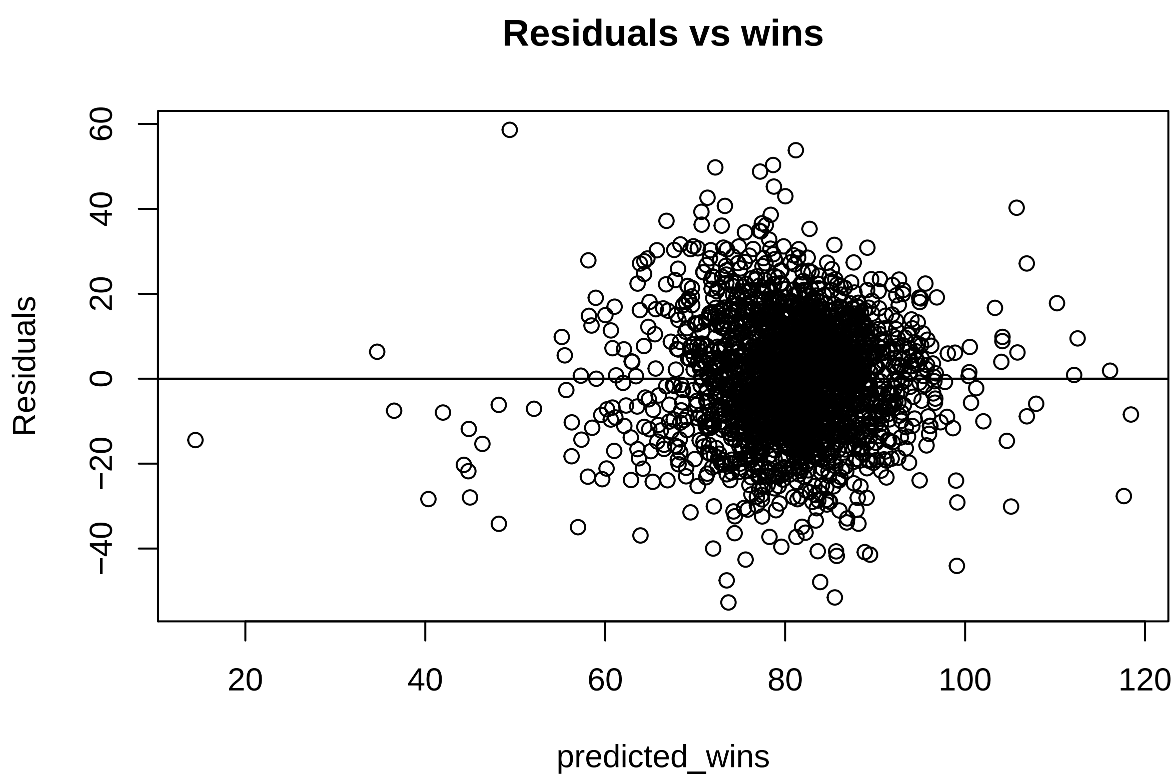
```
## [1] 18372.57
```

```
BIC(m1)
```

```
## [1] 18418.41
```

Model residual data analysis

```r
test<-predict(m1,train_data,type="response")
score<-predict(m1,train_data,type="response")
actual<-train_data$TARGET_WIN


# Analysis of residual with predicted values (residual plot)

res.m1 <- resid(m1)


plot(score, res.m1,  ylab="Residuals", xlab="predicted_wins" ,main="Residuals vs wins")
abline(0, 0)

# Analysis of residual with actual values (not useed verry frequently)

#plot(train_data$TARGET_WIN, res.m1,  ylab="Residuals", xlab="target_wins" ,main="Residuals vs wins")
abline(0, 0)
```



Analysis of RMSE

```r
# Getting RMSE: Typically this measure is used for measuring the absolute quantity of acuracy
```

```r
rmse<-(mean((score-actual)^2))^0.5
rmse
```

```
## [1] 13.64946
```

```r
# Realtive SE to measure accuracy with respect to the baseline ratio of MSE/MSE baseline is used
mu<-mean(actual)
rse<-(mean((score-actual)^2))/(mean((mu-actual)^2))
rse
```

```
## [1] 0.751176
```