Hitters

Tahani Omer

12/12/2019

## Baseball-Data

Major League Baseball Data from the 1986 and 1987 seasons.We wish to predict a baseball playerâs Salary on the basis of various statistics associated with performance in the previous year.

## Packages required

We will Load required packages,and have a look at the “Hitters” data from “MASS” packages.

library(MASS)  
library(ISLR)  
library(leaps)  
  
names(Hitters)

## [1] "AtBat" "Hits" "HmRun" "Runs" "RBI"   
## [6] "Walks" "Years" "CAtBat" "CHits" "CHmRun"   
## [11] "CRuns" "CRBI" "CWalks" "League" "Division"   
## [16] "PutOuts" "Assists" "Errors" "Salary" "NewLeague"

## Exploratory data analysis

#summary statistics  
attach(Hitters)  
summary(Hitters)

## AtBat Hits HmRun Runs   
## Min. : 16.0 Min. : 1 Min. : 0.00 Min. : 0.00   
## 1st Qu.:255.2 1st Qu.: 64 1st Qu.: 4.00 1st Qu.: 30.25   
## Median :379.5 Median : 96 Median : 8.00 Median : 48.00   
## Mean :380.9 Mean :101 Mean :10.77 Mean : 50.91   
## 3rd Qu.:512.0 3rd Qu.:137 3rd Qu.:16.00 3rd Qu.: 69.00   
## Max. :687.0 Max. :238 Max. :40.00 Max. :130.00   
##   
## RBI Walks Years CAtBat   
## Min. : 0.00 Min. : 0.00 Min. : 1.000 Min. : 19.0   
## 1st Qu.: 28.00 1st Qu.: 22.00 1st Qu.: 4.000 1st Qu.: 816.8   
## Median : 44.00 Median : 35.00 Median : 6.000 Median : 1928.0   
## Mean : 48.03 Mean : 38.74 Mean : 7.444 Mean : 2648.7   
## 3rd Qu.: 64.75 3rd Qu.: 53.00 3rd Qu.:11.000 3rd Qu.: 3924.2   
## Max. :121.00 Max. :105.00 Max. :24.000 Max. :14053.0   
##   
## CHits CHmRun CRuns CRBI   
## Min. : 4.0 Min. : 0.00 Min. : 1.0 Min. : 0.00   
## 1st Qu.: 209.0 1st Qu.: 14.00 1st Qu.: 100.2 1st Qu.: 88.75   
## Median : 508.0 Median : 37.50 Median : 247.0 Median : 220.50   
## Mean : 717.6 Mean : 69.49 Mean : 358.8 Mean : 330.12   
## 3rd Qu.:1059.2 3rd Qu.: 90.00 3rd Qu.: 526.2 3rd Qu.: 426.25   
## Max. :4256.0 Max. :548.00 Max. :2165.0 Max. :1659.00   
##   
## CWalks League Division PutOuts Assists   
## Min. : 0.00 A:175 E:157 Min. : 0.0 Min. : 0.0   
## 1st Qu.: 67.25 N:147 W:165 1st Qu.: 109.2 1st Qu.: 7.0   
## Median : 170.50 Median : 212.0 Median : 39.5   
## Mean : 260.24 Mean : 288.9 Mean :106.9   
## 3rd Qu.: 339.25 3rd Qu.: 325.0 3rd Qu.:166.0   
## Max. :1566.00 Max. :1378.0 Max. :492.0   
##   
## Errors Salary NewLeague  
## Min. : 0.00 Min. : 67.5 A:176   
## 1st Qu.: 3.00 1st Qu.: 190.0 N:146   
## Median : 6.00 Median : 425.0   
## Mean : 8.04 Mean : 535.9   
## 3rd Qu.:11.00 3rd Qu.: 750.0   
## Max. :32.00 Max. :2460.0   
## NA's :59

str(Hitters)

## '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 ...

#Check for missing values  
sum(is.na(Hitters))

## [1] 59

# removes all of the rows that have missing values in any variable  
Hitters = na.omit(Hitters)  
dim(Hitters )

## [1] 263 20

sum(is.na(Hitters))

## [1] 0

## Variable Selection

We will do the Best subset technique, to come up with the best linear regression model for the dependent variable Salary.

