Linear Regression Applied to Combined Cycle Power Plant Dataset

Soodabeh Ramezani

October 28, 2019

Combined Cycle Power Plant Dataset

Data Set Information:

Acording to the UCI website, the description of the dataset is as follows: The dataset contains 9568 data points collected from a Combined Cycle Power Plant over 6 years (2006-2011), when the power plant was set to work with full load. Features consist of hourly average ambient variables Temperature (T), Ambient Pressure (AP), Relative Humidity (RH) and Exhaust Vacuum (V) to predict the net hourly electrical energy output (EP) of the plant. A combined cycle power plant (CCPP) is composed of gas turbines (GT), steam turbines (ST) and heat recovery steam generators. In a CCPP, the electricity is generated by gas and steam turbines, which are combined in one cycle, and is transferred from one turbine to another. While the Vacuum is colected from and has effect on the Steam Turbine, he other three of the ambient variables effect the GT performance.

Attribute Information:

Features consist of hourly average ambient variables

- Temperature (T) in the range 1.81°C and 37.11°C,
- Ambient Pressure (AP) in the range 992.89-1033.30 milibar,
- Relative Humidity (RH) in the range 25.56% to 100.16%
- Exhaust Vacuum (V) in teh range 25.36-81.56 cm Hg
- Net hourly electrical energy output (EP) 420.26-495.76 MW

The averages are taken from various sensors located around the plant that record the ambient variables every second. The variables are given without normalization.

The data is available in https://archive.ics.uci.edu

I begin by plotting the dara:

```
setwd("C:/Users/soodr/Documents/STAT5310_Applied_Regression_Analysis/data")
mydata <- read.csv("mydata.csv",header=TRUE)
attach(mydata)
```

```
pairs(PE~AT+V+AP+RH,data=mydata,gap=0.4,cex.labels=0.85)
mtext("Figure 1 Scatter plot matrix of the response variable and each of the predictors", side = 1, lin
```

