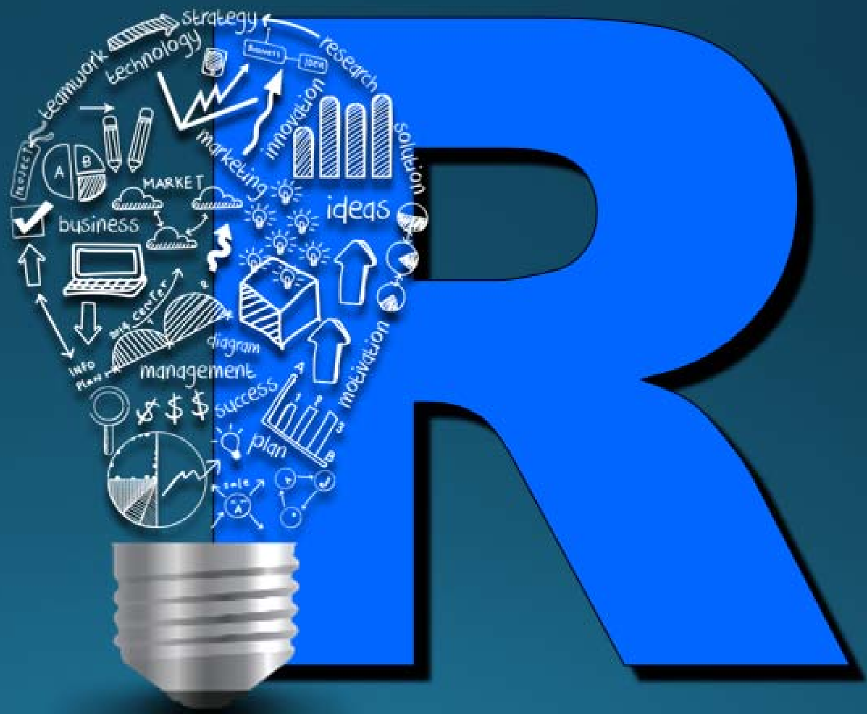
B05

統計模型與
迴歸分析



吳漢銘
國立政治大學 統計學系

http://www.hmwu.idv.tw

本章大綱

- **統計模型配適**
  - 解釋變數(X)，反應變數(Y)，模型公式in R
- **簡單線性迴歸 (Simple Linear Regression)**
  - 最小平方法、最大概似法、變異數表格(ANOVA Table)、信賴區間
- **Extract Information from Model Objects**
- **統計模型檢測**
  - Residual Plots、Normal QQ-plot、A Scale-Location Plot、Cook's Distance vs Row Labels、Residuals vs Leverages、Cook's Distance vs Leverage.
- **多重迴歸 (Multiple Linear Regression) 之模型選擇**: 逐步迴歸變數篩選法
- **解釋變數是類別變數(categorical)**: using dummy variables
- **反應變數是二元變數**: 羅吉斯迴歸 (Logistic Regression)
- **共線性 (Collinearity)**: 變異數膨脹因子(The Variance Inflation Factors)
- **高維度資料問題 (high-dimensional daata)**: large p small n

統計模型配適 (Statistical Modeling)

四個問題:

1. Which of your variables is the **response variable** (反應變數, Y)?
2. Which are the **explanatory variable** (解釋變數, $\boldsymbol{X}$)?
3. Are the explanatory variables **continuous** (連續) or **categorical** (類別), or a **mixture** (混合) of both?
4. What kind of response variable do you have: **continuous** measurement, a **count**, a **proportion**, a **time** at death, or **category**?

配適統計模型的目的

- To determine the values of the **parameters** in a specific model that lead to the best fit of the model to the data.

解釋變數, X

The Explanatory Variable (X)

- **All X's are continuous**: Regression

例如:

$$\begin{aligned}
&\text{Simple linear regression: } y = \beta_0 + \beta_1 x + \epsilon \\
&\text{Multiple linear regression: } y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p + \epsilon \\
&\text{Polynomial regression: } y = \beta_0 + \beta_1 x + + \beta_2 x^2 + \cdots + \beta_d x^d + \epsilon \\
&\text{Nonlinear regression: } y = \theta_0 + \theta_1 (1 - e^{\theta_2 x}) + \epsilon
\end{aligned}$$

- **All X's are categorical**: Analysis of Variance (ANOVA, 變異數分析)

