13. 線性迴歸分析 Regression Analysis

鄒慶士 (Ching-Shih Tsou)
台北商業技術學院資訊與決策科學所
E-mail：cstsou@mail.ntcb.edu.tw

簡單線性迴歸

- 迴歸分析可以用來研究兩個數值變量間的線性關係。
- 下列為線性迴歸的基本模型：

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

- $i = 1,2,\ldots,n$；y_i是因變數；x_i是自變數，被用來解釋或預測y_i值；β_0及β_1是迴歸模型的參數，分別代表母體迴歸線的截距及斜率；ε_i則是隨機誤差項

估計參數β_0及β_1

- 簡單線性迴歸以最小平方法估計參數β_0及β_1。也就是在 $b_0 + b_1 x_i - y_i$ 為最小的目標下，求出參數估計值b_0與b_1的方法。
- 最小平方法先令迴歸係數為b_0及b_1，則估計的迴歸方程式可寫成 $\hat{y}_i = b_0 + b_1 x_i$ 而 $e_i = \widehat{y_i} - y_i$ 則為第i個觀測值的殘差。
- 最小平方法就是要找出(b_0, b_1)使觀察值與估計值的差之平方和 $\sum e_i^2$ 最小。
- 而b_0及b_1可經由下列公式求出：

$$b_1 = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2}, \quad \bar{y} = b_0 + b_1 \bar{x}$$

線性迴歸模型的假設

- 為了進行統計推論，必須對隨機誤差項ε_i做下列假設：
  - 1.隨機誤差間均相互獨立
  - 2.隨機誤差服從常態分配
  - 3.隨機誤差其平均數為 **0**
  - 4.隨機誤差其變異數為常數 σ^2

範例：最大心跳率

- 最大心跳率與年齡的關係為 $\text{Max} = 220 - \text{Age}.$
- 下列為15位受測者其最大心跳率的數據資料

Age	18	23	25	35	65	54	34	56	72	19	23	42	18	39	37
Max Rate	202	186	187	180	156	169	174	172	153	199	193	174	198	183	178

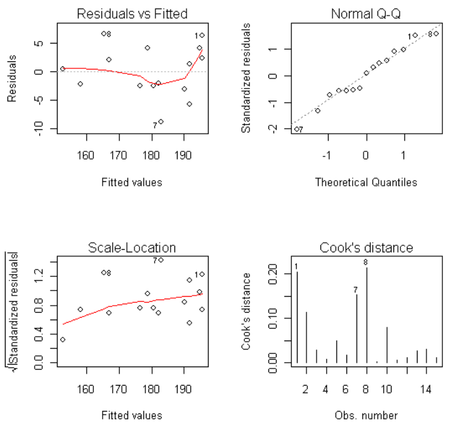
- 請複習如何畫出迴歸線

檢驗模型的假設

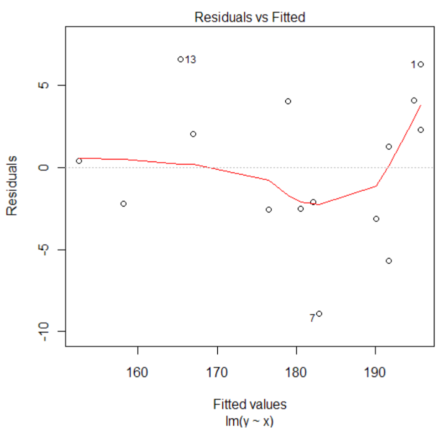
- 模型的有效性與資料是否符合模型假設密切相關，我們可以圖形化的探索式資料分析(EDA)檢查上述假設。
- 直方圖、盒鬚圖和常態機率圖。
- 殘差對於時間或觀測順序的圖表。
- 殘差對於估計值的圖表。

> par(mfrow=c(2,2))

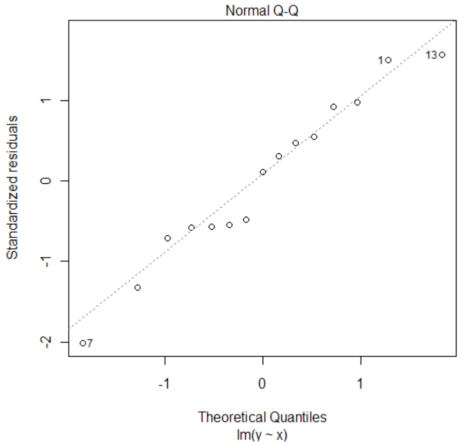
> plot(lm.result)



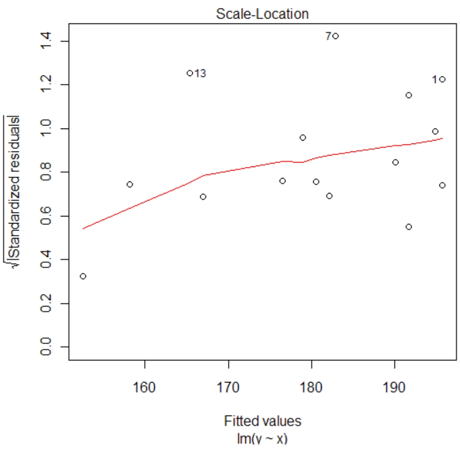
- **Residuals vs. fitted** 為殘差對於估計值 $\hat{Y}$ 的圖表。查看重點在於殘差是否在水平的y=0附近震動而沒有顯著的傾向。



