Regression

Elara

2016 年 *5* 月 *4* 日

线性回归

线性模型

$$Y_i = e^{\beta_1+\beta_2 X_i+\epsilon_i}$$

$$Y_i = \frac{1}{e^{\beta_1+\beta_2 X_i+\epsilon_i}}$$

$$Y_i = \beta_1 + (0.75-\beta_1)e^{-\beta_2(X_i-2)} + \epsilon_i$$

$$Y_i = \beta_1 + \beta_2^3 X_i + \epsilon_i$$

$$Y_i = \beta_1 + \beta_2(\frac{1}{X_i}) + \epsilon_i$$

1. 125 是线性模型
2. 没有截距项的时候 R2 不能用。此时 OLS 的 FOC 没有 β_0 相关，得不到残差和 =0
3. 无法把方差分解成可解释和不可解释部分。
4. 即使截距项不显著也不能去掉。去掉的话一定过原点。
5. R2 受到模型变量数目影响。要用 adj.R2

LM 线性模型估计 OLS

```r
#Y X 线性
options(digits=3)
fit <- lm(weight ~ height, data = women)
summary(fit)
```

```
##
## Call:
## lm(formula = weight ~ height, data = women)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -1.733 -1.133 -0.383  0.742  3.117
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -87.5167     5.9369   -14.7  1.7e-09 ***
## height        3.4500     0.0911    37.9  1.1e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.53 on 13 degrees of freedom
## Multiple R-squared:  0.991,  Adjusted R-squared:  0.99
## F-statistic: 1.43e+03 on 1 and 13 DF,  p-value: 1.09e-14
```

```r
coefficients(fit)
```

```
## (Intercept)      height
##      -87.52        3.45
```

```r
fitted(fit)
```

```
##   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15
## 113 116 119 123 126 130 133 137 140 144 147 151 154 157 161
```

```r
residuals(fit)
```

```
##       1       2       3       4       5       6       7       8       9
##  2.4167  0.9667  0.5167  0.0667 -0.3833 -0.8333 -1.2833 -1.7333 -1.1833
##      10      11      12      13      14      15
## -1.6333 -1.0833 -0.5333  0.0167  1.5667  3.1167
```

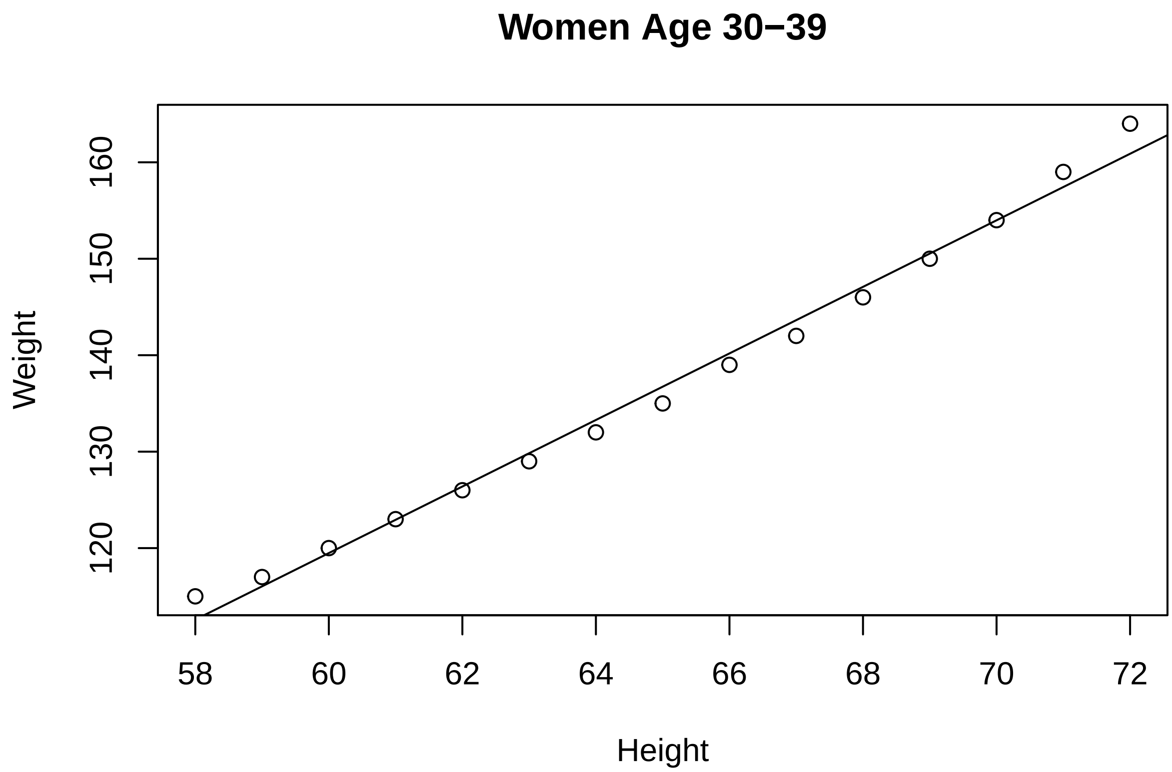
```r
deviance(fit)
```

```
## [1] 30.2
```

```r
# 置信区间 0.99
confint(fit,level=0.99)
```

```
##               0.5 % 99.5 %
## (Intercept) -105.40 -69.63
## height         3.18   3.72
```