例如:

$$\begin{aligned}
y_{ij} &= \mu + \alpha_j + \epsilon_{ij} \\
\boldsymbol{y} &= \boldsymbol{A\theta} + \boldsymbol{\epsilon}
\end{aligned}$$

- **X's are both continuous and categorical:** Analysis of Covariance (ANCOVA，共變異數分析)

例如:

$$y = \beta_0 + \beta_1 x + \theta z + \epsilon, \;\; z = \{0, 1\}$$

反應變數, Y

The Response Variable (Y)

- **Continuous**: Normal Regression, ANOVA or ANCOVA
- **Binary**: Binary Logistic Analysis

例如:

$$P(y_i = 0) = 1 - \pi_i,\ P(y_i = 1) = \pi_i$$

$$\text{Logistic link function: } g(\pi) = \log(\frac{\pi}{1-\pi})$$

$$\text{Logistic regression: } \log(\frac{\pi}{1-\pi}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon$$

- **Ordinal**: proportional-odds model

例如:

$$\gamma_j(\mathbf{x}) = P(Y \leq j|\mathbf{x}), \quad \log\left(\frac{\gamma_j(\mathbf{x})}{1-\gamma_j(\mathbf{x})}\right) = \boldsymbol{\beta}^T\mathbf{x}$$

反應變數, Y

- **Count**: Log-Linear Models

例如:

$$Y \sim Poisson(\mu),\ \mu = E(Y), \quad \log \mu = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p$$

- **Time at death**: Survival Analysis

  - T: survival time with a density function $f(t)$.
  - $1 - F(t)$: survival function (i.e., $F(t) = \int_{-\infty}^{t} f(s)\, ds$).
  - $h(t) = \dfrac{f(t)}{1 - F(t)}$: hazard function.
  - $h(t)\delta t$: the probability of dying in the next small interval δt given survival to time t
  - Proportional-hazards model: $h(t; \mathbf{x}) = \lambda(t) \exp(\beta^T \mathbf{x})$

模式寫法 (Model Formulae in R)

- The structure of the model: `response_variable ~ explanatory_variables`
  - Example: `fm <- formula(y ~ x)`
  - Example: `lm(fm), lm(y ~ x);  aov(y ~ x); glm(y ~ x)`
- `~`: "is modelled as a function of"
  - Example: `lm(y ~ x)`
- `+` : inclusion of an explanatory variable in the model (not addition);
  - Example: `lm(y ~ x1 + x2)`
- `-` : deletion of an explanatory variable from the model (not subtraction);
  - Example: `lm(y ~ x1 - 1)`
- `*` : inclusion of explanatory variables and interactions (not multiplication);
  - Example: `lm(y ~ x1 * x2)`
- `/`: nesting of explanatory variables in the model (not division);
  - Example: `lm(y ~ x1 / x2) # x1因子的各分類下，再細分出x2因子的分類`

Examples

```
> y <- rnorm(50)
> x1 <- rnorm(50)
> x2 <- rnorm(50)
> x3 <- rnorm(50)
> lm(y ~ x1 + x2)
Call:
lm(formula = y ~ x1 + x2)

Coefficients:
(Intercept)           x1
   -0.13024      0.05576
         x2
    0.02093
> lm(y ~ x1 - 1)
Call:
lm(formula = y ~ x1 - 1)

Coefficients:
     x1
0.03885
> lm(y ~ x1 * x2)
Call:
lm(formula = y ~ x1 * x2)

Coefficients:
(Intercept)           x1
   -0.05122     -0.03178
         x2        x1:x2
    0.05614      0.26850
```

```
> y <- rnorm(50)
> school <- as.factor(sample(c("a", "b", "c"), 50, replace = T))
> gender <- as.factor(sample(c("f", "m"), 50, replace = T))
> table(school, gender)
      gender
school  f  m
     a 10 12
     b  4  9
     c  6  9
> lm(y ~ school / gender)
Call:
lm(formula = y ~ school/gender)

Coefficients:
    (Intercept)          schoolb          schoolc
         0.1198           0.1504           1.0190
schoola:genderm  schoolb:genderm  schoolc:genderm
         0.1192          -0.0647          -1.3472

> lm(y ~ gender / school)

Call:
lm(formula = y ~ gender/school)

Coefficients:
    (Intercept)          genderm  genderf:schoolb
         0.1198           0.1192           0.1504
genderm:schoolb  genderf:schoolc  genderm:schoolc
        -0.0335           1.0190          -0.4475
```

