Sports Analytics: A case study in baseball

Final project report

By:

Olusegun Stephen Omotunde

Contents

1. Introduction

1.1. Motivation

Baseball is the second highest-paid sport in the world according to a ranking done by Rulesofsport.com in 2021, following Basketball closely. The American League and the National League are the two major professional leagues in the United States. There is one major difference between the American and National Leagues, and that is the American League has a specific hitter known as the "Designated Hitter" who does not play in the field but instead bats for the pitcher. There is no such person in the National League. The placement of a player in both of these leagues is very important since both are crucial to the team's success.

The majority of Americans love baseball, so there is a high turnover in the industry. The players, and especially owners of sports teams, are interested in estimating what influences salaries and placements of players in American (A) and National (N) leagues. They are willing to use these estimates in their negotiations for their next contracts.

1.2. Objectives

To identify the most significant variables and models that can be used to predict the salary of baseball players.

1.3. Hitters dataset

There are 20 variables and 322 observations in the Hitters dataset from the ISLR package. Twenty variables are used in the analysis: 18 have been used as predictors and two as dependents. Salary is a continuous dependent, whereas New League is a categorical dependent. There are 18 independent variables, 16 of which are continuous and two of which are categorical (League and Division).

Regression models are estimated using the entire set of independent variables. The dependent variable for the New League is dropped from regression models. However, the league variable is not used for the classification models, and 17 other independent variables are used instead. The salary variable is not used in this model.

2. Methods

2.1. Full Model

At the first step, I fit a multiple linear regression model containing all regressors (all independent variables).

Y = 0+1*x1+ ... + k*xk+   k= number of regressors

Here, the response variable (Y) is Salary that the plan is to model or predict.

Regressor variables (x) are used to evaluate their influence on the amount of Salary.

Regression Coefficients (k) show how regressors impact the response.

The error (*) represents dependent variables that cannot be explained by the regression.

2.2. Assumptions of Linearity

There are four assumptions for linear regression.

• The relationship between response y and the regressors is linear, at least approximately

• The random error term has zero mean and constant variance

• The errors are uncorrelated

• The errors are normally distributed

For the full model, I check three of these assumptions (1,3,4)

2.3. Multicollinearity

One of the problems with fitted models is multicollinearity. A regressor can have a strong correlation with more than one variable and reduce the accuracy of the model. In order to check the multicollinearity, I measure the variance inflation factor (VIF). The threshold for VIF is 4. A VIF greater than 4 indicates multicollinearity in the model. If the VIF was greater than 10, the model suffers from severe multicollinearity.

2.4. Pearson Correlation Coefficient

By using VIF I just know whether the model suffers from multicollinearity or not. To calculate the correlation between regressors I evaluate the Pearson Correlation Coefficient. This value is between 1 and -1 and shows how regressors are correlated to each other.

Pearson Correlation Coefficient = 0 ⇒ No correlation

Pearson Correlation Coefficient 1 ⇒ Highly positively correlated

Pearson Correlation Coefficient -1 ⇒ Highly negatively correlated

2.5. Variable Selection

2.5.1. Analysis of Variance (ANOVA)

In order to get rid of multicollinearity, I should drop some correlated variables from the model. To select which variable should be dropped, ANOVA type II is one of the best methods to use it. With ANOVA type II I can find out the amount of each variable contribution in the model. Consequently, variables with the highest contributions, highest SSR, can stay in the model and I can drop variables with the lowest contributions.

2.6. Identification of Influential Points

In the dataset, there are some observations that may influence the model noticeably. To find these points I use three methods to identify the potential influential points. These three methods are as follows:

- Difference in Fits (DFFITS):

  The ithdata point is considered to be influential if:

  DIFFITSi> 2 k+2n-k-2

- Cook's distance (cook's D):

  If cd > 1 I consider the point as an influential one.

- COVRATIO:

  The ithdata point is considered to be influential if:

$$cv>1+3(k+1)n$$

2.7. Transformation

There are some variables that do not have a linear relationship with the response. I need to perform a proper transformation to assure the linearity assumption is validated.

3. Results

3.1. Construction of the Final Model

3.1.1. Full Model

Before fitting a regression model I must handle missing values. I had 59 missing values in the Salary column which have been removed from the dataset. At the next step, I fit a multiple linear regression with 19 regressors. The equation of the full MLR is as follows:

Salary=

163.10359 - 1.97987*AtBat + 7.50077*Hits + 4.33088*HmRun - 2.37621*Runs-1.04496*RBI +

6.23129*Walks - 3.48905*Years - 0.17134*CAtBat + 0.13399*CHits - 0.17286*CHmRun +

1.45430*CRuns + 0.80771*CRBI - 0.81157*CWalks + 62.59942*LeagueN - 116.84925*DivisionW +

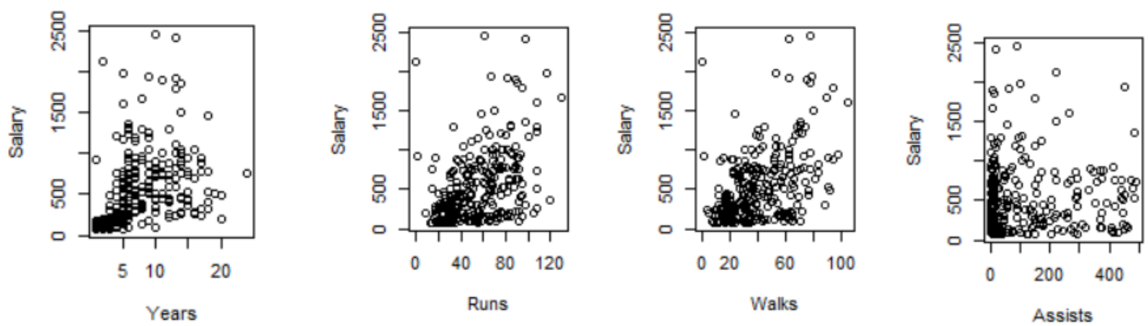
0.28189*PutOuts + 0.37107*Assists - 3.36076*Errors - 24.76233*NewLeagueN

In this model, the value of R-squared is 0.5461 and Adjusted R-squared has a value of 0.5106 (See Appendix A for the summary of the full model).

3.1.2. Checking Linearity Assumptions

3.1.2.1. Checking Linearity Between Regressors and Response

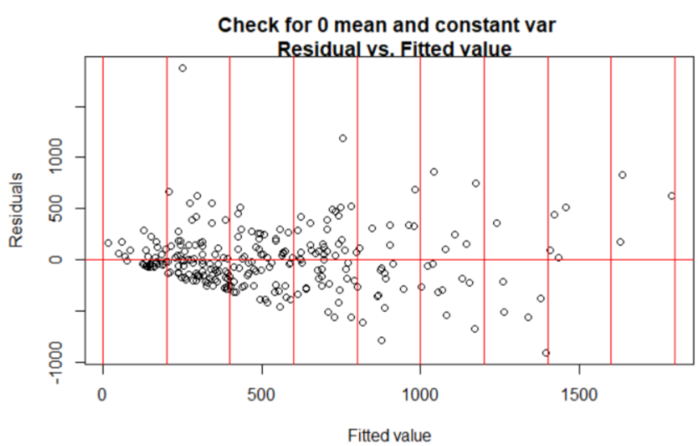
In order to check the linearity between regressors and response I plotted each regressor against response. There were some variables like Years, Runs, Walks, and Assists.



These plots show that there is not a linear association between these variables and Salary.