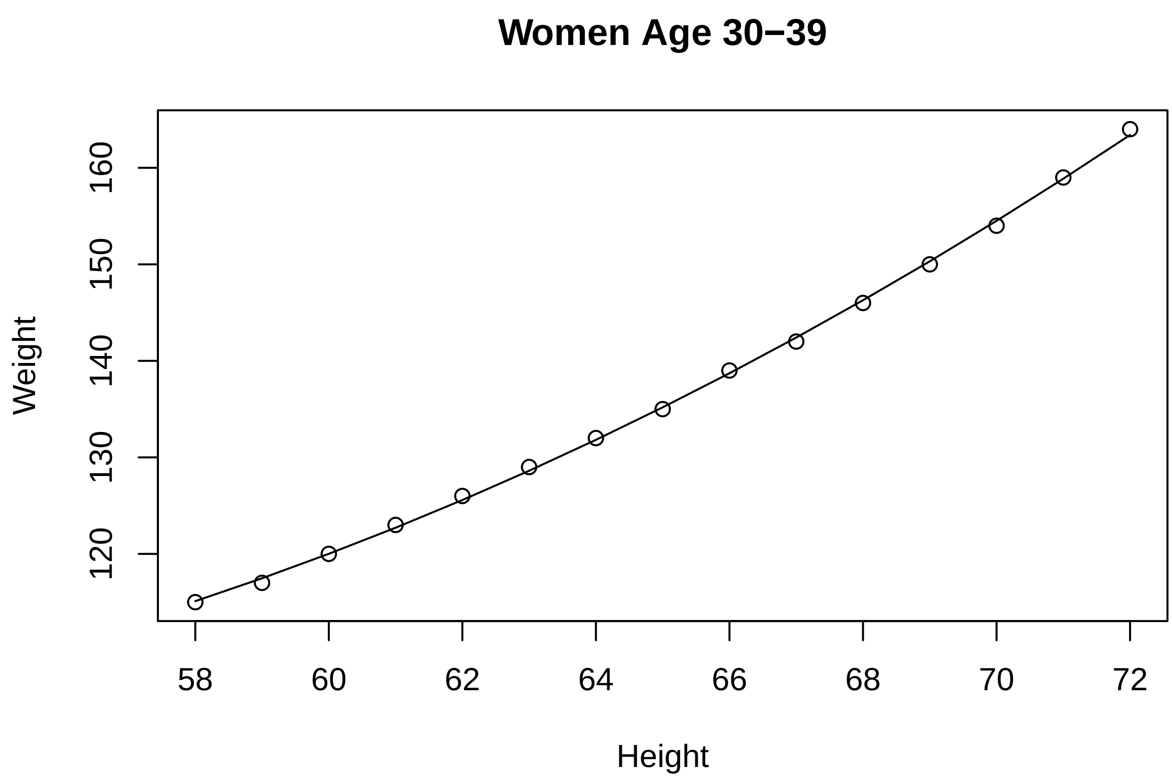
```r
plot(women$height,women$weight,main="Women Age 30-39",xlab="Height",ylab="Weight")

abline(fit)
```




```r
#x 和 y 非线性
fit2 <- lm(weight ~ height + I(height^2), data=women)
summary(fit2)
```

```
##
## Call:
## lm(formula = weight ~ height + I(height^2), data = women)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.5094 -0.2961 -0.0094  0.2862  0.5971
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) 261.87818   25.19677   10.39  2.4e-07 ***
## height       -7.34832    0.77769   -9.45  6.6e-07 ***
## I(height^2)   0.08306    0.00598   13.89  9.3e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.384 on 12 degrees of freedom
## Multiple R-squared:  0.999,  Adjusted R-squared:  0.999
## F-statistic: 1.14e+04 on 2 and 12 DF,  p-value: <2e-16
```

```r
plot(women$height, women$weight, main = "Women Age 30-39",
xlab = "Height", ylab = "Weight")
lines(women$height, fitted(fit2))
```




```r
Anscombe<-data.frame(
X =c(10.0, 8.0, 13.0, 9.0, 11.0, 14.0, 6.0, 4.0, 12.0, 7.0, 5.0),
Y1=c(8.04, 6.95, 7.58, 8.81, 8.33, 9.96, 7.24, 4.26, 10.84, 4.82, 5.68),
Y2=c(9.14, 8.14, 8.74, 8.77, 9.26, 8.10, 6.13, 3.10, 9.13, 7.26, 4.74),
Y3=c(7.46, 6.77, 12.74, 7.11, 7.81, 8.84, 6.08, 5.39, 8.15, 6.44, 5.73),
X4=c(rep(8,7), 19, rep(8,3)),
Y4=c(6.58, 5.76, 7.71, 8.84, 8.47, 7.04, 5.25, 12.50, 5.56, 7.91, 6.89)
)
summary(lm(Y1~X, data=Anscombe))
```

```
##
## Call:
## lm(formula = Y1 ~ X, data = Anscombe)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9213 -0.4558 -0.0414  0.7094  1.8388
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.000      1.125    2.67   0.0257 *
```

```
## X               0.500      0.118    4.24   0.0022 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.24 on 9 degrees of freedom
## Multiple R-squared:  0.667,  Adjusted R-squared:  0.629
## F-statistic:   18 on 1 and 9 DF,  p-value: 0.00217
```

```r
summary(lm(Y2~X, data=Anscombe))
```

```
##
## Call:
## lm(formula = Y2 ~ X, data = Anscombe)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -1.901 -0.761  0.129  0.949  1.269
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.001      1.125    2.67   0.0258 *
## X               0.500      0.118    4.24   0.0022 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.24 on 9 degrees of freedom
## Multiple R-squared:  0.666,  Adjusted R-squared:  0.629
## F-statistic:   18 on 1 and 9 DF,  p-value: 0.00218
```

```r
summary(lm(Y3~X, data=Anscombe))
```

```
##
## Call:
## lm(formula = Y3 ~ X, data = Anscombe)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -1.159 -0.616 -0.232  0.151  3.241
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.008      1.124    2.67   0.0254 *
## X               0.499      0.118    4.24   0.0022 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.24 on 9 degrees of freedom
## Multiple R-squared:  0.666,  Adjusted R-squared:  0.629
## F-statistic: 17.9 on 1 and 9 DF,  p-value: 0.00218
```

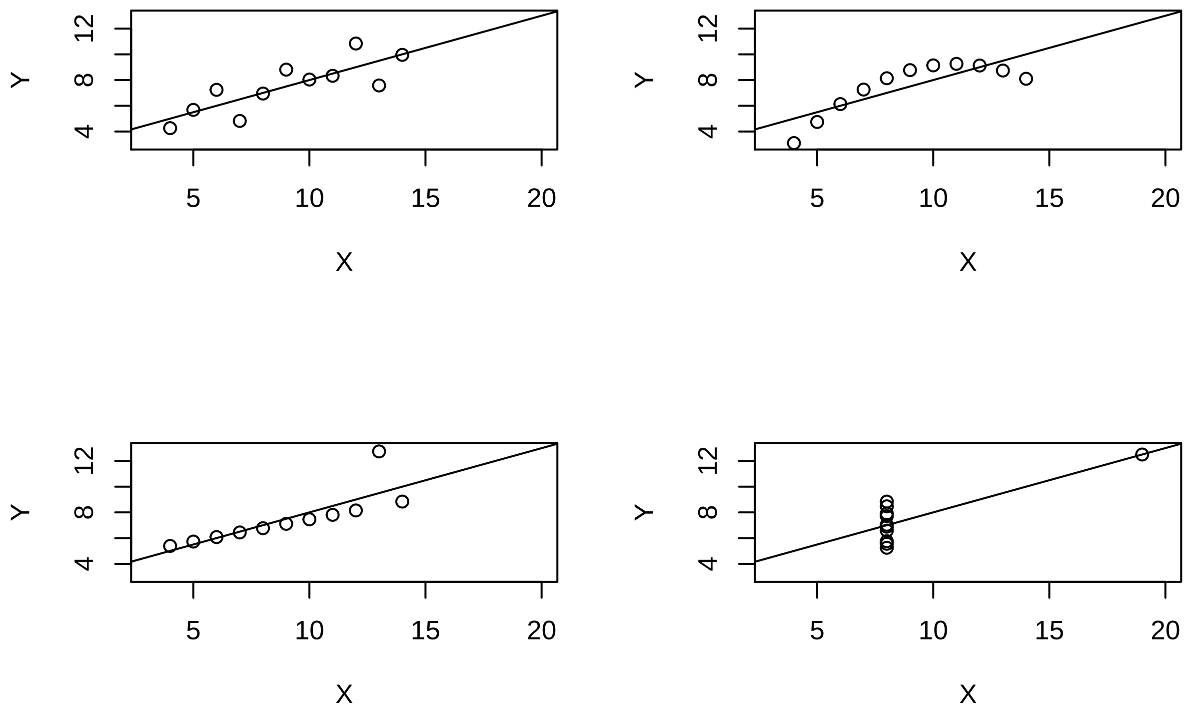
```r
summary(lm(Y4~X4,data=Anscombe))
```

```
##
## Call:
## lm(formula = Y4 ~ X4, data = Anscombe)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -1.751 -0.831  0.000  0.809  1.839
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.002      1.124    2.67   0.0256 *
## X4             0.500      0.118    4.24   0.0022 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.24 on 9 degrees of freedom
## Multiple R-squared:  0.667,  Adjusted R-squared:  0.63
## F-statistic:   18 on 1 and 9 DF,  p-value: 0.00216
```

```r
head(Anscombe)
```

```
##    X   Y1   Y2    Y3 X4   Y4
## 1 10 8.04 9.14  7.46  8 6.58
## 2  8 6.95 8.14  6.77  8 5.76
## 3 13 7.58 8.74 12.74  8 7.71
## 4  9 8.81 8.77  7.11  8 8.84
## 5 11 8.33 9.26  7.81  8 8.47
## 6 14 9.96 8.10  8.84  8 7.04
```

```r
attach(Anscombe)
par(mfrow = c(2,2))
plot(c(3,20), c(3,13), type="n", xlab = "X", ylab = "Y"); points(X,Y1); abline(lm(Y1~X))
plot(c(3,20), c(3,13), type="n", xlab = "X", ylab = "Y"); points(X,Y2); abline(lm(Y2~X))
plot(c(3,20), c(3,13), type="n", xlab = "X", ylab = "Y"); points(X,Y3); abline(lm(Y3~X))
plot(c(3,20), c(3,13), type="n", xlab = "X", ylab = "Y"); points(X4,Y4); abline(lm(Y4~X4))
```