模式寫法 (Model Formulae in R)

- `|`: indicates **conditioning** (not "or"), so that $y \sim x \mid z$ is read as "y as a function of x given z". Example: `lm(y ~ x | z)`

- "`:`" : a colon denotes an **interaction**
  - `A:B` means the two-way interaction between `A` and `B`
  - `N:P:K:Mg` means the four-way interaction between `N, P, K` and `Mg`.

```
> lm(y ~ x1 | x2)

Call:
lm(formula = y ~ x1 | x2)

Coefficients:
(Intercept)   x1 | x2TRUE
    -0.1216            NA
```

```
> lm(y ~ x1:x2:x3)

Call:
lm(formula = y ~ x1:x2:x3)

Coefficients:
(Intercept)     x1:x2:x3
   -0.08602     -0.20145
```

```
> # Create a formula for a model with a large number of variables:
> xnam <- paste("x", 1:25, sep="")
> (fmla <- as.formula(paste("y ~ ", paste(xnam, collapse= "+"))))
y ~ x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 + x9 + x10 + x11 +
    x12 + x13 + x14 + x15 + x16 + x17 + x18 + x19 + x20 + x21 +
    x22 + x23 + x24 + x25
```

模式寫法 (Model Formulae in R)

- `A * B * C` is the same as `A+B+C+A:B+A:C+B:C+A:B:C`
- `A/B/C` is the same as `A+B%in%A+C%in%B%in%A`
- `(A + B + C)^3` is the same as `A * B * C`
- `(A + B + C)^2` is the same as `A * B * C – A : B : C`

```
> y <- rnorm(50)
> A <- rnorm(50)
> B <- rnorm(50)
> C <- rnorm(50)
```

```
> lm(y ~ A * B * C)

Call:
lm(formula = y ~ A * B * C)

Coefficients:
(Intercept)            A            B
    0.20776     -0.04336      0.01105
          C          A:B          A:C
   -0.06969      0.14857     -0.02269
        B:C        A:B:C
   -0.06689      0.08850

> lm(y ~ A/B/C)

Call:
lm(formula = y ~ A/B/C)

Coefficients:
(Intercept)            A          A:B
    0.21586     -0.06219      0.12840
      A:B:C
    0.07229
```

```
> lm(y ~ (A + B + C) ^ 3)

Call:
lm(formula = y ~ (A + B + C) ^ 3)

Coefficients:
(Intercept)            A            B            C
    0.20776     -0.04336      0.01105     -0.06969
        A:B          A:C          B:C        A:B:C
    0.14857     -0.02269     -0.06689      0.08850

> lm(y ~ (A + B + C)^2)

Call:
lm(formula = y ~ (A + B + C) ^ 2)

Coefficients:
(Intercept)            A            B            C
    0.21990     -0.03953      0.02210     -0.05622
        A:B          A:C          B:C
    0.15181     -0.05379     -0.03787
```

Model Formula 例子(1)

Table 9.3. Examples of R model formulae. In a model formula, the function I case i) stands for 'as is' and is used for generating sequences I(1:10) or calculating quadratic terms I(x^2).

Model	Model formula	Comments
Null	y ~ 1	1 is the intercept in regression models, but here it is the overall mean *y*
Regression	y ~ x	*x* is a continuous explanatory variable
Regression through origin	y ~ x-1	Do not fit an intercept y ~ 0 + x
One-way ANOVA	y ~ sex	sex is a two-level categorical variable
One-way ANOVA	y ~ sex-1	as above, but do not fit an intercept (gives two means rather than a mean and a difference)
Two-way ANOVA	y ~ sex + genotype	genotype is a four-level categorical variable
Factorial ANOVA	y ~ N * P * K	*N*, *P* and *K* are two-level factors to be fitted along with all their interactions

Source: Crawley, M. J. , 2007, The R Book, Wiley.

Model Formula 例子(2)

Table 9.3. (Continued)

Model	Model formula	Comments
Three-way ANOVA	y ~ N*P*K – N:P:K	As above, but don't fit the three-way interaction
Analysis of covariance	y ~ x + sex	A common slope for *y* against *x* but with two intercepts, one for each sex
Analysis of covariance	y ~ x * sex	Two slopes and two intercepts
Nested ANOVA	y ~ a/b/c	Factor *c* nested within factor *b* within factor *a*
Split-plot ANOVA	y ~ a*b*c+Error(a/b/c)	A factorial experiment but with three plot sizes and three different error variances, one for each plot size
Multiple regression	y ~ x + z	Two continuous explanatory variables, flat surface fit
Multiple regression	y ~ x * z	Fit an interaction term as well (x + z + x:z)

Source: Crawley, M. J. , 2007, The R Book, Wiley.