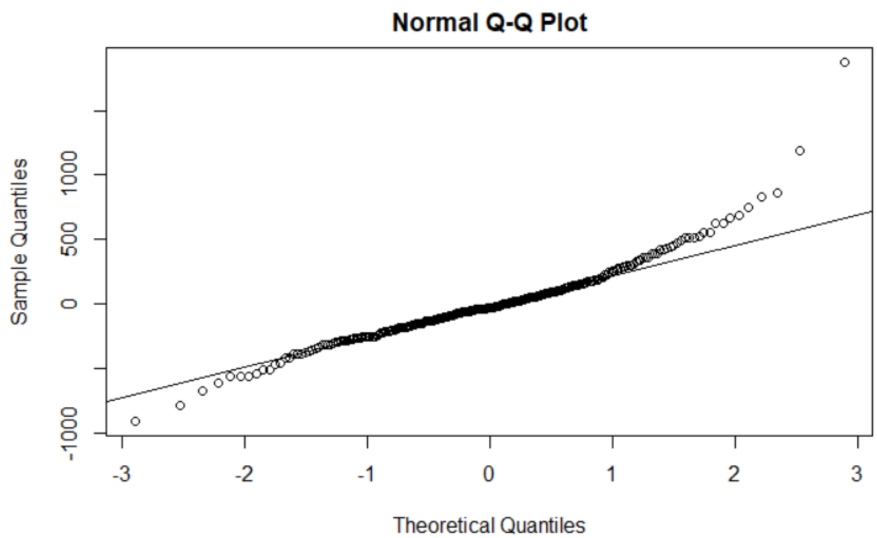
3.1.2.2. Checking Zero Mean and Constant Variance of Error Terms

After plotting residual against fitted value I realized this assumption is violated too. According to the figure, if I scan thin vertical stripes from left to right the vertical average of the residuals will change from strictly positive to strictly negative and then again to positive.



3.1.2.3. Checking for the assumption of normality of error terms

By using two methods I can find out whether error terms are distributed normally or not. First, I can use the normal probability plot (also called the q-q plot) for checking the assumption of normality of error terms. The Normal Q-Q plot shows that some of the points do not fall on the straight line, explaining that errors do not follow a normal distribution.



Also, I can use the Shapiro-Wilk test for testing normality. Based on the result of the Shapro-Wilk test, the test statistic is W = 0.92899 and the p-value is 6.602e-10 which is so small and the null hypothesis is rejected. So, I conclude that the error terms are not distributed normally.

```
	Shapiro-Wilk normality test

data:  mlr1$residuals
W = 0.92899, p-value = 6.602e-10
```

To handle linearity assumptions, I check the multicollinearity of the model and then use selection variables and transformation techniques to improve the model.

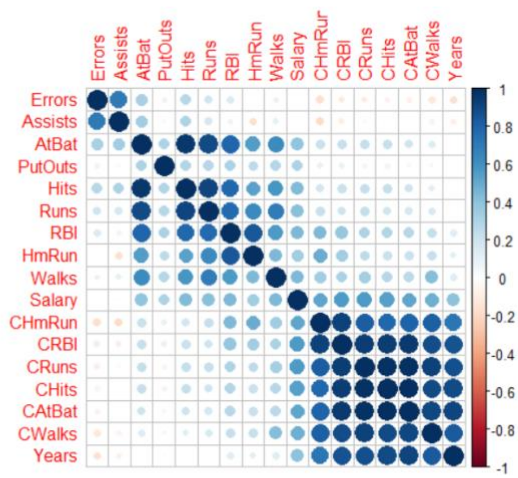
3.1.3. Multicollinearity

By using the VIF function I can find out whether the model suffers from multicollinearity or not. The VIF results show that the variance inflation factor is high for some variables. The VIF value for CHits is more than 500 which is so high. This result explains that the model suffers from severe multicollinearity.

```
     AtBat       Hits      HmRun       Runs        RBI      Walks      Years     CAtBat      CHits     CHmRun 
 22.944366  30.281255   7.758668  15.246418  11.921715   4.148712   9.313280 251.561160 502.954289  46.488462 
     CRuns       CRBI     CWalks     League   Division    PutOuts    Assists     Errors  NewLeague 
162.520810 131.965858  19.744105   4.134115   1.075398   1.236317   2.709341   2.214543   4.099063 
```

3.1.4. Pearson Correlation Coefficient

To find out the correlation between variables I check the Pearson Correlation Coefficient. The correlation coefficients show that some variables are highly correlated together.



3.1.5. Analysis of Variance (ANOVA)

To get rid of the multicollinearity I should drop some variables from the model. Based on ANOVA type II, variables with the lowest incremental sum of square are dropped from the model. Hmrun, Runs, RBI, Years, CHits, CHmRun, Assists, Errors, and NewLeague removed from the model (See ANOVA type II result in the Appendix).

3.1.6. Transformation

After selecting variables with the highest contribution to the model I realized that some variables do not have a linear association with the response. So, I use transformation to figure out this problem. By using Box-Cox I found a proper transformation for Salary which was 0.25. Then I used the log transformation for CRBI, since this variable had an exponential association with the fourth root of Salary (Salary^0.25).

3.1.7. Identification of Influential Points

In this project, three methods were used to identify influential points. I used the Deffits, cooks.distance, and covratio methods and then dropped common influential points. 8 observations were removed from the dataset.

3.1.8. Final Model

After removing influential points I fitted the new model with 10 variables. According to the summary, there is a big improvement in the model and adjusted R-Square increased to 0.7528 (See the appendix).

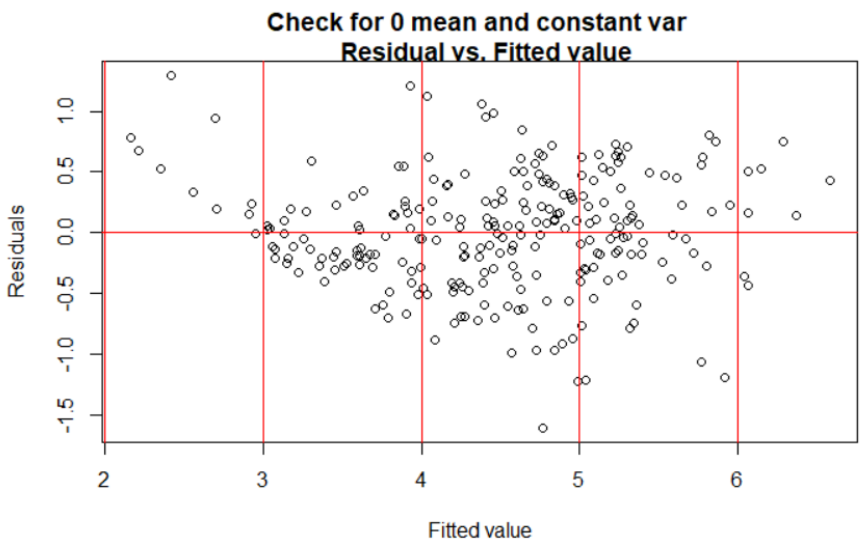
Equation of fitted model

$$\widehat{Salary}^{0.25} = \\ 0.4513324 - 0.0022757 * AtBat + 0.0086190 * Hits \\ +0.0071524 * Walks - 0.0003134 * CAtBat + 0.0027842 * CRuns \\ +0.7165028 * log(CRBI) - 0.0012647 * CWalks \\ +0.1309695 * LeagueN - 0.0893950 * DivisionW + 0.0003267 * PutOuts$$

3.2. Linearity Assumptions for Final Model

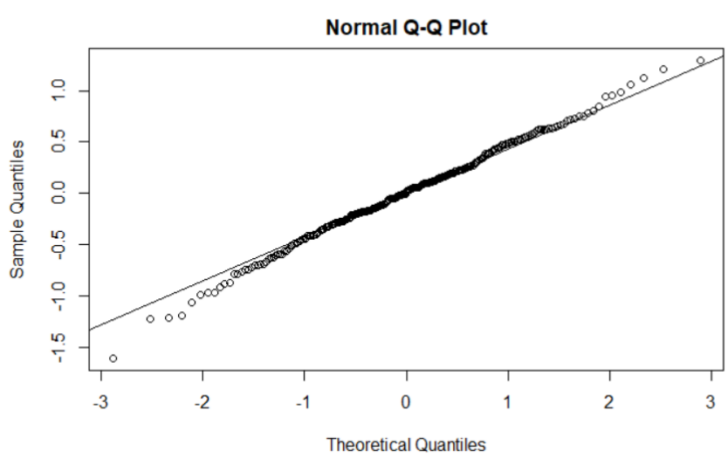
3.2.1. Assumption of Zero Mean and Constant Variance of Error Terms

If I check the residual vs. fitted value, I realize that there is an improvement in the plot and points are distributed more symmetrically on both sides of the reference line.