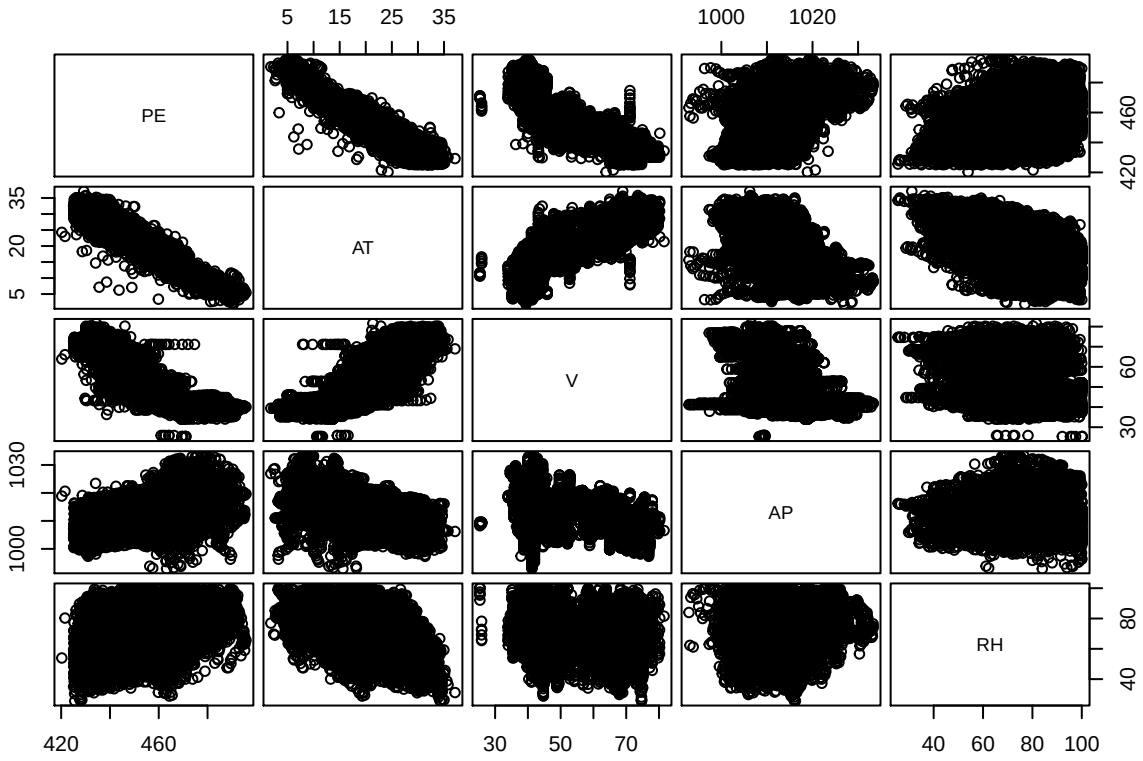


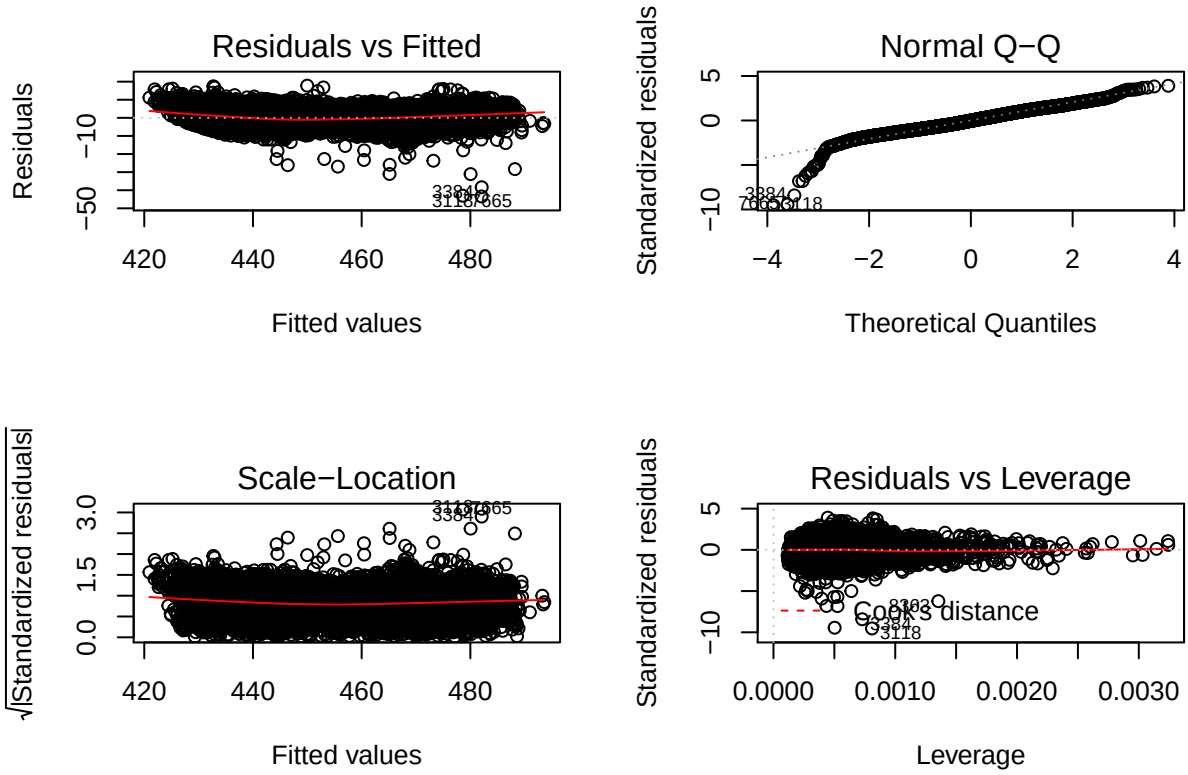
Figure 1 Scatter plot matrix of the response variable and each of the predictors

```
#model1
m1 <- lm(PE~AT+V+AP+RH,data=mydata)
```

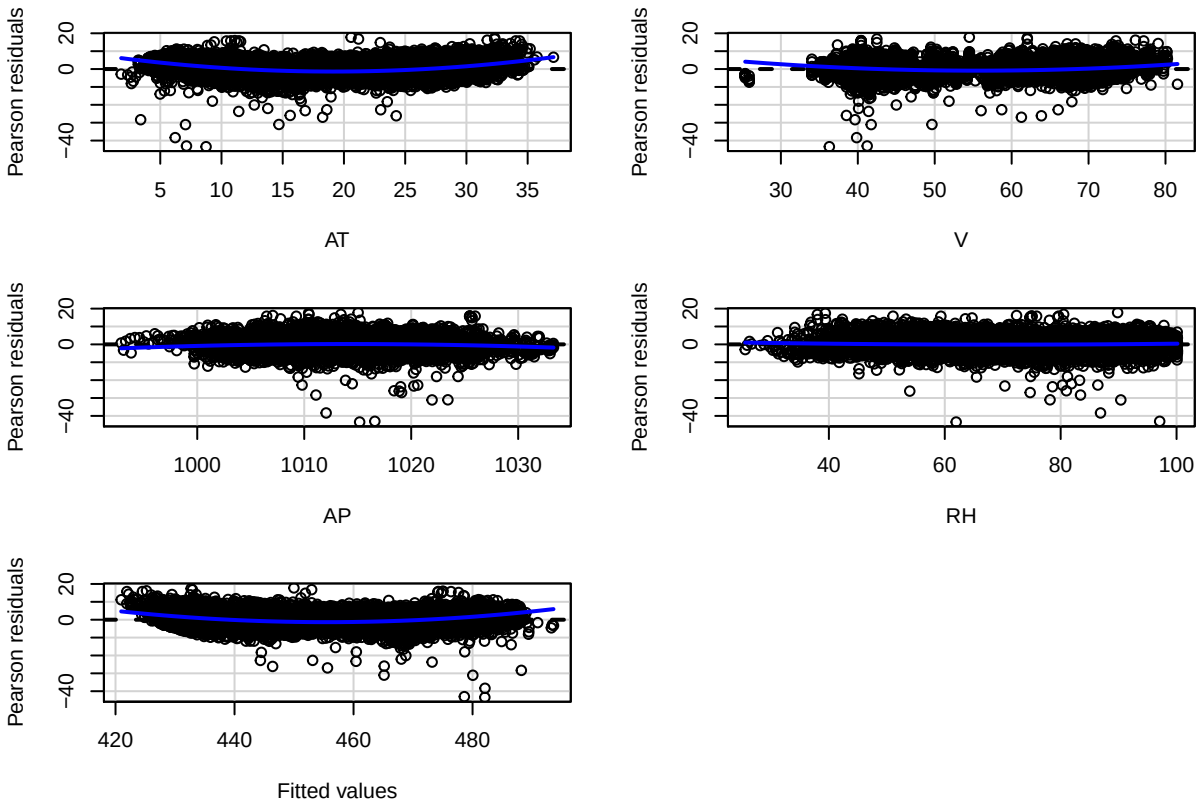
```
library(car)
```

```
## Loading required package: carData
```

```
par(mfrow=c(2,2))
plot(m1)
```

