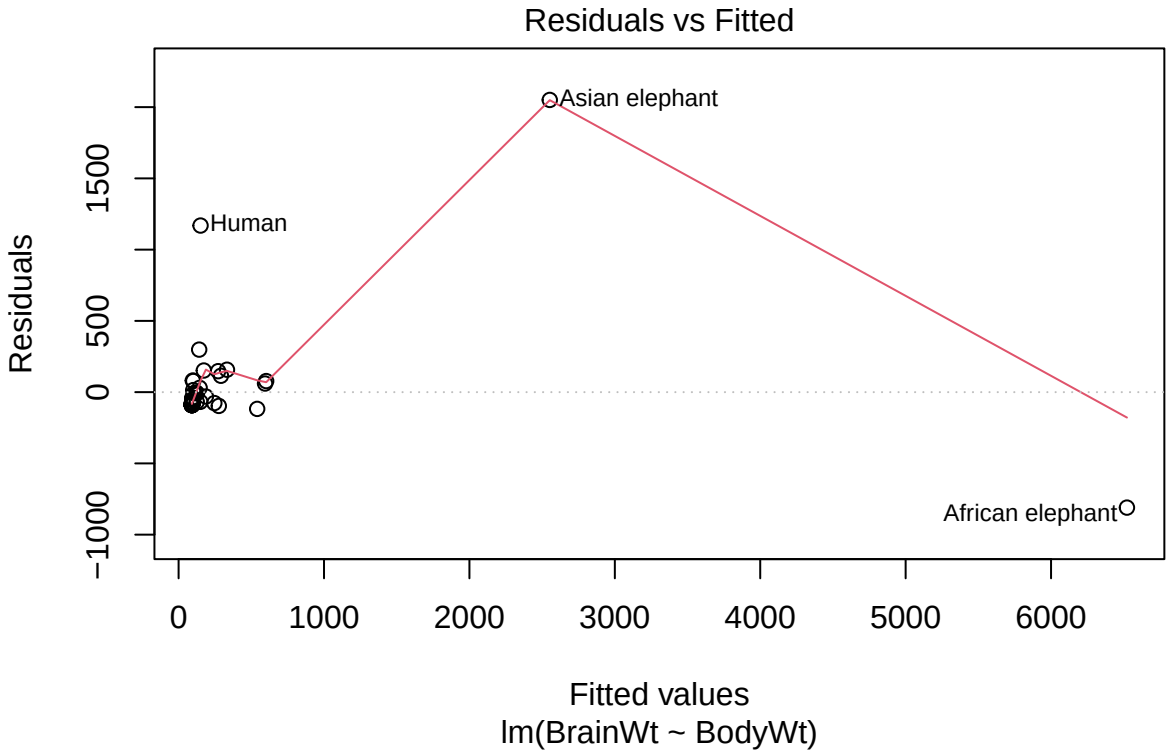
lab 4

1 Box-Cox

1

(a) $BrainWt = \beta_0 + \beta_1 BodyWt + \epsilon$.

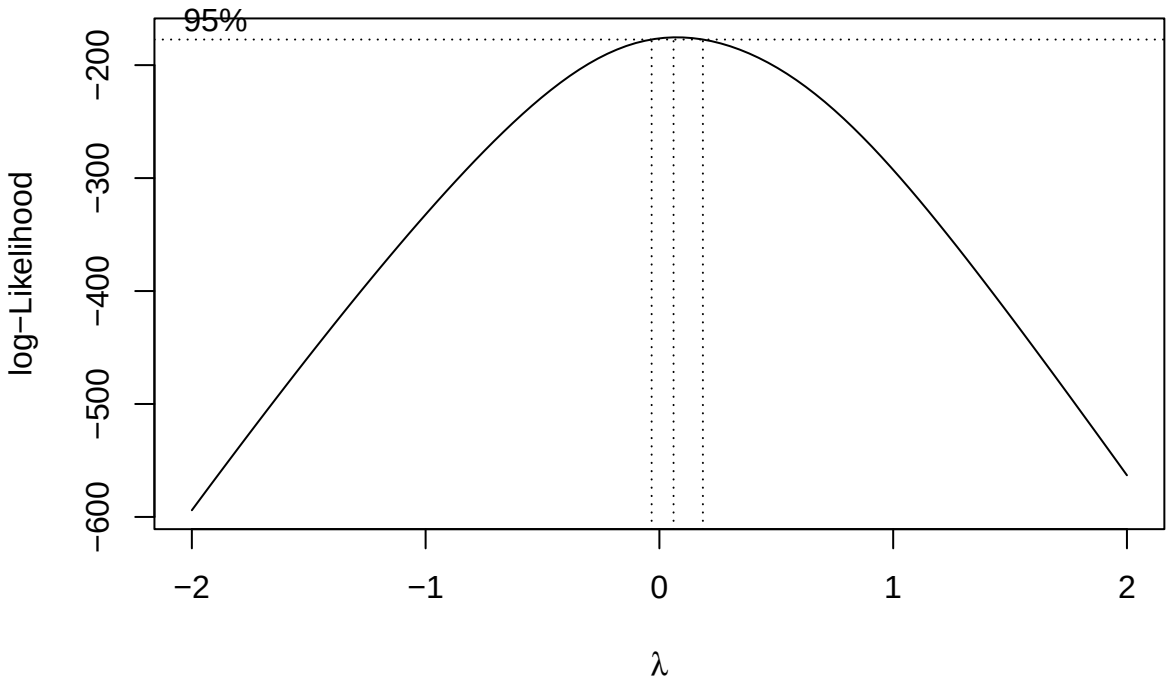
```
myfit=lm(BrainWt~BodyWt,data=brains)
plot(myfit,which=1)
```



We can see that the residual plot isn't ideal: there are some data points that are far from the others.

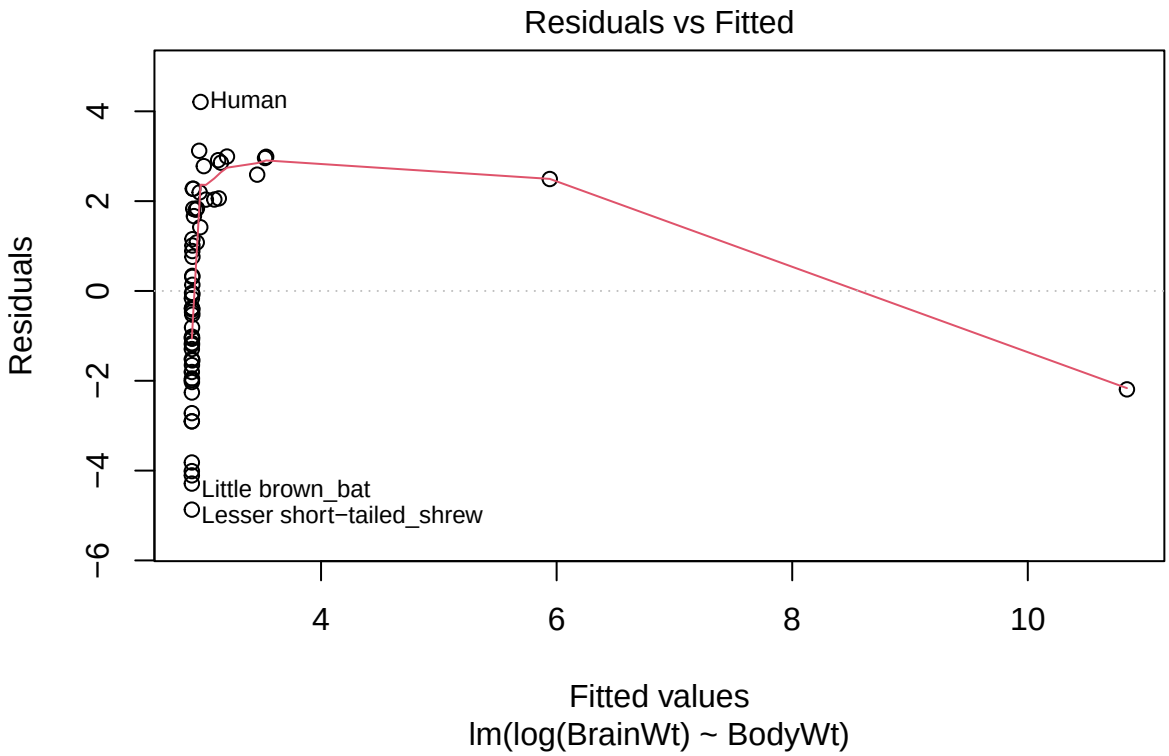
(b) Consider Box-Cox transformation.

```
library(MASS)
boxcox(BrainWt~BodyWt,data=brains)
```



From the plot, select $\lambda = 0, i.e.$ logarithmic transform.

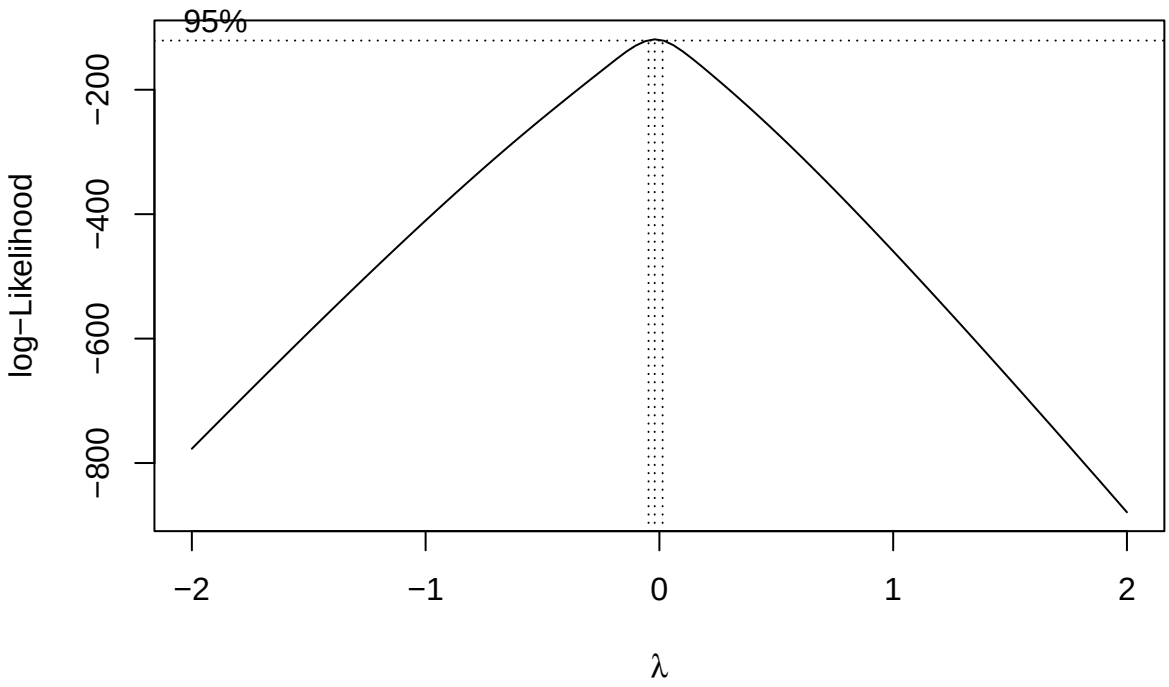
```
myfit_2=lm(log(BrainWt)~BodyWt,data=brains)
plot(myfit_2,which=1)
```



We can see that the trend(red line) is not parallel. The result is still unsatisfactory.