## Best subset selection

We will use package “leaps” to evalute all best-subset model that contains a given number of predictors, where best is quantiï¬ed using RSS

library (leaps)  
regfit.full=regsubsets (Salaryâ¼.,Hitters)  
summary (regfit.full) #outputs the best set of variables for each model size

## Subset selection object  
## Call: regsubsets.formula(Salary ~ ., Hitters)  
## 19 Variables (and intercept)  
## Forced in Forced out  
## AtBat FALSE FALSE  
## Hits FALSE FALSE  
## HmRun FALSE FALSE  
## Runs FALSE FALSE  
## RBI FALSE FALSE  
## Walks FALSE FALSE  
## Years FALSE FALSE  
## CAtBat FALSE FALSE  
## CHits FALSE FALSE  
## CHmRun FALSE FALSE  
## CRuns FALSE FALSE  
## CRBI FALSE FALSE  
## CWalks FALSE FALSE  
## LeagueN FALSE FALSE  
## DivisionW FALSE FALSE  
## PutOuts FALSE FALSE  
## Assists FALSE FALSE  
## Errors FALSE FALSE  
## NewLeagueN FALSE FALSE  
## 1 subsets of each size up to 8  
## Selection Algorithm: exhaustive  
## AtBat Hits HmRun Runs RBI Walks Years CAtBat CHits CHmRun CRuns  
## 1 ( 1 ) " " " " " " " " " " " " " " " " " " " " " "   
## 2 ( 1 ) " " "\*" " " " " " " " " " " " " " " " " " "   
## 3 ( 1 ) " " "\*" " " " " " " " " " " " " " " " " " "   
## 4 ( 1 ) " " "\*" " " " " " " " " " " " " " " " " " "   
## 5 ( 1 ) "\*" "\*" " " " " " " " " " " " " " " " " " "   
## 6 ( 1 ) "\*" "\*" " " " " " " "\*" " " " " " " " " " "   
## 7 ( 1 ) " " "\*" " " " " " " "\*" " " "\*" "\*" "\*" " "   
## 8 ( 1 ) "\*" "\*" " " " " " " "\*" " " " " " " "\*" "\*"   
## CRBI CWalks LeagueN DivisionW PutOuts Assists Errors NewLeagueN  
## 1 ( 1 ) "\*" " " " " " " " " " " " " " "   
## 2 ( 1 ) "\*" " " " " " " " " " " " " " "   
## 3 ( 1 ) "\*" " " " " " " "\*" " " " " " "   
## 4 ( 1 ) "\*" " " " " "\*" "\*" " " " " " "   
## 5 ( 1 ) "\*" " " " " "\*" "\*" " " " " " "   
## 6 ( 1 ) "\*" " " " " "\*" "\*" " " " " " "   
## 7 ( 1 ) " " " " " " "\*" "\*" " " " " " "   
## 8 ( 1 ) " " "\*" " " "\*" "\*" " " " " " "

Comments:

this output indicates that the best two-variable model contains only Hits and CRBI. By default, but regsubsets() only reports results up to the best eight-variable model

# ï¬t up to a 19-variable model  
regfit.full=regsubsets (Salaryâ¼.,data=Hitters ,nvmax=19)   
reg.summary =summary (regfit.full)   
reg.summary

