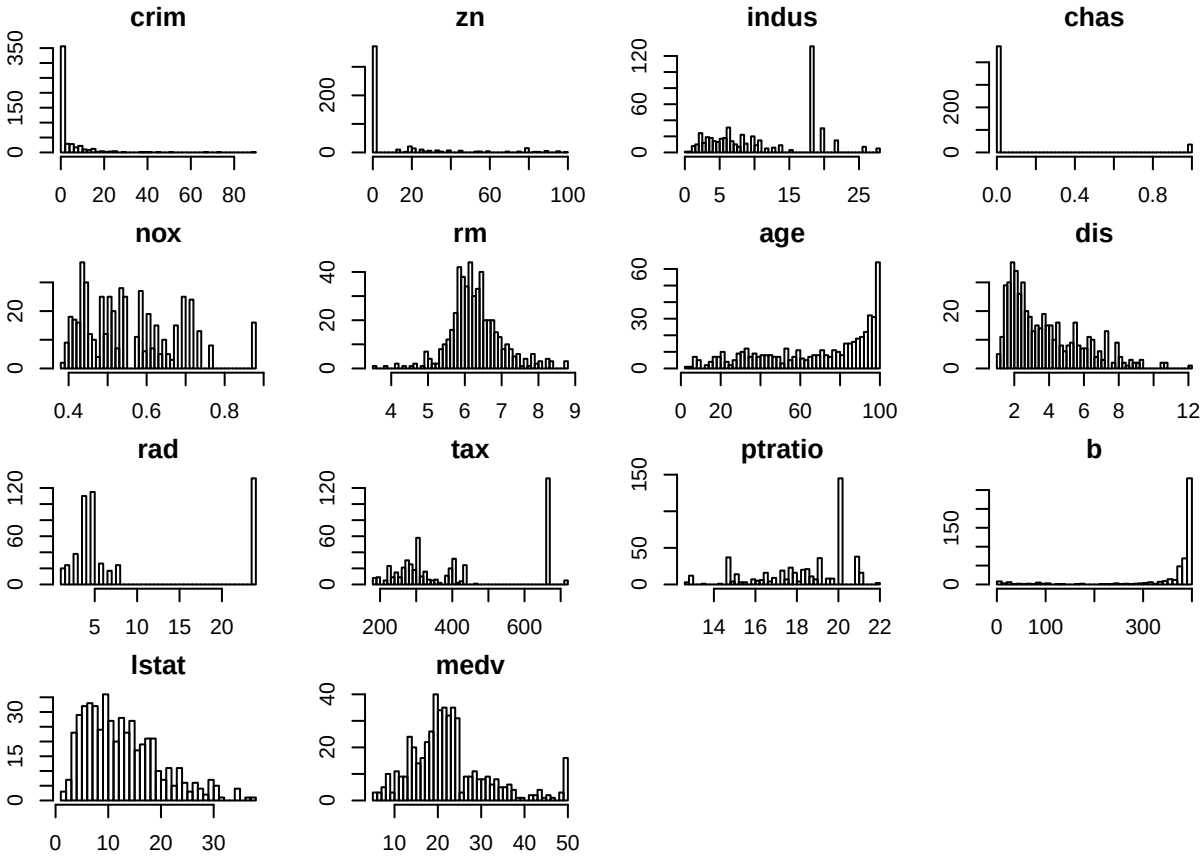
Midterm II_R code

Huidi Wang

Part 2

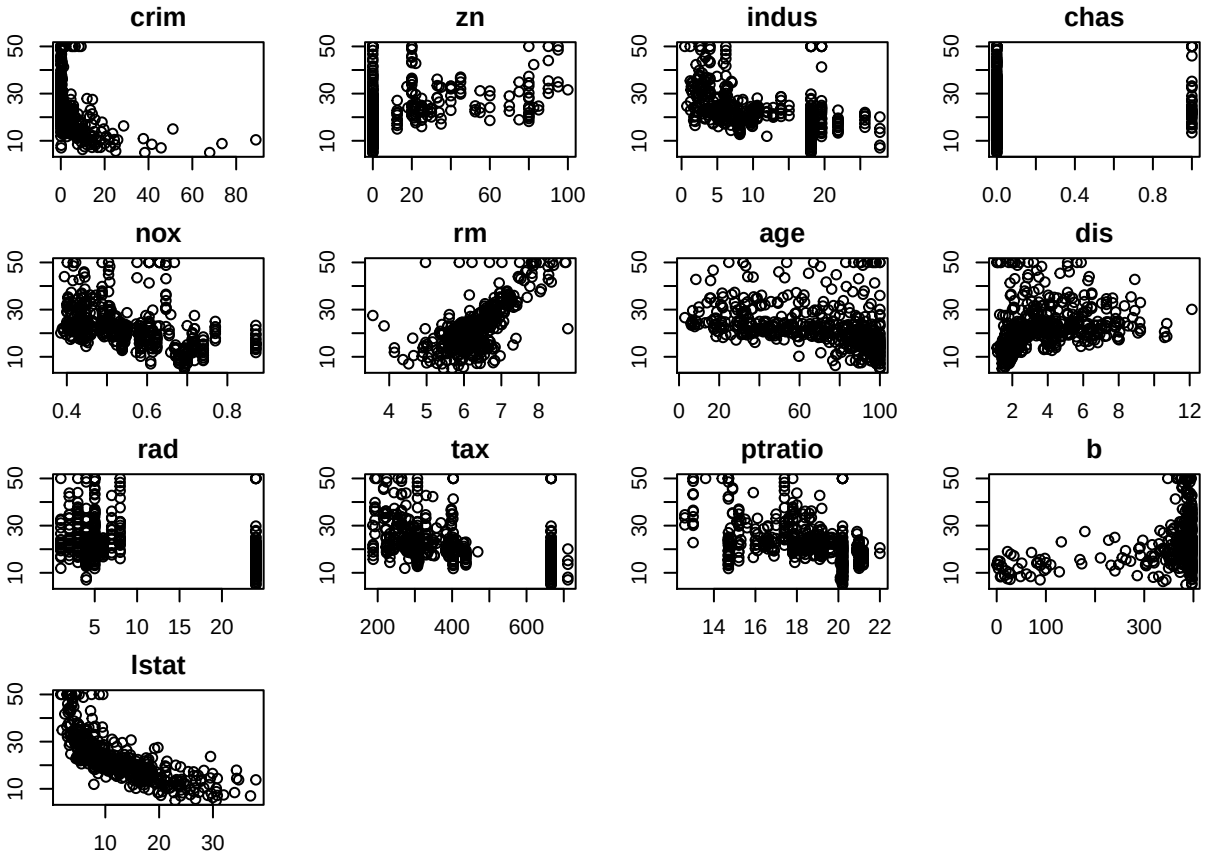
2)

```
library(mlbench)
library(leaps)
library(faraway)
data(BostonHousing)
BH=BostonHousing
BH$chas=as.numeric(BH$chas)
BH$chas=as.numeric(BH$chas==2)  #To change "level" to "numeric" for BH$chas.
par(mfrow = c(4, 4))            #Graph each variable's histogram
par(mar = rep(2, 4))
for(i in 1:14)
{
  hist(BH[,i], xlab = "", main = names(BH)[i], breaks = 50)
}
```



The relationship between medv with each explanatory variable

```
par(mfrow = c(4,4))
par(mar = rep(2, 4))
for(i in 1:13)
{
  plot(BH[,i], BH[,14], main = names(BH)[i])
}
par(mfrow = c(1,1))
```