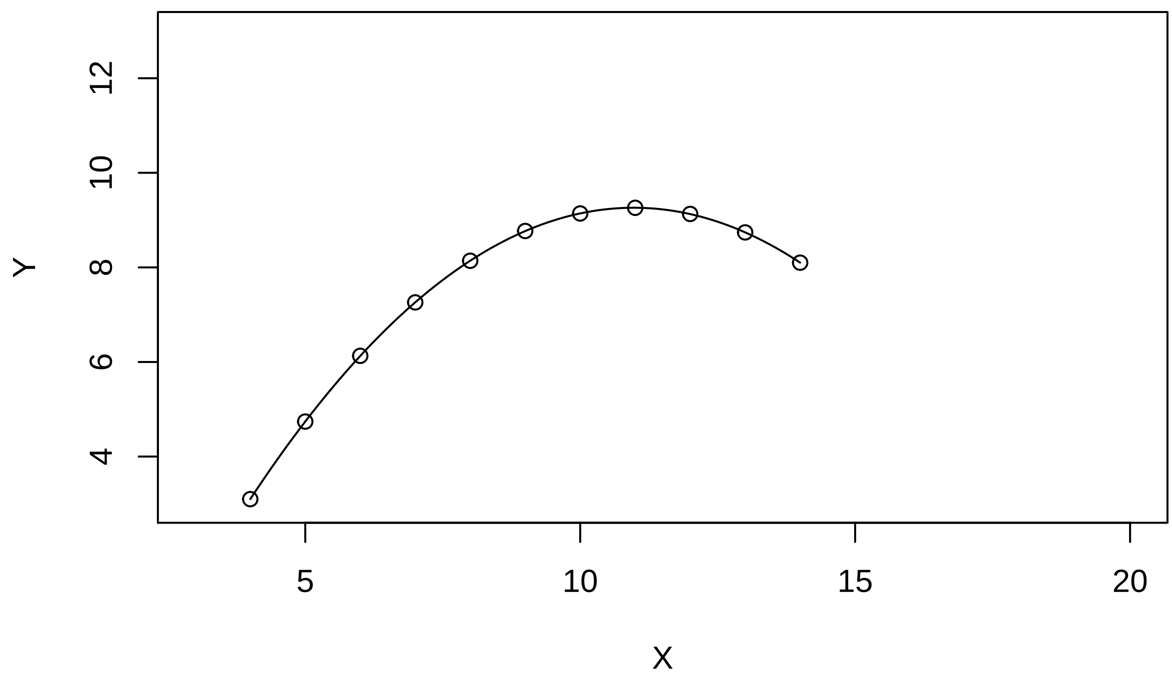
系数都是 3 和 0.5 并且都显著。可是作图结果形状完全不一致 2 是曲线 3 有异常值 4 除了一个点以外都是同一个竖线上

```
#1 没有问题
par(mfrow = c(1,1))
#2 是个曲线，加入平方拟合
X2<-X^2
# 存放用平方拟合的系数
lm2.sol<-lm(Y2~X+X2)
summary(lm2.sol)
```

```
##
## Call:
## lm(formula = Y2 ~ X + X2)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.001329 -0.001189 -0.000629  0.000874  0.002378
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -6.00e+00   4.33e-03   -1385   <2e-16 ***
## X            2.78e+00   1.04e-03    2674   <2e-16 ***
## X2          -1.27e-01   5.71e-05   -2219   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.00167 on 8 degrees of freedom
## Multiple R-squared:      1,    Adjusted R-squared:      1
## F-statistic: 7.38e+06 on 2 and 8 DF,  p-value: <2e-16
```

```
# 作图用 x
x<-seq(min(X), max(X), by=0.1)
# 作图用系数
b<-coef(lm2.sol)
y<-b[1]+b[2]*x+b[3]*x^2
plot(c(3,20), c(3,13), type="n", xlab = "X", ylab = "Y")
#plot 原图
points(X,Y2)
lines(x,y)
```

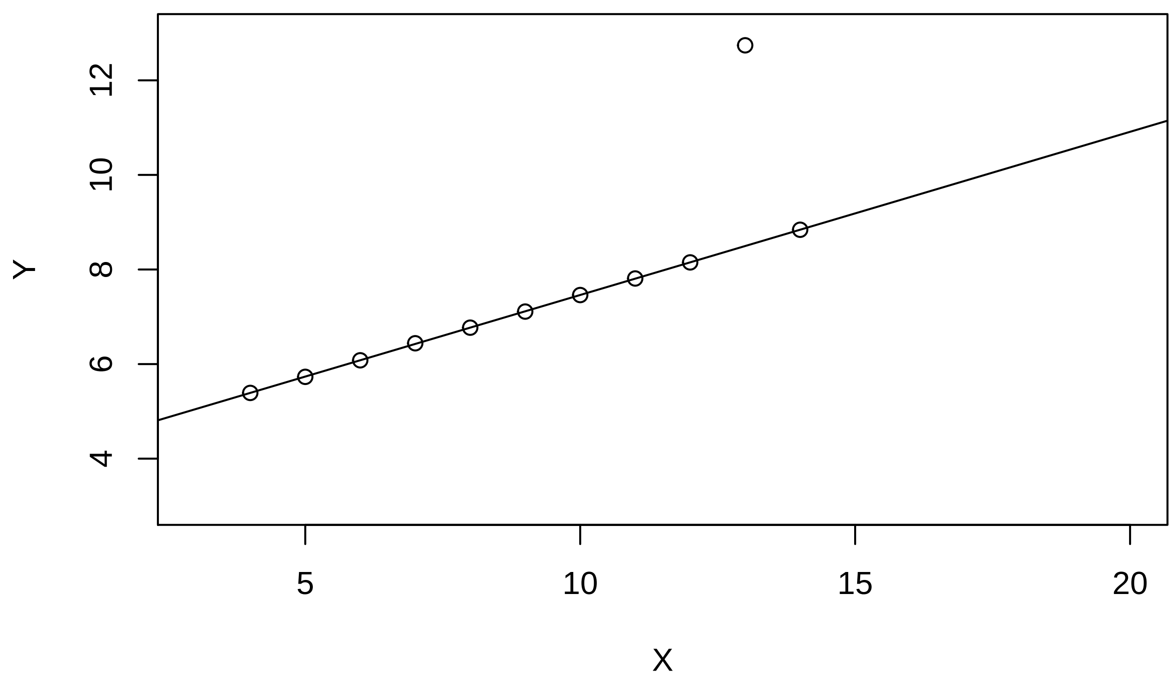



```
#3
# 去掉第三个（异常值）
i<-1:11; Y31<-Y3[i!=3]; X3<-X[i!=3]
lm3.sol<-lm(Y31~X3)
summary(lm3.sol)
```

```
##
## Call:
## lm(formula = Y31 ~ X3)
##
```

```
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.006017 -0.001212 -0.001017 -0.000823  0.014069
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4.010628   0.005711     702   <2e-16 ***
## X3          0.345043   0.000626     551   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.00602 on 8 degrees of freedom
## Multiple R-squared:     1,  Adjusted R-squared:     1
## F-statistic: 3.04e+05 on 1 and 8 DF,  p-value: <2e-16
```

```r
plot(c(3,20), c(3,13), type="n", xlab = "X", ylab = "Y")
points(X,Y3)
abline(lm3.sol)
```




```r
detach(Anscombe)
```