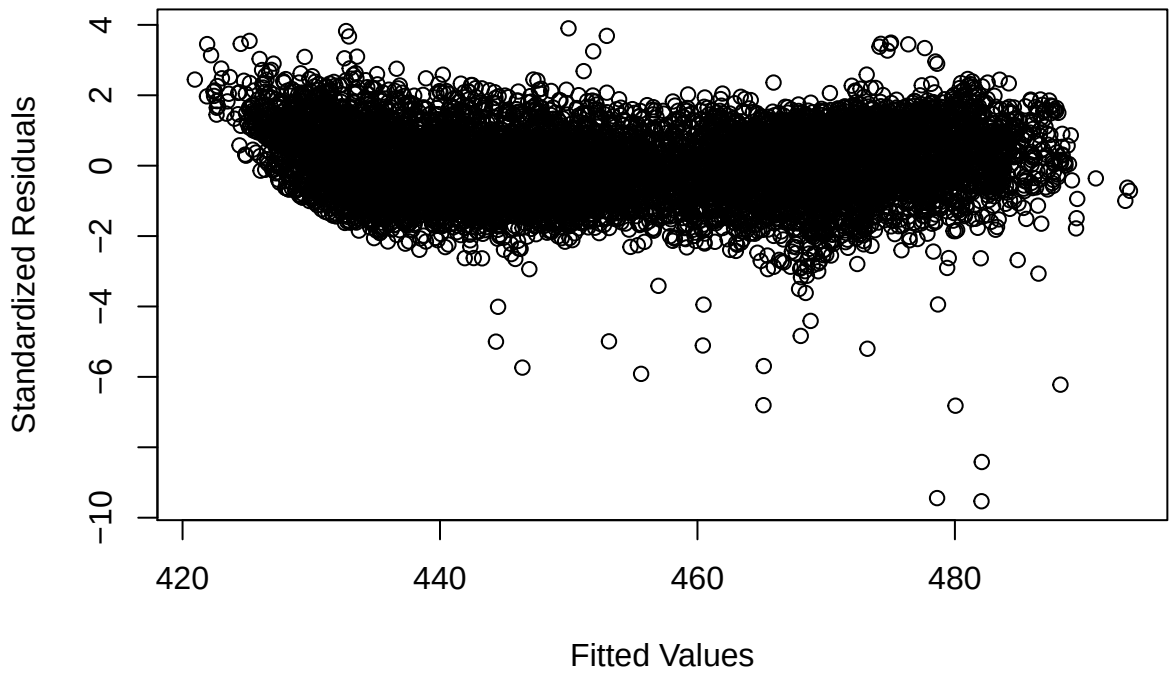



```
#Plots of standardized residuals against each predictor and the fitted value:
par(mfrow=c(1,1))
StanRes1 <- rstandard(m1)
residualPlots(m1)
```

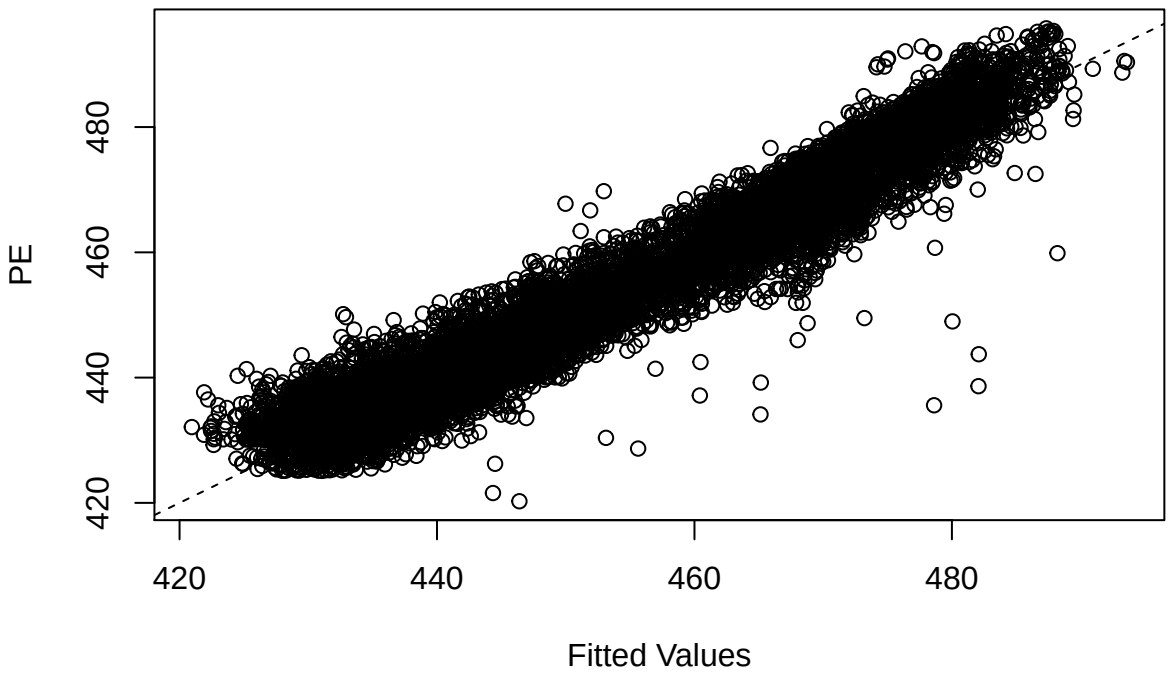



```
##             Test stat Pr(>|Test stat|)
## AT            32.2396        < 2.2e-16 ***
## V             14.4880        < 2.2e-16 ***
## AP            -5.8061        6.599e-09 ***
## RH             3.0722         0.002131 **
## Tukey test    29.3154        < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

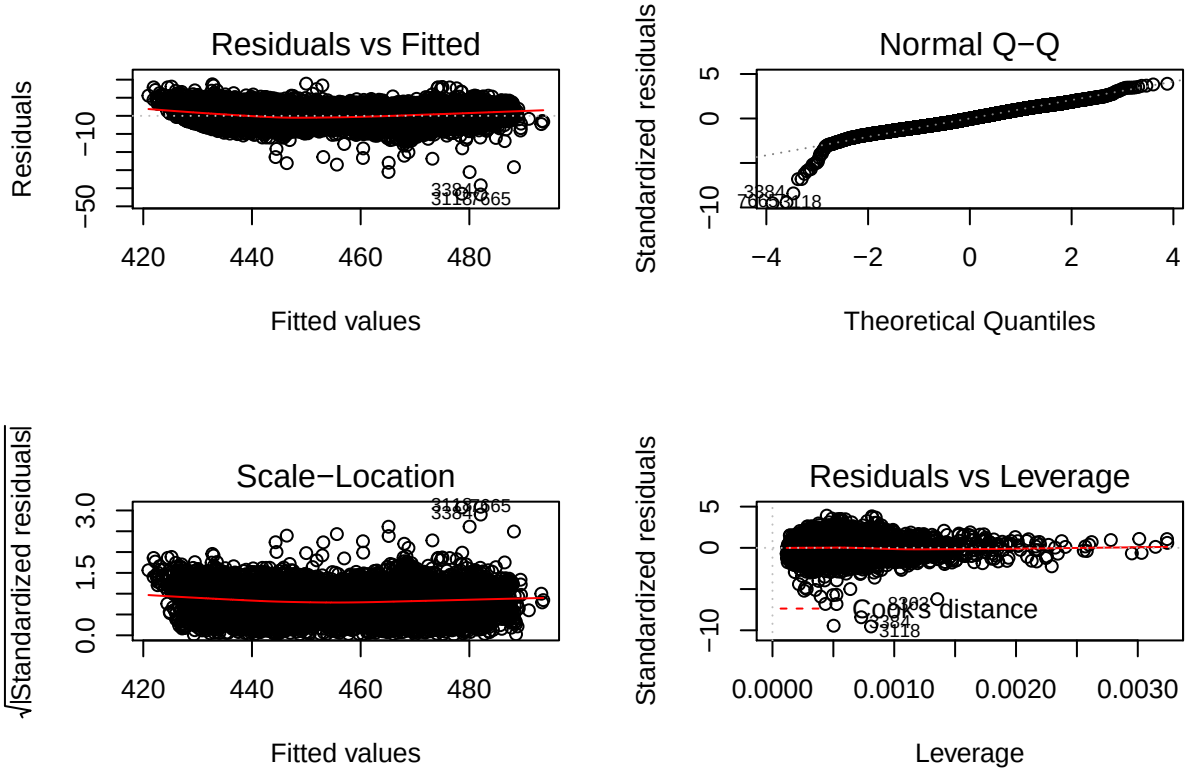
```
plot(m1$fitted.values,StanRes1,ylab="Standardized Residuals",xlab="Fitted Values")
```