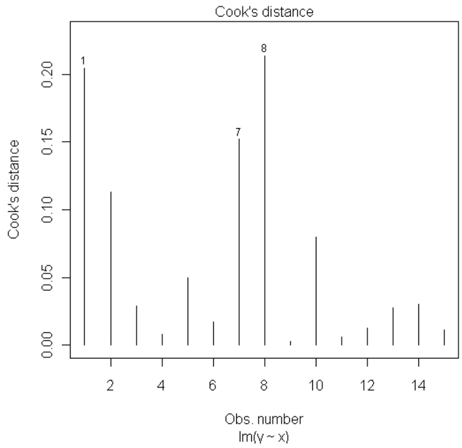
- **Normal qqplot** 如果點接近直線，則殘差是常態分佈的。



- **Scale-Location** 此圖顯示標準化殘差的平方根對於估計值的圖表，最高點是最大的殘差。



- **Cook's distance:** A combined measure of the "unusualness" of a case's predictors and response.



統計推論

- 若模型與資料的配適情形良好，則可進行β_0、β_1與σ的統計推論。
- 反應變項的條件分配為平均數等於$\beta_0+\beta_1 X$，標準差等於σ的常態分配。

估計迴歸母數

- 母數β_0、β_1與σ通常為未知，而必須由樣本資料來進行估計。
- 母體迴歸線上的截距β_0與斜率β_1之點估計式分別可由樣本迴歸線的截距 b_0與斜率 b_1求得。

統計推論 —σ

- 變異數(或標準差)相等的情況稱為變異數同質性(homoscedasticity)；不符合上述情況時，則稱為變異數異質性(heteroscedasticity)。
- 殘差的平方和可用來估計誤差項的變異數。

$$s^2 = \frac{1}{n-2}\sum(\widehat{y}_i - y_i)^2 = \frac{1}{n-2}\sum e_i^2.$$

統計推論 —β_1

- b_1是樣本迴歸線的斜率，也是 β_1 的不偏點估計式。
- b_1標準誤的求法是

$$\mathrm{SE}(b_1) = \frac{s}{\sqrt{\sum(x_i - \bar{x})^2}}$$

- 標準化後的b_1服從自由度為n-2的t分配

$$t = \frac{b_1 - \beta_1}{\mathrm{SE}(b_1)}$$

統計推論 —β_1

- 在 df＝n-2 的 t 分配下進行下列假說檢定。
- 虛無假設是H_0：β_1＝a
- 對立假設為H_1：$\beta_1 \neq$ a，檢定統計量的公式為：

$$t = \frac{b_1 - a}{\mathrm{SE}(b_1)}$$

- 計算 p 值。

案例：最大心跳率

- 檢定斜率是否為 -1， $H_0：\beta_1 = -1$ ， $H_1：\beta_1 \neq -1$
- 檢定統計量和 p 值 (雙尾)
  - es=resid(lm.result)
  - b1=(coef(lm.result))[['x']]
  - s=sqrt(sum(es^2)/(15-2))
  - SE=s/sqrt(sum((x-mean(x))^2))
  - t=(b1-(-1))/SE
  - pt(t,13,lower.tail=FALSE)
- pt(t, df, lower.tail = TRUE)
  - lower.tail = TRUE ，則機率為 $P[T <= \mathrm{t}]$；若 lower.tail = FALSE ，則機率為 $P[T > \mathrm{t}]$。

統計推論 — β_0

- b_0 是樣本迴歸線的截距，也是 β_0 的不偏點估計式。
- b_0的標準誤為

$$\mathrm{SE}(b_0) = s\sqrt{\frac{\sum x_i^2}{n\sum(x_i-\bar{x})^2}} = s\sqrt{\frac{1}{n}+\frac{\bar{x}^2}{\sum(x_i-\bar{x})^2}}$$

- 標準化後的b_0服從自由度為n-2的t分佈

$$t = \frac{b_0 - \beta_0}{SE(b_0)}$$

案例：最大心跳率

- 在 df＝n-2 的 t 分配下，截距是220。
- H_0：β_0＝220
- H_1：β_0＜220
- 利用先前的 s ，計算檢定統計量和 p 值 (左尾)。

```
> SE=s*sqrt(sum(x^2)/(n*sum((x-mean(x))^2)))
> b0=210.04846
> t=(b0-220)/SE
> pt(t,13,lower.tail=TRUE)
```