3.2.2. Assumption of Normality in the Distribution of Error Terms

The Noramal Q-Q plot for the final model shows most of the points fall on the straight line and the error terms are distributed normally.



Also, the p-value for the Shapiro-Wilk test is 0.7332, explaining that I cannot reject the null hypothesis and the error terms follow a normal distribution.

```
        Shapiro-Wilk normality test

data:  mlr5$residuals
W = 0.99585, p-value = 0.7332
```

3.3. Prediction

In order to check the accuracy of the model, I predicted the Salary for two observations. First observation is the median. I put the median of regressors in the model and compare the predicted value with the real median value.

AtBat=413, Hits=103, Walks = 37, CAtBat=1928 , CRuns=274,

CRBI=226, CWalks=172, League= "A", Division= "W", PutOuts=222,

Real value Salary = 416

Predicted Salary = 400

Error = 416-400 = 16 ⇒ ((416-400) / 416) * 100 ~ **4%**

Second observation was a player that was randomly chosen from the dataset.

Rance Mulliniks:

AtBat=348, Hits=90, Walks = 43, CAtBat=2288 , CRuns=295,

CRBI=273, CWalks=269, League= "A", Division= "E", PutOuts=450

Real value Salary = 450

Predicted Salary = 476.8

Error = 476.8 - 450= 16.8 $\Rightarrow$ ((476.8 - 450) / 450) * 100 ~ **4%**

As it is clear, the model has fitted well and has a high level of accuracy with a low percentage of error.

4. Conclusion

4.1. Conclusion

The full model suffered from severe multicollinearity which has been reduced by the variable selection method. Using the transformation technique and then removing influential points improved the model adjusted R-square from 0.5106 to 0.7528. Based on the summary of the final model, Log(CRBI) has the most contribution in the final model. The final model has a reasonable accuracy for predicting Salary and also has 10 variables instead of 19 variables. So, the interpretation of the model will be easier than the full model.

4.2. Limitations of My Research

The Hitters dataset was not complete and there were other variables that were not included in the model. As a result of that, the final model just explains 75% of the Salary variability.

Appendix: Hitters dataset and R codes

Hitters {ISLR} R Documentation Baseball Data
Description: Major League Baseball Data from the 1986 and 1987 seasons.

Variable	Description
AtBat	Number of times at bat in 1986
Hits	Number of hits in 1986
HmRun	Number of home runs in 1986
Runs	Number of runs in 1986
RBI	Number of runs batted in in 1986
Walks	Number of walks in 1986
Years	Number of years in the major leagues
CAtBat	Number of times at bat during his career
CHits	Number of hits during his career
CHmRun	Number of home runs during his career
CRuns	Number of runs during his career
CRBI	Number of runs batted in during his career
CWalks	Number of walks during his career
League	A factor with levels A and N indicating player's league at the end of 1986
Division	A factor with levels E and W indicating player's division at the end of 1986
PutOuts	Number of put outs in 1986
Assists	Number of assists in 1986
Errors	Number of errors in 1986
Salary	1987 annual salary on opening day in thousands of dollars
NewLeague	A factor with levels A and N indicating player's league at the beginning of 1987

'data.frame': **322 obs. of 20 variables**:
$ AtBat : int 293 315 479 496 321 594 185 298 323 401 ...
$ Hits : int 66 81 130 141 87 169 37 73 81 92 ...
$ HmRun : int 1 7 18 20 10 4 1 0 6 17 ...
$ Runs : int 30 24 66 65 39 74 23 24 26 49 ...
$ RBI : int 29 38 72 78 42 51 8 24 32 66 ...
$ Walks : int 14 39 76 37 30 35 21 7 8 65 ...
$ Years : int 1 14 3 11 2 11 2 3 2 13 ...
$ CAtBat : int 293 3449 1624 5628 396 4408 214 509 341 5206 ...
$ CHits : int 66 835 457 1575 101 1133 42 108 86 1332 ...
$ CHmRun : int 1 69 63 225 12 19 1 0 6 253 ...
$ CRuns : int 30 321 224 828 48 501 30 41 32 784 ...
$ CRBI : int 29 414 266 838 46 336 9 37 34 890 ...
$ CWalks : int 14 375 263 354 33 194 24 12 8 866 ...
$ League : Factor w/ 2 levels "A","N": 1 2 1 2 2 1 2 1 2 1 ...
$ Division : Factor w/ 2 levels "E","W": 1 2 2 1 1 2 1 2 2 1 ...
$ PutOuts : int 446 632 880 200 805 282 76 121 143 0 ...
$ Assists : int 33 43 82 11 40 421 127 283 290 0 ...
$ Errors : int 20 10 14 3 4 25 7 9 19 0 ...
$ **Salary** : num NA 475 480 500 91.5 750 70 100 75 1100 ...
$ **NewLeague**: Factor w/ 2 levels "A","N": 1 2 1 2 2 1 1 1 2 1 ...

Project_vesion6

Group 20: Iman, Mostafa, Olusegun, Virtus

11/30/2021

Table of Contents

Meet Data: "Hitters"

```
rm(list = ls())
library(ISLR)
df1 = Hitters
df1 = df1[complete.cases(df1),]

Table = data.frame(table(df1$League), table(df1$Division), table(df1$NewLeagu
e))
names(Table)[1] = "League"
names(Table)[3] = "Division"
```

```r
names(Table)[5] = "NewLeague"
#View(Table)
Table
```

```
##   League Freq Division Freq.1 NewLeague Freq.2
## 1      A  139        E    129         A    141
## 2      N  124        W    134         N    122
```

```r
str(df1)
```

```
## 'data.frame':    263 obs. of  20 variables:
##  $ AtBat    : int  315 479 496 321 594 185 298 323 401 574 ...
##  $ Hits     : int  81 130 141 87 169 37 73 81 92 159 ...
##  $ HmRun    : int  7 18 20 10 4 1 0 6 17 21 ...
##  $ Runs     : int  24 66 65 39 74 23 24 26 49 107 ...
##  $ RBI      : int  38 72 78 42 51 8 24 32 66 75 ...
##  $ Walks    : int  39 76 37 30 35 21 7 8 65 59 ...
##  $ Years    : int  14 3 11 2 11 2 3 2 13 10 ...
##  $ CAtBat   : int  3449 1624 5628 396 4408 214 509 341 5206 4631 ...
##  $ CHits    : int  835 457 1575 101 1133 42 108 86 1332 1300 ...
##  $ CHmRun   : int  69 63 225 12 19 1 0 6 253 90 ...
##  $ CRuns    : int  321 224 828 48 501 30 41 32 784 702 ...
##  $ CRBI     : int  414 266 838 46 336 9 37 34 890 504 ...
##  $ CWalks   : int  375 263 354 33 194 24 12 8 866 488 ...
##  $ League   : Factor w/ 2 levels "A","N": 2 1 2 2 1 2 1 2 1 1 ...
##  $ Division : Factor w/ 2 levels "E","W": 2 2 1 1 2 1 2 2 1 1 ...
##  $ PutOuts  : int  632 880 200 805 282 76 121 143 0 238 ...
##  $ Assists  : int  43 82 11 40 421 127 283 290 0 445 ...
##  $ Errors   : int  10 14 3 4 25 7 9 19 0 22 ...
##  $ Salary   : num  475 480 500 91.5 750 ...
##  $ NewLeague: Factor w/ 2 levels "A","N": 2 1 2 2 1 1 1 2 1 1 ...
```

- I have 16 integer variables
- Also, I have 3 categorical variables.