```
#Plot of y against fitted value
par(mfrow=c(1,1))
fit1 <- m1$fitted.values
plot(fit1,PE,xlab="Fitted Values")
abline(lsfit(m1$fitted.values,PE),lty=2)
```




```
#Diagnostic plots
par(mfrow=c(2,2))
plot(m1)
```

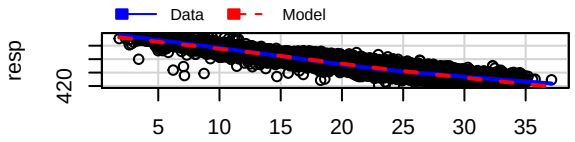


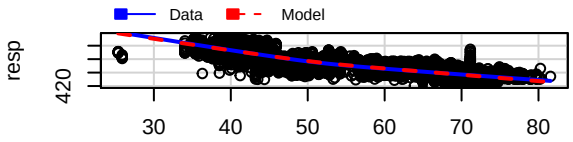

```
summary(m1)
```

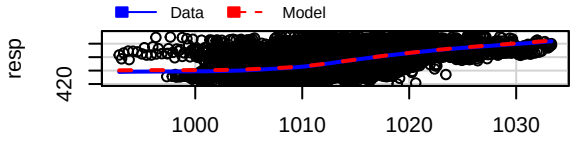
```
##
## Call:
## lm(formula = PE ~ AT + V + AP + RH, data = mydata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -43.435  -3.166  -0.118   3.201  17.778
##
## Coefficients:
##              Estimate Std. Error  t value Pr(>|t|)
## (Intercept) 454.609274   9.748512   46.634  < 2e-16 ***
## AT           -1.977513   0.015289 -129.342  < 2e-16 ***
## V            -0.233916   0.007282  -32.122  < 2e-16 ***
## AP            0.062083   0.009458    6.564 5.51e-11 ***
## RH           -0.158054   0.004168  -37.918  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.558 on 9563 degrees of freedom
## Multiple R-squared:  0.9287, Adjusted R-squared:  0.9287
## F-statistic: 3.114e+04 on 4 and 9563 DF,  p-value: < 2.2e-16
```

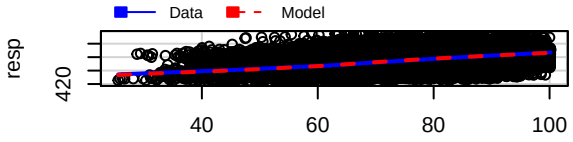
```
library(alr3)
par(mfrow=c(3,2))
mmp(m1,AT,key=NULL)
```

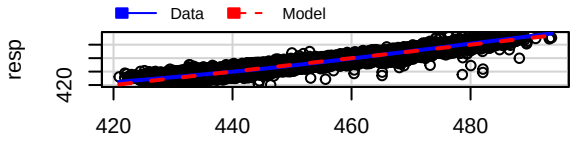
```
mmp(m1,V,key=NULL)
mmp(m1,AP,key=NULL)
mmp(m1,RH,key=NULL)
mmp(m1,m1$fitted.values,xlab="Fitted Values",key=NULL)
```







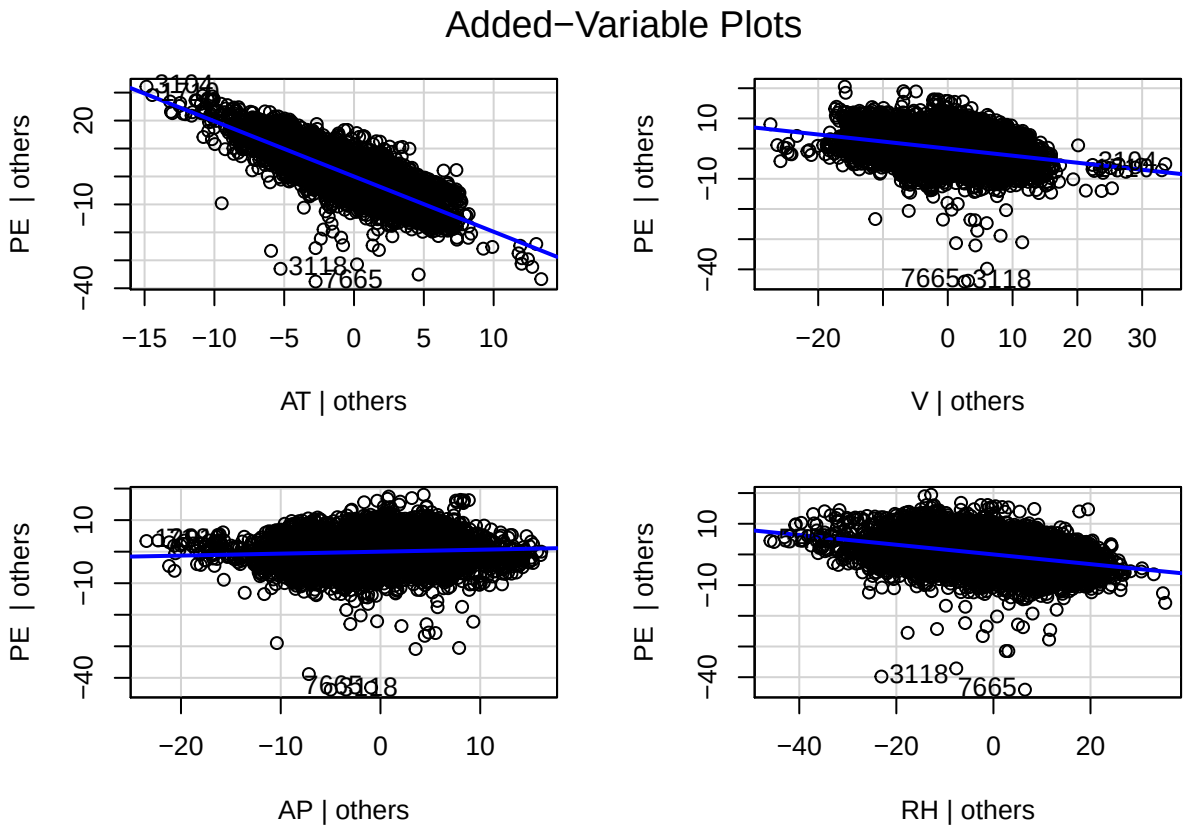





```
library(car)
vif(m1)
```

```
##        AT         V        AP        RH
## 5.977602 3.943003 1.452639 1.705290
```

```
#Added-variable plots for model
avPlots(m1)
```



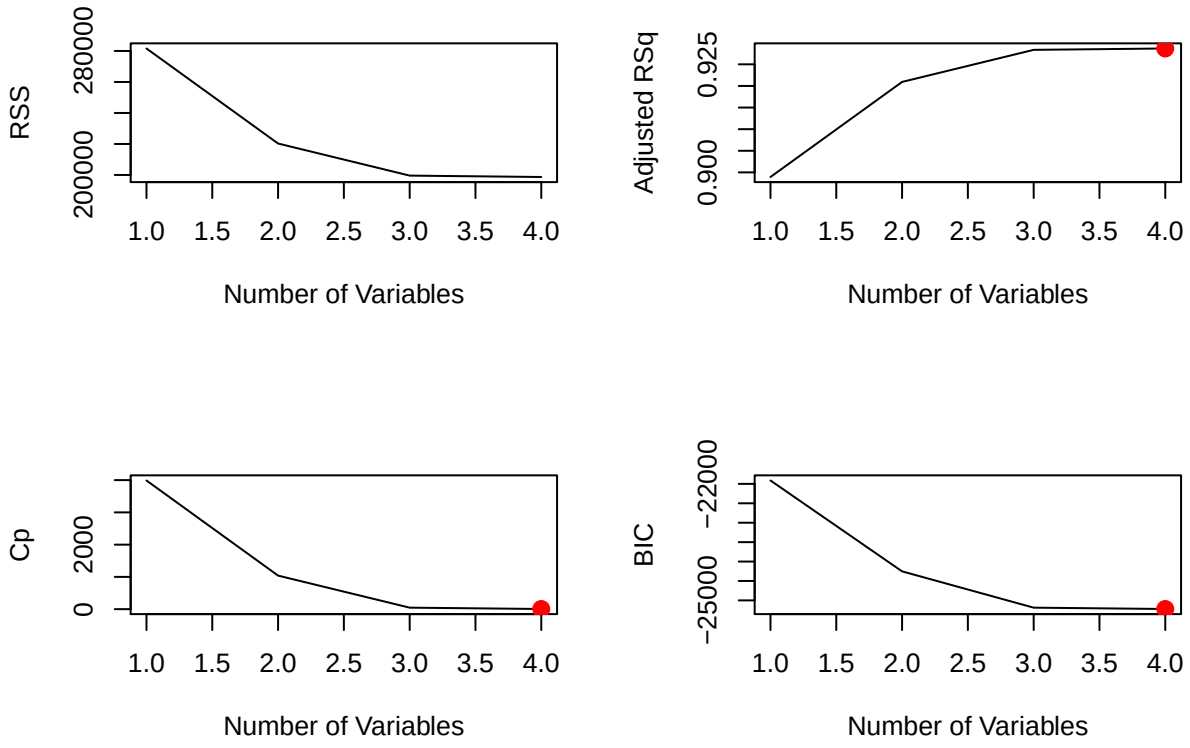
Identify the optimal model or models based on adj R^2 , AIC, AICC, BIC from the approach based on all possible subsets.