## Subset selection object  
## Call: regsubsets.formula(Salary ~ ., data = Hitters, nvmax = 19)  
## 19 Variables (and intercept)  
## Forced in Forced out  
## AtBat FALSE FALSE  
## Hits FALSE FALSE  
## HmRun FALSE FALSE  
## Runs FALSE FALSE  
## RBI FALSE FALSE  
## Walks FALSE FALSE  
## Years FALSE FALSE  
## CAtBat FALSE FALSE  
## CHits FALSE FALSE  
## CHmRun FALSE FALSE  
## CRuns FALSE FALSE  
## CRBI FALSE FALSE  
## CWalks FALSE FALSE  
## LeagueN FALSE FALSE  
## DivisionW FALSE FALSE  
## PutOuts FALSE FALSE  
## Assists FALSE FALSE  
## Errors FALSE FALSE  
## NewLeagueN FALSE FALSE  
## 1 subsets of each size up to 19  
## Selection Algorithm: exhaustive  
## AtBat Hits HmRun Runs RBI Walks Years CAtBat CHits CHmRun CRuns  
## 1 ( 1 ) " " " " " " " " " " " " " " " " " " " " " "   
## 2 ( 1 ) " " "\*" " " " " " " " " " " " " " " " " " "   
## 3 ( 1 ) " " "\*" " " " " " " " " " " " " " " " " " "   
## 4 ( 1 ) " " "\*" " " " " " " " " " " " " " " " " " "   
## 5 ( 1 ) "\*" "\*" " " " " " " " " " " " " " " " " " "   
## 6 ( 1 ) "\*" "\*" " " " " " " "\*" " " " " " " " " " "   
## 7 ( 1 ) " " "\*" " " " " " " "\*" " " "\*" "\*" "\*" " "   
## 8 ( 1 ) "\*" "\*" " " " " " " "\*" " " " " " " "\*" "\*"   
## 9 ( 1 ) "\*" "\*" " " " " " " "\*" " " "\*" " " " " "\*"   
## 10 ( 1 ) "\*" "\*" " " " " " " "\*" " " "\*" " " " " "\*"   
## 11 ( 1 ) "\*" "\*" " " " " " " "\*" " " "\*" " " " " "\*"   
## 12 ( 1 ) "\*" "\*" " " "\*" " " "\*" " " "\*" " " " " "\*"   
## 13 ( 1 ) "\*" "\*" " " "\*" " " "\*" " " "\*" " " " " "\*"   
## 14 ( 1 ) "\*" "\*" "\*" "\*" " " "\*" " " "\*" " " " " "\*"   
## 15 ( 1 ) "\*" "\*" "\*" "\*" " " "\*" " " "\*" "\*" " " "\*"   
## 16 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" " " "\*" "\*" " " "\*"   
## 17 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" " " "\*" "\*" " " "\*"   
## 18 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*" "\*" "\*" " " "\*"   
## 19 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*" "\*" "\*" "\*" "\*"   
## CRBI CWalks LeagueN DivisionW PutOuts Assists Errors NewLeagueN  
## 1 ( 1 ) "\*" " " " " " " " " " " " " " "   
## 2 ( 1 ) "\*" " " " " " " " " " " " " " "   
## 3 ( 1 ) "\*" " " " " " " "\*" " " " " " "   
## 4 ( 1 ) "\*" " " " " "\*" "\*" " " " " " "   
## 5 ( 1 ) "\*" " " " " "\*" "\*" " " " " " "   
## 6 ( 1 ) "\*" " " " " "\*" "\*" " " " " " "   
## 7 ( 1 ) " " " " " " "\*" "\*" " " " " " "   
## 8 ( 1 ) " " "\*" " " "\*" "\*" " " " " " "   
## 9 ( 1 ) "\*" "\*" " " "\*" "\*" " " " " " "   
## 10 ( 1 ) "\*" "\*" " " "\*" "\*" "\*" " " " "   
## 11 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" " " " "   
## 12 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" " " " "   
## 13 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*" " "   
## 14 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*" " "   
## 15 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*" " "   
## 16 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*" " "   
## 17 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*" "\*"   
## 18 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*" "\*"   
## 19 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*" "\*"

names(reg.summary)

## [1] "which" "rsq" "rss" "adjr2" "cp" "bic" "outmat" "obj"

comments:

The summary() function also returns R2, RSS, adjusted R2, Cp, and BIC We can examine these to try to select the best overall model

reg.summary$rsq

## [1] 0.3214501 0.4252237 0.4514294 0.4754067 0.4908036 0.5087146 0.5141227  
## [8] 0.5285569 0.5346124 0.5404950 0.5426153 0.5436302 0.5444570 0.5452164  
## [15] 0.5454692 0.5457656 0.5459518 0.5460945 0.5461159

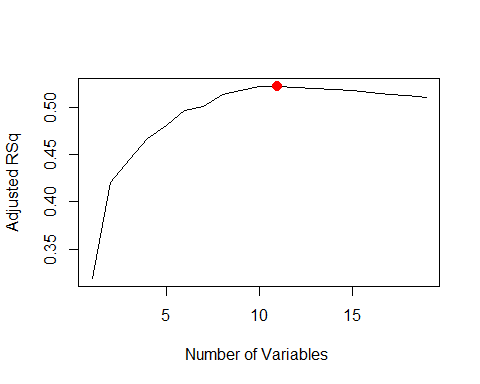
comments:

The R2 statistic increases as more variables are included

# identify the location of the maximum point of a vector  
which.max(reg.summary$adjr2)

## [1] 11

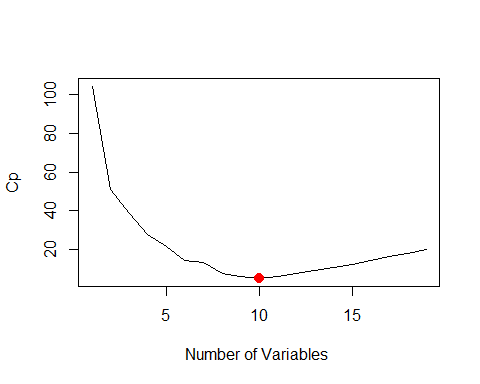
plot(reg.summary$adjr2 ,xlab="Number of Variables ", ylab="Adjusted RSq",type="l")  
  
points (11,reg.summary$adjr2[11], col="red",cex=2,pch=20)



which.min(reg.summary$cp )