Full MLR model

```r
mlr1 = lm(Salary ~ . , data = df1)
summary(mlr1)
```

```
##
## Call:
## lm(formula = Salary ~ ., data = df1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -907.62 -178.35  -31.11  139.09 1877.04
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  163.10359   90.77854   1.797 0.073622 .
## AtBat         -1.97987    0.63398  -3.123 0.002008 **
```

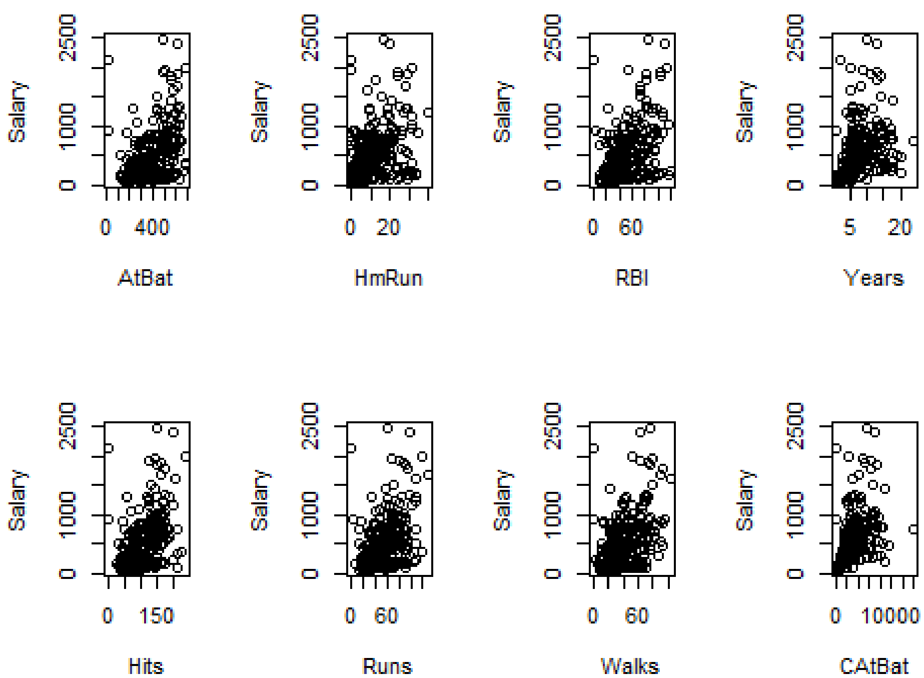
```
## Hits           7.50077    2.37753   3.155 0.001808 **
## HmRun          4.33088    6.20145   0.698 0.485616
## Runs          -2.37621    2.98076  -0.797 0.426122
## RBI           -1.04496    2.60088  -0.402 0.688204
## Walks          6.23129    1.82850   3.408 0.000766 ***
## Years         -3.48905   12.41219  -0.281 0.778874
## CAtBat        -0.17134    0.13524  -1.267 0.206380
## CHits          0.13399    0.67455   0.199 0.842713
## CHmRun        -0.17286    1.61724  -0.107 0.914967
## CRuns          1.45430    0.75046   1.938 0.053795 .
## CRBI           0.80771    0.69262   1.166 0.244691
## CWalks        -0.81157    0.32808  -2.474 0.014057 *
## LeagueN       62.59942   79.26140   0.790 0.430424
## DivisionW   -116.84925   40.36695  -2.895 0.004141 **
## PutOuts        0.28189    0.07744   3.640 0.000333 ***
## Assists        0.37107    0.22120   1.678 0.094723 .
## Errors        -3.36076    4.39163  -0.765 0.444857
## NewLeagueN   -24.76233   79.00263  -0.313 0.754218
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 315.6 on 243 degrees of freedom
## Multiple R-squared:  0.5461, Adjusted R-squared:  0.5106
## F-statistic: 15.39 on 19 and 243 DF,  p-value: < 2.2e-16
```

Check linearity assumptions

scatter plots for response variable vs. the regressors

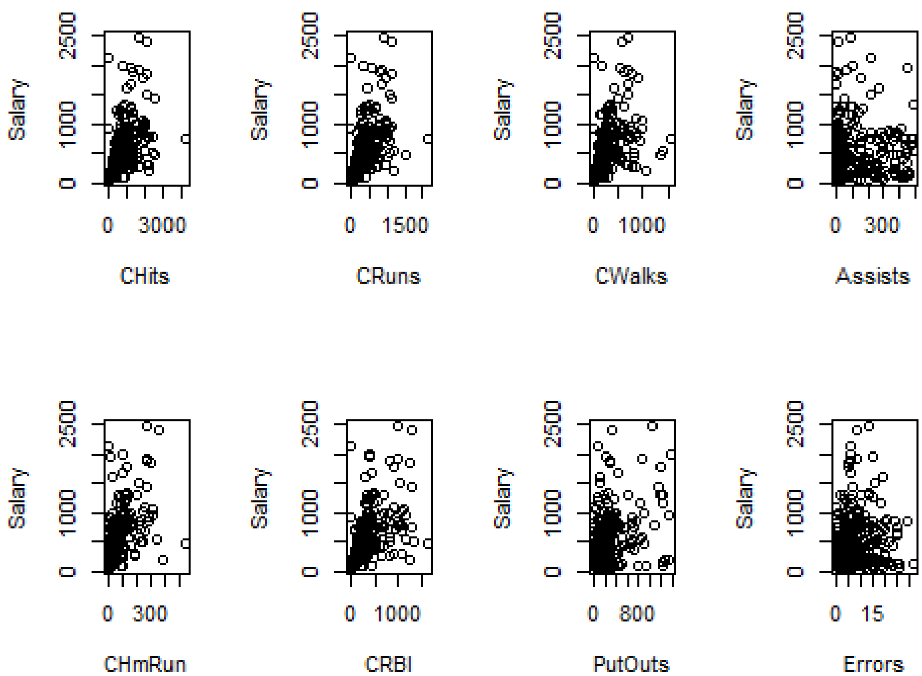
```
#pairs(df1)
# pairs(df1[ , c(17, 1:4)])
# pairs(df1[ , c(17, 5:8)])
# pairs(df1[ , c(17, 9:12)])
# pairs(df1[ , c(17, 13:16)])

#
par(mfcol = c(2, 4))
plot(df1$AtBat, df1$Salary,
     xlab = "AtBat",
     ylab = "Salary")
plot(df1$Hits, df1$Salary,
     xlab = "Hits",
     ylab = "Salary")
plot(df1$HmRun, df1$Salary,
     xlab = "HmRun",
     ylab = "Salary")
plot(df1$Runs, df1$Salary,
     xlab = "Runs",
     ylab = "Salary")
plot(df1$RBI, df1$Salary,
     xlab = "RBI",
```

```
     ylab = "Salary")
plot(df1$Walks, df1$Salary,
     xlab = "Walks",
     ylab = "Salary")
plot(df1$Years, df1$Salary,
     xlab = "Years",
     ylab = "Salary")
plot(df1$CAtBat, df1$Salary,
     xlab = "CAtBat",
     ylab = "Salary")
```




```
plot(df1$CHits, df1$Salary,
     xlab = "CHits",
     ylab = "Salary")
plot(df1$CHmRun, df1$Salary,
     xlab = "CHmRun",
     ylab = "Salary")
plot(df1$CRuns, df1$Salary,
     xlab = "CRuns",
     ylab = "Salary")
plot(df1$CRBI, df1$Salary,
     xlab = "CRBI",
     ylab = "Salary")
plot(df1$CWalks, df1$Salary,
     xlab = "CWalks",
     ylab = "Salary")
plot(df1$PutOuts, df1$Salary,
```

```
     xlab = "PutOuts",
     ylab = "Salary")
plot(df1$Assists, df1$Salary,
     xlab = "Assists",
     ylab = "Salary")
plot(df1$Errors, df1$Salary,
     xlab = "Errors",
     ylab = "Salary")
```