```
library(leaps)
regfit=regsubsets(PE~AT+V+AP+RH,data=mydata)
summary(regfit)
```

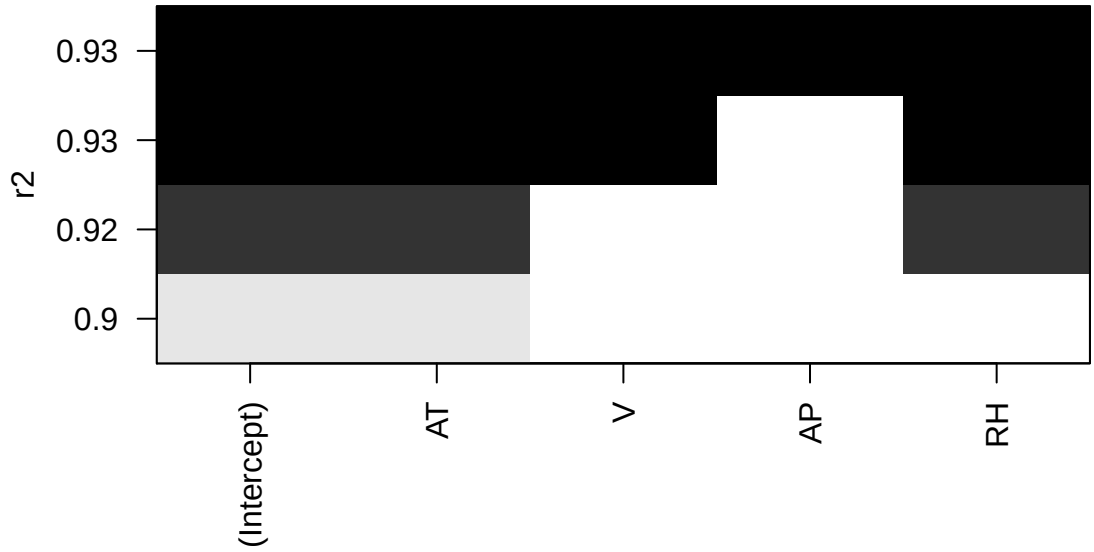
```
## Subset selection object
## Call: regsubsets.formula(PE ~ AT + V + AP + RH, data = mydata)
## 4 Variables  (and intercept)
##    Forced in Forced out
## AT     FALSE      FALSE
## V      FALSE      FALSE
## AP     FALSE      FALSE
## RH     FALSE      FALSE
## 1 subsets of each size up to 4
## Selection Algorithm: exhaustive
##          AT  V   AP  RH
## 1  ( 1 ) "*" " " " " " "
## 2  ( 1 ) "*" " " " " "*"
## 3  ( 1 ) "*" "*" " " "*"
## 4  ( 1 ) "*" "*" "*" "*"
```