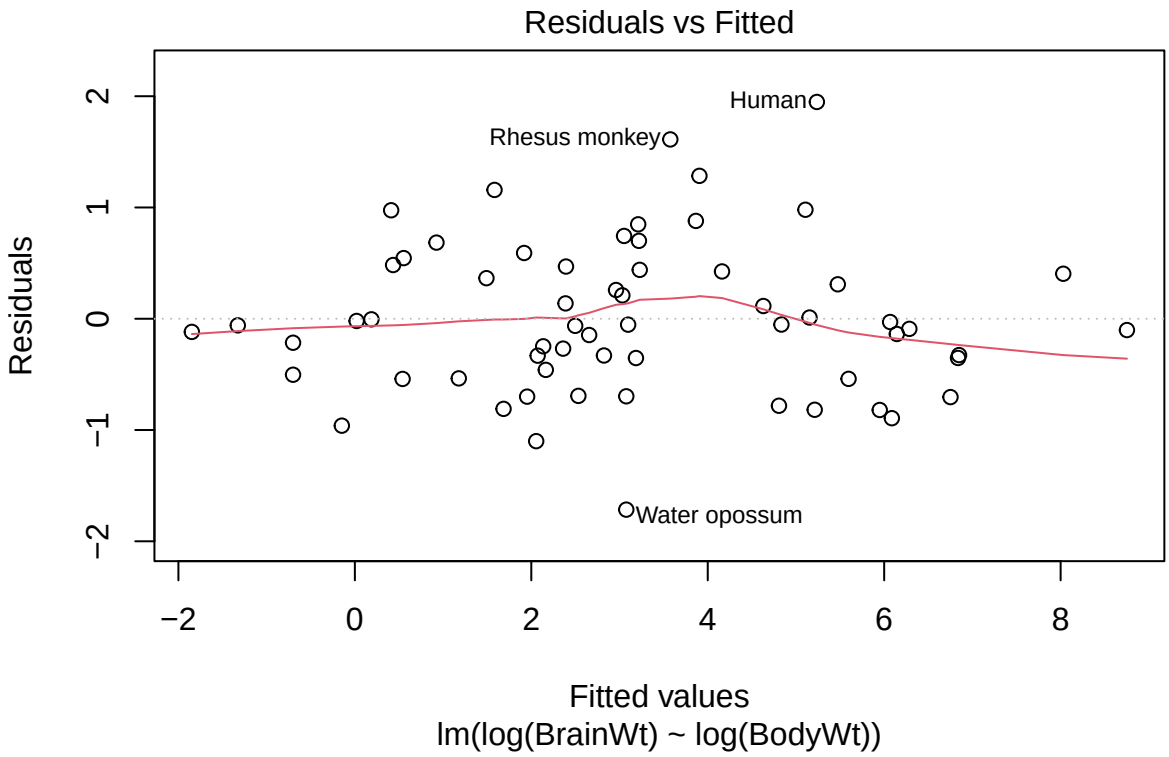
(c) Consider performing Box-Cox transformation on bodyweight.

```
boxcox(BodyWt~log(BrainWt),data=brains)
```



From the plot, still select logarithmic transform on bodyweight.

```
myfit_3=lm(log(BrainWt)~log(BodyWt),data=brains)
plot(myfit_3,which=1)
```



The residual plot is much better, as the red line is close to parallel, and the data points scatter along the red line.

2 Regression Diagnostics

2

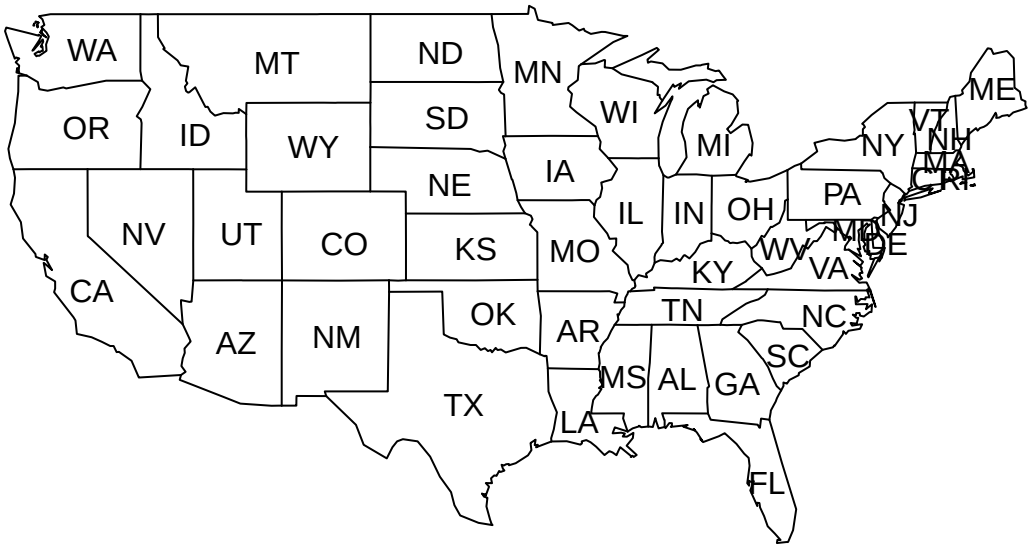
```
edu=read.table("http://staff.ustc.edu.cn/~ynyang/2025/lab/edu.xls",head=T,row.name=1)
install.packages("maps",, repos = "https://cloud.r-project.org")
```

```
## package 'maps' successfully unpacked and MD5 sums checked

## Warning: cannot remove prior installation of package 'maps'

## Warning in file.copy(savedcopy, lib, recursive = TRUE): problem copying
## D:\360Downloads\Software\R-4.5.1\library\00LOCK\maps\libs\x64\maps.dll to
## D:\360Downloads\Software\R-4.5.1\library\maps\libs\x64\maps.dll: Permission
## denied

## Warning: restored 'maps'

##
## The downloaded binary packages are in
##  D:\Temp\RtmpQruXN7\downloaded_packages
```

```
library(maps)
```

```
## Warning: package 'maps' was built under R version 4.5.2
```

```
map("state")
text(state.center,state.abb)
```

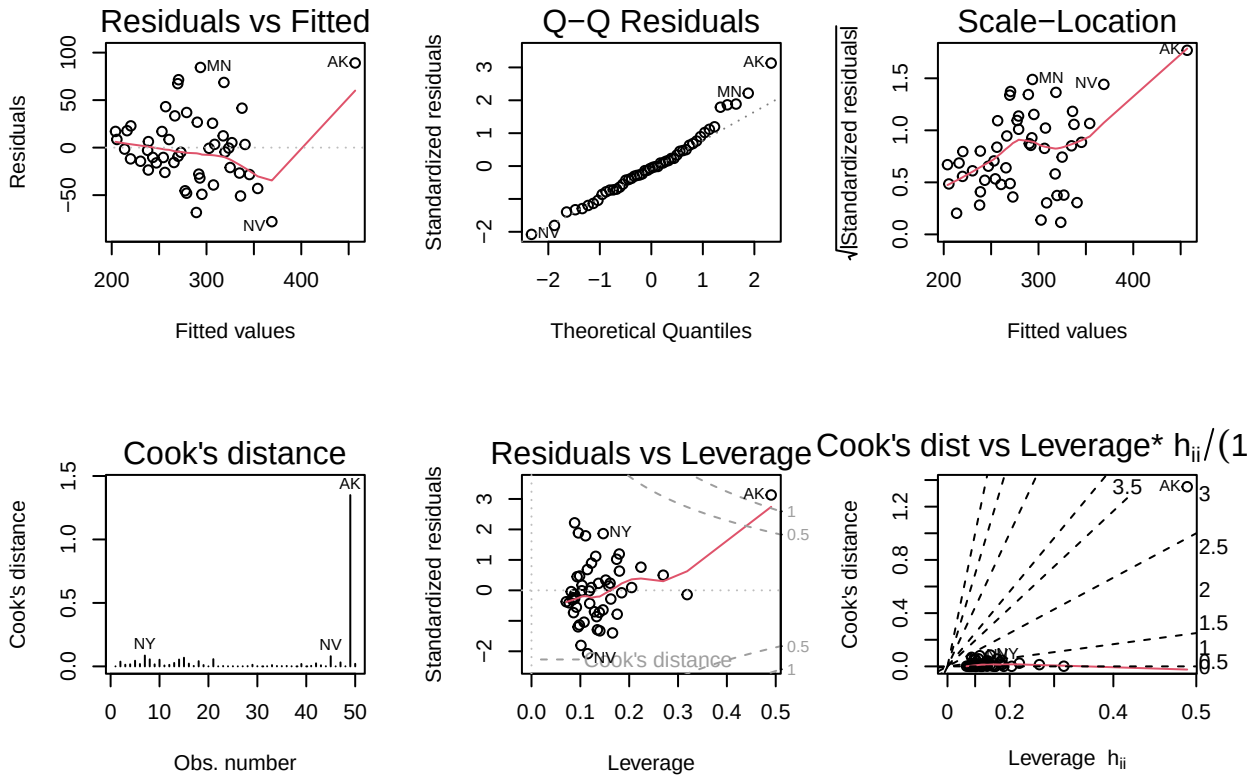



```
#edu
edu[,"Region"]=as.factor(edu[,"Region"])
edu[,"Region"]
```

```
##  [1] 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 4
## [39] 4 4 4 4 4 4 4 4 4 4 4 4
## Levels: 1 2 3 4
```

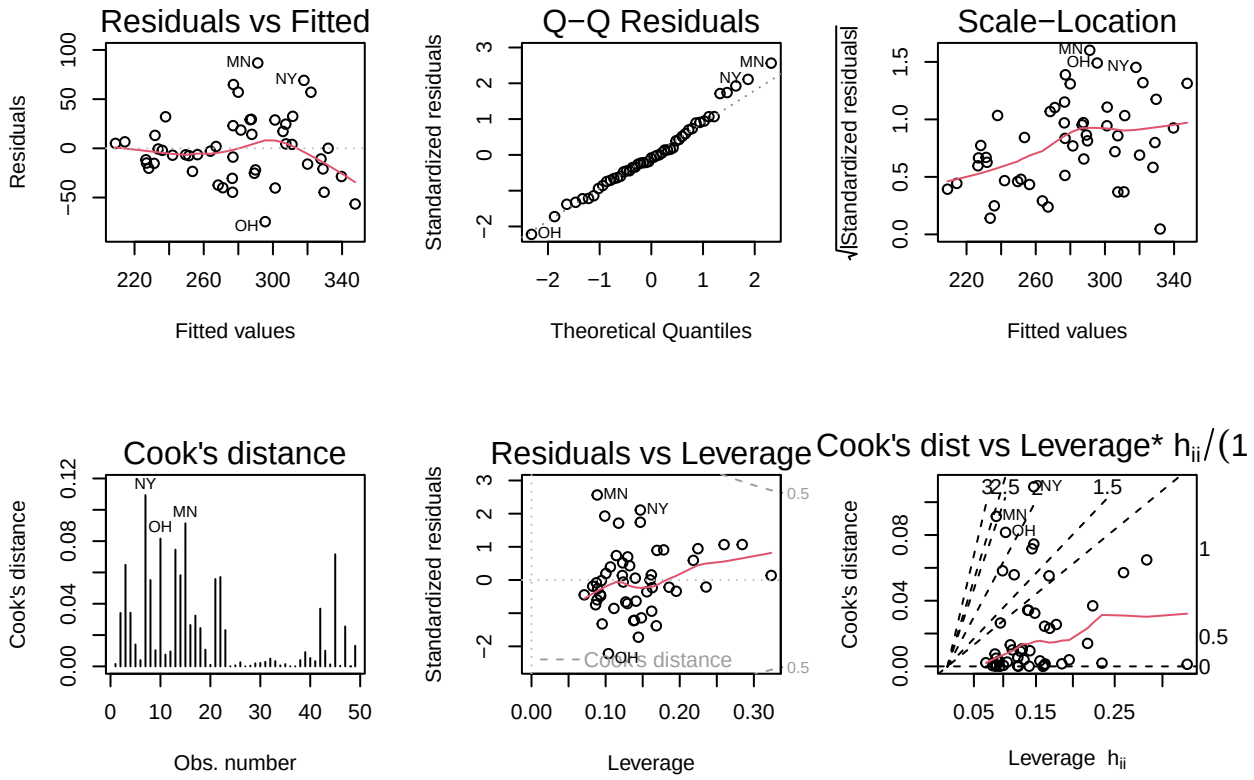
We hope to find the relation between expenditure and other variables. Assume $Expenditure_i = \beta_0 + \beta_1 \times Income_i + \beta_3 \times Urban_i + \sum_{k=1}^{4} \alpha_k I_{(Region_i=k)} + \epsilon_i$.

```
fit1=lm(Expenditure~.,data=edu)
par(mfrow=c(2,3))
plot(fit1,which=1:6)
```



From the result, we can see that AK is a high-influence point. Therefore, we consider deleting AK.