异常值检测

1.diffits 指标

$$DFFITS = \frac{\hat{y_i} - \hat{y_{i(i)}}}{s_(i)\sqrt{h_{ii}}}\sqrt{h_{ii}/(1-h_{ii})}$$

h 是帽子矩阵，y 尖 =hy

```
attach(Anscombe)
p<-1; n<-length(X);d<-dffits(lm(Y3~X, data=Anscombe))
cf<-1:n; cf[d>2*sqrt((p+1)/n)]
```

```
## [1] 3
```

```
# 取出 1 到 n 里面满足 dffits 大于 2 根号((p+1)/n)
detach(Anscombe)
```

返回了异常值位置 3

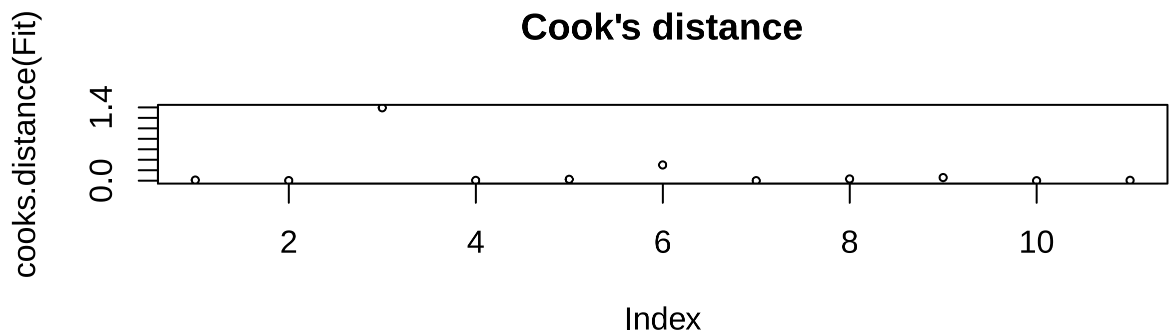
2.Cook's distance

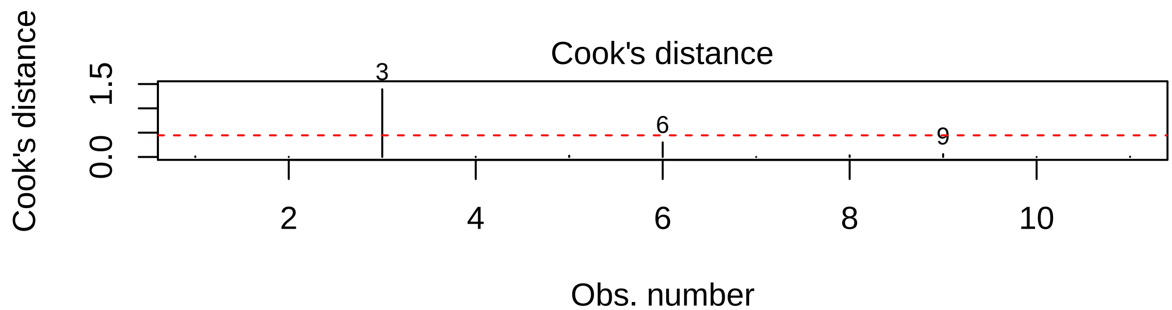
$$D_i = \frac{(\hat{\beta} - \hat{\beta}^{(-i)})^T(X^TX)(\hat{\beta} - \hat{\beta}^{(-i)})}{(1+p)s^2}$$

```
Fit<-lm(Y3~X, data=Anscombe)
cooks.distance(Fit)
```

```
##        1        2        3        4        5        6        7        8
## 0.011831 0.002183 1.392828 0.005525 0.026072 0.300634 0.000480 0.033194
##        9       10       11
## 0.059650 0.000218 0.006752
```

```
par(mfrow=c(2,1))
# 散点图
plot(cooks.distance(Fit),main="Cook's distance",cex=0.5)
# 线图，红线表示警戒线
Np<-length(coefficients(Fit))-1# 变量数
N<-length(fitted(Fit))
# 红线算法
CutLevel<-4/(N-Np-1)
plot(Fit,which=4)
abline(CutLevel,0,lty=2,col="red")
```





summary

```
# 可以直接算 dffit 和 cook, 有问题的会带星号
influence.measures(lm(Y3~X, data=Anscombe))
```

```
## Influence measures of
##   lm(formula = Y3 ~ X, data = Anscombe) :
##
##        dfb.1_     dfb.X    dffit    cov.r   cook.d    hat inf
## 1  -4.64e-03 -4.43e-02  -0.1468 1.34e+00 0.011831 0.1000
## 2  -3.75e-02  1.88e-02  -0.0624 1.39e+00 0.002183 0.1000
## 3  -1.83e+02  2.69e+02 342.7851 7.36e-10 1.392828 0.2364   *
## 4  -3.31e-02 -2.66e-18  -0.0997 1.36e+00 0.005525 0.0909
## 5   4.92e-02 -1.17e-01  -0.2197 1.34e+00 0.026072 0.1273
## 6   4.90e-01 -6.67e-01  -0.7898 1.36e+00 0.300634 0.3182
## 7   2.60e-02 -2.01e-02   0.0292 1.53e+00 0.000480 0.1727
## 8   2.39e-01 -2.07e-01   0.2449 1.80e+00 0.033194 0.3182   *
## 9   1.38e-01 -2.32e-01  -0.3365 1.34e+00 0.059650 0.1727
## 10 -1.54e-02  1.05e-02  -0.0197 1.45e+00 0.000218 0.1273
## 11  1.04e-01 -8.62e-02   0.1098 1.64e+00 0.006752 0.2364
```

最大似然估计

The following function is called a likelihood function, denoted by LF(β_1; β_2; σ^2)

$$LF(\beta_1, \beta_2, \sigma^2) = f(Y_1, Y_2, \dots, Y_n|\beta_1 + \beta_2 X_i, \sigma^2) = \frac{1}{\sigma^n(\sqrt{2\pi})^n} exp(\frac{1}{2}\sum\frac{(Y_i - \beta_1 - \beta_2 X_i)^2}{\sigma^2})$$

where β_1; β_2; σ^2 are not known. The method of maximum likelihood, as the name indicates, consists in estimating the unknown parameters in such a manner that the probability of observing the given Y's is as high (or maximum) as possible. Therefore, we have to find the maximum of the function 6. For differentiation it is easier to express 6 in the log term as follows:

$$lnLF = -nln\sigma - \frac{n}{2}ln(2\pi) - \frac{1}{2}\sum\frac{(Y_i - \beta_1 - \beta_2 X_i)^2}{\sigma^2} = -\frac{n}{2}ln\sigma^2 - \frac{n}{2}ln(2\pi) - \frac{1}{2}\sum\frac{(Y_i - \beta_1 - \beta_2 X_i)^2}{\sigma^2}$$