Model Formula 例子(3)

Table 9.3. (Continued)

Model	Model formula	Comments
Multiple regression	y ~ x + I(x^2) + z + I(z^2)	Fit a quadratic term for both x and z
Multiple regression	y <- poly(x,2) + z	Fit a quadratic polynomial for x and linear z
Multiple regression	y ~ (x + z + w)^2	Fit three variables plus all their interactions up to two-way
Non-parametric model	y ~ s(x) +s(z)	y is a function of smoothed x and z in a generalized additive model
Transformed response and explanatory variables	log(y) ~ I(1/x) + sqrt(z)	All three variables are transformed in the model

the function I case i) stands
for 'as is' and is used for generating sequences I(1:10)
or calculating quadratic terms I(x^2).

The second order regression model.

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_{11} x_{i1}^2 + \beta_{22} x_{i2}^2 + \beta_{12} x_{i1} x_{i2} + \varepsilon_i$$

```
lm(Y ~ X1 * X2 + I(X1^2) + I(X2^2))
```

Source: Crawley, M. J. , 2007, The R Book, Wiley.

Statistical Models in R

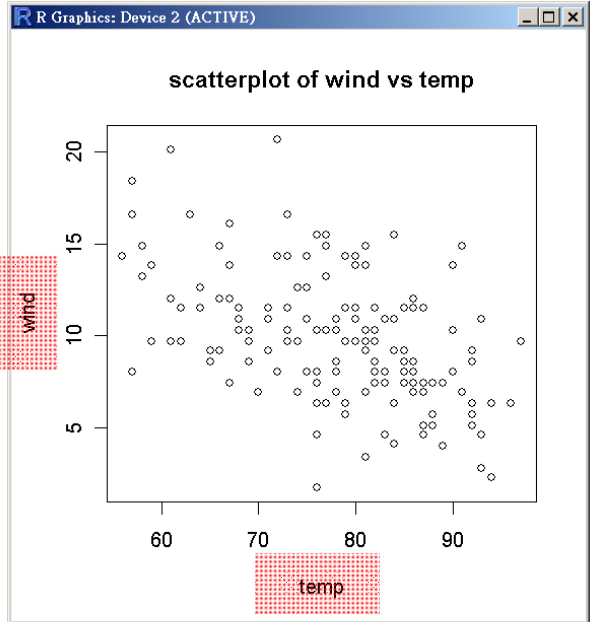
lm	fits a linear model with normal errors and constant variance; generally this is used for regression analysis using continuous explanatory variables.
aov	fits analysis of variance with normal errors, constant variance and the identity link; generally used for categorical explanatory variables or ANCOVA with a mix of categorical and continuous explanatory variables.
glm	fits generalized linear models to data using categorical or continuous explanatory variables, by specifying one of a family of **error structures** (e.g. Poisson for count data or binomial for proportion data) and a particular **link function**.
gam	fits generalized additive models
lme	and lmer fit linear mixed-effects models
nls	fits a non-linear regression model via least squares
nlme	fits a specified non-linear function in a mixed-effects model
loess	fits a local regression model
tree	fits a regression tree model using binary recursive partitioning

Source: Crawley, M. J. , 2007, The R Book, Wiley.

簡單線性迴歸 (Simple Linear Regression) (參數估計: 最小平方法)

```
> dim(airquality)
[1] 153   6
> head(airquality)
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6
```