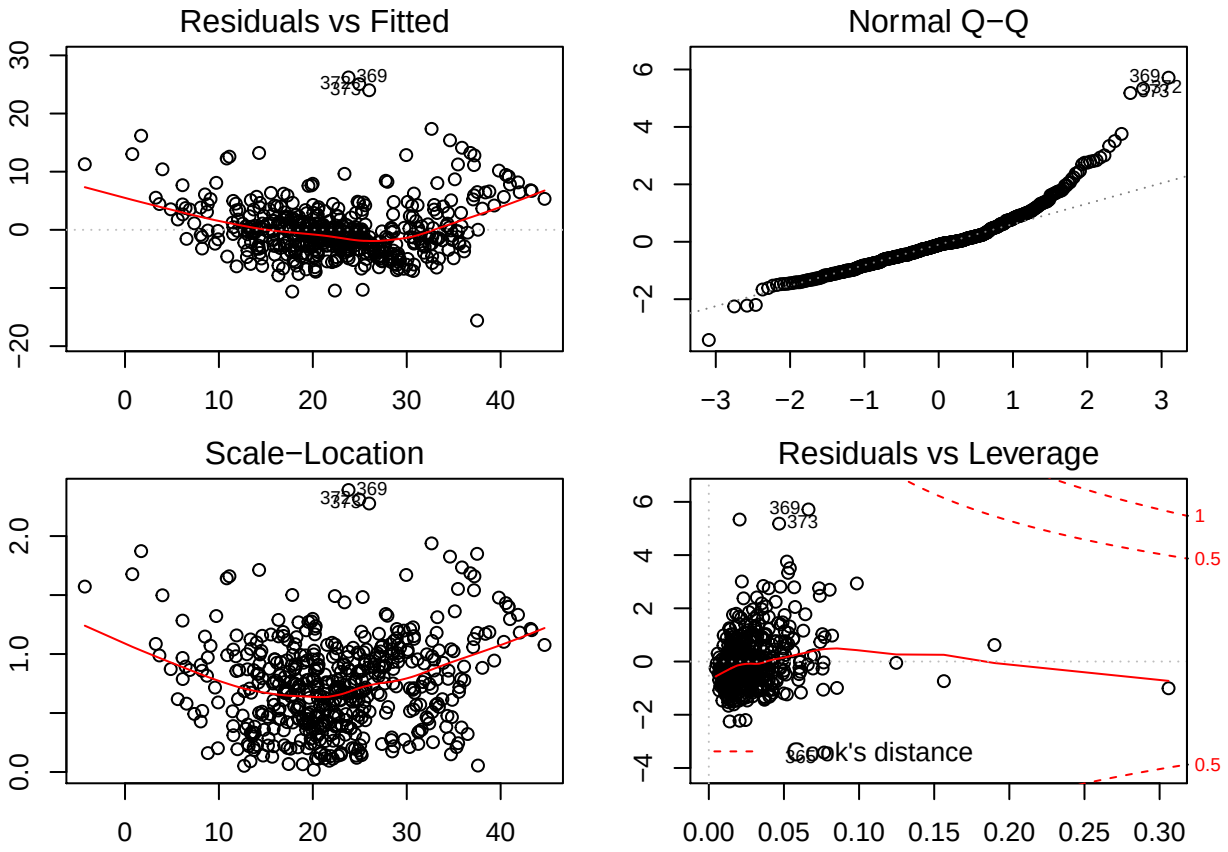
Find out outliers from previous graphs.

```
o1=which(BH$crim > 60)
o4=which(BH$dis > 12)
outliers=c(o1,o4)
table(outliers)
```

```
## outliers
## 354 381 406 419
##   1   1   1   1
```

3)

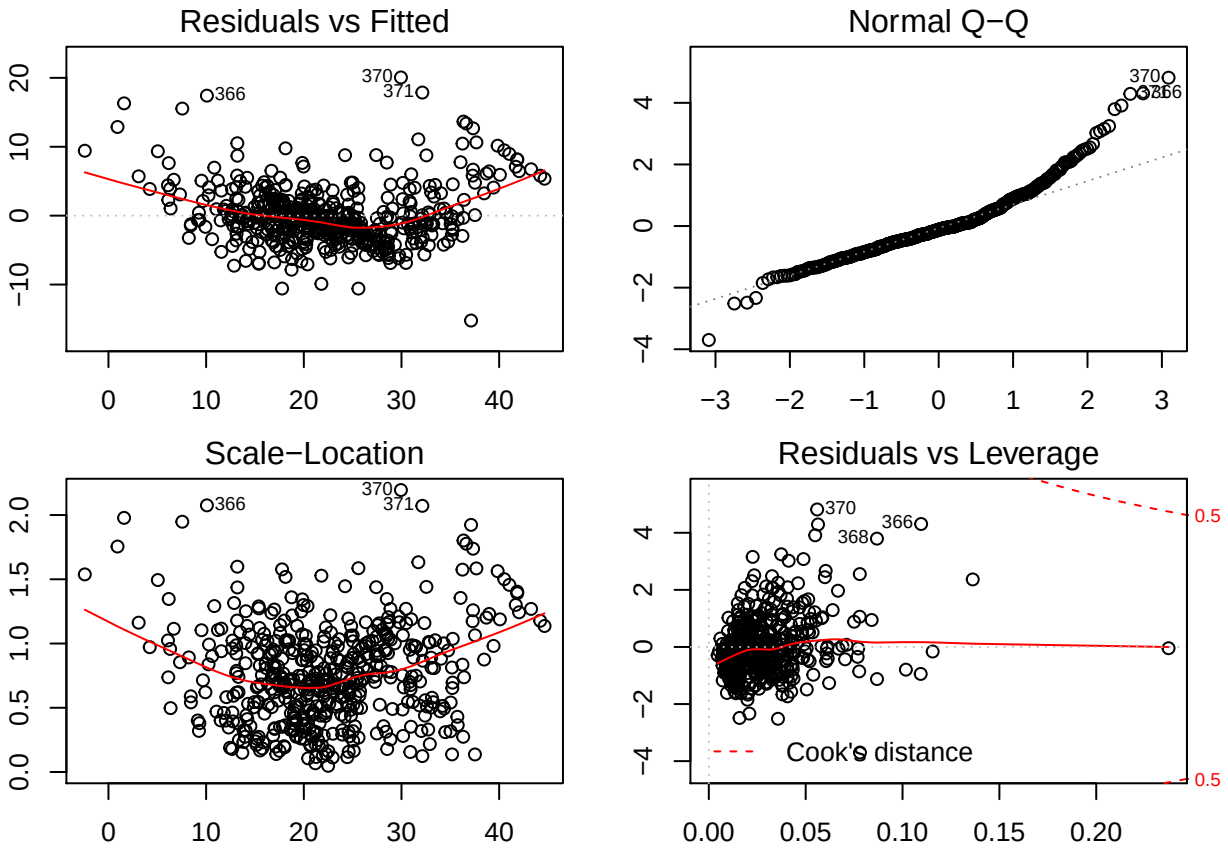
```
BBHH = lm(BH$medv ~ ., BH)  #Do linear model and plot the four graphs.
par(mfrow=c(2,2))
par(mar = rep(2, 4))
plot(BBHH)                  #By observation, outliers are #369,372,373
```




```
outliers=c(outliers,369,372,373)
```