```
summ=summary(regfit)
```

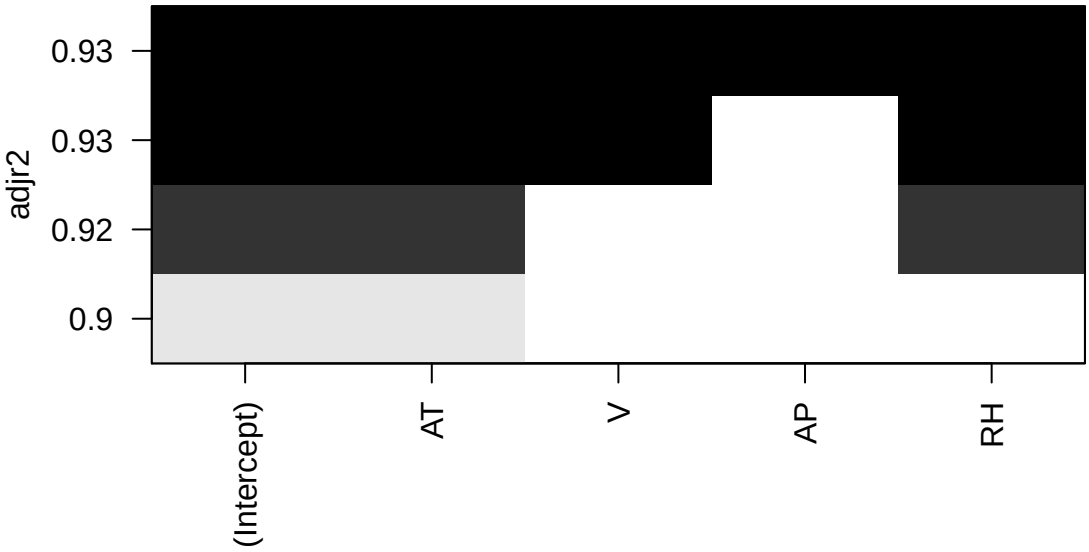
```
par(mfrow=c(2,2))
plot(summ$rss, xlab="Number of Variables", ylab="RSS", type="l")
plot(summ$adjr2, xlab="Number of Variables", ylab="Adjusted RSq", type="l")
points(which.max(summ$adjr2),summ$adjr2[which.max(summ$adjr2)], col="red", cex=2, pch=20)
plot(summ$cp, xlab="Number of Variables", ylab="Cp", type='l')
points(which.min(summ$cp),summ$cp[which.min(summ$cp)], col="red", cex=2, pch=20)
plot(summ$bic, xlab="Number of Variables", ylab="BIC", type='l')
points(which.min(summ$bic),summ$bic[which.min(summ$bic)], col="red", cex=2, pch=20)
```



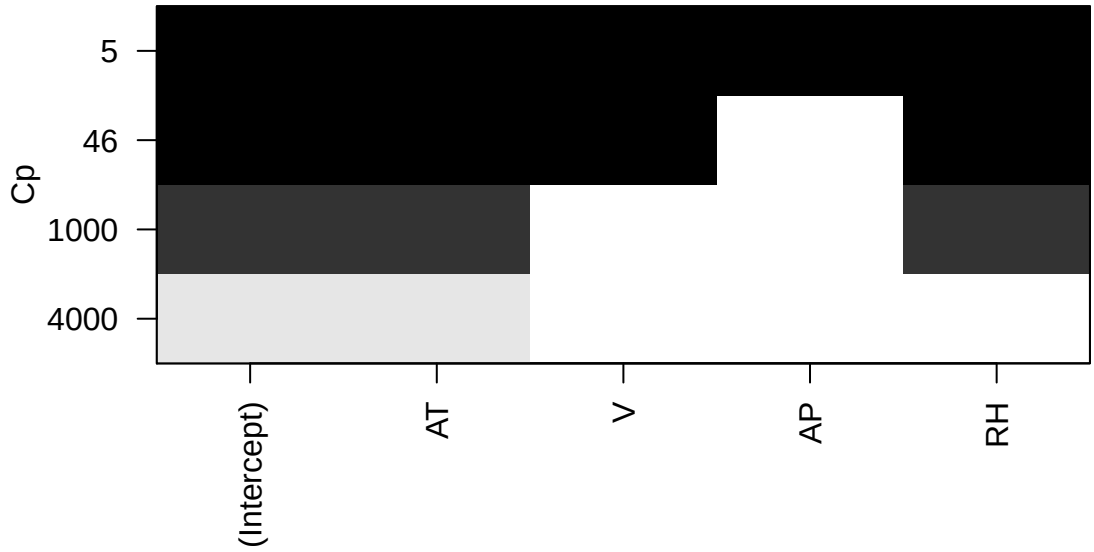

```
plot(regfit, scale = "r2")
```



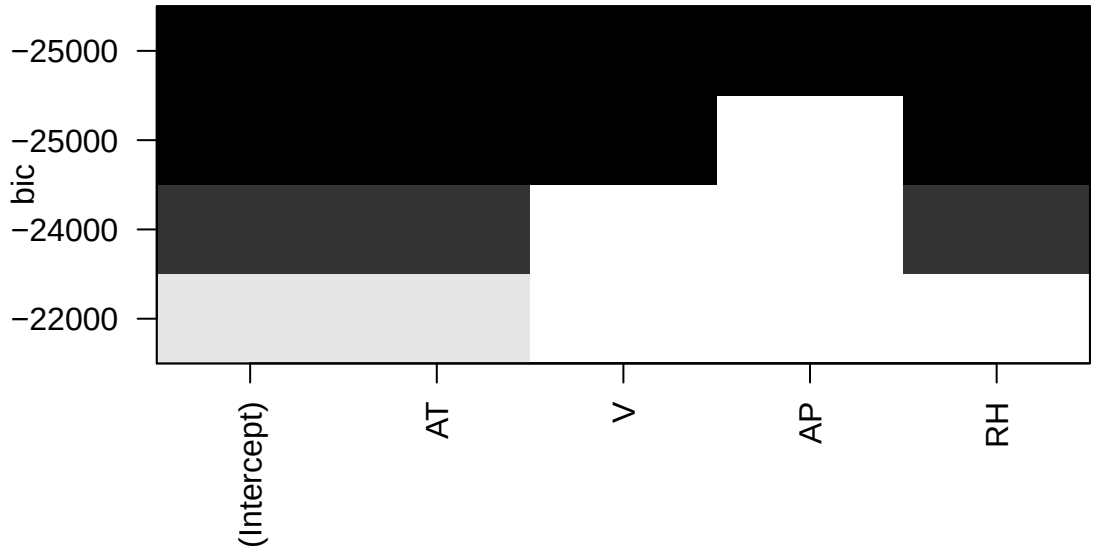

```
plot(regfit, scale = "adjr2")
```




```
plot(regfit, scale = "Cp")
```




```
plot(regfit, scale = "bic")
```




```
m1=lm(PE~AT,data=mydata)
m2=lm(PE~AT+RH,data=mydata)
m3=lm(PE~AT+V+RH,data=mydata)
m4=lm(PE~AT+V+AP+RH,data=mydata)