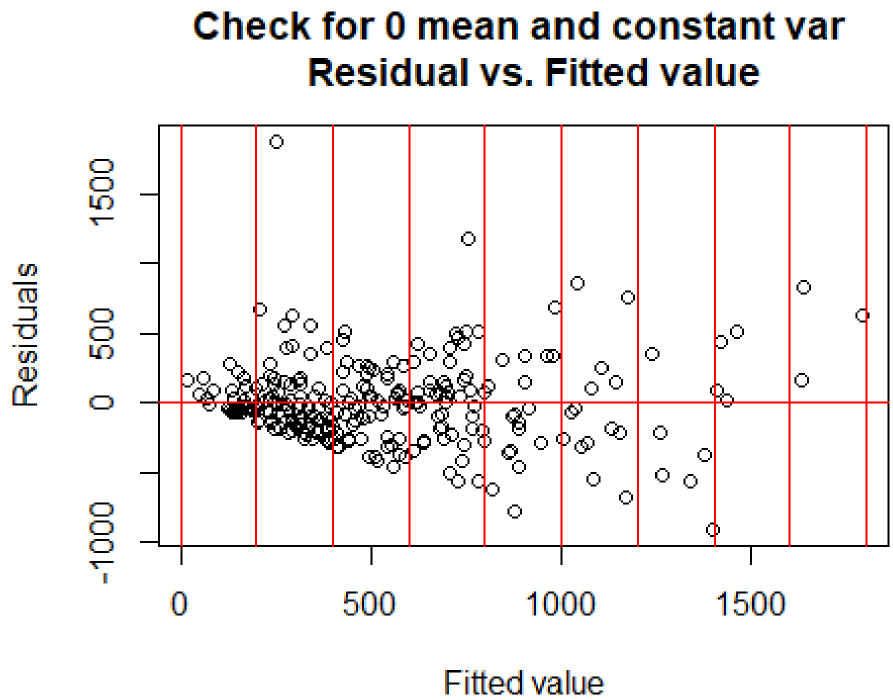


####
interpretation Variables like Years, Runs, Walks, and Assists do not have a linear relationship with the response (Salary)

Checking for zero mean and constant variance of error terms

Residual errors vs. the Fitted values

```
plot(mlr1$fitted.values, mlr1$residuals,
     xlab = "Fitted value",
     ylab = "Residuals",
     main = "Check for 0 mean and constant var \n  Residual vs. Fitted value"
)
list1 = seq(0,2000,200)
abline(h=0, v = list1, col="red")
```

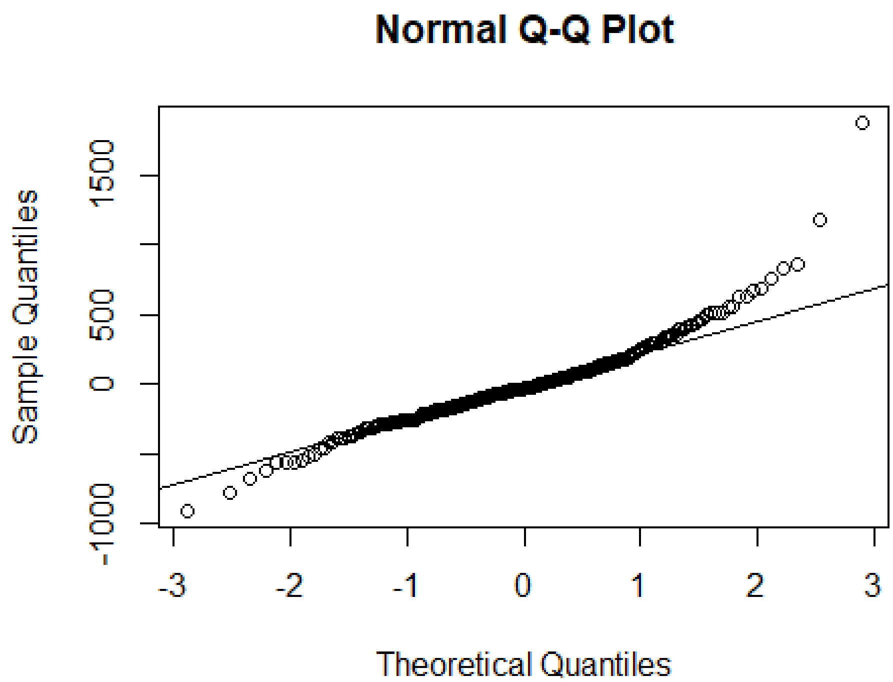


Comments

- If I imagine some vertical strips I realize that the average mean is not zero
- If I scan the plot from left to right that variance is not constant.

Normal Probability Plot

```
qqnorm(mlr1$residuals)
qqline(mlr1$residuals)
```



Comments
the dataset is not normal, there are some point that far away from from the straight line.

Perform Shapiro-Wilk test for normality

```
shapiro.test(mlr1$residuals)

##
##  Shapiro-Wilk normality test
##
## data:  mlr1$residuals
## W = 0.92899, p-value = 6.602e-10
```

Comments

According to the Shapiro-Wilk test, the p-value is 6.602e-10 which is less than 0.05. So I reject the null hypothesis and conclude that the dataset does not follow a normal distribution.

VIF

```
library(car)

## Loading required package: carData

vif(mlr1)

##     AtBat      Hits     HmRun      Runs       RBI     Walks      Yea
rs
##  22.944366  30.281255  7.758668  15.246418  11.921715  4.148712  9.3132
80
```