空氣品質資料



抽10個觀察值
出來看, 比較清楚

數學模型 $y = \beta_0 + \beta_1 x$

$$(y_1, x_1), \cdots, (y_n, x_n)$$

參數估計: 最小平方法

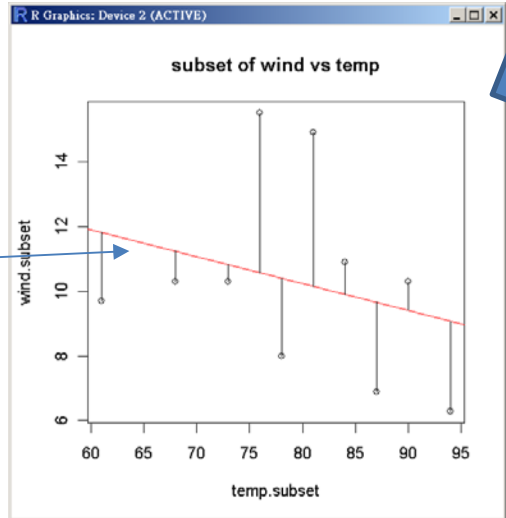
$$S(\beta_0, \beta_1) = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2$$

可當成評估指標

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$$



參數估計: 最小平方法

$$(y_1, x_1), \cdots, (y_n, x_n)$$

$$S(\beta_0, \beta_1) = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2$$

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$$

```
> y <- airquality$Wind
> x <- airquality$Temp
> xbar <- mean(x) ; xbar
[1] 77.88235
> ybar <- mean(y) ; ybar
[1] 9.957516

> beta1.num <- sum((x - xbar) * (y - ybar))
> beta1.den <- sum((x - xbar) ^ 2)
> (beta1.hat <- beta1.num/beta1.den)
[1] -0.1704644

> (beta0.hat <- ybar - beta1.hat * xbar)
[1] 23.23369
> yhat <- beta0.hat + beta1.hat * x
```

$$S_{xx} = \sum_{i=1}^{n} (x_i - \bar{x})^2$$

$$S_{yy} = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

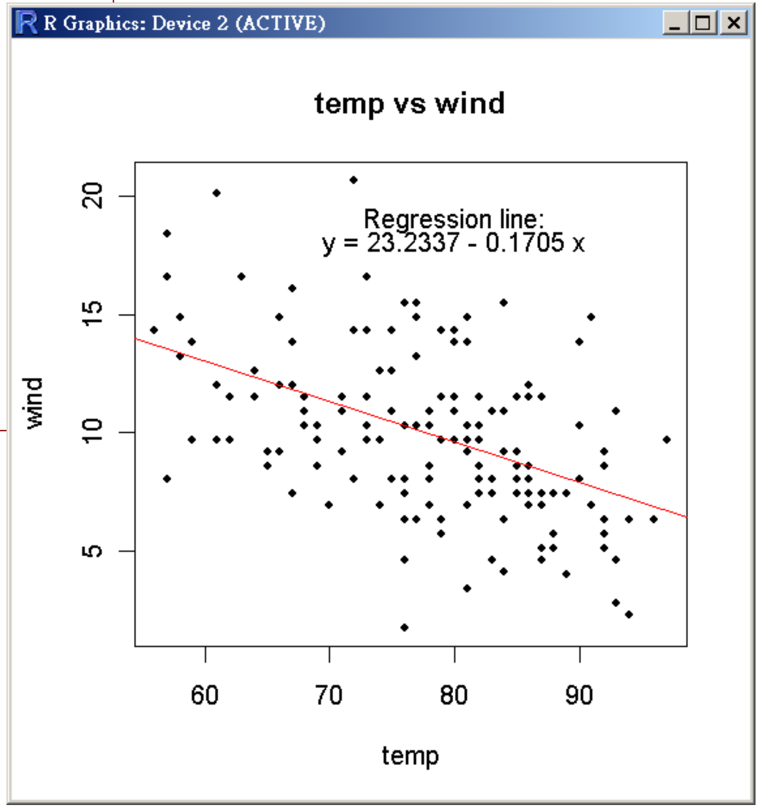
$$S_{xy} = \sum_{i=1}^{n} y_i (x_i - \bar{x})$$

$$\hat{\beta}_1 = \frac{S_{xy}}{S_{xx}}$$

```
> Sxy <- sum(y*(x-xbar)) ; Sxy
[1] -2321.365
> Sxx <- sum((x-xbar)^2) ; Sxx
[1] 13617.88
> Syy <- sum((y-ybar)^2) ; Syy
[1] 1886.554
> beta1.hat2 <- Sxy/Sxx ; beta1.hat2
[1] -0.1704644
```

lsfit: Find the Least Squares Fit

```
> model_fit <- lsfit(temp, wind)
> ls.print(model_fit)
Residual Standard Error=3.1422
R-Square=0.2098
F-statistic (df=1, 151)=40.0795
p-value=0

          Estimate Std.Err t-value Pr(>|t|)
Intercept  23.2337  2.1124 10.9987        0
X          -0.1705  0.0269 -6.3308        0


> plot(temp, wind, main = "temp vs wind", pch = 20)
> abline(model_fit, col = "red")
> text(80, 19, "Regression line:")
> text(80, 18, "y = 23.2337 - 0.1705 x")
```