Differentiating 7 partially with respect to β_1; β_2, and σ^2, we can obtain the ML estimators.

```
install.packages(maxLik)
```

```
library("maxLik")
```

```
## Loading required package: miscTools
```

```
##
## Please cite the 'maxLik' package as:
## Henningsen, Arne and Toomet, Ott (2011). maxLik: A package for maximum likelihood estimation in R. Computa
##
## If you have questions, suggestions, or comments regarding the 'maxLik' package, please use a forum or 'tra
## https://r-forge.r-project.org/projects/maxlik/
```

```
indfood<-read.csv(file="C:\\Users\\44180\\Documents\\sourcetree\\elara7\\soe\\Rmarkdown\\Chap_9\\Indfoo
# 抽取数据
foodexp<-indfood[,1]
totalexp<-indfood[,2]
#OLS 回归
lm_r <- lm(foodexp~totalexp)
summary(lm_r)
```

```
##
## Call:
## lm(formula = foodexp ~ totalexp)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -153.77  -46.61    7.75   37.70  171.59
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  94.2088    50.8563    1.85     0.07 .
## totalexp      0.4368     0.0783    5.58  8.5e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 66.9 on 53 degrees of freedom
## Multiple R-squared:  0.37,	Adjusted R-squared:  0.358
## F-statistic: 31.1 on 1 and 53 DF,  p-value: 8.45e-07
```

```
# 最大似然估计
# 对数似然函数
loglik=function (para){
N=length(foodexp)# 样本量
e=foodexp-para[1]-para[2]*totalexp# 残差项表达式, para 是参数估计量
ll=-0.5*N*log(2*pi)-0.5*N*log(para[3]^2)-0.5*sum(e^2/para[3]^2)# 对数似然函数, 注意有个参数 3
return(ll)
}
# 需要 1, log 后的似然函数, 初始值
mle1=maxLik(loglik,start=c(0.1,1,1))#3 个参数, 1 2, 方差
coef(mle1)
```

```
## [1]  94.266   0.437 -65.601
```

多元线性回归

OLS 是线性无偏中方差最小的。如果有一个有偏估计方差很小也可以用

```
class(mtcars)
```

```
## [1] "data.frame"
```

```
mtcar <- as.data.frame(mtcars[,c("mpg", "cyl",
"disp", "hp", "wt")])
cor(mtcar)
```

```
##         mpg    cyl   disp     hp     wt
## mpg   1.000 -0.852 -0.848 -0.776 -0.868
## cyl  -0.852  1.000  0.902  0.832  0.782
## disp -0.848  0.902  1.000  0.791  0.888
## hp   -0.776  0.832  0.791  1.000  0.659
## wt   -0.868  0.782  0.888  0.659  1.000
```

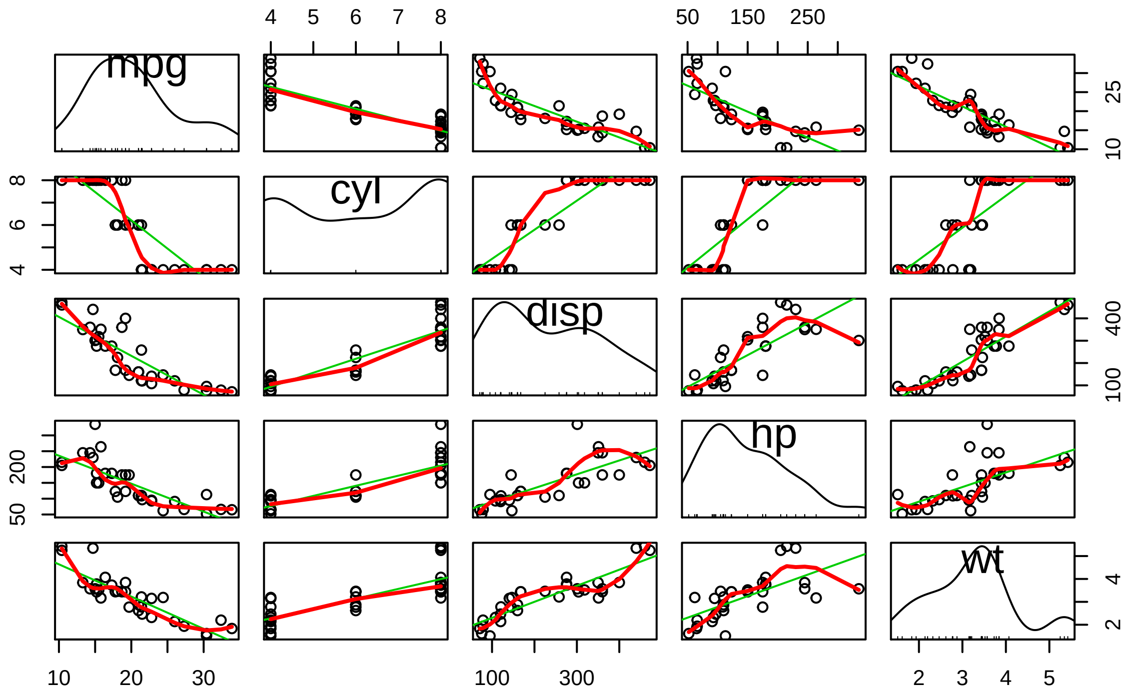
```
library(car)
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked _by_ '.GlobalEnv':
##
##     Anscombe
```

```
scatterplotMatrix(mtcar, spread=FALSE, main="Scatter Plot Matrix")
```

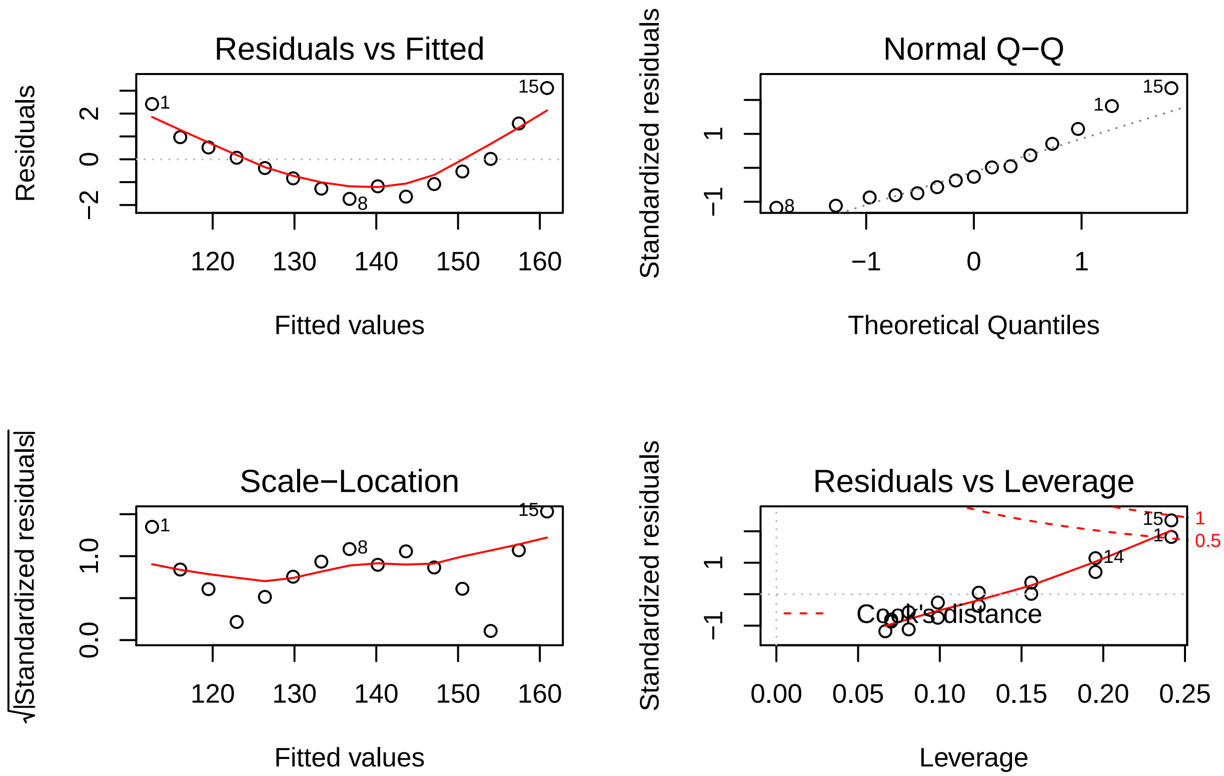
Scatter Plot Matrix




```
fit3 <- lm(mpg ~ hp + wt + hp:wt, data = mtcar)
summary(fit3)
```

```
##
## Call:
## lm(formula = mpg ~ hp + wt + hp:wt, data = mtcar)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -3.063 -1.649 -0.736  1.421  4.551
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 49.80842    3.60516   13.82  5.0e-14 ***
## hp          -0.12010    0.02470   -4.86  4.0e-05 ***
## wt          -8.21662    1.26971   -6.47  5.2e-07 ***
## hp:wt        0.02785    0.00742    3.75  0.00081 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.15 on 28 degrees of freedom
## Multiple R-squared:  0.885,  Adjusted R-squared:  0.872
## F-statistic: 71.7 on 3 and 28 DF,  p-value: 2.98e-13
```