#Subset size=1-----------------------------------
n <- length(m1$residuals)
npar <- length(m1$coefficients) +1
#Calculate AIC
AIC_1=extractAIC(m1,k=2)
#Calculate AICc
AICc_1=extractAIC(m1,k=2)+2*npar*(npar+1)/(n-npar-1)
#Subset size=2-----------------------------------
npar <- length(m2$coefficients) +1
#Calculate AIC
AIC_2=extractAIC(m2,k=2)
#Calculate AICc
AICc_2=extractAIC(m2,k=2)+2*npar*(npar+1)/(n-npar-1)
#Subset size=3-----------------------------------
npar <- length(m3$coefficients) +1
#Calculate AIC
AIC_3=extractAIC(m3,k=2)
#Calculate AICc
AICc_3=extractAIC(m3,k=2)+2*npar*(npar+1)/(n-npar-1)
#Subset size=4-----------------------------------
npar <- length(m4$coefficients) +1
```

```
#Calculate AIC
AIC_4=extractAIC(m4,k=2)
#Calculate AICc
AICc_4=extractAIC(m4,k=2)+2*npar*(npar+1)/(n-npar-1)


#Creating a table of criteria
vec1=c(summ$adjr2[1],AIC_1[2],AICc_1[2],summ$bic[2], summ$cp[1],
     summ$adjr2[2],AIC_2[2],AICc_2[2],summ$bic[2], summ$cp[2],
     summ$adjr2[3],AIC_3[2],AICc_3[2],summ$bic[3], summ$cp[3],
     summ$adjr2[4],AIC_4[2],AICc_4[2],summ$bic[4], summ$cp[4])


AllModels<- matrix(vec1,ncol=5,byrow=TRUE)
colnames(AllModels) <- c("Adj.R2","AIC","AICc","BIC","PC")
rownames(AllModels) <- c("M1","M2","M3","M4")
AllModels <- as.table(AllModels)
AllModels
```

```
##           Adj.R2           AIC          AICc           BIC            PC
## M1  8.989370e-01  3.236367e+04  3.236367e+04 -2.425274e+04  3.988751e+03
## M2  9.209315e-01  3.001640e+04  3.001640e+04 -2.425274e+04  1.040133e+03
## M3  9.283524e-01  2.907443e+04  2.907444e+04 -2.518754e+04  4.608722e+01
## M4  9.286663e-01  2.903342e+04  2.903343e+04 -2.522138e+04  5.000000e+00
```

Identify the optimal model or models based on AIC and BIC from the approach based on backward selection.

```
backAIC <- step(m4,direction="backward", data=mydata)
```

```
## Start:  AIC=29033.42
## PE ~ AT + V + AP + RH
##
##        Df Sum of Sq    RSS   AIC
## <none>              198702 29033
## - AP    1       895 199598 29074
## - V     1     21440 220142 30012
## - RH    1     29875 228578 30372
## - AT    1    347607 546309 38708
```

```
backBIC <- step(m4,direction="backward", data=mydata, k=log(n))
```

```
## Start:  AIC=29069.25
## PE ~ AT + V + AP + RH
##
##        Df Sum of Sq    RSS   AIC
## <none>              198702 29069
## - AP    1       895 199598 29103
## - V     1     21440 220142 30040
## - RH    1     29875 228578 30400
## - AT    1    347607 546309 38737
```

Identify the optimal model or models based on AIC and BIC from the approach based on forward selection.

```
# Fit the null model
null.model <- lm(PE~1, data = mydata)

#AIC from the approach based on forward selection-----------------------------------------------------
forwardAIC <- step(null.model, direction = "forward", scope = list(lower = null.model, upper = m4))
```

```
## Start:  AIC=54292.64
## PE ~ 1
##
##        Df Sum of Sq     RSS   AIC
## + AT    1   2505095  281603 32364
## + V     1   2108187  678511 40778
## + AP    1    748977 2037721 51300
## + RH    1    423409 2363289 52718
## <none>               2786698 54293
##
## Step:  AIC=32363.67
## PE ~ AT
##
##        Df Sum of Sq    RSS   AIC
## + RH    1     61309 220294 30016
## + V     1     46766 234837 30628
## + AP    1      5196 276406 32187
## <none>              281603 32364
##
## Step:  AIC=30016.4
## PE ~ AT + RH
##
##        Df Sum of Sq    RSS   AIC
## + V     1   20696.1 199598 29074
## + AP    1     151.7 220142 30012
## <none>              220294 30016
##
## Step:  AIC=29074.43
## PE ~ AT + RH + V
##
##        Df Sum of Sq    RSS   AIC
## + AP    1    895.28 198702 29033
## <none>              199598 29074
##
## Step:  AIC=29033.42
## PE ~ AT + RH + V + AP
```

```
#BIC from the approach based on forward selection-----------------------------------------------------
forwardBIC <- step(null.model, direction = "forward", scope = list(lower = null.model, upper = m4),k=log
```

```
## Start:  AIC=54299.8
## PE ~ 1
##
##        Df Sum of Sq     RSS   AIC
## + AT    1   2505095  281603 32378
```

```
## + V     1   2108187  678511 40792
## + AP    1    748977 2037721 51314
## + RH    1    423409 2363289 52732
## <none>               2786698 54300
##
## Step:  AIC=32378
## PE ~ AT
##
##        Df Sum of Sq    RSS   AIC
## + RH    1     61309 220294 30038
## + V     1     46766 234837 30650
## + AP    1      5196 276406 32209
## <none>              281603 32378
##
## Step:  AIC=30037.89
## PE ~ AT + RH
##
##        Df Sum of Sq    RSS   AIC
## + V     1   20696.1 199598 29103
## <none>              220294 30038
## + AP    1     151.7 220142 30041
##
## Step:  AIC=29103.1
## PE ~ AT + RH + V
##
##        Df Sum of Sq    RSS   AIC
## + AP    1    895.28 198702 29069
## <none>              199598 29103
##
## Step:  AIC=29069.25
## PE ~ AT + RH + V + AP
```