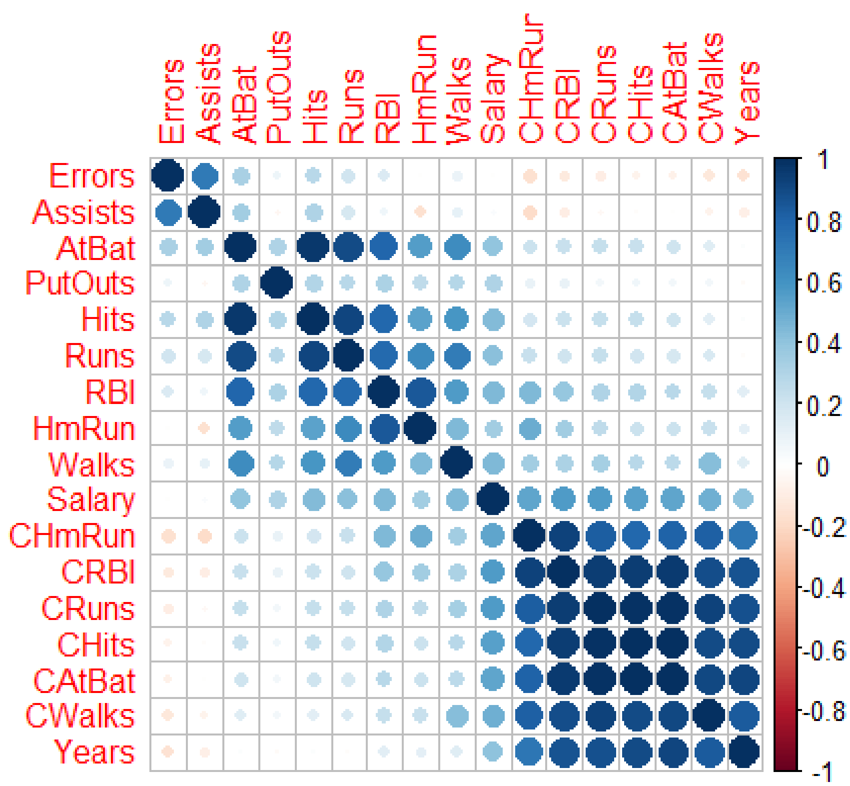
```
##     CAtBat      CHits     CHmRun      CRuns       CRBI     CWalks      Leag
ue 
## 251.561160 502.954289  46.488462 162.520810 131.965858  19.744105   4.1341
15 
##   Division    PutOuts    Assists     Errors  NewLeague 
##   1.075398   1.236317   2.709341   2.214543   4.099063
```

Comments

The model suffers from high multicollinearity. there are some variables that their VIF value is more than 250. As I know the threshold for VIF is 4.

Pearson Correlation

```
library(corrplot)

## corrplot 0.90 loaded

df_1 = df1[ , -c(14,15,20)]
M = cor(df_1)
#corrplot.mixed(M, order = 'AOE')
corrplot(M, order = 'AOE')
```



Comments

- ATBat has a strong correlation with Hits, Runs, RBI, and Walk.
- there is a significant correlation between RBI and AtBat,Hits, Run and HmRun.
- Variables like CRBI, CRuns, Chits, CAtBat, CWalk, and years have a strong correlation together.

Variable Selection

Anova Type II

```
Anova(mlr1, type = 2)
```

```
## Anova Table (Type II tests)
##
## Response: Salary
##              Sum Sq  Df F value    Pr(>F)
## AtBat        971288   1  9.7527 0.0020077 **
## Hits         991242   1  9.9531 0.0018082 **
## HmRun         48572   1  0.4877 0.4856158
## Runs          63291   1  0.6355 0.4261225
## RBI           16076   1  0.1614 0.6882042
## Walks       1156606   1 11.6135 0.0007662 ***
## Years          7869   1  0.0790 0.7788736
## CAtBat       159864   1  1.6052 0.2063804
## CHits          3930   1  0.0395 0.8427129
## CHmRun         1138   1  0.0114 0.9149671
## CRuns        374007   1  3.7554 0.0537951 .
## CRBI         135439   1  1.3599 0.2446905
## CWalks       609408   1  6.1191 0.0140574 *
## League        62121   1  0.6238 0.4304236
## Division     834491   1  8.3791 0.0041408 **
## PutOuts     1319628   1 13.2504 0.0003329 ***
## Assists      280263   1  2.8141 0.0947232 .
## Errors        58324   1  0.5856 0.4448566
## NewLeague      9784   1  0.0982 0.7542178
## Residuals  24200700 243
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Droping Some Variables manually

I drop variables with the lowest contribution in the model

```
rm(list = ls())
df1 = Hitters
df1 = df1[complete.cases(df1),]
df2 = df1 [ , -c(3,4,5,7,9,10,17,18,20)]
mlr2 = lm(Salary ~ . , data = df2)
summary(mlr2)
```

```
##
## Call:
## lm(formula = Salary ~ ., data = df2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -901.64 -178.98  -26.72  130.38 1967.02
##
```

```
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  116.85209   70.73807   1.652 0.099801 .
## AtBat         -1.90375    0.52301  -3.640 0.000331 ***
## Hits           6.67814    1.64590   4.057 6.62e-05 ***
## Walks          5.39152    1.59087   3.389 0.000814 ***
## CAtBat        -0.11138    0.05389  -2.067 0.039759 *
## CRuns          1.31303    0.38511   3.409 0.000758 ***
## CRBI           0.68061    0.20130   3.381 0.000837 ***
## CWalks        -0.77199    0.26289  -2.937 0.003626 **
## LeagueN       48.82127   39.97367   1.221 0.223100
## DivisionW   -113.78174   39.33427  -2.893 0.004154 **
## PutOuts        0.26984    0.07417   3.638 0.000333 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 312.9 on 252 degrees of freedom
## Multiple R-squared:  0.5374, Adjusted R-squared:  0.519
## F-statistic: 29.27 on 10 and 252 DF,  p-value: < 2.2e-16
```

The adjusted r-square increased from 0.5106 to 0.519 but did not change significantly

Testing whether the present of categorical variables have significant impact on Salary or not

```
linearHypothesis(mlr2, c("DivisionW = 0"))

## Linear hypothesis test
##
## Hypothesis:
## DivisionW = 0
##
## Model 1: restricted model
## Model 2: Salary ~ AtBat + Hits + Walks + CAtBat + CRuns + CRBI + CWalks +
##     League + Division + PutOuts
##
##   Res.Df      RSS Df Sum of Sq      F   Pr(>F)
## 1    253 25487134
## 2    252 24668034  1    819100 8.3676 0.004154 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

linearHypothesis(mlr2, c("LeagueN  = 0"))

## Linear hypothesis test
##
## Hypothesis:
## LeagueN = 0
##
## Model 1: restricted model
## Model 2: Salary ~ AtBat + Hits + Walks + CAtBat + CRuns + CRBI + CWalks +
```

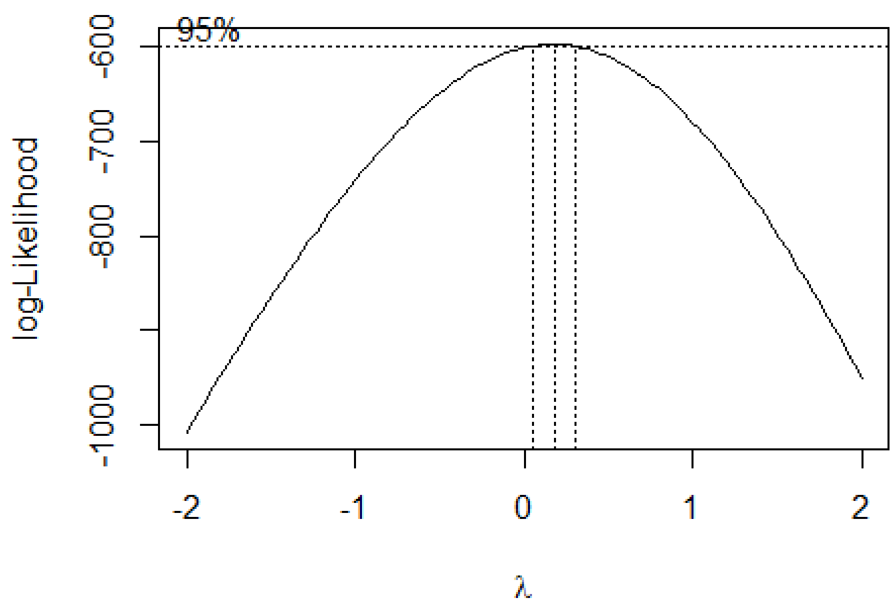
```
##     League + Division + PutOuts
##
##   Res.Df      RSS Df Sum of Sq      F Pr(>F)
## 1    253 24814051
## 2    252 24668034  1    146017 1.4917 0.2231
```

The p_value for both categorical variables is more than 0.05. I conclude that these variables do not have a significant impact on the response (Salary)

Transformation

Using box-cox