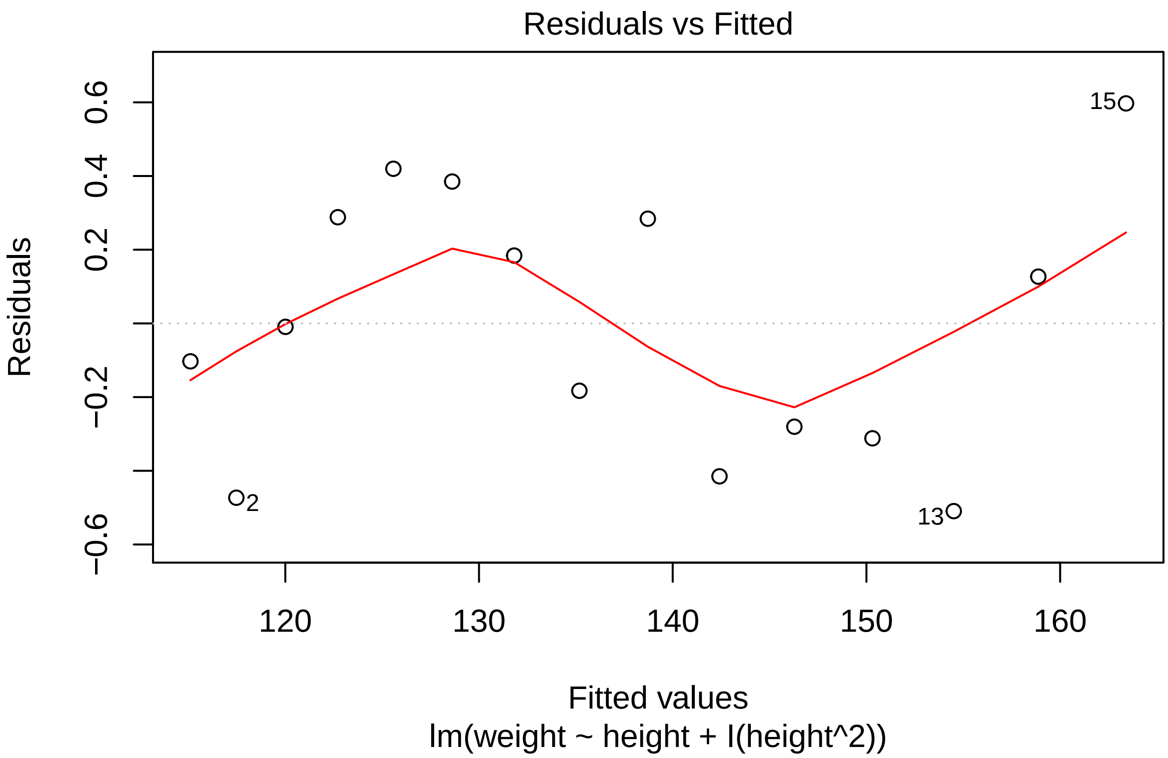
```
fit <- lm(weight ~ height, data=women)
par(mfrow=c(2,2))
plot(fit)
```