Remove outliers and plot again

```
BH1=BH[-outliers,]
BBHH1 = lm(BH1$medv ~ ., BH1)
par(mar = rep(2, 4))#Re-do linear model w/o outliers and plot the four graphs.
par(mfrow=c(2,2))
plot(BBHH1)                    #Plots w/o outliers
```



Part 3

```
BH$medv=log(BH$medv)
BBHH = lm(BH$medv ~ ., BH)      #Since here BH w/log(medv)
summary(BBHH)                   #Initial Model
```

```
##
## Call:
## lm(formula = BH$medv ~ ., data = BH)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.73361 -0.09747 -0.01657  0.09629  0.86435
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.1020423  0.2042726  20.081  < 2e-16 ***
## crim        -0.0102715  0.0013155  -7.808 3.52e-14 ***
## zn           0.0011725  0.0005495   2.134 0.033349 *
## indus        0.0024668  0.0024614   1.002 0.316755
## chas         0.1008876  0.0344859   2.925 0.003598 **
## nox         -0.7783993  0.1528902  -5.091 5.07e-07 ***
## rm           0.0908331  0.0167280   5.430 8.87e-08 ***
## age          0.0002106  0.0005287   0.398 0.690567
## dis         -0.0490873  0.0079834  -6.149 1.62e-09 ***
## rad          0.0142673  0.0026556   5.373 1.20e-07 ***
## tax         -0.0006258  0.0001505  -4.157 3.80e-05 ***
```

```
## ptratio     -0.0382715  0.0052365  -7.309 1.10e-12 ***
## b            0.0004136  0.0001075   3.847 0.000135 ***
## lstat       -0.0290355  0.0020299 -14.304  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1899 on 492 degrees of freedom
## Multiple R-squared:  0.7896, Adjusted R-squared:  0.7841
## F-statistic: 142.1 on 13 and 492 DF,  p-value: < 2.2e-16
```