簡單線性迴歸 (Simple Linear Regression) (參數估計: 最大概似法)

統計模型

$$y = \beta_0 + \beta_1 x + \epsilon \qquad E(\epsilon) = 0 \qquad Var(\epsilon) = \sigma^2$$

$$y = X\beta + \varepsilon, \qquad \varepsilon \sim N(0, \sigma^2 I) \qquad y \sim N(X\beta, \sigma^2 I)$$

參數估計: 最大概似法

$$\prod_{i=1}^{n} p(y_i | x_i; \beta_0, \beta_1, \sigma^2) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i - (\beta_0 + \beta_1 x_i))^2}{2\sigma^2}}$$

likelihood

$$\begin{aligned} L(\beta_0, \beta_1, \sigma^2) &= \log \prod_{i=1}^{n} p(y_i | x_i; \beta_0, \beta_1, \sigma^2) \\ &= \sum_{i=1}^{n} \log p(y_i | x_i; \beta_0, \beta_1, \sigma^2) \\ &= -\frac{n}{2} \log 2\pi - n \log \sigma - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - (\beta_0 + \beta_1 x_i))^2 \end{aligned}$$

$$\Rightarrow$$

$$\begin{aligned} \hat{\beta}_1 &= \frac{\sum_{i=1}^{n} (x_i - \overline{x})(y_i - \overline{y})}{\sum_{i=1}^{n} (x_i - \overline{x})^2} \\ \hat{\beta}_0 &= \overline{y} - \hat{\beta}_1 \overline{x} \\ \hat{\sigma}^2 &= \frac{1}{n} \sum_{i=1}^{n} (y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i))^2 \end{aligned}$$

統計推論: 信賴區間、假設檢定、近似理論

Testing just one predictor $H_0 : \beta_i = 0.$ **Test of all predictors** $H_0 : \beta_1 = \ldots \beta_{p-1} = 0$

統計模型檢測 (Model Checking): 殘差分析(Residual Analysis)

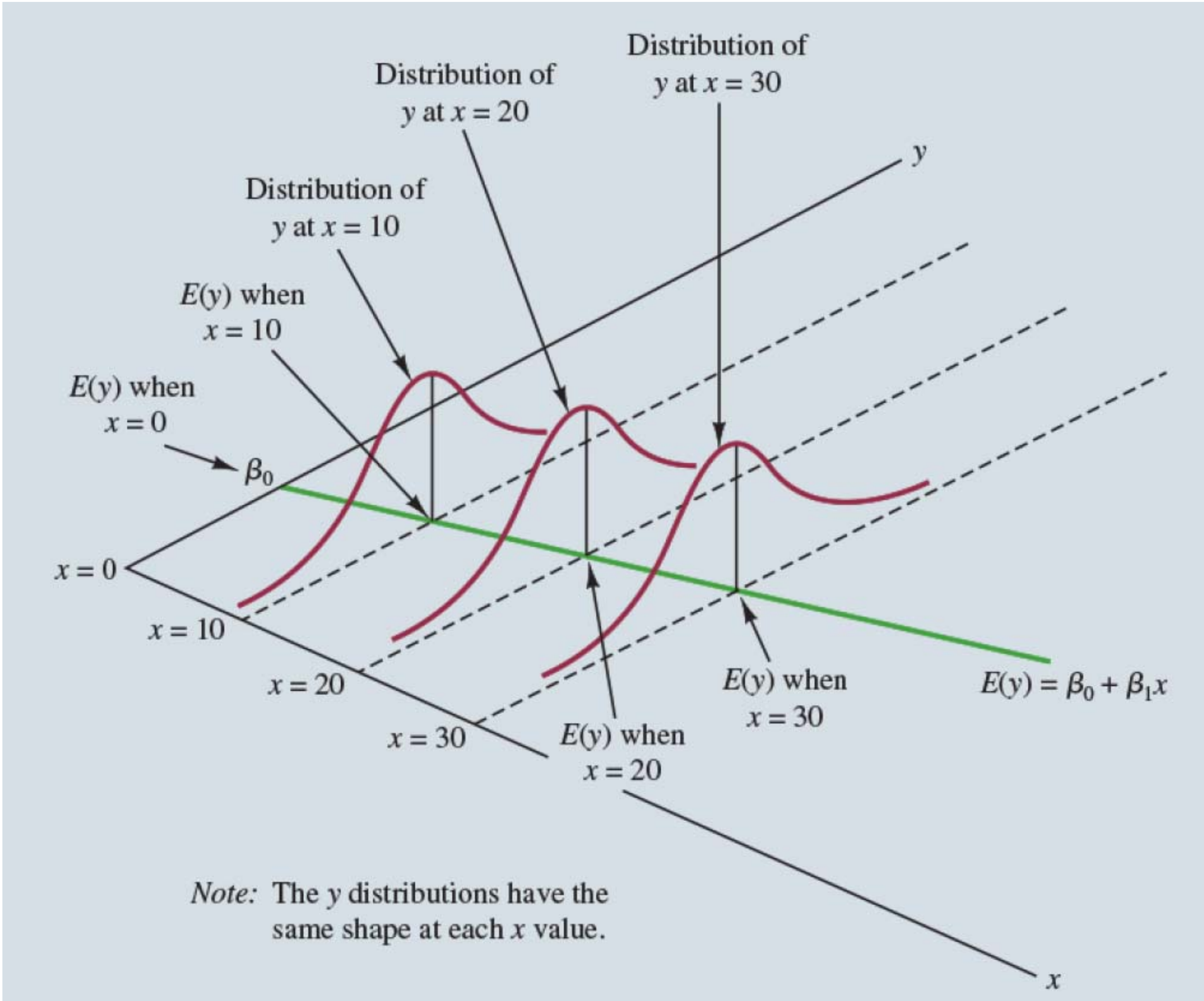
Assumptions About The Error Term ϵ in the Regression Model