信賴區間與預測區間

- 迴歸線可在既定的自變數值下，預測應變數的期望值，或預測應變數的值。
- 前述預測值正確性如何呢？預測區間與信賴區間估計可以回答這個問題。
- 因為y_i 期望值的變異數小於 y_i 個別值的變異數，所以雖然兩者的公式看起來很像，但所建構出的區間是不同的。

$$b_0 + b_1 x\_i \pm t^*SE.$$

信賴區間與預測區間

- 給定 x ，應變數 y 的期望值多以 $\mu_{y|x}$ 表示，其標準誤為

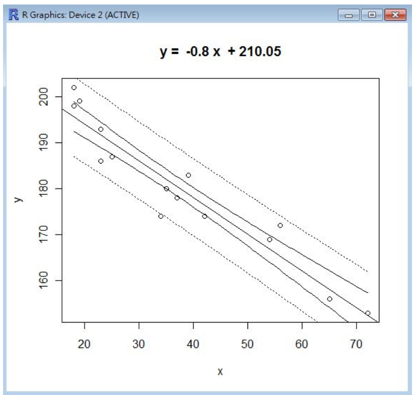
$$SE = s\sqrt{\frac{1}{n} + \frac{(x-\bar{x})^2}{\sum(x_i-\bar{x})^2}}$$

- 個別 y 值的標準誤則為

$$SE = s\sqrt{1 + \frac{1}{n} + \frac{(x-\bar{x})^2}{\sum(x_i-\bar{x})^2}}$$

信賴區間與預測區間

```
> simple.lm(x,y,show.ci=TRUE,conf.level=0.90)
```



R Basics: the low-level R commands

```
> lm.result = lm(y ~ x)
> summary(lm.result)
> plot(x,y)
> abline(lm.result)
> resid(lm.result)
> coef(lm.result)
> coef(lm.result)[1]
> coef(lm.result)['x'] or [['x']]
> fitted(lm.result)
> coefficients(lm.result)
> coefficients(summary(lm.result))[2, 2]
> coefficients(summary(lm.result))['x', 'Std. Error']
```


R Basics: the low-level R commands

#產生預測值

> predict(lm.result,data.frame(x= c(50,60))) # x 須與自變數同名

#產生應變數期望值的信賴區間並畫出圖形

> predict(lm.result,data.frame(x=sort(x)), level=.9,interval="confidence")

	fit	lwr	upr
1	195.6894	192.5083	198.8705
2	195.6894	192.5083	198.8705
3	194.8917	191.8028	197.9805
4	191.7007	188.9557	194.4458
5	191.7007	188.9557	194.4458
6	190.1053	187.5137	192.6969
7	182.9258	180.7922	185.0593
8	182.1280	180.0149	184.2411
9	180.5326	178.4390	182.6262
10	178.9371	176.8337	181.0405
11	176.5439	174.3723	178.7155
12	166.9712	164.0309	169.9116
13	165.3758	162.2564	168.4952
14	158.1962	154.1798	162.2127
15	152.6121	147.8341	157.3902

R Basics: the low-level R commands

>plot(x,y)

>abline(lm.result)

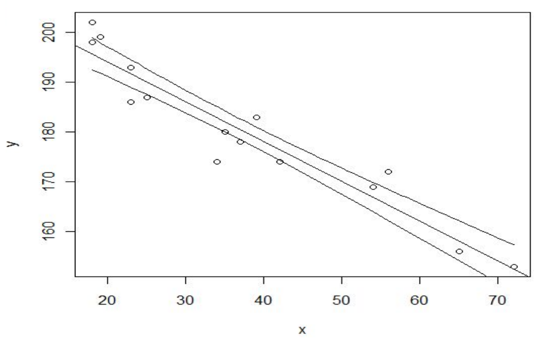
>ci.lwr=predict(lm.result,data.frame(x=sort(x)),level=.9,interval="confidence")[,2]

>points(sort(x),ci.lwr,type="l")

>curve(predict(lm.result,data.frame(x=x),level=.9,interval="confidence")[,3],add=T) # x 無須排序

R Basics: the low-level R commands

應變數期望值的信賴區間圖形

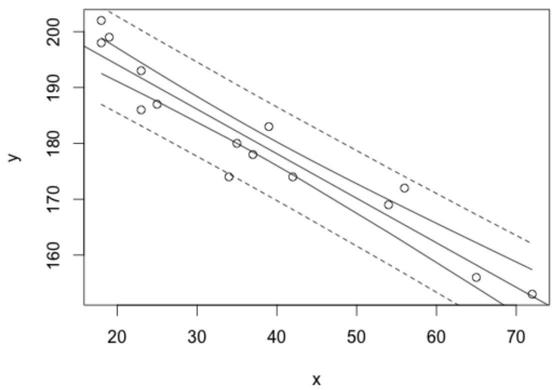


R Basics: the low-level R commands

```
>pi.lwr=predict(lm.result,data.frame(x=sort(x)),level=.9,interval="prediction")[,2]
>pi.upr=predict(lm.result,data.frame(x=sort(x)),level=.9,interval="prediction")[,3]
>points(sort(x),pi.lwr,type="l",lty=2)
>points(sort(x),pi.upr,type="l",lty=2)
>#curve(predict(lm.result,data.frame(x=x),level=.9,interval="prediction")[,3],add=T,lty=2)  # x 無須排序
```


R Basics: the low-level R commands

加上個別應變數的預測區間圖形(虛線)




The End