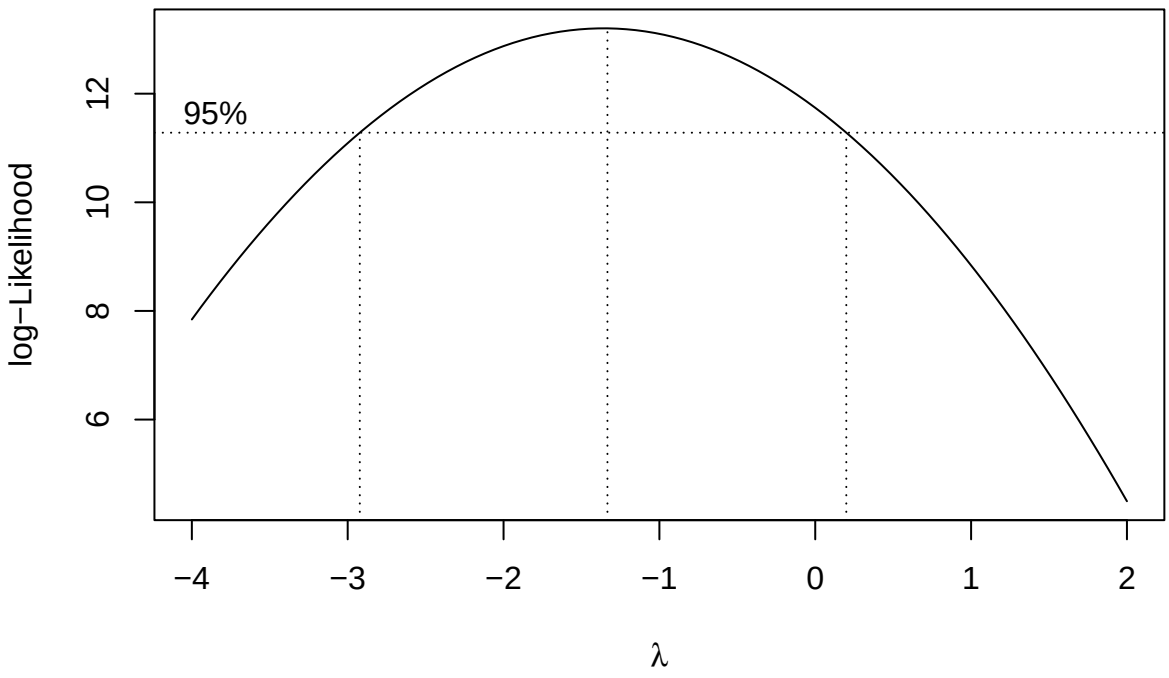
```
edu_noAK <- edu[rownames(edu) != "AK", ]
fit_noAK=lm(Expenditure~.,data=edu_noAK)
par(mfrow=c(2,3))
plot(fit_noAK,which=1:6)
```



We can still see that the result is heteroscedastic and nonlinear, implying AK isn't the only reason.

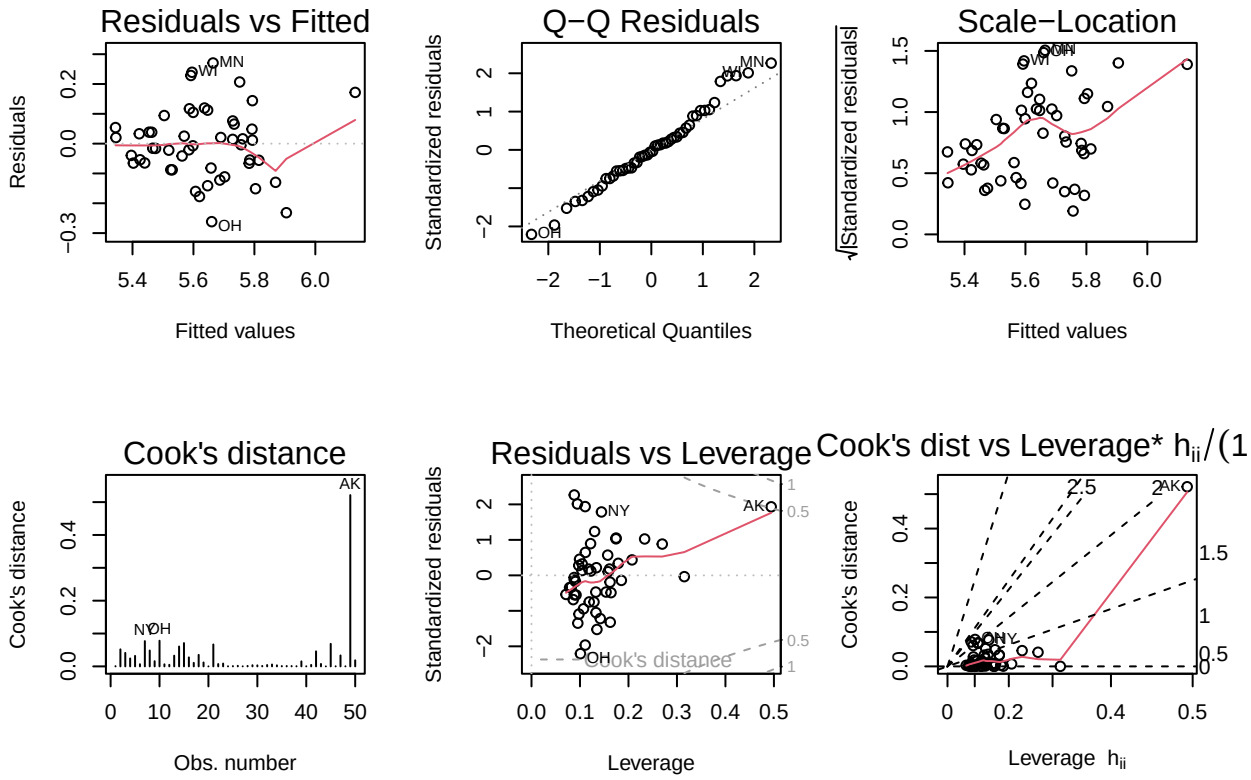
Next,consider Box-Cox transformation on Expenditure and Income.

```
boxcox(Expenditure~.,data=edu_noAK,lambda = seq(-4, 2, by = 1))
```




```
#boxcox(Income~)
```

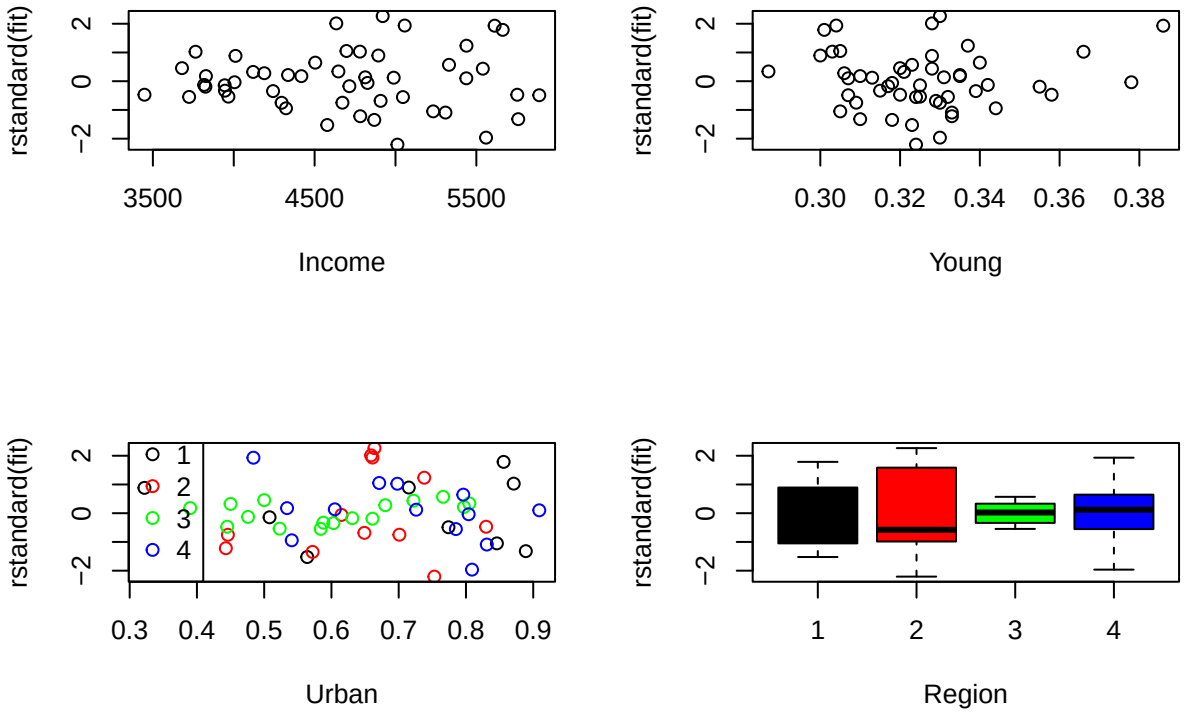
```
fit2=lm(log(Expenditure)~log(Income)+Young+Urban+Region,data=edu)
par(mfrow=c(2,3))
plot(fit2,which=1:6)
```



The result shows that heteroscedasticity and nonlinearity still exist after box-cox transformation.

```
z=rstandard(fit2)
par(mfrow = c(2, 2))
plot(edu$Income, z,
     xlab = "Income",
     ylab = "rstandard(fit)",
     pch = 1)

plot(edu$Young, z,
     xlab = "Young",
     ylab = "rstandard(fit)",
     pch = 1)

cols <- c("black", "red", "green", "blue")
plot(edu$Urban, z,
     col = cols[edu$Region],
     pch = 1,
     xlab = "Urban",
     ylab = "rstandard(fit)")


legend("bottomleft",
       legend = levels(edu$Region),
       col = cols,
       pch = 1)
boxplot(z ~ edu$Region,
```

```
        xlab = "Region",
        ylab = "rstandard(fit)",
        col = cols)
```

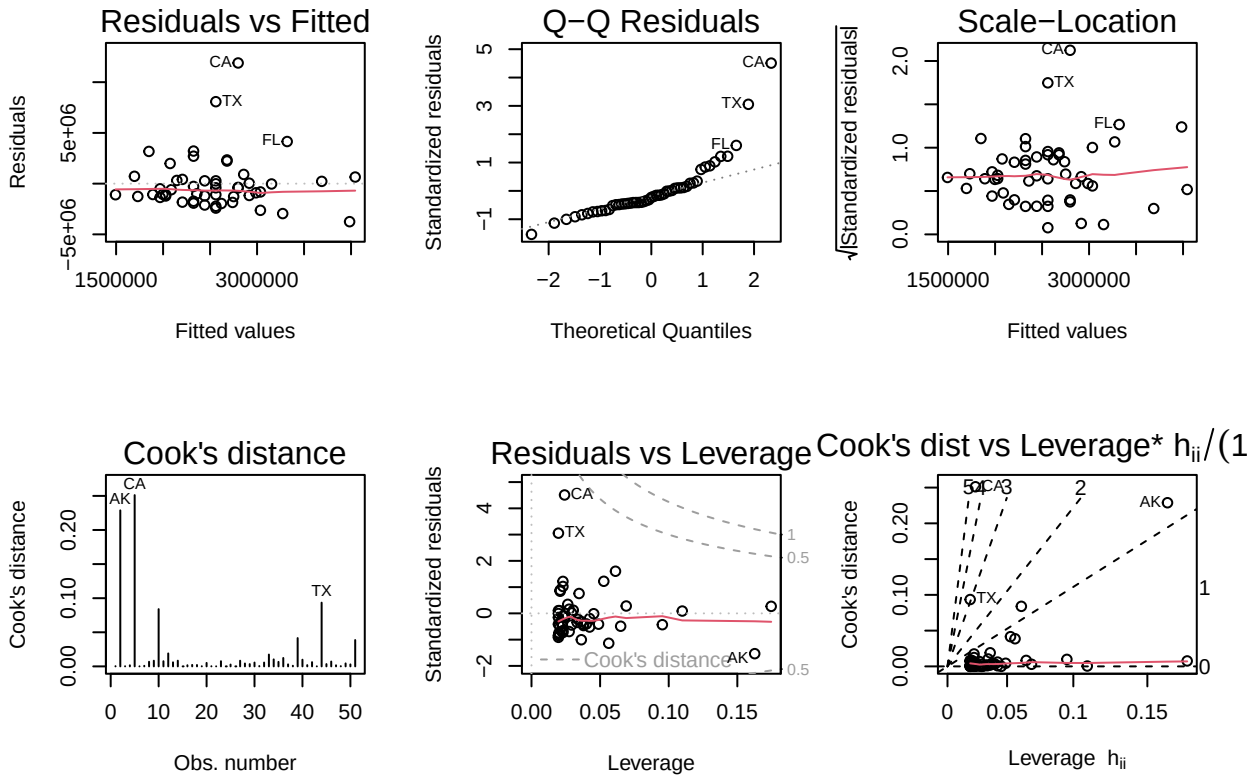


From the fourth plot we can see that there is a significant difference of variance of residuals between each region. Therefore, we should assume that the variance of error in each region differs.