(1) Model Selection

```
#1 Backward elimination using individual p-value
#at each stage we remove the predictor with the largest p-value over 0.05
summary(BBHH)
```

```
##
## Call:
## lm(formula = BH$medv ~ ., data = BH)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.73361 -0.09747 -0.01657  0.09629  0.86435
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.1020423  0.2042726  20.081  < 2e-16 ***
## crim        -0.0102715  0.0013155  -7.808 3.52e-14 ***
## zn           0.0011725  0.0005495   2.134 0.033349 *
## indus        0.0024668  0.0024614   1.002 0.316755
## chas         0.1008876  0.0344859   2.925 0.003598 **
## nox         -0.7783993  0.1528902  -5.091 5.07e-07 ***
## rm           0.0908331  0.0167280   5.430 8.87e-08 ***
## age          0.0002106  0.0005287   0.398 0.690567
## dis         -0.0490873  0.0079834  -6.149 1.62e-09 ***
## rad          0.0142673  0.0026556   5.373 1.20e-07 ***
## tax         -0.0006258  0.0001505  -4.157 3.80e-05 ***
## ptratio     -0.0382715  0.0052365  -7.309 1.10e-12 ***
## b            0.0004136  0.0001075   3.847 0.000135 ***
## lstat       -0.0290355  0.0020299 -14.304  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1899 on 492 degrees of freedom
## Multiple R-squared:  0.7896, Adjusted R-squared:  0.7841
## F-statistic: 142.1 on 13 and 492 DF,  p-value: < 2.2e-16
```

```
BBHH<- update(BBHH, . ~ . - age)
summary(BBHH)
```

```
##
```

```
## Call:
## lm(formula = BH$medv ~ crim + zn + indus + chas + nox + rm +
##     dis + rad + tax + ptratio + b + lstat, data = BH)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.73345 -0.09809 -0.01744  0.09653  0.86552
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.0951779  0.2033706  20.137  < 2e-16 ***
## crim        -0.0102698  0.0013143  -7.814 3.38e-14 ***
## zn           0.0011461  0.0005450   2.103 0.035978 *
## indus        0.0024679  0.0024593   1.003 0.316129
## chas         0.1015851  0.0344120   2.952 0.003307 **
## nox         -0.7622525  0.1472924  -5.175 3.32e-07 ***
## rm           0.0922108  0.0163525   5.639 2.89e-08 ***
## dis         -0.0500137  0.0076307  -6.554 1.41e-10 ***
## rad          0.0141871  0.0026457   5.362 1.27e-07 ***
## tax         -0.0006240  0.0001503  -4.151 3.90e-05 ***
## ptratio     -0.0381084  0.0052160  -7.306 1.12e-12 ***
## b            0.0004163  0.0001072   3.883 0.000117 ***
## lstat       -0.0287597  0.0019066 -15.085  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1898 on 493 degrees of freedom
## Multiple R-squared:  0.7896, Adjusted R-squared:  0.7845
## F-statistic: 154.2 on 12 and 493 DF,  p-value: < 2.2e-16
```

```
BBHH<- update(BBHH, . ~ . - indus)
summary(BBHH)
```

```
##
## Call:
## lm(formula = BH$medv ~ crim + zn + chas + nox + rm + dis + rad +
##     tax + ptratio + b + lstat, data = BH)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.73400 -0.09460 -0.01771  0.09782  0.86290
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.0836823  0.2030491  20.112  < 2e-16 ***
## crim        -0.0103187  0.0013134  -7.856 2.49e-14 ***
## zn           0.0010874  0.0005418   2.007 0.045308 *
## chas         0.1051484  0.0342285   3.072 0.002244 **
## nox         -0.7217440  0.1416535  -5.095 4.97e-07 ***
## rm           0.0906728  0.0162807   5.569 4.20e-08 ***
## dis         -0.0517059  0.0074420  -6.948 1.18e-11 ***
## rad          0.0134457  0.0025405   5.293 1.82e-07 ***
## tax         -0.0005579  0.0001351  -4.129 4.28e-05 ***
## ptratio     -0.0374259  0.0051715  -7.237 1.77e-12 ***
```

```
## b            0.0004127  0.0001071   3.852 0.000133 ***
## lstat       -0.0286039  0.0019002 -15.053  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1898 on 494 degrees of freedom
## Multiple R-squared:  0.7891, Adjusted R-squared:  0.7844
## F-statistic: 168.1 on 11 and 494 DF,  p-value: < 2.2e-16
```

```
#BBHH~crim,zn,chas,nox,rm,dis,rad,tax,ptratio,b,lstat with all having P-value less than 0.05,
#which means they have more predictive power, should be selected.

#2 Forward Selection using p-value
#at each stage we add the predictor with the smallest p-value less than 0.05
BBHH = lm(BH$medv ~ ., BH)

for( i in 1:13)
{
  g1 <- lm(BH$medv~ ., BH[,c(i,14)])
  print((summary(g1))$coef)
}
```

```
##                Estimate  Std. Error   t value     Pr(>|t|)
## (Intercept)  3.12517153 0.016758213 186.48597 0.000000e+00
## crim        -0.02508871 0.001797724 -13.95582 1.166239e-37
##                Estimate   Std. Error    t value     Pr(>|t|)
## (Intercept) 2.962148180 0.0188544263 157.106248 0.000000e+00
## zn          0.006368093 0.0007273285   8.755457 3.085079e-17
##                Estimate  Std. Error   t value   Pr(>|t|)
## (Intercept)  3.39386621 0.029175828 116.32459 0.0000e+00
## indus       -0.03226726 0.002231136 -14.46226 6.6786e-40
##              Estimate Std. Error    t value     Pr(>|t|)
## (Intercept) 3.016879 0.01861514 162.065879 0.0000000000
## chas        0.254935 0.07077951   3.601819 0.0003473109
##               Estimate Std. Error   t value      Pr(>|t|)
## (Intercept)  4.033594  0.0765541  52.68945 3.916668e-207
## nox         -1.801135  0.1351004 -13.33183  6.039086e-35
##               Estimate Std. Error   t value    Pr(>|t|)
## (Intercept) 0.7237362 0.12699229  5.699056 2.05217e-08
## rm          0.3676867 0.02008192 18.309338 8.76474e-58
##                 Estimate   Std. Error   t value      Pr(>|t|)
## (Intercept)  3.486027386 0.0427295176  81.58359 1.388394e-292
## age         -0.006584253 0.0005765155 -11.42077  5.033204e-27
##                Estimate Std. Error   t value      Pr(>|t|)
## (Intercept) 2.78199100 0.03524573 78.931303 7.121251e-286
## dis         0.06653993 0.00812286  8.191688  2.135908e-15
##                Estimate  Std. Error  t value     Pr(>|t|)
## (Intercept)  3.25057598 0.023666738 137.3479 0.000000e+00
## rad         -0.02262581 0.001832168 -12.3492 8.581607e-31
##                 Estimate   Std. Error   t value      Pr(>|t|)
## (Intercept)  3.590423805 3.947721e-02  90.94928 7.377290e-315
## tax         -0.001361735 8.939685e-05 -15.23247  2.261646e-43
##                Estimate  Std. Error   t value      Pr(>|t|)
## (Intercept)  4.78280316 0.135182357  35.38038 1.072037e-138
```

```
## ptratio     -0.09472987 0.007274975 -13.02133  1.288226e-33
##                Estimate  Std. Error   t value      Pr(>|t|)
## (Intercept) 2.391930910 0.067216149 35.585658 1.334106e-139
## b           0.001801594 0.000182578  9.867533  4.079766e-21
##                Estimate  Std. Error   t value      Pr(>|t|)
## (Intercept)  3.61757152 0.021970708 164.65430  0.000000e+00
## lstat       -0.04608043 0.001512547 -30.46545 2.229076e-116
```

```
# The first selected predictor: lstat
for( i in 1:12)
{
  g2 <- lm(BH$medv~ ., BH[,c(i,13,14)])
  print((summary(g2))$coef)
}
```

```