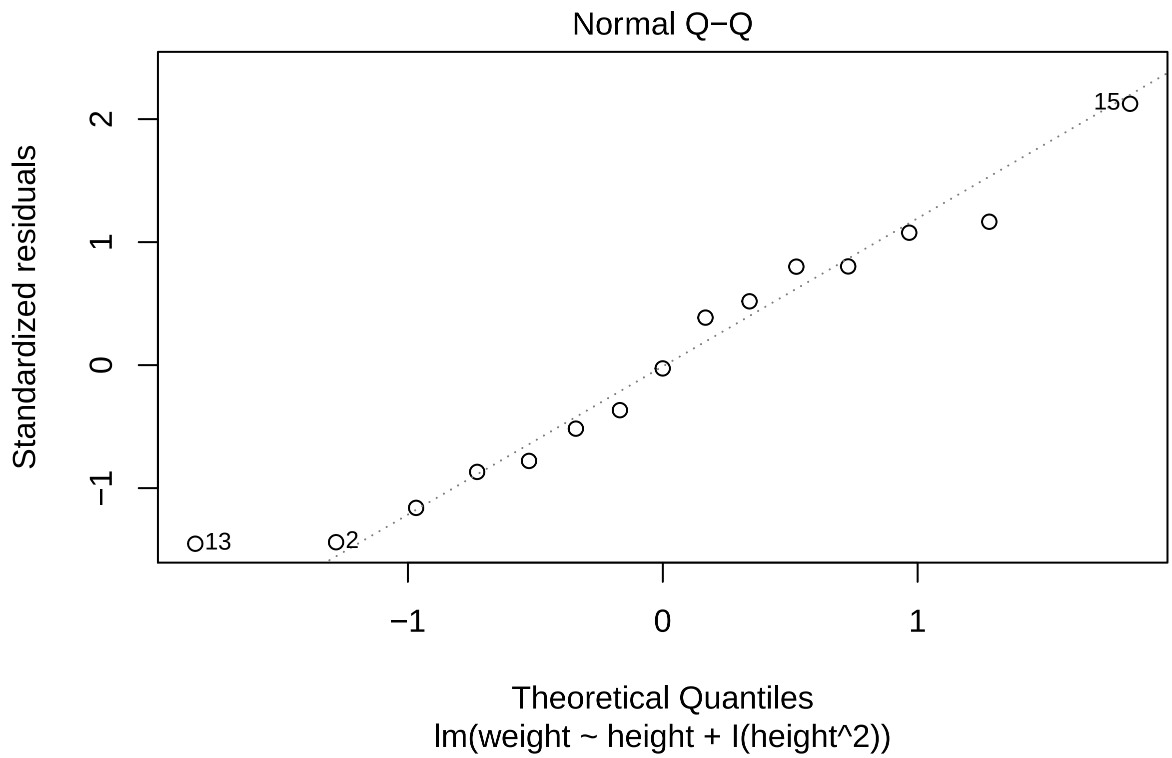


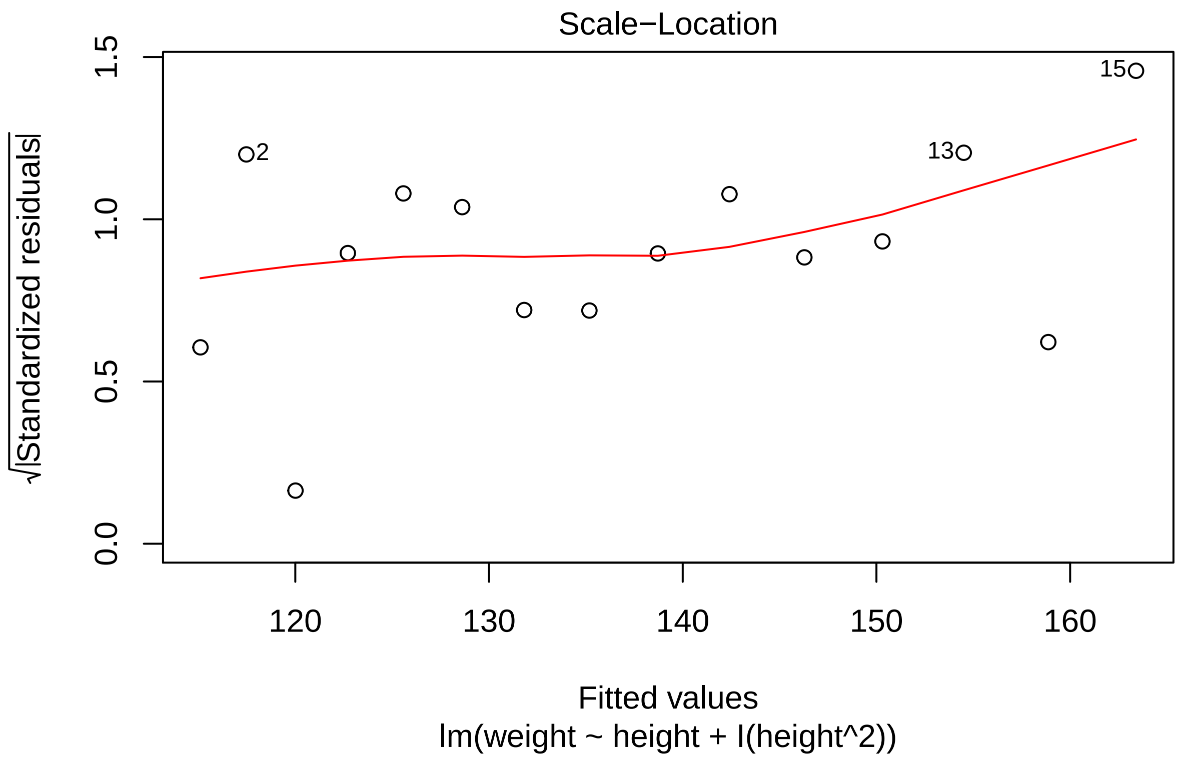
残差图如果是左右开口的喇叭状很可能有异方差

第四图：高杠杆有离群点，强影响（红实线是警戒线）

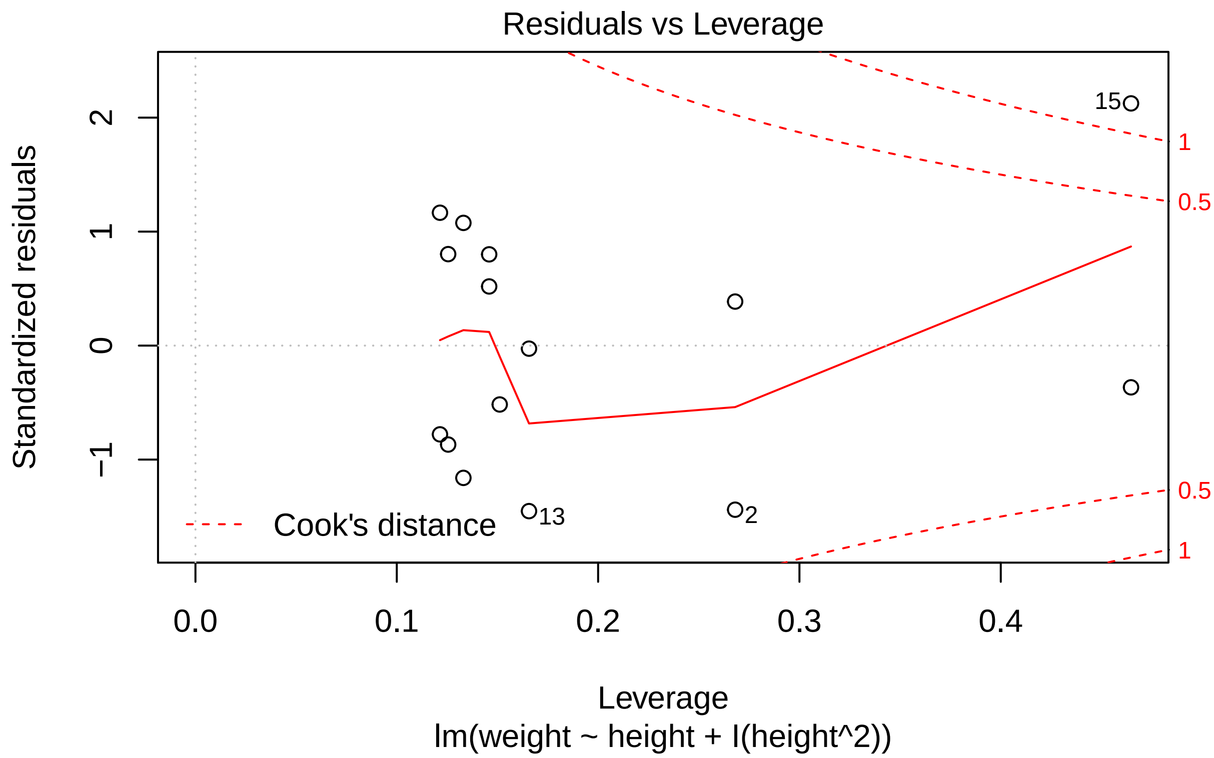
```
# 加入平方项回归
fit2 <- lm(weight ~ height + I(height^2), data=women)
plot(fit2)
```


Residuals vs Fitted
Residuals
0.6
0.4
0.2
−0.2
−0.6
15
2
13
120
130
140
150
160
Fitted values
lm(weight ~ height + I(height^2))


Normal Q−Q
Standardized residuals
2
1
0
−1
15
13
2
−1
0
1
Theoretical Quantiles
lm(weight ~ height + I(height^2))



Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
120
130
140
150
160
Fitted values
lm(weight ~ height + I(height^2))
2
13
15



系数之间相关影响实验

In order to explain the meaning of coefficients ,we have the following step.
Regression model:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + u_i$$

Step 1:

$$w_i = y_i - \hat{\alpha}_0 - \hat{\alpha}_1 x_{i2}$$

Step 2:

$$v_i = x_{i1} - \hat{b}_0 - \hat{b}_1 x_{i2}$$

Step 3:

$$\bar{\beta}_1 = \frac{\sum v_i w_i}{\sum v_i^2}$$

```
mtcar <- as.data.frame(mtcars[,c("mpg", "cyl", "disp", "hp", "wt")])
fit <- lm(mpg~wt+disp, data=mtcar)
summary(fit)
```

```
##
## Call:
## lm(formula = mpg ~ wt + disp, data = mtcar)
##
```

```
## Residuals:
##    Min     1Q Median     3Q    Max
## -3.409 -2.324 -0.768  1.772  6.348
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 34.96055    2.16454   16.15  4.9e-16 ***
## wt          -3.35083    1.16413   -2.88   0.0074 **
## disp        -0.01772    0.00919   -1.93   0.0636 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.92 on 29 degrees of freedom
## Multiple R-squared:  0.781,  Adjusted R-squared:  0.766
## F-statistic: 51.7 on 2 and 29 DF,  p-value: 2.74e-10
```

```r
fit1 <- lm(mpg~disp, data=mtcar)
fit2 <- lm(wt~disp, data=mtcar)
fit3 <- lm(fit1$residuals~fit2$residuals-1)# 没常数项用-1
summary(fit3)
```

```
##
## Call:
## lm(formula = fit1$residuals ~ fit2$residuals - 1)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -3.409 -2.324 -0.768  1.772  6.348
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## fit2$residuals    -3.35       1.13   -2.98   0.0056 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.82 on 31 degrees of freedom
## Multiple R-squared:  0.222,  Adjusted R-squared:  0.197
## F-statistic: 8.86 on 1 and 31 DF,  p-value: 0.00562
```