## [1] 10

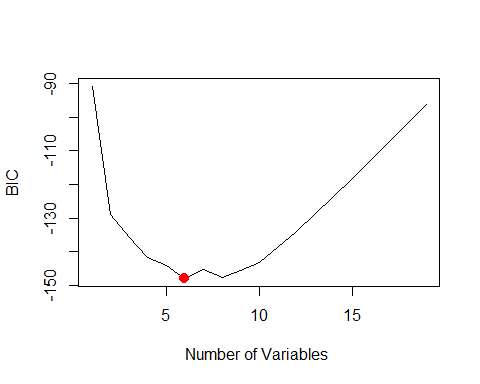
plot(reg.summary$cp ,xlab="Number of Variables ",ylab="Cp", type="l")   
  
points (10,reg.summary$cp [10],col="red",cex=2,pch=20)



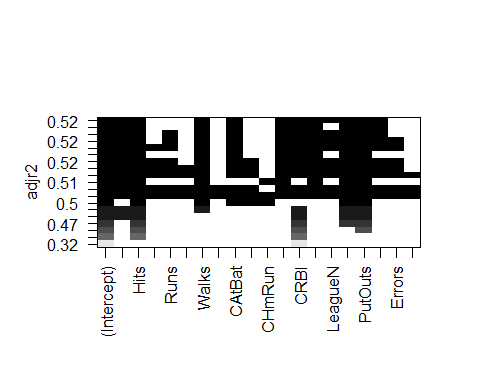
which.min(reg.summary$bic )

## [1] 6

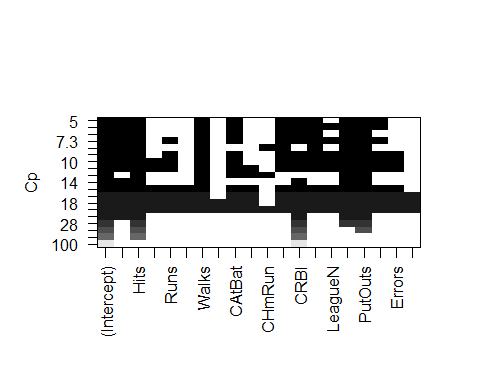
plot(reg.summary$bic ,xlab="Number of Variables ",ylab="BIC", type="l")  
  
points(6,reg.summary$bic [6],col="red",cex=2,pch =20)



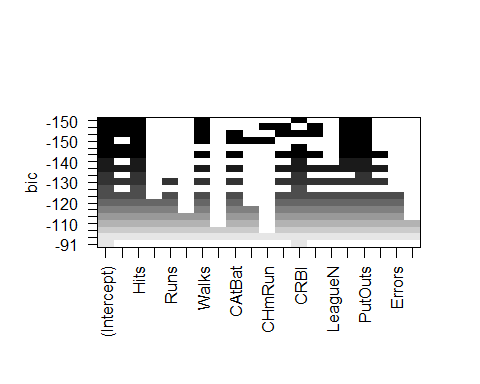
plot(regfit.full ,scale="adjr2")



plot(regfit.full ,scale="Cp")



plot(regfit.full ,scale="bic")



Comments:

plot() command display the selected variables for the best model with a given number of predictors, ranked according to the BIC, Cp,adjusted R2

#to see the coeï¬cient estimates associated with the model with the lowest BIC which has six variable  
  
coef(regfit.full ,6)

## (Intercept) AtBat Hits Walks CRBI   
## 91.5117981 -1.8685892 7.6043976 3.6976468 0.6430169   
## DivisionW PutOuts   
## -122.9515338 0.2643076

Comments:

The model contains only AtBat, Hits, Walks, CRBI, DivisionW, and PutOuts

## Fitting Linear-Regression

we will Split data set into 80:20 train and test data And fit the linear regression model

set.seed(2)  
index <- sample(nrow(Hitters), nrow(Hitters) \* 0.80)  
Hitters.train <- Hitters[index, ]  
Hitters.test <- Hitters[-index, ]  
model1 = lm( Salary ~ ., data = Hitters.train)  
model1.sum = summary(model1)  
model1.sum