##                 Estimate  Std. Error   t value     Pr(>|t|)
## (Intercept)  3.585383164 0.021416911 167.40898 0.000000e+00
## crim        -0.009664611 0.001344696  -7.18721 2.403098e-12
## lstat       -0.040776449 0.001619712 -25.17513 4.151208e-91
##                  Estimate   Std. Error    t value      Pr(>|t|)
## (Intercept)  3.5990266036 0.0262838377 136.929266  0.000000e+00
## zn           0.0006522914 0.0005081916   1.283554  1.998890e-01
## lstat       -0.0452006064 0.0016597356 -27.233619 5.023353e-101
##                 Estimate  Std. Error    t value     Pr(>|t|)
## (Intercept)  3.637324207 0.023078111 157.609270 0.000000e+00
## indus       -0.005201936 0.001963427  -2.649416 8.316771e-03
## lstat       -0.043062973 0.001886247 -22.829980 1.067892e-79
##                Estimate  Std. Error    t value      Pr(>|t|)
## (Intercept)  3.60022834 0.021925156 164.205367  0.000000e+00
## chas         0.18560643 0.041818662   4.438364  1.114737e-05
## lstat       -0.04572441 0.001487411 -30.740946 1.424856e-117
##                Estimate  Std. Error    t value      Pr(>|t|)
## (Intercept)  3.69959669 0.054588218  67.772806 4.428703e-255
## nox         -0.18927048 0.115345194  -1.640905  1.014423e-01
## lstat       -0.04426568 0.001871701 -23.649971  1.072701e-83
##                Estimate  Std. Error    t value     Pr(>|t|)
## (Intercept)  2.7103318 0.132976881  20.381978 9.141937e-68
## rm           0.1287085 0.018628063   6.909387 1.478049e-11
## lstat       -0.0383073 0.001832836 -20.900560 2.743945e-70
##                  Estimate   Std. Error    t value     Pr(>|t|)
## (Intercept)  3.5899264742 0.0287016962 125.077154 0.000000e+00
## age          0.0007174471 0.0004801163   1.494319 1.357192e-01
## lstat       -0.0477838806 0.0018925377 -25.248575 1.829617e-91
##                Estimate  Std. Error    t value      Pr(>|t|)
## (Intercept)  3.70104851 0.039764793  93.073501 3.335141e-319
## dis         -0.01477636 0.005880299  -2.512859  1.228737e-02
## lstat       -0.04824592 0.001733945 -27.824376 7.580043e-104
##                 Estimate  Std. Error    t value     Pr(>|t|)
## (Intercept)  3.628550793 0.021850674 166.061276 0.000000e+00
## rad         -0.005462255 0.001402231  -3.895403 1.112812e-04
## lstat       -0.042825726 0.001709772 -25.047617 1.722635e-90
##                  Estimate   Std. Error    t value     Pr(>|t|)
## (Intercept)  3.7221618185 2.802228e-02 132.828659 0.000000e+00
## tax         -0.0004255336 7.406207e-05  -5.745634 1.587976e-08
```

```
## lstat       -0.0406170686 1.747948e-03 -23.236995 1.104964e-81
##                 Estimate  Std. Error    t value      Pr(>|t|)
## (Intercept)  4.36712736 0.087768059  49.757593 2.056908e-196
## ptratio     -0.04403791 0.005014224  -8.782597  2.515468e-17
## lstat       -0.04108660 0.001520155 -27.027897 4.856453e-100
##                  Estimate   Std. Error    t value      Pr(>|t|)
## (Intercept)  3.3860535048 0.0562145361  60.234483 3.805221e-232
## b            0.0005566764 0.0001248192   4.459862  1.012414e-05
## lstat       -0.0434750568 0.0015957503 -27.244273 4.466858e-101
```

```
# The second selected predictor: crim

for( i in c(2:5,6:12))
{
  g3 <- lm(BH$medv~ ., BH[,c(i,1,13,14)])
  print((summary(g3))$coef)
}
```

```
##                  Estimate   Std. Error    t value     Pr(>|t|)
## (Intercept)  3.5684211424 0.0254185643 140.386416 0.000000e+00
## zn           0.0005995715 0.0004845409   1.237401 2.165167e-01
## crim        -0.0096393761 0.0013441410  -7.171403 2.675155e-12
## lstat       -0.0399815806 0.0017416477 -22.956181 2.868742e-80
##                 Estimate  Std. Error    t value     Pr(>|t|)
## (Intercept)  3.597208830 0.022878828 157.228718 0.000000e+00
## indus       -0.002790434 0.001913801  -1.458058 1.454498e-01
## crim        -0.009295336 0.001366862  -6.800495 2.972120e-11
## lstat       -0.039360476 0.001886985 -20.858925 4.741929e-70
##                 Estimate  Std. Error    t value     Pr(>|t|)
## (Intercept)  3.569662641 0.021337798 167.292927 0.000000e+00
## chas         0.175543565 0.039899777   4.399613 1.325200e-05
## crim        -0.009459727 0.001321631  -7.157614 2.930905e-12
## lstat       -0.040552169 0.001591758 -25.476337 1.650817e-92
##                 Estimate  Std. Error     t value      Pr(>|t|)
## (Intercept)  3.595561887 0.054262961  66.2618079 2.378707e-250
## nox         -0.023030447 0.112788374  -0.2041917  8.382865e-01
## crim        -0.009605193 0.001377074  -6.9750715  9.689666e-12
## lstat       -0.040588238 0.001864959 -21.7636121  1.864961e-74
##                 Estimate  Std. Error    t value     Pr(>|t|)
## (Intercept)  2.58700925 0.125716353  20.578144 1.098978e-68
## rm           0.14122196 0.017552303   8.045779 6.228427e-15
## crim        -0.01054416 0.001271537  -8.292447 1.025048e-15
## lstat       -0.03176489 0.001892858 -16.781443 1.693018e-50
##                 Estimate   Std. Error    t value     Pr(>|t|)
## (Intercept)  3.542065811 0.0280153225 126.433162 0.000000e+00
## age          0.001093612 0.0004589046   2.383092 1.753894e-02
## crim        -0.010018231 0.0013466847  -7.439181 4.421777e-13
## lstat       -0.043178961 0.0019014866 -22.708002 4.647550e-79
##                 Estimate  Std. Error    t value     Pr(>|t|)
## (Intercept)  3.71577942 0.037535702  98.993205 0.000000e+00
## dis         -0.02374424 0.005656375  -4.197784 3.188547e-05
## crim        -0.01078862 0.001349838  -7.992528 9.146392e-15
## lstat       -0.04363933 0.001733398 -25.175596 4.701204e-91
##                  Estimate  Std. Error     t value     Pr(>|t|)
```

```
## (Intercept)  3.5874693828 0.022235051 161.3429769 0.000000e+00
## rad         -0.0005602014 0.001586657  -0.3530703 7.241839e-01
## crim        -0.0093763090 0.001574205  -5.9562175 4.858916e-09
## lstat       -0.0406008722 0.001695681 -23.9437015 4.464791e-85
##                  Estimate   Std. Error    t value     Pr(>|t|)
## (Intercept)  3.6507032152 3.065990e-02 119.070930 0.000000e+00
## tax         -0.0002389461 8.083408e-05  -2.956007 3.263106e-03
## crim        -0.0076858345 1.492956e-03  -5.148063 3.784812e-07
## lstat       -0.0387946161 1.741612e-03 -22.275120 5.989496e-77
##                 Estimate  Std. Error    t value      Pr(>|t|)
## (Intercept)  4.263216981 0.086049515  49.543766 1.994633e-195
## ptratio     -0.039530439 0.004878629  -8.102776  4.119641e-15
## crim        -0.008163876 0.001279280  -6.381615  3.988254e-10
## lstat       -0.037117368 0.001590153 -23.342017  3.784832e-82
##                  Estimate   Std. Error    t value      Pr(>|t|)
## (Intercept)  3.4425872007 0.0549606634  62.637293 2.085461e-239
## b            0.0003515787 0.0001247739   2.817726  5.027232e-03
## crim        -0.0086367579 0.0013844365  -6.238464  9.395770e-10
## lstat       -0.0396950682 0.0016537985 -24.002360  2.315105e-85
```

```
# The third selected predictor: rm

for( i in c(2:5,7:12))
{
  g4 <- lm(BH$medv~ ., BH[,c(i,6,1,13,14)])
  #print((summary(g4))$coef)
}