Exercise 1

The goal is to study whether a state with higher tax has less fuel consumption. Here FuelC is response variable, and Tax is independent variable.

```
#head(fuel2001)
#help(fuel2001)
fit_fuel=lm(FuelC~Tax,data=fuel2001)
par(mfrow=c(2,3))
plot(fit_fuel,which=1:6)
```



In plot 1, the red line is close to parallel, implying that the residuals are independent of data points(fitted values). Therefore, there is no need for Box-Cox transformation.

However, from plot 1, we can see that there are several outliers(whose residuals are higher than others). From plot 2, we can see several high-leverage points(that deviate from the line y=x). From Plot 4 and 5, there is no high-influence point.

Since there is no high-influence point, I think it is unnecessary to delete any data.

```
summary(fit_fuel)
```

```
##
## Call:
## lm(formula = FuelC ~ Tax, data = fuel2001)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3749415 -1268820  -599344   365998 11893315
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4933916    1715628   2.876  0.00595 **
## Tax          -118638      83077  -1.428  0.15962
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2670000 on 49 degrees of freedom
```

```
## Multiple R-squared:  0.03996,    Adjusted R-squared:  0.02036
## F-statistic: 2.039 on 1 and 49 DF,  p-value: 0.1596
```

Since the coefficient<0, indeed a state with higher tax has less fuel consumption. If the tax increases 1 cent/gallon, then the fuel consumption would decrease 118638 thousand gallons. However, the p-value is 0.1596>0.05. Therefore, the effect of tax on fuel consumption is not significant.

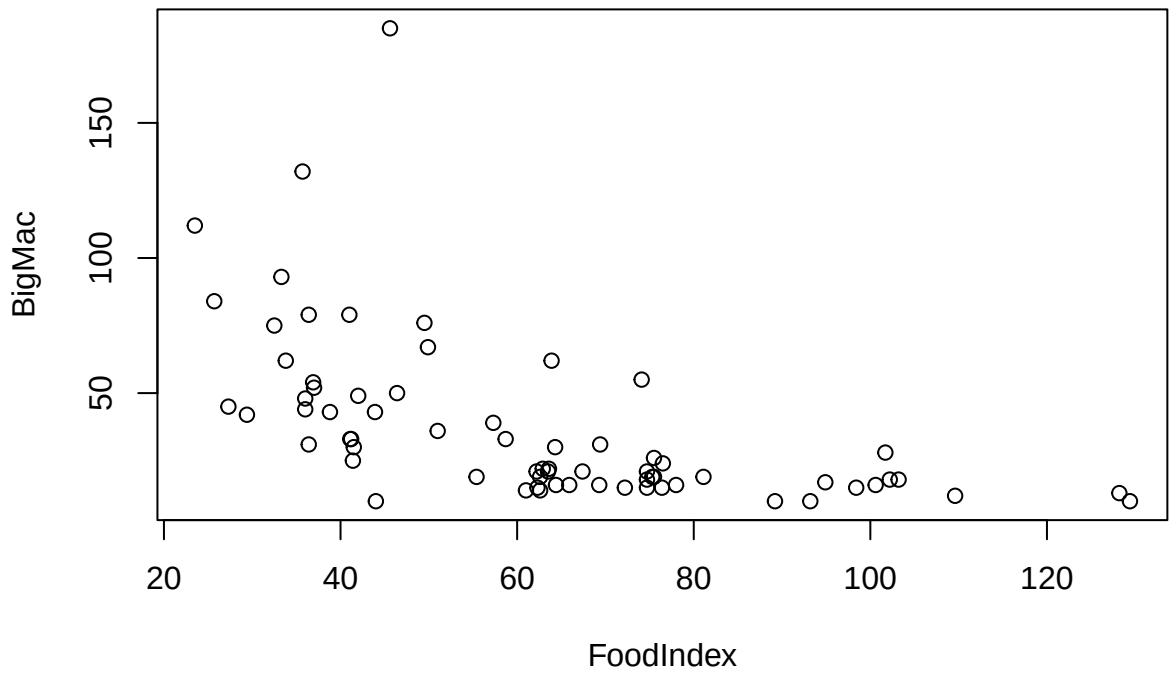
Exercise 2

The goal is to find the relation between price and other factors.

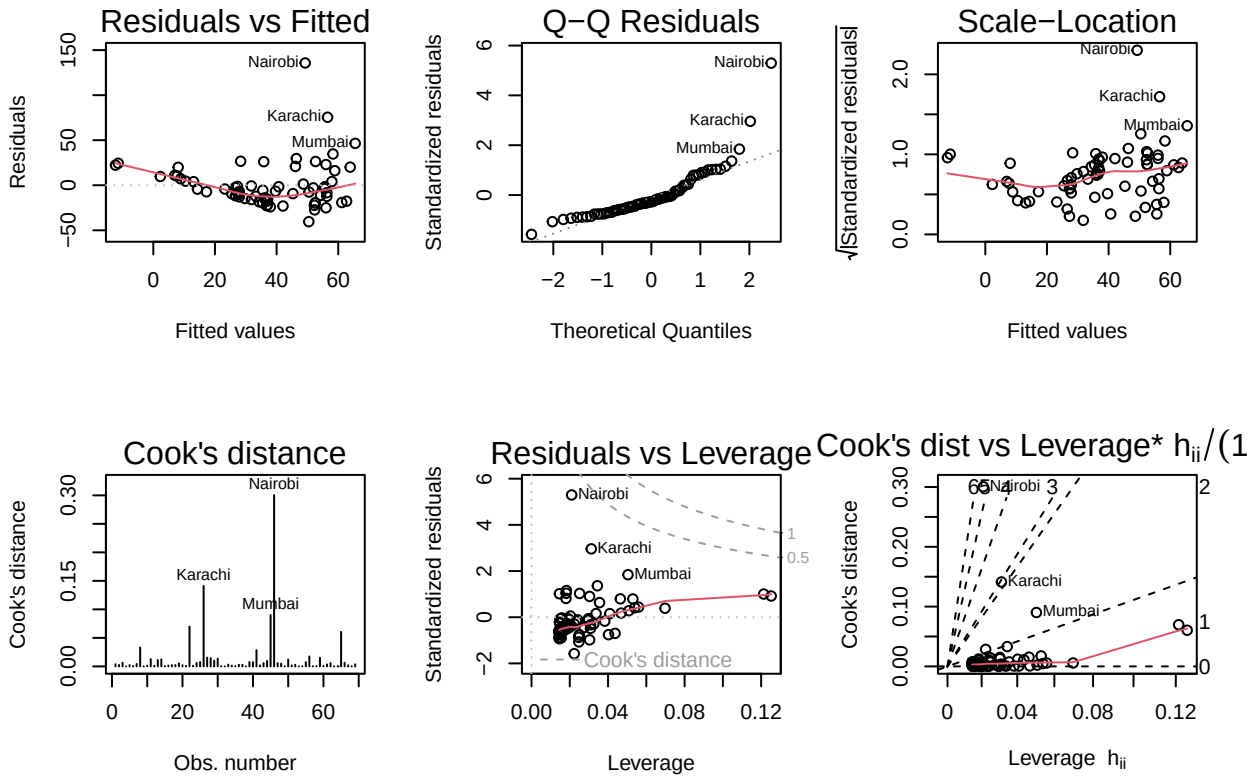
```
#head(BigMac2003)
data=BigMac2003
#data
```

(a)

```
plot(data$FoodIndex,data$BigMac,
     xlab="FoodIndex",
     ylab="BigMac")
```

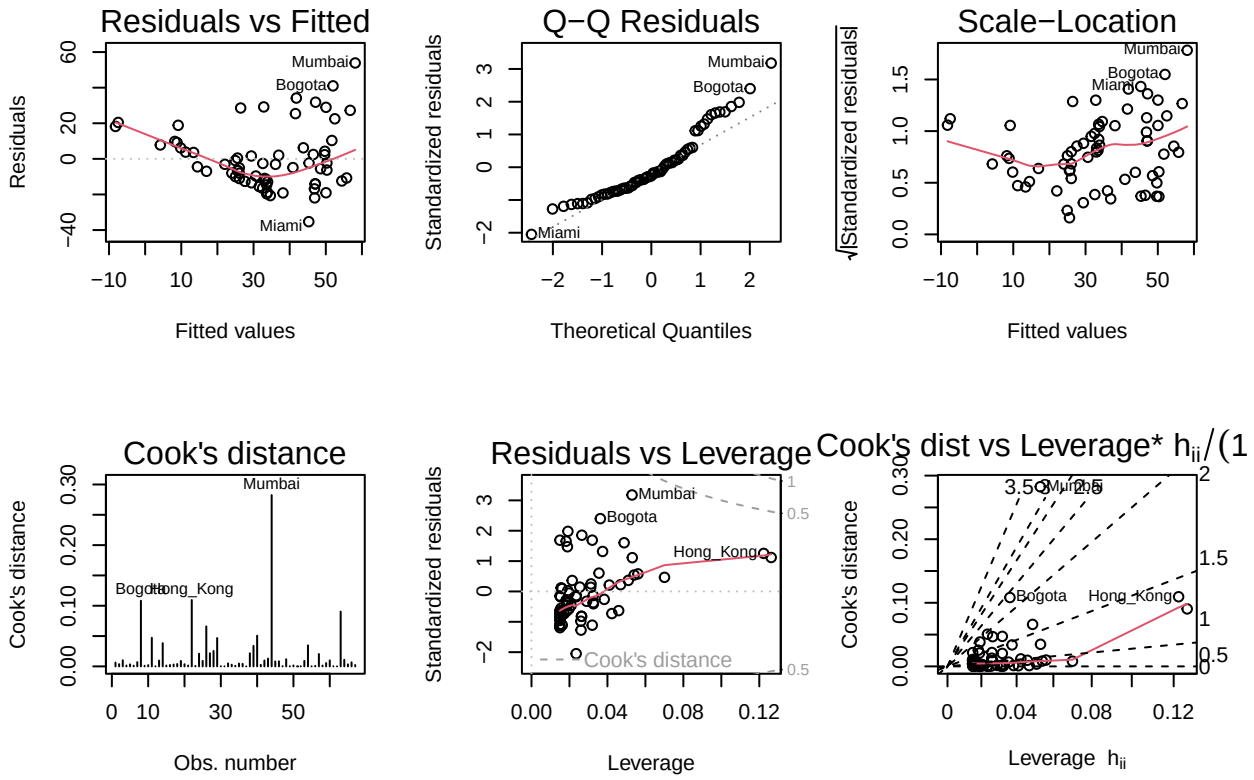



```
fit_mac=lm(BigMac~FoodIndex,data=data)
par(mfrow=c(2,3))
plot(fit_mac,which=1:6)
```