```
library(MASS)
boxcox(mlr2)
```




```
mlr3 <- lm((Salary)^(0.25) ~ ., data = df2)
summary(mlr3)

##
## Call:
## lm(formula = (Salary)^(0.25) ~ ., data = df2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.2502 -0.4904  0.0421  0.4258  3.3671
##
## Coefficients:
```

```
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.3458623  0.1519300  22.022  < 2e-16 ***
## AtBat       -0.0034607  0.0011233  -3.081 0.002293 **
## Hits         0.0136414  0.0035350   3.859 0.000145 ***
## Walks        0.0115757  0.0034168   3.388 0.000817 ***
## CAtBat       0.0001099  0.0001157   0.950 0.343234
## CRuns        0.0011782  0.0008271   1.424 0.155557
## CRBI         0.0005637  0.0004323   1.304 0.193445
## CWalks      -0.0013025  0.0005646  -2.307 0.021878 *
## LeagueN      0.0935701  0.0858547   1.090 0.276813
## DivisionW   -0.2131271  0.0844815  -2.523 0.012260 *
## PutOuts      0.0003866  0.0001593   2.427 0.015918 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.672 on 252 degrees of freedom
## Multiple R-squared:  0.5498, Adjusted R-squared:  0.5319
## F-statistic: 30.77 on 10 and 252 DF,  p-value: < 2.2e-16
```

Transformation

I used power transformation p=0.25 for response and log transformation for `CRBI` variable

```
mlr4 = lm(Salary^0.25 ~ AtBat + Hits + Walks + CAtBat + CRuns + log(CRBI) +
    CWalks + League + Division + PutOuts   , data = df2)
summary(mlr4)
```

```
##
## Call:
## lm(formula = Salary^0.25 ~ AtBat + Hits + Walks + CAtBat + CRuns +
##     log(CRBI) + CWalks + League + Division + PutOuts, data = df2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0637 -0.3792 -0.0196  0.3219  4.1144
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.6332048  0.3034827   5.382 1.69e-07 ***
## AtBat       -0.0031707  0.0010488  -3.023  0.00276 **
## Hits         0.0101516  0.0033438   3.036  0.00265 **
## Walks        0.0093572  0.0032074   2.917  0.00385 **
## CAtBat      -0.0001883  0.0001152  -1.635  0.10323
## CRuns        0.0026005  0.0007971   3.263  0.00126 **
## log(CRBI)    0.4764034  0.0756917   6.294 1.36e-09 ***
## CWalks      -0.0014534  0.0005256  -2.765  0.00611 **
## LeagueN      0.1349779  0.0802378   1.682  0.09376 .
## DivisionW   -0.1631771  0.0791438  -2.062  0.04025 *
## PutOuts      0.0004230  0.0001474   2.869  0.00447 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.6268 on 252 degrees of freedom
## Multiple R-squared:  0.6083, Adjusted R-squared:  0.5928
## F-statistic: 39.14 on 10 and 252 DF,  p-value: < 2.2e-16
```

The value of adjusted R-square increased significantly to 0.5928.

removing influential points

```
k= ncol(df2)-1
n= nrow(df2)

which(abs(dffits(mlr4)) > 2*sqrt((k+2)/(n-k-2))) #influential if abs(dffits)
> 2*sqrt((k+2)/(n-k-2))

##    -Bill Buckner   -Darrell Evans     -Glenn Davis   -Graig Nettles
##               21               55               85               92
## -Jeffrey Leonard    -Jim Sundberg    -Mike Schmidt     -Ozzie Smith
##              120              133              173              183
##       -Pete Rose  -Reggie Jackson   -Steve Balboni       -Sid Bream
##              189              201              220              222
##       -Steve Sax   -Terry Kennedy    -Wally Joyner
##              230              241              258

which(cooks.distance(mlr4) > 1) #influential if Cook's distance > 1

## named integer(0)

which(covratio(mlr4) > (1 + 3*(k+1)/n)) #influential if covratio > 1 + 3*(k+1
)/n OR covratio < 1 - 3*(k+1)/n

##    -Bill Buckner    -Chris Speier   -Darrell Evans   -Don Mattingly
##               21               47               55               62
##  -Darrell Porter -Keith Hernandez       -Pete Rose    -Steve Garvey
##               67              140              189              226
##  -Tony Fernandez      -Wade Boggs   -Willie Upshaw
##              234              256              262

which(covratio(mlr4) < (1 - 3*(k+1)/n))

## -Jeffrey Leonard    -Mike Schmidt     -Ozzie Smith       -Steve Sax
##              120              173              183              230
##   -Terry Kennedy
##              241

### Finding common influential points
intersect(which(abs(dffits(mlr4)) > 2*sqrt((k+2)/(n-k-2))), which(covratio(ml
r4) > (1 + 3*(k+1)/n)))

## [1]  21  55 189

intersect(which(abs(dffits(mlr4)) > 2*sqrt((k+2)/(n-k-2))), which(covratio(ml
r4) < (1 - 3*(k+1)/n)))
```

```
## [1] 120 173 183 230 241
```

Removing Influential Points

```
df3 = df2[ -c(21,55,120,173,183,189,230,241), ]

mlr5 = lm(Salary^0.25 ~ AtBat + Hits + Walks + CAtBat + CRuns + log(CRBI) +
    CWalks + League + Division + PutOuts   , data = df3)
summary(mlr5)

##
## Call:
## lm(formula = Salary^0.25 ~ AtBat + Hits + Walks + CAtBat + CRuns +
##     log(CRBI) + CWalks + League + Division + PutOuts, data = df3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.60490 -0.28408  0.02481  0.29432  1.29683
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.4513324  0.2650747   1.703 0.089905 .
## AtBat       -0.0022757  0.0008249  -2.759 0.006244 **
## Hits         0.0086190  0.0026425   3.262 0.001266 **
## Walks        0.0071524  0.0025329   2.824 0.005138 **
## CAtBat      -0.0003134  0.0000940  -3.333 0.000991 ***
## CRuns        0.0027842  0.0006403   4.348 2.02e-05 ***
## log(CRBI)    0.7165028  0.0659958  10.857  < 2e-16 ***
## CWalks      -0.0012647  0.0004395  -2.877 0.004365 **
## LeagueN      0.1309695  0.0626683   2.090 0.037664 *
## DivisionW   -0.0893950  0.0621505  -1.438 0.151613
## PutOuts      0.0003267  0.0001183   2.762 0.006173 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4802 on 244 degrees of freedom
## Multiple R-squared:  0.7625, Adjusted R-squared:  0.7528
## F-statistic: 78.34 on 10 and 244 DF,  p-value: < 2.2e-16
```

Adjusted R-Square

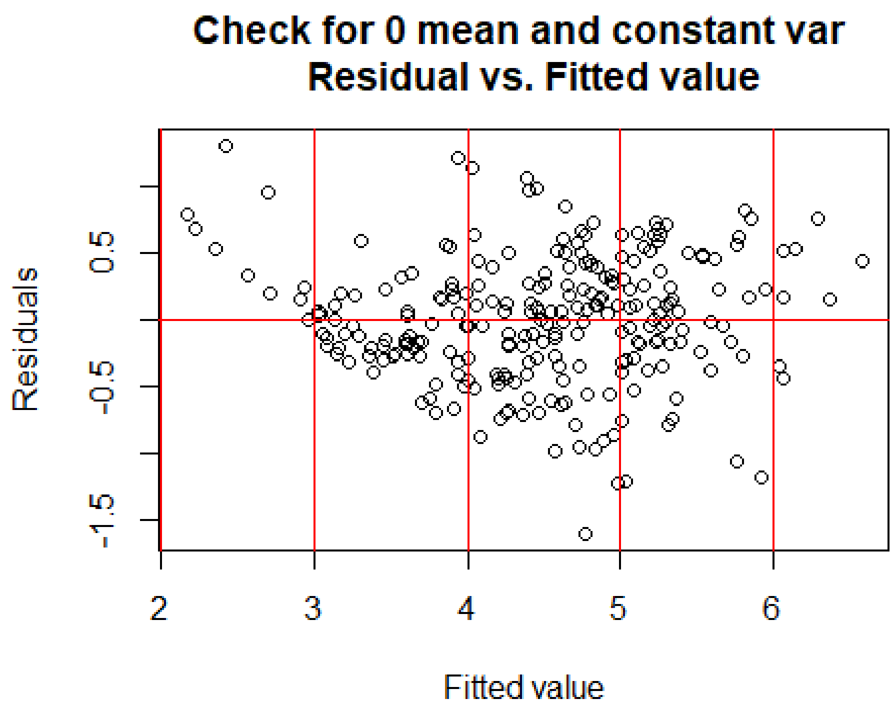
For the final model the value of adjusted R-square increased significantly to 0.7528.

Linearity Assumptions

Checking for zero mean and constant variance of error terms