##   
## Call:  
## lm(formula = Salary ~ ., data = Hitters.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -845.79 -156.96 -29.68 132.32 1794.13   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 218.47282 105.77352 2.065 0.040235 \*   
## AtBat -2.50625 0.71121 -3.524 0.000533 \*\*\*  
## Hits 8.26145 2.74309 3.012 0.002951 \*\*   
## HmRun 5.86302 7.16173 0.819 0.414006   
## Runs -1.72520 3.37922 -0.511 0.610272   
## RBI -0.03787 2.93895 -0.013 0.989733   
## Walks 5.64202 2.05738 2.742 0.006684 \*\*   
## Years -6.56279 13.96463 -0.470 0.638924   
## CAtBat -0.14621 0.14960 -0.977 0.329648   
## CHits -0.06647 0.75300 -0.088 0.929757   
## CHmRun -0.46256 1.75565 -0.263 0.792474   
## CRuns 1.66148 0.84913 1.957 0.051850 .   
## CRBI 0.77080 0.74555 1.034 0.302509   
## CWalks -0.65956 0.38139 -1.729 0.085369 .   
## LeagueN 5.09592 95.83307 0.053 0.957648   
## DivisionW -135.86765 46.17509 -2.942 0.003662 \*\*   
## PutOuts 0.35658 0.09398 3.794 0.000199 \*\*\*  
## Assists 0.59023 0.24284 2.431 0.016006 \*   
## Errors -5.90033 4.90354 -1.203 0.230366   
## NewLeagueN 37.15662 96.84965 0.384 0.701665   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 315 on 190 degrees of freedom  
## Multiple R-squared: 0.5709, Adjusted R-squared: 0.528   
## F-statistic: 13.3 on 19 and 190 DF, p-value: < 2.2e-16

Comments:

linear model were fitted to the training data. \* Using lm function resulted in model with 6 variables are significant and 13 variables are insignificant based on p-value

#Remove the insignificant variable based on best subset selection  
  
model2 = lm( Salary ~ AtBat+ Hits+Walks+CRBI+Division+PutOuts, data = Hitters.train)  
model2.sum = summary(model2)  
model2.sum

##   
## Call:  
## lm(formula = Salary ~ AtBat + Hits + Walks + CRBI + Division +   
## PutOuts, data = Hitters.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -822.83 -189.42 -33.13 126.97 2032.21   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 99.05957 73.39895 1.350 0.178646   
## AtBat -2.06378 0.59148 -3.489 0.000594 \*\*\*  
## Hits 8.08902 1.83375 4.411 1.67e-05 \*\*\*  
## Walks 4.52634 1.37364 3.295 0.001161 \*\*   
## CRBI 0.59836 0.07326 8.168 3.31e-14 \*\*\*  
## DivisionW -150.09844 44.97061 -3.338 0.001005 \*\*   
## PutOuts 0.32132 0.09150 3.512 0.000549 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 319.3 on 203 degrees of freedom  
## Multiple R-squared: 0.529, Adjusted R-squared: 0.5151   
## F-statistic: 38.01 on 6 and 203 DF, p-value: < 2.2e-16

## Model selection

#R-squared  
model1.sum$r.squared

## [1] 0.5709058

model2.sum$r.squared

## [1] 0.5290406

#AIC  
AIC(model1)

## [1] 3033.013

AIC(model2)

## [1] 3026.563

#BIC  
BIC(model1)

## [1] 3103.302

BIC(model2)

## [1] 3053.34

#Test error (MSSE)  
model1.pred.test <- predict(model1, newdata = Hitters.test)  
model1.msse = mean((model1.pred.test - Hitters.test$Salary) ^ 2)  
model1.msse

## [1] 110838.4

model2.pred.test <- predict(model2, newdata = Hitters.test)  
model2.mspe <- mean((model2.pred.test - Hitters.test$Salary) ^ 2)  
model2.mspe

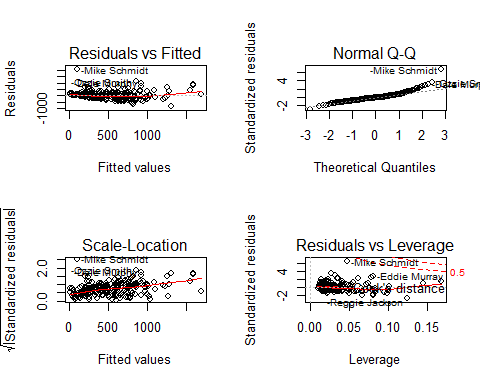
## [1] 108080

Comments: Based on AIC,Bic criteria and R square values, model 2 is slightly better than model 1, also based on prediction error, model 2 is slightly better than model 1. we choose model2 for the best model for linear-regression

## Model Assessment

Model Diagnostics for model 2

par(mfrow = c(2,2))  
plot(model2)



par(mfrow = c(1,1))