# The fourth selected predictor: ptratio
for( i in c(2:3,4,5,7,8:10,12))
{
  g5 <- lm(BH$medv~ ., BH[,c(i,11,6,1,13,14)])
  #print((summary(g5))$coef)
}
# The fifth selected predictor: dis
for( i in c(2:5,7,9:10,12))
{
  g6 <- lm(BH$medv~ ., BH[,c(i,8,11,6,1,13,14)])
  #print((summary(g6))$coef)
}
# The sixth selected predictor: nox
for( i in c(2:4,7,9:10,12))
{
  g7 <- lm(BH$medv~ ., BH[,c(i,5,8,11,6,1,13,14)])
  #print((summary(g7))$coef)
}
# The seventh selected predictor: b
for( i in c(2:4,7,9:10))
{
  g8 <- lm(BH$medv~ ., BH[,c(i,12,5,8,11,6,1,13,14)])
  #print((summary(g8))$coef)
}
# The eighth selected predictor: rad
for( i in c(2:4,7,10))
{
```

```
  g9 <- lm(BH$medv~ ., BH[,c(i,9,12,5,8,11,6,1,13,14)])
  #print((summary(g9))$coef)
}
# The ninth selected predictor: tax
for( i in c(2:4,7))
{
  g10 <- lm(BH$medv~ ., BH[,c(i,10,9,12,5,8,11,6,1,13,14)])
  #print((summary(g10))$coef)
}
#The tenth selected predictor: chas
for( i in c(2,3,7))
{
  g11 <- lm(BH$medv~ ., BH[,c(i,4,10,9,12,5,8,11,6,1,13,14)])
  #print((summary(g11))$coef)
}
#The eleventh selected predictor: zn
#11 variables except indus and age

#3 Adjusted R^2

Ad<-regsubsets(BH$medv ~ ., BH,nvmax = 13)

AD=summary(Ad)
AD$which
```

```
##    (Intercept)  crim    zn indus  chas   nox    rm   age   dis   rad   tax
## 1         TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## 2         TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## 3         TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## 4         TRUE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
## 5         TRUE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE
## 6         TRUE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE
## 7         TRUE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE
## 8         TRUE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE
## 9         TRUE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE
## 10        TRUE  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE
## 11        TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE
## 12        TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE
## 13        TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##    ptratio     b lstat
## 1    FALSE FALSE  TRUE
## 2     TRUE FALSE  TRUE
## 3     TRUE FALSE  TRUE
## 4     TRUE FALSE  TRUE
## 5     TRUE FALSE  TRUE
## 6     TRUE FALSE  TRUE
## 7     TRUE  TRUE  TRUE
## 8     TRUE FALSE  TRUE
## 9     TRUE  TRUE  TRUE
## 10    TRUE  TRUE  TRUE
## 11    TRUE  TRUE  TRUE
## 12    TRUE  TRUE  TRUE
## 13    TRUE  TRUE  TRUE
```

```r
AD$adjr2
```

```
##  [1] 0.6473817 0.6936576 0.7160805 0.7406047 0.7480246 0.7622027 0.7676207
##  [8] 0.7728762 0.7794784 0.7831296 0.7844479 0.7844509 0.7840825
```

```r
which(AD$adjr2==max(AD$adjr2)) #12th, which means 12 variables are most predictive except age.
```

```
## [1] 12
```

```r
#4 AIC
AIC=lm(BH$medv ~ ., BH)
step(AIC)
```

```
## Start:  AIC=-1667.19
## BH$medv ~ crim + zn + indus + chas + nox + rm + age + dis + rad +
##     tax + ptratio + b + lstat
##
##           Df Sum of Sq    RSS     AIC
## - age      1    0.0057 17.755 -1669.0
## - indus    1    0.0362 17.786 -1668.2
## <none>                 17.749 -1667.2
## - zn       1    0.1643 17.914 -1664.5
## - chas     1    0.3088 18.058 -1660.5
## - b        1    0.5339 18.283 -1654.2
## - tax      1    0.6235 18.373 -1651.7
## - nox      1    0.9351 18.684 -1643.2
## - rad      1    1.0413 18.791 -1640.3
## - rm       1    1.0637 18.813 -1639.7
## - dis      1    1.3639 19.113 -1631.7
## - ptratio  1    1.9270 19.676 -1617.0
## - crim     1    2.1995 19.949 -1610.1
## - lstat    1    7.3809 25.130 -1493.2
##
## Step:  AIC=-1669.03
## BH$medv ~ crim + zn + indus + chas + nox + rm + dis + rad + tax +
##     ptratio + b + lstat
##
##           Df Sum of Sq    RSS     AIC
## - indus    1    0.0363 17.791 -1670.0
## <none>                 17.755 -1669.0
## - zn       1    0.1593 17.914 -1666.5
## - chas     1    0.3138 18.069 -1662.2
## - b        1    0.5431 18.298 -1655.8
## - tax      1    0.6205 18.376 -1653.7
## - nox      1    0.9645 18.720 -1644.3
## - rad      1    1.0356 18.791 -1642.3
## - rm       1    1.1452 18.900 -1639.4
## - dis      1    1.5471 19.302 -1628.8
## - ptratio  1    1.9224 19.677 -1619.0
## - crim     1    2.1988 19.954 -1612.0
## - lstat    1    8.1949 25.950 -1479.0
##
```

```
## Step:  AIC=-1670
## BH$medv ~ crim + zn + chas + nox + rm + dis + rad + tax + ptratio +
##     b + lstat
##
##           Df Sum of Sq    RSS     AIC
## <none>                  17.791 -1670.0
## - zn       1    0.1451 17.936 -1667.9
## - chas     1    0.3399 18.131 -1662.4
## - b        1    0.5344 18.326 -1657.0
## - tax      1    0.6139 18.405 -1654.8
## - nox      1    0.9350 18.726 -1646.1
## - rad      1    1.0088 18.800 -1644.1
## - rm       1    1.1171 18.909 -1641.2
## - dis      1    1.7385 19.530 -1624.8
## - ptratio  1    1.8862 19.678 -1621.0
## - crim     1    2.2229 20.014 -1612.4
## - lstat    1    8.1604 25.952 -1481.0

##
## Call:
## lm(formula = BH$medv ~ crim + zn + chas + nox + rm + dis + rad +
##     tax + ptratio + b + lstat, data = BH)
##
## Coefficients:
## (Intercept)         crim           zn         chas          nox
##   4.0836823   -0.0103187    0.0010874    0.1051484   -0.7217440
##          rm          dis          rad          tax      ptratio
##   0.0906728   -0.0517059    0.0134457   -0.0005579   -0.0374259
##           b        lstat
##   0.0004127   -0.0286039
```

```
#BH$medv ~ crim + zn + chas + nox + rm + dis + rad +
    #tax + ptratio + b + lstat

#5 BIC

BH_bic=AD$bic
which(BH_bic==min(BH_bic))
```

```
## [1] 10
```

```
#The result is 10. It means that the 10 variables combined w/crim  + chas + nox + rm + dis + rad
    #   +tax + ptratio + b + lstat

#6 Mallow's Cp
BH_cp=AD$cp
which(BH_cp==min(BH_cp))
```

```
## [1] 11
```

```
# the result is 11. It means that the 11 variables combined w/

#Cross-Validation
m1=lm(BH$medv ~ crim + zn + chas + nox + rm + dis + rad + tax + ptratio + b + lstat,BH)
m2=lm(BH$medv ~ crim + zn + chas + nox + rm + dis + rad + tax + ptratio + b + lstat,BH)
m3=lm(BH$medv ~ crim + zn + chas + nox + rm + dis + rad + tax + ptratio + b + lstat+indus,BH)
m4=lm(BH$medv ~ crim + zn + chas + nox + rm + dis + rad + tax + ptratio + b + lstat,BH)
m5=lm(BH$medv ~ crim  + chas + nox + rm + dis + rad + tax + ptratio + b + lstat,BH)
m6=lm(BH$medv ~ crim + zn + chas + nox + rm + dis + rad + tax + ptratio + b + lstat,BH)
cv.scores = rep(-999, 6)
cv.scores[1] = sum((m1$residuals^2)/((1 - influence(m1)$hat)^2))
cv.scores[2] = sum((m2$residuals^2)/((1 - influence(m2)$hat)^2))
cv.scores[3] = sum((m3$residuals^2)/((1 - influence(m3)$hat)^2))
cv.scores[4] = sum((m4$residuals^2)/((1 - influence(m4)$hat)^2))
cv.scores[5] = sum((m5$residuals^2)/((1 - influence(m5)$hat)^2))
cv.scores[6] = sum((m6$residuals^2)/((1 - influence(m6)$hat)^2))
cv.scores
```