```
plot(mlr5$fitted.values, mlr5$residuals,
     xlab = "Fitted value",
     ylab = "Residuals",
     main = "Check for 0 mean and constant var \n  Residual vs. Fitted value"
)
```

```
list1 = seq(2,7,1)
abline(h=0, v = list1, col="red")
```

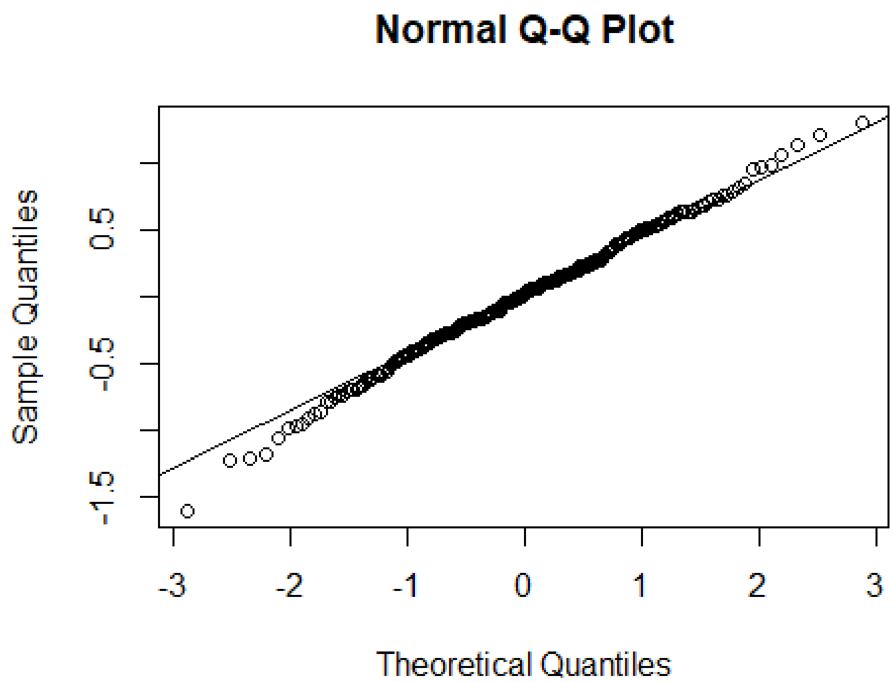


Interpretation

- If I compare the new plot with previous one I can realize that there is an improvement. Except the firs part of plot, I can see that the points are distributed more symmetrically than before.

Normal Probability Plot

```
qqnorm(mlr5$residuals)
qqline(mlr5$residuals)
```



####

Interpretation
As I can see most of the points fall on the straight line that indicates data distributed normaly.

ShapiroWilk test

```
shapiro.test(mlr5$residuals)

##
##  Shapiro-Wilk normality test
##
## data:  mlr5$residuals
## W = 0.99585, p-value = 0.7332
```

Interpretation

p-value = 0.7332 ==> I fail to reject the null ==> The errors foloow a normal distribution.

Final Model

Equation of fitted model

$$\widehat{Salary}^{0.25} = 0.4513324 - 0.0022757 * AtBat + 0.0086190 * Hits$$
$$+0.0071524 * Walks - 0.0003134 * CAtBat + 0.0027842 * CRuns$$
$$+0.7165028 * log(CRBI) - 0.0012647 * CWalks$$
$$+0.1309695 * LeagueN - 0.0893950 * DivisionW + 0.0003267 * PutOuts$$

ANOVA II

```
Anova(mlr5, type = 2)

## Anova Table (Type II tests)
##
## Response: Salary^0.25
##            Sum Sq  Df  F value    Pr(>F)
## AtBat       1.755   1   7.6101 0.0062437 **
## Hits        2.453   1  10.6387 0.0012656 **
## Walks       1.838   1   7.9738 0.0051379 **
## CAtBat      2.562   1  11.1115 0.0009913 ***
## CRuns       4.359   1  18.9073 2.017e-05 ***
## log(CRBI)  27.177   1 117.8700 < 2.2e-16 ***
## CWalks      1.909   1   8.2795 0.0043647 **
## League      1.007   1   4.3676 0.0376640 *
## Division    0.477   1   2.0689 0.1516125
## PutOuts     1.759   1   7.6312 0.0061734 **
## Residuals  56.258 244
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Predicting a Salary

```
#summary(df4)

# A random point (Midian points) # Median point for Salary = 416
x0 = data.frame(AtBat=413, Hits=103, Walks = 37, CAtBat=1928 , CRuns=274 , CR
BI=226 , CWalks=172 , League= "A", Division= "W",  PutOuts=222)
(predict(mlr5, x0, interval = "prediction", level = 0.95))^4

##        fit      lwr     upr
## 1 399.9629 153.0518 867.288

# predicting Rance Mulliniks Salary # real value is 450
#                  AtBat Hits Walks CAtBat CRuns CRBI CWalks League Division
PutOuts Salary
# Rance Mulliniks   348   90    43   2288   295  273    269      A        E
60    450
x1 = data.frame(AtBat=348, Hits=90, Walks = 43, CAtBat=2288 , CRuns=295 , CRB
I=273 , CWalks=269 , League= "A", Division= "E",  PutOuts=450)
#(predict(mlr5, x1, interval = "prediction", level = 0.95))^4
(predict(mlr5, x1))^4

##        1
## 476.8772
```

- For median point the predicted Salary is 400 which is so closed to real value, 416.
- Also, the predicted value for Rance Mulliniks, one of the observation, is 476 that is so closed to real value, 450
- I can conclude that the final model can predict Salary with a high precision.

Check significance of categorical variables

```
linearHypothesis(mlr5, c("DivisionW = 0"))

## Linear hypothesis test
##
## Hypothesis:
## DivisionW = 0
##
## Model 1: restricted model
## Model 2: Salary^0.25 ~ AtBat + Hits + Walks + CAtBat + CRuns + log(CRBI) +
##     CWalks + League + Division + PutOuts
##
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    245 56.735
## 2    244 56.258  1   0.47701 2.0689 0.1516

linearHypothesis(mlr5, c("LeagueN  = 0"))

## Linear hypothesis test
##
## Hypothesis:
## LeagueN = 0
##
## Model 1: restricted model
## Model 2: Salary^0.25 ~ AtBat + Hits + Walks + CAtBat + CRuns + log(CRBI) +
##     CWalks + League + Division + PutOuts
##
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1    245 57.265
## 2    244 56.258  1     1.007 4.3676 0.03766 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```