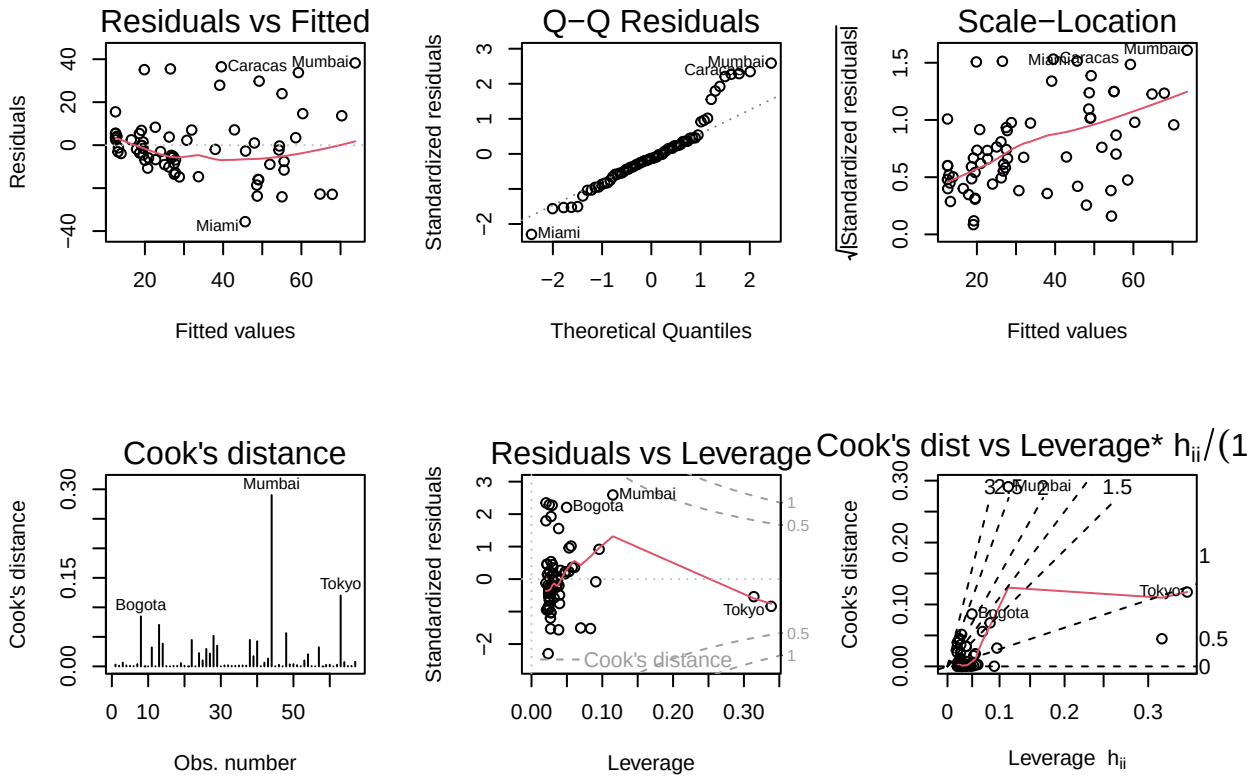
From plot 1, Nairobi and Karachi are outliers. Consider deleting them:

```
data_delete=data[!row.names(data) %in% c("Nairobi", "Karachi"), ]
fit_mac=lm(BigMac~FoodIndex,data=data_delete)
par(mfrow=c(2,3))
plot(fit_mac,which=1:6)
```



Still, the result is unsatisfactory. Since the red line in plot 1 is U-shaped, consider adding a squared term:

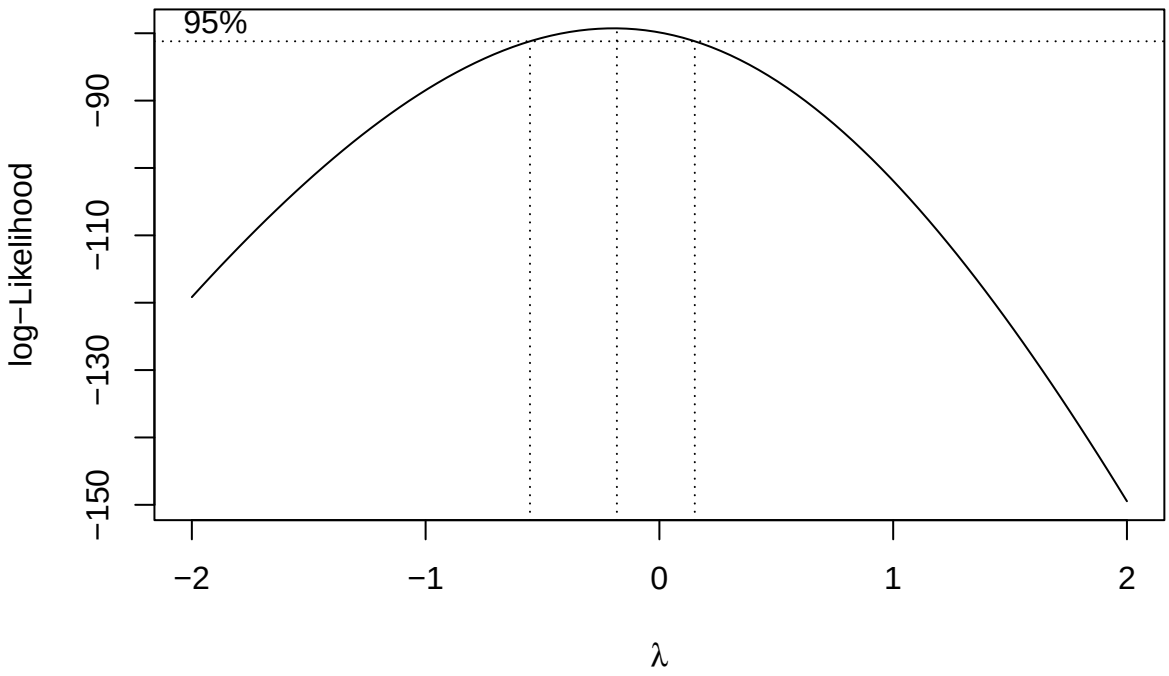
```
fit_mac2=lm(BigMac~FoodIndex+I(FoodIndex^2),data=data_delete)
par(mfrow=c(2,3))
plot(fit_mac2,which=1:6)
```



The result is better, but still the red line in plot 1 isn't straight enough.

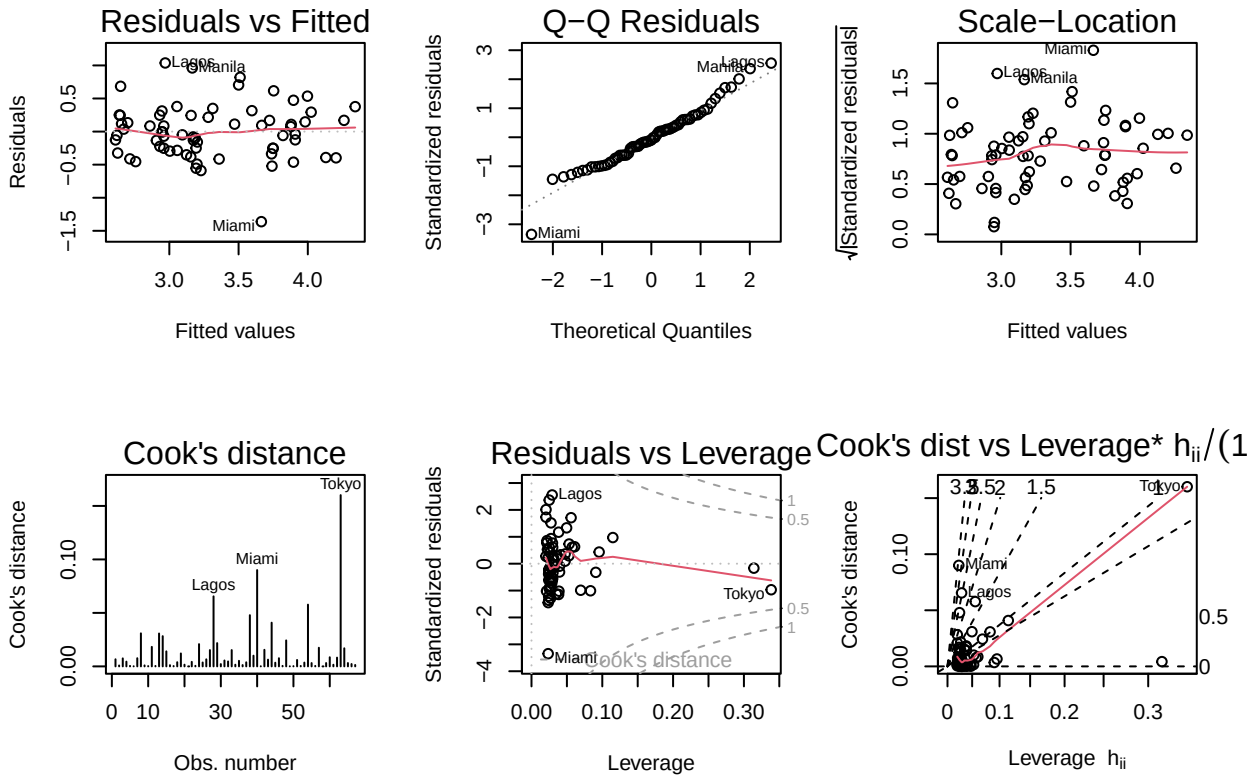
Consider Box-Cox transformation:

```
boxcox(BigMac~FoodIndex+I(FoodIndex^2),data=data_delete)
```



From the result, take log transform on BigMac:

```
fit_mac3=lm(log(BigMac)~FoodIndex+I(FoodIndex^2),data=data_delete)
par(mfrow=c(2,3))
plot(fit_mac3,which=1:6)
```

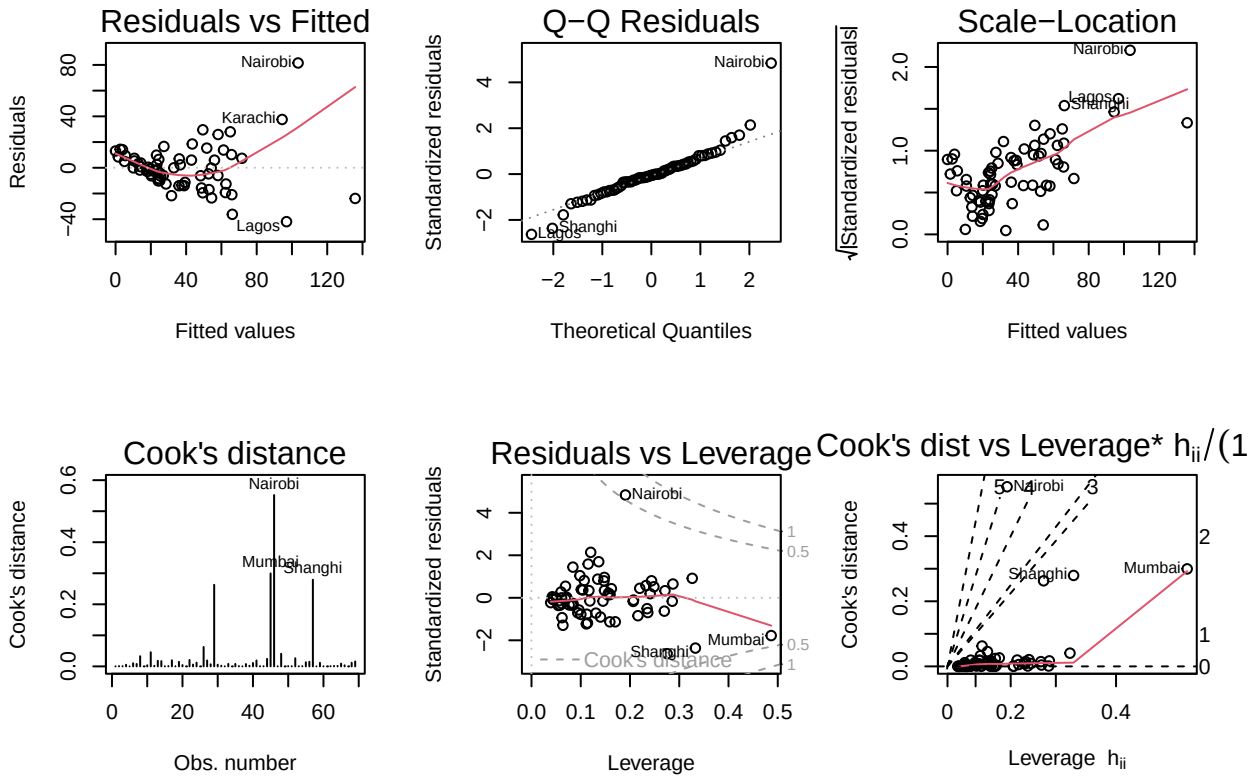


Now the result is much better.