说明 x2 对 x1 系数没有影响

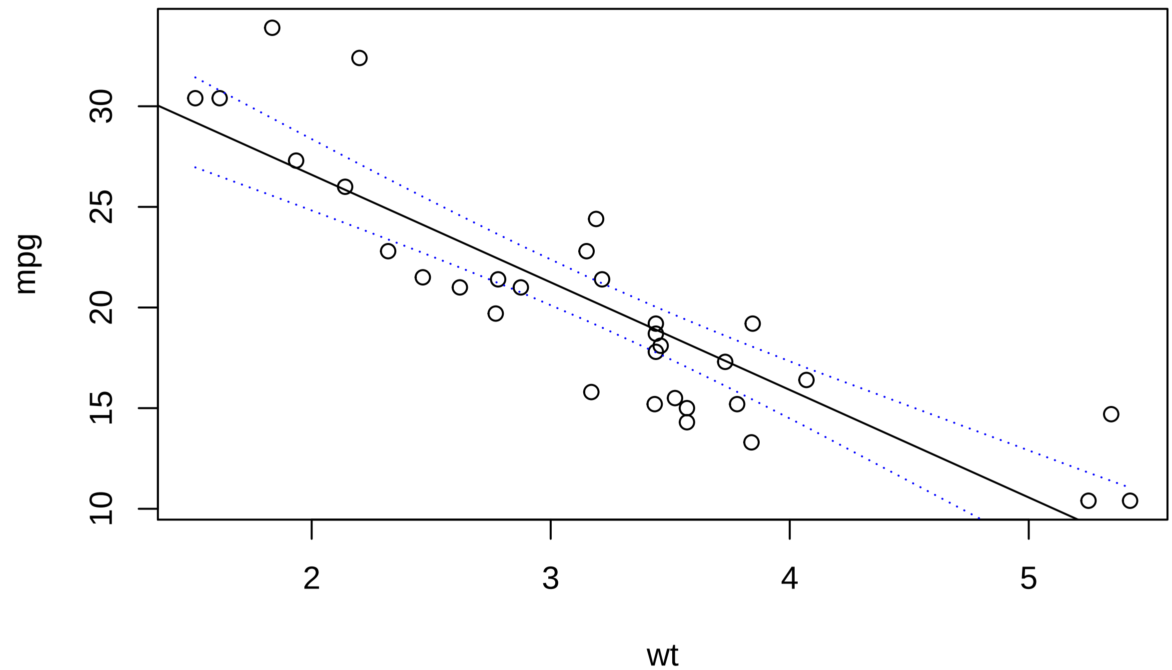
置信区间

```r
mtcar <- as.data.frame(mtcars[,c("mpg", "cyl", "disp", "hp", "wt")])
mtcarn<-mtcar[order(mtcar$wt),]
fit <- lm(mpg~wt, data=mtcarn)
conf=predict(fit,interval="confidence",level=0.95)
conf
```

```
##                       fit   lwr  upr
## Lotus Europa        29.20 26.96 31.4
```

```
## Honda Civic         28.65 26.52 30.8
## Toyota Corolla      27.48 25.55 29.4
## Fiat X1-9           26.94 25.11 28.8
## Porsche 914-2       25.85 24.20 27.5
## Fiat 128            25.53 23.93 27.1
## Datsun 710          24.89 23.38 26.4
## Toyota Corona       24.11 22.72 25.5
## Mazda RX4           23.28 21.99 24.6
## Ferrari Dino        22.48 21.27 23.7
## Volvo 142E          22.43 21.22 23.6
## Mazda RX4 Wag       21.92 20.75 23.1
## Merc 230            20.45 19.35 21.6
## Ford Pantera L      20.34 19.24 21.4
## Merc 240D           20.24 19.14 21.3
## Hornet 4 Drive      20.10 19.00 21.2
## AMC Javelin         18.93 17.80 20.1
## Hornet Sportabout   18.90 17.77 20.0
## Merc 280            18.90 17.77 20.0
## Merc 280C           18.90 17.77 20.0
## Valiant             18.79 17.66 19.9
## Dodge Challenger    18.47 17.32 19.6
## Duster 360          18.21 17.03 19.4
## Maserati Bora       18.21 17.03 19.4
## Merc 450SL          17.35 16.10 18.6
## Merc 450SLC         17.08 15.81 18.4
## Camaro Z28          16.76 15.45 18.1
## Pontiac Firebird    16.74 15.42 18.0
## Merc 450SE          15.53 14.06 17.0
## Cadillac Fleetwood   9.23  6.66 11.8
## Chrysler Imperial    8.72  6.05 11.4
## Lincoln Continental  8.30  5.55 11.0
```

```r
plot(mpg~wt, data=mtcarn)
abline(fit)
lines(mtcarn$wt,conf[,2],lty=3,col="blue")
lines(mtcarn$wt,conf[,3],lty=3,col="blue")
```



假设检验

```
mtcar <- as.data.frame(mtcars[,c("mpg", "cyl",
"disp", "hp", "wt")])
library(car)
fit <- lm(mpg ~ hp + wt, data = mtcar)
summary(fit)
```

```
##
## Call:
## lm(formula = mpg ~ hp + wt, data = mtcar)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -3.941 -1.600 -0.182  1.050  5.854
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 37.22727    1.59879   23.28  < 2e-16 ***
## hp          -0.03177    0.00903   -3.52   0.0015 **
## wt          -3.87783    0.63273   -6.13  1.1e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 2.59 on 29 degrees of freedom
## Multiple R-squared:  0.827,  Adjusted R-squared:  0.815
## F-statistic: 69.2 on 2 and 29 DF,  p-value: 9.11e-12
```

```r
linearHypothesis(fit, "hp = 0")# 变量 hp 的系数 =0
```

```
## Linear hypothesis test
##
## Hypothesis:
## hp = 0
##
## Model 1: restricted model
## Model 2: mpg ~ hp + wt
##
##   Res.Df RSS Df Sum of Sq    F Pr(>F)
## 1     30 278
## 2     29 195  1      83.3 12.4 0.0015 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
linearHypothesis(fit, "hp = -0.5")
```

```
## Linear hypothesis test
##
## Hypothesis:
## hp = - 0.5
##
## Model 1: restricted model
## Model 2: mpg ~ hp + wt
##
##   Res.Df   RSS Df Sum of Sq    F Pr(>F)
## 1     30 18280
## 2     29   195  1     18085 2689 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
linearHypothesis(fit, "hp - wt= 0")#hp 和 wt 相等
```

```
## Linear hypothesis test
##
## Hypothesis:
## hp - wt = 0
##
## Model 1: restricted model
## Model 2: mpg ~ hp + wt
##
##   Res.Df RSS Df Sum of Sq    F  Pr(>F)
## 1     30 439
## 2     29 195  1       244 36.3 1.5e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

F 检验的 f 值，总是对应假设中 T 检验 t 值的平方（要在相同原假设下采用正确形式）