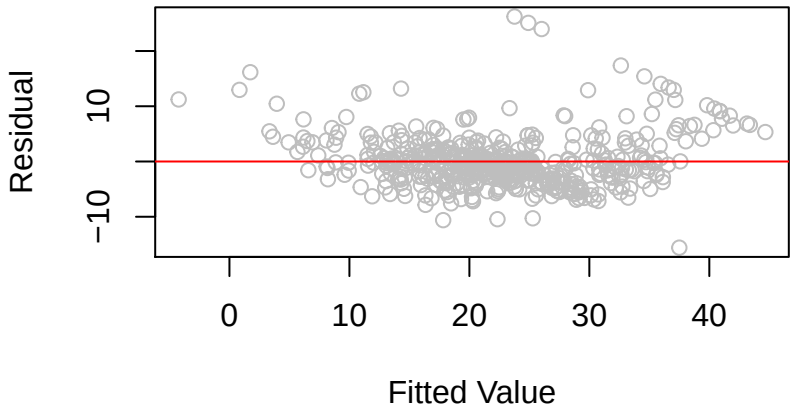
```
## [1] 19.12844 19.12844 19.12499 19.12844 19.22828 19.12844
```

```
#The smallest cv.scores 19.12844 belongs to 11 variables.
```

(2) Regression Diagnostics

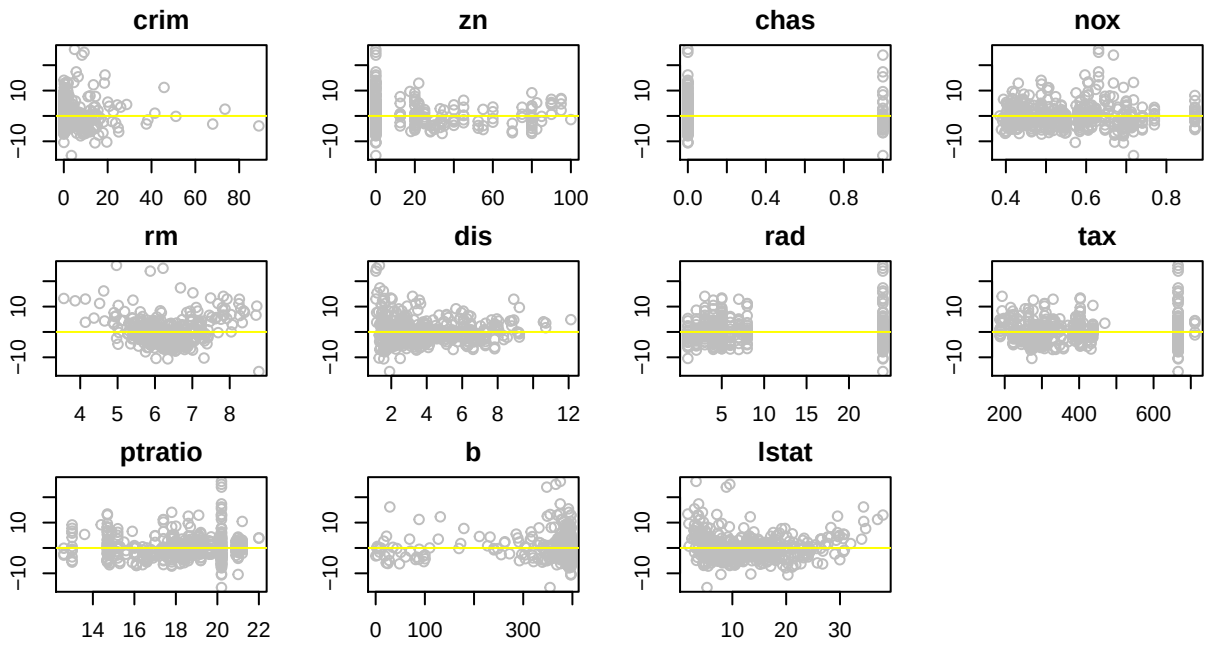
Step1: **%%plot of residuals against fitted values**

```
BH=BostonHousing
BH=BH[,-c(3,7)]          #Use the selected model
BH$chas=as.numeric(BH$chas)
BH$chas=as.numeric(BH$chas==2)
BBHH = lm(BH$medv ~ ., BH)
res=residuals(BBHH)
fitt=BBHH$fitted.values
par(mfrow = c(1, 1))
par(mar = rep(8, 4))
plot(fitt, res,xlab="Fitted Value",ylab="Residual",col="grey")
abline(a=0,b=0,col="red")
```



%%plot of residuals against explanatory variables

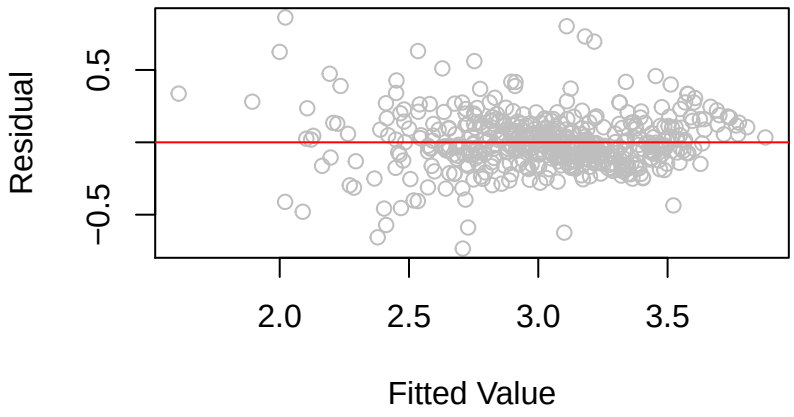
```
par(mfrow = c(4, 4))
par(mar = rep(2, 4))
for(i in 1:11)
{
  plot(BH[,i], BBHH$res, main = names(BH)[i],col="grey")
  abline(0,0,col="yellow")
}
par(mfrow = c(1, 1))
```




```
par(mar = rep(8, 4))
```

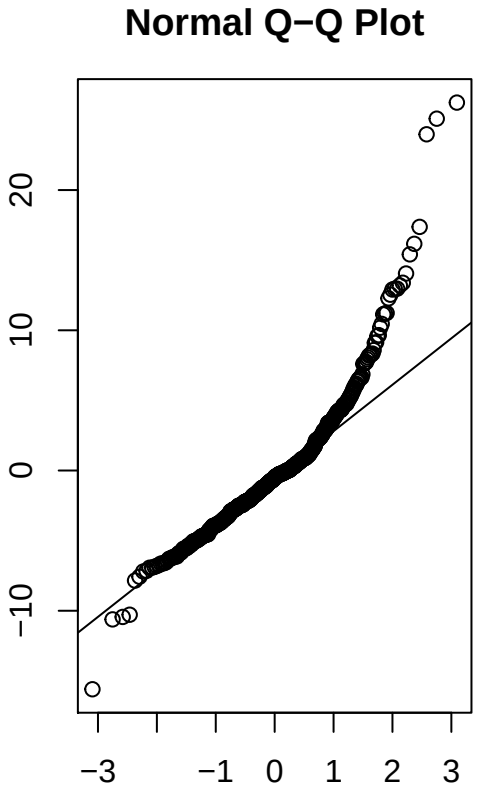
Comparing to take log

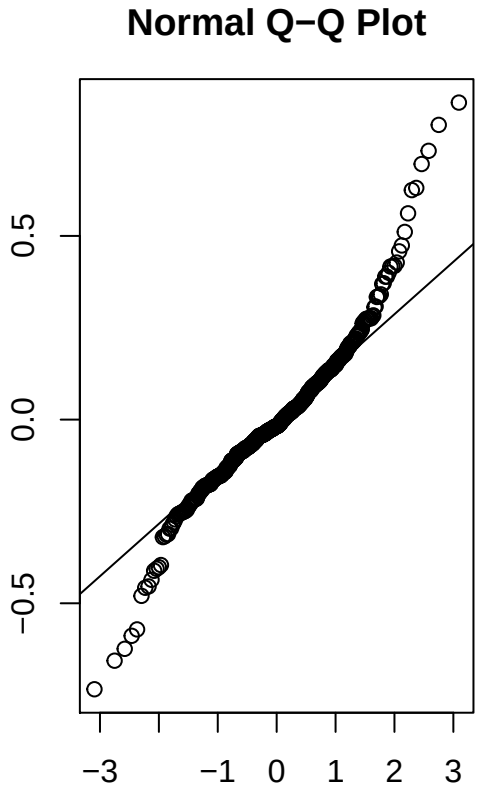
```
BBHHs = lm(log(BH$medv) ~ ., BH)
res1=residuals(BBHHs)
fitt1=BBHHs$fitted.values
par(mfrow = c(1, 1))
par(mar = rep(8, 4))
plot(fitt1, res1,xlab="Fitted Value",ylab="Residual",col="grey")
abline(a=0,b=0,col="red")
```



2. Normality:–qqnorm

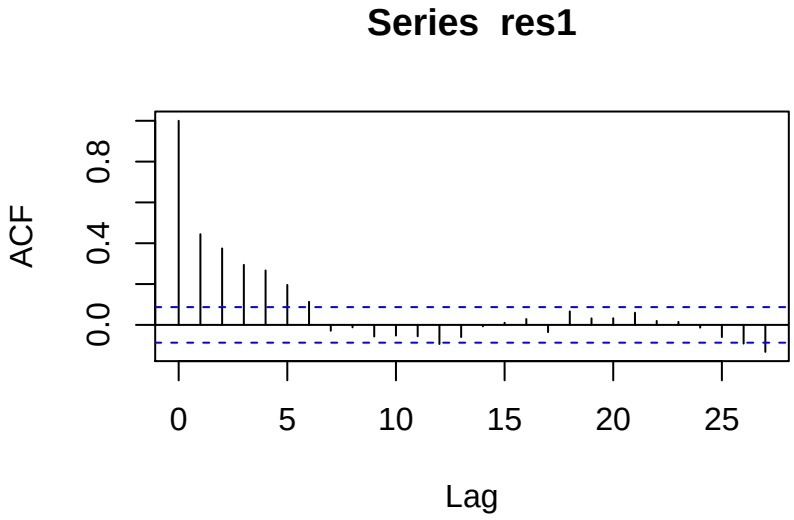
```
par(mfrow = c(1, 2))
par(mar = rep(3, 4))
qqnorm(res) #w/o log
qqline(res)
qqnorm(res1)  #w/ log
qqline(res1)
```