Therefore, eventually consider $log(BigMac) = \beta_0 + \beta_1 FoodIndex + \beta_2 FoodIndex^2 + \epsilon$.

(b)

```
fit_all=lm(BigMac~.,data=data)
par(mfrow=c(2,3))
plot(fit_all,which=1:6)
```




```
summary(fit_all)
```

```
##
## Call:
## lm(formula = BigMac ~ ., data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -41.916 -10.053  -1.024   7.359  81.512
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 40.359743  16.884721   2.390   0.0200 *
## Bread        0.387238   0.183319   2.112   0.0389 *
## Rice         0.965387   0.182620   5.286  1.9e-06 ***
## FoodIndex   -0.512792   0.194416  -2.638   0.0107 *
## Bus         -0.229961   4.533740  -0.051   0.9597
## Apt          0.003929   0.007795   0.504   0.6161
## TeachGI      1.848863   1.363304   1.356   0.1802
## TeachNI     -2.287929   1.830213  -1.250   0.2162
## TaxRate     -0.775878   0.397161  -1.954   0.0555 .
## TeachHours   0.295898   0.335860   0.881   0.3819
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 18.73 on 59 degrees of freedom
## Multiple R-squared:  0.6918, Adjusted R-squared:  0.6448
## F-statistic: 14.71 on 9 and 59 DF,  p-value: 3.744e-12
```

We can see that the p-values of Bus, Apt, TeachGI, TeachNI and TeachHours are relatively large. Consider deleting these variables and use anova to check:

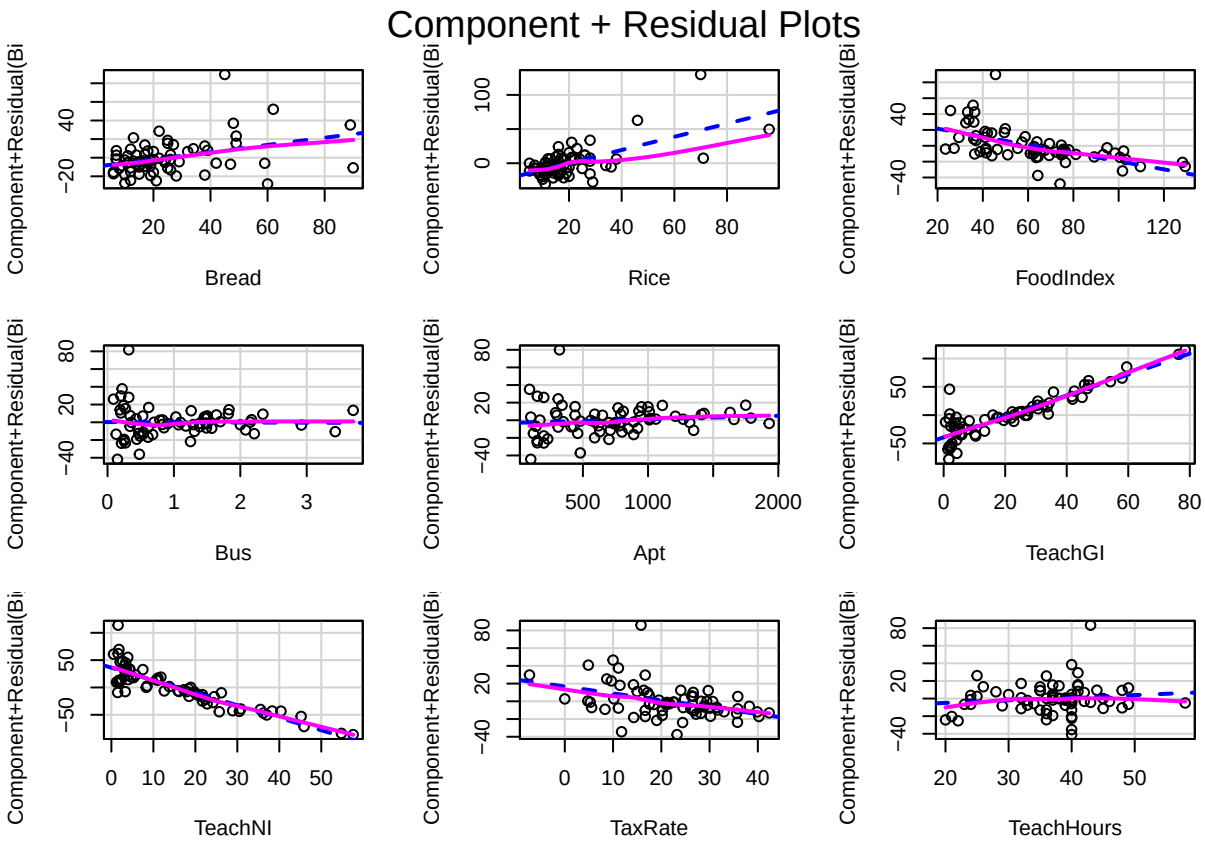
```
fit_reduced=lm(BigMac~Bread+Rice+FoodIndex+TaxRate,data=data)
anova(fit_reduced,fit_all)
```

```
## Analysis of Variance Table
##
## Model 1: BigMac ~ Bread + Rice + FoodIndex + TaxRate
## Model 2: BigMac ~ Bread + Rice + FoodIndex + Bus + Apt + TeachGI + TeachNI +
##     TaxRate + TeachHours
##   Res.Df   RSS Df Sum of Sq      F Pr(>F)
## 1     64 21811
## 2     59 20692  5    1119.1 0.6382 0.6714
```

Since the p-value is 0.6714, it makes sense to omit these variables.

Use crPlots to check if we should add nonlinear terms:

```
library(car)
crPlots(fit_all)
```

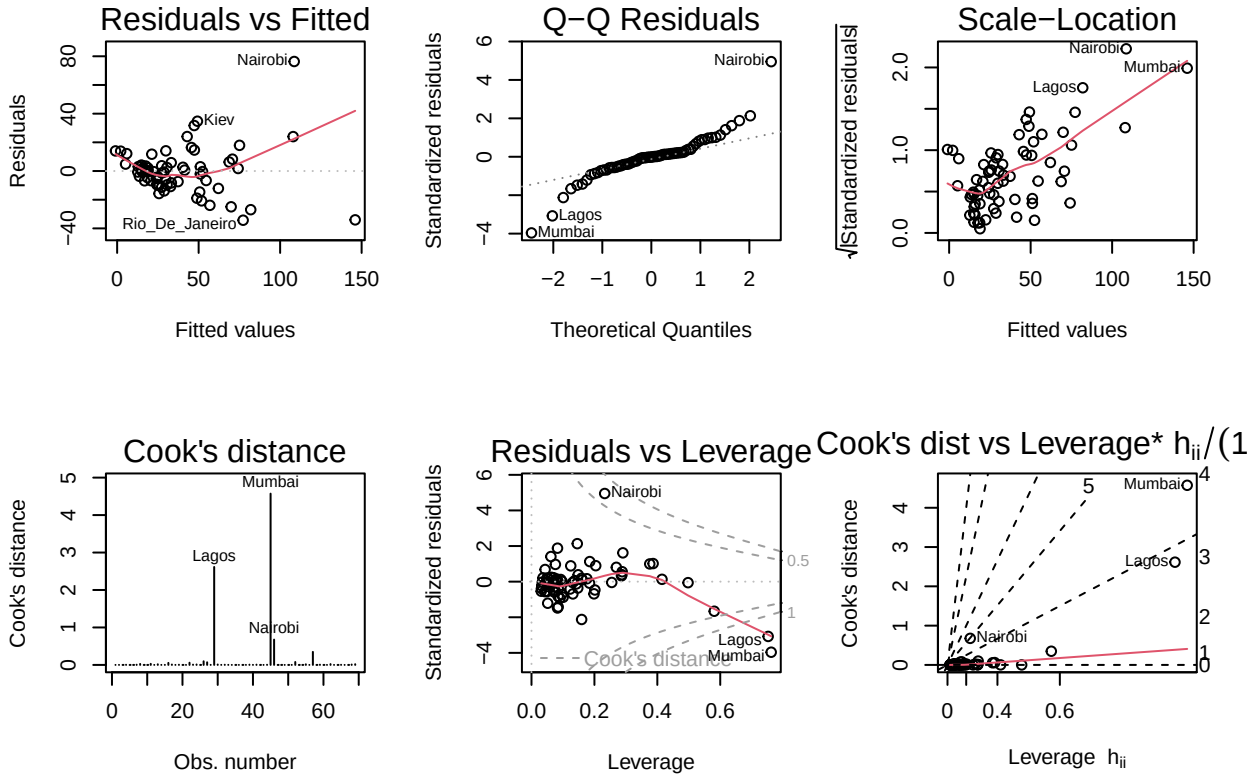


Since the red lines are all close to linear, consider interaction between each two variables instead of squared term of each variable.

```r
# interaction
fit_interaction=lm(BigMac~Bread+Rice+FoodIndex+TaxRate
                   +Bread:Rice
                   +Bread:FoodIndex
                   +Bread:TaxRate
                   +Rice:FoodIndex
                   +Rice:TaxRate
                   +FoodIndex:TaxRate
                   ,data=data)
anova(fit_reduced,fit_interaction)
```

```
## Analysis of Variance Table
##
## Model 1: BigMac ~ Bread + Rice + FoodIndex + TaxRate
## Model 2: BigMac ~ Bread + Rice + FoodIndex + TaxRate + Bread:Rice + Bread:FoodIndex +
##     Bread:TaxRate + Rice:FoodIndex + Rice:TaxRate + FoodIndex:TaxRate
##   Res.Df   RSS Df Sum of Sq      F  Pr(>F)
## 1     64 21811
## 2     58 17978  6    3833.3 2.0612 0.07186 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

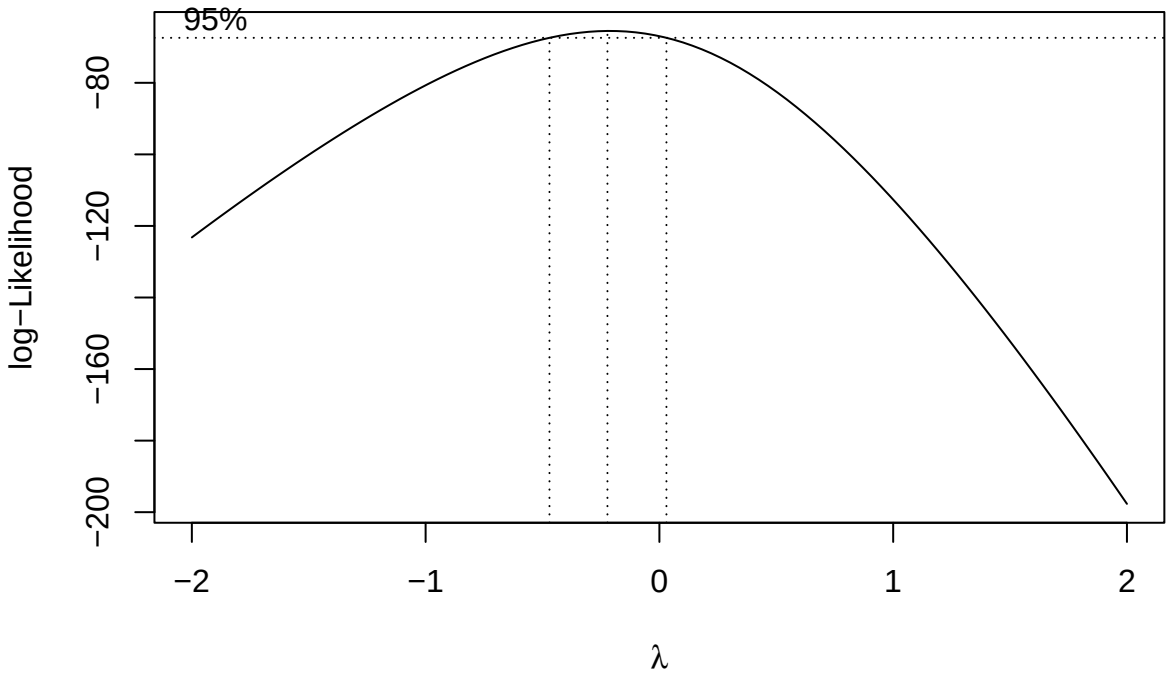
```r
par(mfrow=c(2,3))
plot(fit_interaction,which=1:6)
```



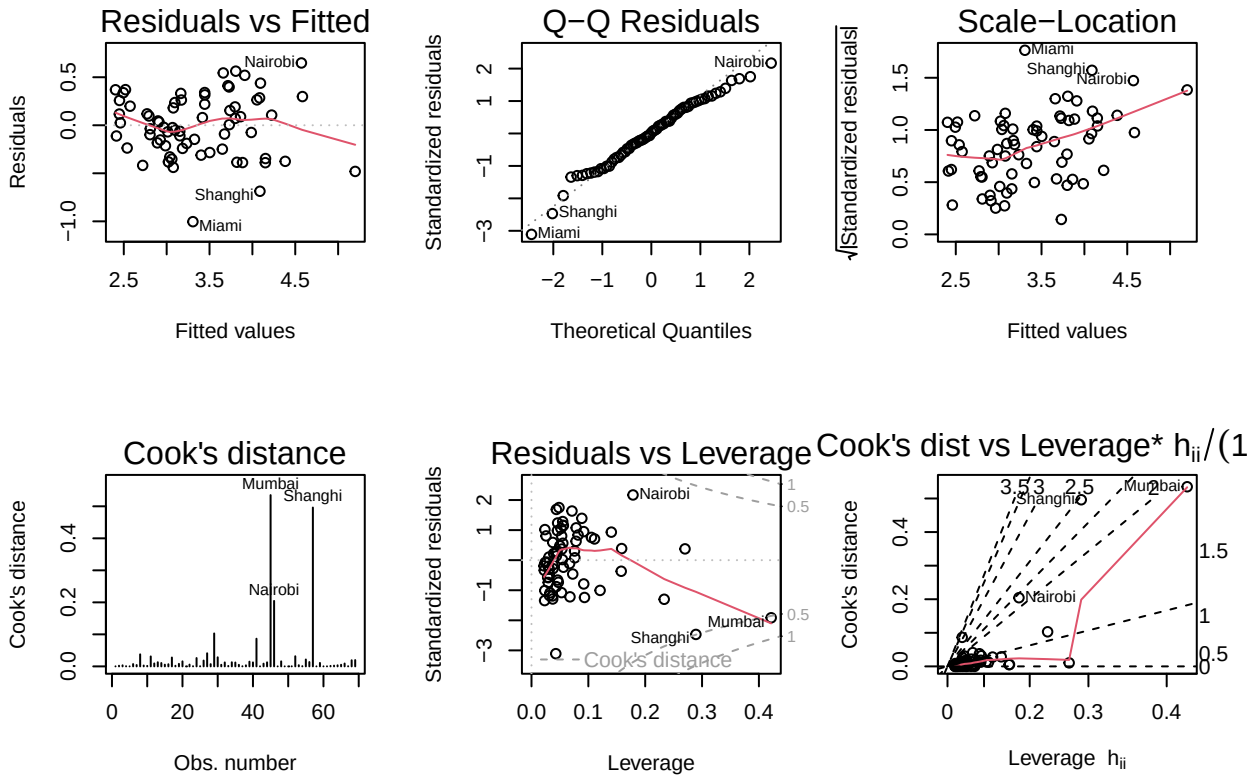
By comparing the interaction model with reduced model, the p-value is greater than 0.05. Also the residual plot is still unsatisfactory. These suggest that interaction is not a good option.

Next, consider box-cox transformation on BigMac:

```
boxcox(BigMac~Bread+Rice+FoodIndex+TaxRate,data=data)
```




```
fit_log=lm(log(BigMac)~Bread+Rice+FoodIndex+TaxRate
                    ,data=data)
par(mfrow=c(2,3))
plot(fit_log,which=1:6)
```



In the residual plot, the data plots scatter more evenly, and the red line is closer to parallel. Comparing with previous models, this is the best model so far.

```
summary(fit_log)
```

```
##
## Call:
## lm(formula = log(BigMac) ~ Bread + Rice + FoodIndex + TaxRate,
##     data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.00365 -0.23689  0.00657  0.25823  0.64848
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.894962   0.183572  21.218  < 2e-16 ***
## Bread        0.009843   0.002702   3.643 0.000541 ***
## Rice         0.014530   0.003037   4.784 1.05e-05 ***
## FoodIndex   -0.013727   0.001775  -7.733 9.55e-11 ***
## TaxRate     -0.009954   0.004423  -2.250 0.027874 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3299 on 64 degrees of freedom
## Multiple R-squared:  0.7835, Adjusted R-squared:   0.77
## F-statistic: 57.91 on 4 and 64 DF,  p-value: < 2.2e-16
```

Eventually, we propose $BigMac = exp(3.894962 + 0.009843 \times Bread + 0.014530 \times Rice - 0.013727 \times FoodIndex - 0.009954 \times TaxRate)$.

Exercise 3

The question is: whether GS and RI can effectively reduce COST.

```
#head(drugcost)
```

```
fit_all=lm(COST~.,data=drugcost)
summary(fit_all)
```

```
##
## Call:
## lm(formula = COST ~ ., data = drugcost)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.142888 -0.050521 -0.003367  0.047232  0.122523
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.851e+00  7.636e-01   2.424 0.024488 *
## RXPM         2.241e-02  1.100e-02   2.037 0.054483 .
## GS          -1.137e-02  2.830e-03  -4.018 0.000622 ***
## RI           3.341e-04  2.089e-03   0.160 0.874468
## COPAY        1.472e-02  1.870e-02   0.787 0.439791
## AGE         -3.754e-02  1.491e-02  -2.517 0.020012 *
## F            1.297e-02  9.712e-03   1.335 0.196148
## MM           2.908e-08  4.163e-08   0.699 0.492505
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08276 on 21 degrees of freedom
## Multiple R-squared:  0.5758, Adjusted R-squared:  0.4344
## F-statistic: 4.072 on 7 and 21 DF,  p-value: 0.00572
```

Since the p-value of RI is greater than 0.05, RI shouldn't be considered as a significant factor.

Next, we move to GS, and check that its p-value is <0.05, and its coefficient is negative.

To check if GS "effectively" reduces COST, check how much COST is reduced when GS increases 10%.

```
mean=mean(drugcost$COST)
coef=coef(fit_all)["GS"]
coef*10/mean
```

```
##          GS
## -0.09220068
```

If GS increases 10%, then COST will decrease around 9.2%(of mean COST).

Overall, I think RI does not effectively reduce COST, but GS can effectively reduce COST.

Exercise 4

```
#cloud
```

Theoretically, since A is determined randomly, A should not be associated with other independent variables. However, as the data size is small(24), let's check whether there is correlation between A and other independent variables.

```
fit=manova(cbind(D,S,C,P,E)~A,data=cloud)
summary(fit)
```

```
##           Df   Pillai approx F num Df den Df Pr(>F)
## A          1 0.074117  0.28818      5     18 0.9134
## Residuals 22
```

The p-value is 0.9134(extremely large), suggesting that there is no significant correlation between A and other independent variables. Therefore, it is reasonable to assume that other factors do not differ when A=0 or 1.

Use t-test on A and Rain:

```
t.test(Rain~A,data=cloud)
```

```
##
##  Welch Two Sample t-test
##
## data:  Rain by A
## t = -0.3574, df = 20.871, p-value = 0.7244
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -3.154691  2.229691
## sample estimates:
## mean in group 0 mean in group 1
##        4.171667        4.634167
```

The result from t-test suggests that there is no significant difference between A=0 and A=1. Therefore, it can be concluded that cloud seeding(A=1) doesn't significantly increase Rain.

However, since the small sample size may still raise concern about whether other factors are truly the same in A=0 and A=1, we can apply linear regression on all the other factors. Before that, since the value of D isn't really meaningful, we can first divide D into three periods: early, mid and late. In this way, D is changed into a factor.

```
range(cloud$D)
```

```
## [1]  0 83
```

```
cuts=quantile(cloud$D, probs = c(0, 1/3, 2/3, 1))


cloud$D_ =cut(
```

```
  cloud$D,
  breaks = cuts,
  include.lowest = TRUE,
  labels = c(0, 1, 2)
)


cloud$D_ <- as.factor(as.character(cloud$D_))
is.factor(cloud$D_)
```

```
## [1] TRUE
```

```
fit_all=lm(Rain~A+S+C+P+E+D_,data=cloud)
summary(fit_all)
```

```
##
## Call:
## lm(formula = Rain ~ A + S + C + P + E + D_, data = cloud)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.0392 -1.1497 -0.0702  0.3047  5.3869
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.1374     3.1639   0.359   0.7239
## A             0.7098     1.0993   0.646   0.5276
## S            -0.9082     0.6617  -1.372   0.1889
## C             0.0239     0.1064   0.225   0.8251
## P             2.3586     2.4174   0.976   0.3437
## E             3.7184     1.4729   2.525   0.0225 *
## D_1           2.2623     1.4713   1.538   0.1437
## D_2          -1.2092     1.5010  -0.806   0.4323
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.554 on 16 degrees of freedom
## Multiple R-squared:  0.5307, Adjusted R-squared:  0.3254
## F-statistic: 2.585 on 7 and 16 DF,  p-value: 0.05486
```

We can still see that the p-value of A is 0.7239>0.5.

Overall, cloud seeding(A=1) doesn't significantly increase Rain.

3 2 continued

Now, consider $log(Expenditure)_i = \beta_0 + \beta_1 \times log(Income)_i + \beta_3 \times Urban_i + \sum_{k=2}^{4} \alpha_k I(Region_i = k) + \epsilon_i, var(\epsilon) = \sigma^2 G_0 = \sigma^2 diag(4..4, 9..9, 0.5..0.5, 1..1)$.

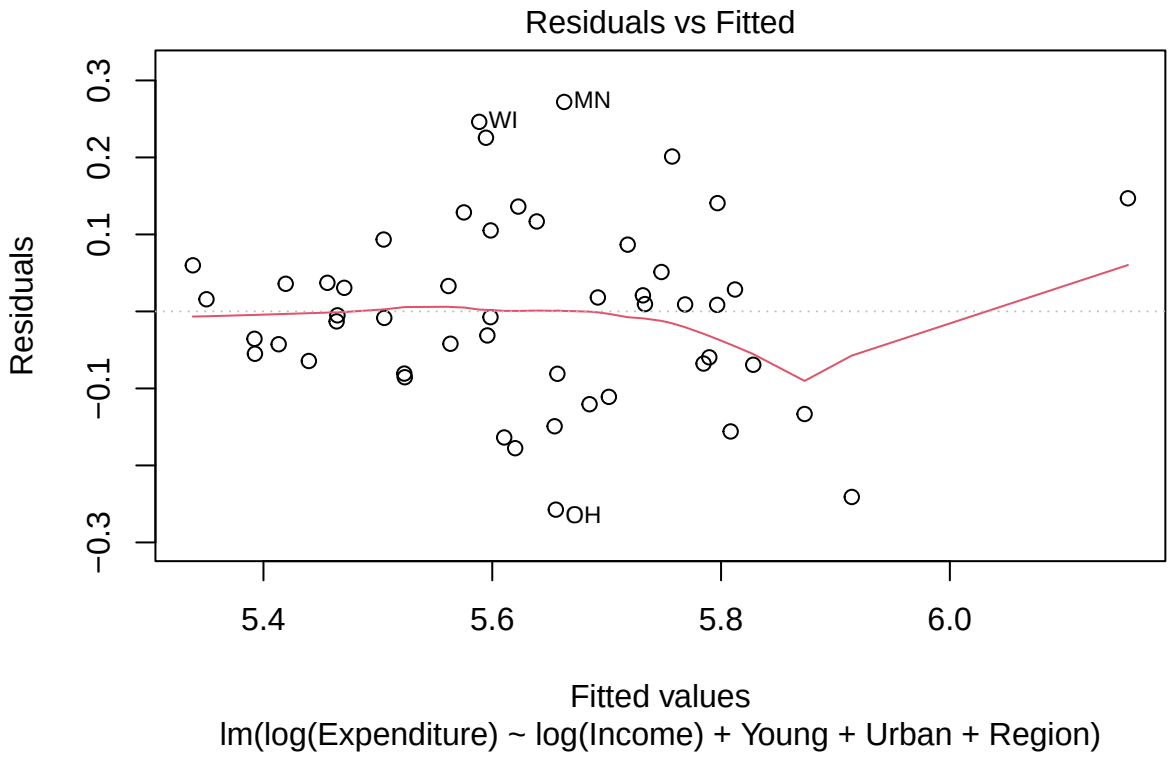
Use GLS:

```r
#edu$Region
G0=rep(0,50)
G0[edu$Region==1]=4
G0[edu$Region==2]=9
G0[edu$Region==3]=0.5
G0[edu$Region==4]=1
w=1/G0
fit3=lm(log(Expenditure)~log(Income)+Young+Urban+Region,data=edu,weights=w)
summary(fit3)
```

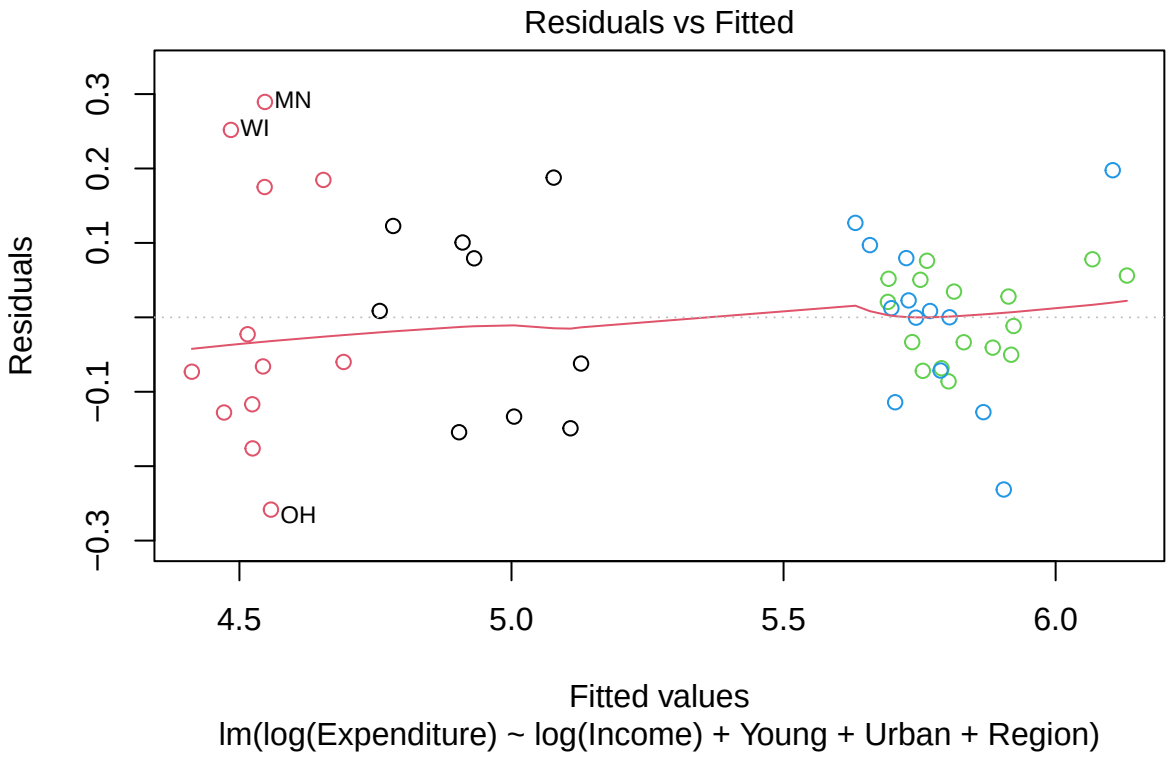
```
##
## Call:
## lm(formula = log(Expenditure) ~ log(Income) + Young + Urban +
##     Region, data = edu, weights = w)
##
## Weighted Residuals:
##       Min        1Q    Median        3Q       Max
## -0.240954 -0.053498  0.003031  0.052131  0.146931
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4.96659    1.02579  -4.842 1.70e-05 ***
## log(Income)  1.11390    0.11669   9.546 3.47e-12 ***
## Young        3.80337    0.65395   5.816 6.77e-07 ***
## Urban       -0.05493    0.11058  -0.497    0.622
## Region2     -0.05878    0.08898  -0.661    0.512
## Region3     -0.03065    0.05799  -0.529    0.600
## Region4      0.06461    0.06084   1.062    0.294
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08094 on 43 degrees of freedom
## Multiple R-squared:  0.8485, Adjusted R-squared:  0.8274
## F-statistic: 40.15 on 6 and 43 DF,  p-value: 4.598e-16
```

Check $y^* = G_0^{-1/2} y = G_0^{-1/2} X\beta + G_0^{-1/2}\epsilon = X^*\beta + \epsilon^*$.

```r
plot(fit3,which=1)
```




```
edu.standard=edu
edu.standard[,1:4]=edu.standard[,1:4]/sqrt(G0)
fit3.standard=lm(log(Expenditure)~log(Income)+Young+Urban+Region,data=edu.standard)
plot(fit3.standard,which=1,col=edu$Region)
```



We can see that the variance of each region is still different in the new model.

Use IRLS:

Assume $var(\epsilon_i) = \sigma_k^2, if\ Region_i = k, k = 1...4; i = 1...50.$

Now we don't know the value of σ_k^2. Therefore update estimate of β and σ_k in turn.

```
fit.ini=fit=lm(log(Expenditure)~.,data=edu)
repeat{
  beta=coef(fit)
  res=resid(fit)
  #sigma_hat^2=sum residual^2/(n-1)
  sigmasq1=sum(res[1:9]^2)/(9-1)
  sigmasq2=sum(res[10:21]^2)/(12-1)
  sigmasq3=sum(res[22:37]^2)/(16-1)
  sigmasq4=sum(res[38:50]^2)/(13-1)
  #covariance matrix
  sigma_sq=c(rep(sigmasq1,9),rep(sigmasq2,12),rep(sigmasq3,16),rep(sigmasq4,13))
  w=1/sigma_sq
  fit=lm(log(Expenditure)~log(Income)+Young+Urban+Region,data=edu,weights=w)
  beta.new=coef(fit)
  beta.new
  delta=sum(abs(beta.new-beta))
  print(delta)
  if(delta<13-10) break
}
```

```
## [1] 9.936183
```

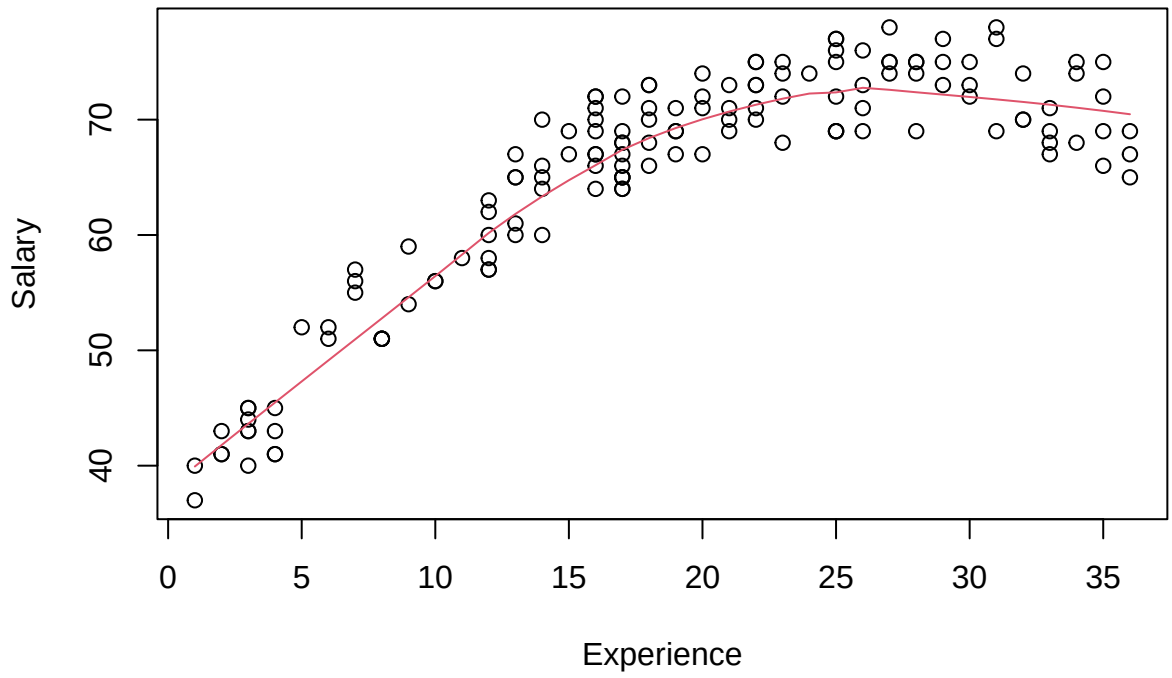
```
## [1] 0.3004868
```

```
fit3=fit
unique(sigma_sq)
```

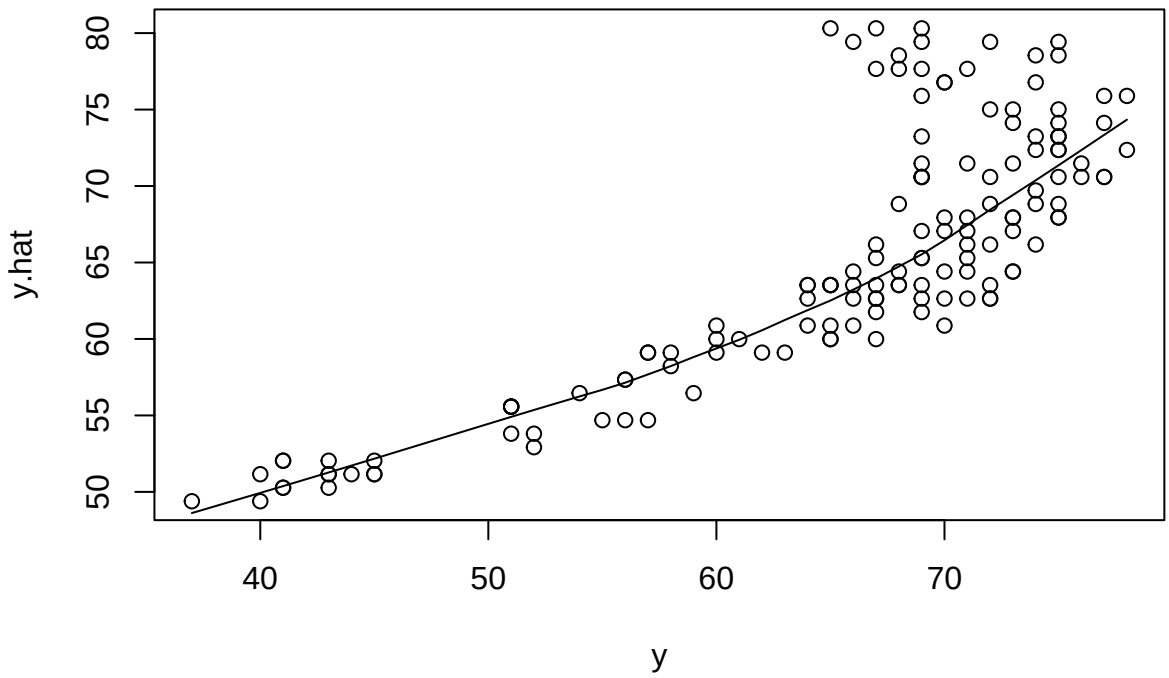
```
## [1] 0.018995624 0.031129937 0.001349336 0.014224593
```

4 lowess,IRP

```
se=read.table("http://staff.ustc.edu.cn/~ynyang/2025/lab/salary-experience.txt",
              head=T,row.names=1)
se=se[,2:1]
plot(se)
lowess.fit=lowess(se,f=2/3)
lines(lowess.fit,col=2)
```




```
y=se[,"Salary"]
myfit=lm(Salary~Experience,data=se)
y.hat=fitted(myfit)
plot(y,y.hat)
lowess.fit=lowess(y,y.hat,f=2/3)
lines(lowess.fit)
```


y.hat
50
55
60
65
70
75
80
40
50
60
70
y