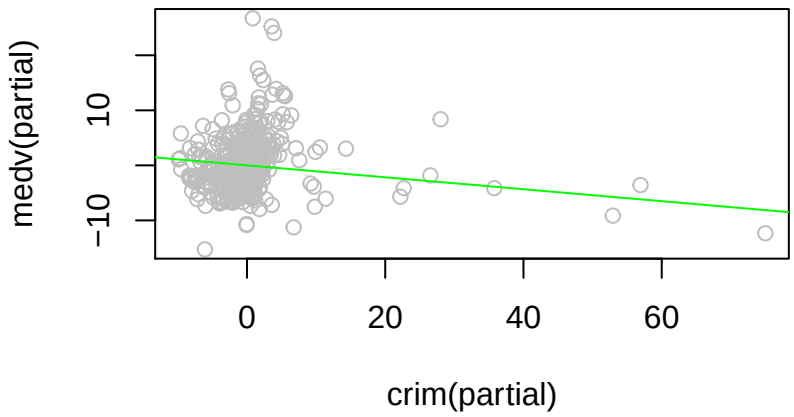
3. Correlated Error

```
par(mfrow = c(1, 1))
par(mar = rep(8, 4))
acf(res1, na.action = na.pass)
```



#Step2: detect linearity: partial regression

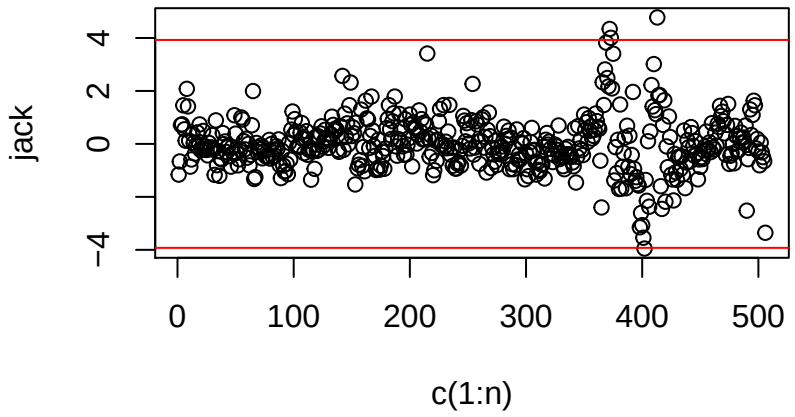
```
d1 = residuals(lm(medv ~ ., BH[,-1]))
m1 = residuals(lm(crim ~ ., BH[,-14]))
par(mfrow = c(1, 1))
par(mar = rep(8, 4))
plot(m1, d1, xlab = "crim(partial)", ylab = "medv(partial)",col="grey")
dd=coef(lm(d1 ~ m1))
abline(0, coef(BBHH)['crim'],col="green")
```



#Step3: Use three methods to detect unusual obervations

1) predicted residuals:

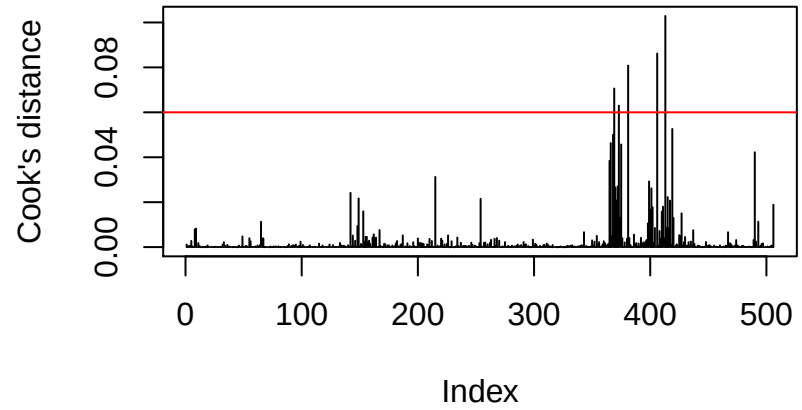
```
jack=rstudent(BBHHs)
p = 13
n = nrow(BH)
h=qt(0.05/(n*2), (n- p -2))
par(mfrow = c(1, 1))
par(mar = rep(8, 4))
plot(c(1:n),jack)
abline(a=h,b=0,col="red")
abline(a=-h,b=0,col="red")
```




```
unusual=c(which(jack>=-h) ,which(jack<h))
outliers=c(outliers,unusual)
```

2) Cooks' distance:

```
cook = cooks.distance(BBHHs)
par(mfrow = c(1, 1))
par(mar = rep(8, 4))
plot(cook, type = "h",ylab="Cook's distance")
abline(a=0.06,b=0,col="red")
```




```
outliers=c(outliers,which(cook>0.06))
```

#3) Check for leverges:

```
lev=influence(BBHHs)$hat
hi.lev=which(lev>0.04)
ratio=sum(outliers%in%hi.lev)/13
Hinf=influence(BBHHs)
#Hinf$coefficient[hi.lev,]
outliers
```

```
##                             372 373 413 402 369 373 381 406 413
## 381 406 419 354 369 372 373 372 373 413 402 369 373 381 406 413
```

Part 4

```
BBHH_new=lm(log(medv)~.,BH[-outliers,])
summary(BBHH_new)
```

```
##
## Call:
## lm(formula = log(medv) ~ ., data = BH[-outliers, ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.65631 -0.08939 -0.01445  0.09164  0.64094
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.7877629  0.1897035  19.967  < 2e-16 ***
## crim        -0.0111000  0.0018972  -5.851 9.02e-09 ***
## zn           0.0009070  0.0004939   1.836  0.06693 .
## chas         0.0819159  0.0314310   2.606  0.00944 **
## nox         -0.6233285  0.1306061  -4.773 2.41e-06 ***
## rm           0.1148987  0.0151796   7.569 1.91e-13 ***
## dis         -0.0457021  0.0069148  -6.609 1.02e-10 ***
## rad          0.0124414  0.0024133   5.155 3.69e-07 ***
## tax         -0.0005590  0.0001229  -4.548 6.84e-06 ***
## ptratio     -0.0369705  0.0046963  -7.872 2.29e-14 ***
## b            0.0005139  0.0001006   5.107 4.72e-07 ***
## lstat       -0.0260544  0.0017851 -14.595  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1718 on 485 degrees of freedom
## Multiple R-squared:  0.8154, Adjusted R-squared:  0.8112
## F-statistic: 194.8 on 11 and 485 DF,  p-value: < 2.2e-16
```

Question 4

```
Q4=BostonHousing
Q4=Q4[,c(6,13,14)]
q4=lm(medv~.,Q4)
summary(q4)
```

```
##
## Call:
## lm(formula = medv ~ ., data = Q4)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.076  -3.516  -1.010   1.909  28.131
##
## Coefficients:
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.35827    3.17283  -0.428    0.669
## rm           5.09479    0.44447  11.463   <2e-16 ***
## lstat       -0.64236    0.04373 -14.689   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.54 on 503 degrees of freedom
## Multiple R-squared:  0.6386, Adjusted R-squared:  0.6371
## F-statistic: 444.3 on 2 and 503 DF,  p-value: < 2.2e-16
```