If we want to make any confidence intervals or perform any hypothesis tests (**statistical inference**), we need to assume a normal error term.

$$y = X\beta + \varepsilon,$$

$$\varepsilon \sim N(0, \sigma^2 I)$$

$$y \sim N(X\beta, \sigma^2 I)$$



Source: Anderson et al., 2019, Statistics for Business & Economics (14th Edition), Cengage Learning Ltd. (ISBN: 0357114485).

t Test for Significance in SLR

$$\hat{\beta} = (X^T X)^{-1} X^T y \sim N(\beta, (X^T X)^{-1} \sigma^2)$$

unknown

t Test for Significance in Simple Linear Regression

(a) Hypothesis:

$$H_0 : \beta_1 = 0, \qquad H_a : \beta_1 \neq 0$$

(b) Test Statistic: $t = \dfrac{b_1}{s_{b_1}}$

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

(c) Rejection Rule:

i. p-value approach: Reject H_0 if p-value $\leq \alpha$

ii. Critical value approach: Reject H_0 if $t \leq -t_{\alpha/2}$ or if $t \geq t_{\alpha/2}$.

where $t_{\alpha/2}$ is based on a t distribution with $n-2$ degrees of freedom.

Confidence Interval for β_1

$$b_1 \pm t_{\alpha/2, n-2} s_{b_1}$$

Testing just one predictor $\quad H_0 : \beta_i = 0. \qquad t_i = \hat{\beta}_i / se(\hat{\beta}_i)$

F Test for Significance in SLR

F **Test for Significance in Simple Linear Regression**

(a) Hypothesis: $H_0: \beta_1 = 0, \qquad H_a: \beta_1 \neq 0$

(b) Test Statistic: $F = \frac{MSR}{MSE}$

(c) Rejection Rule:

i. p-value approach: Reject H_0 if p-value $\leq \alpha$

ii. Critical value approach: Reject H_0 if $\underline{F \geq F_{\alpha,1,n-2}}$

where F_α is based on an F distribution with 1 degree of freedom in the numerator and $n-2$ degrees of freedom in the denominator.

$$e_i = y_i - \hat{y}_i$$

$$SS_E = \sum_{i=1}^{n} e_i^2 \qquad MS_E = \frac{SS_E}{n-2} = \hat{\sigma}^2$$

$$SS_R = \hat{\beta}_1 S_{xy} \qquad MS_R = SS_R/1$$

$$F_0 = MS_R/MS_E$$

F Test for Significance in MLR

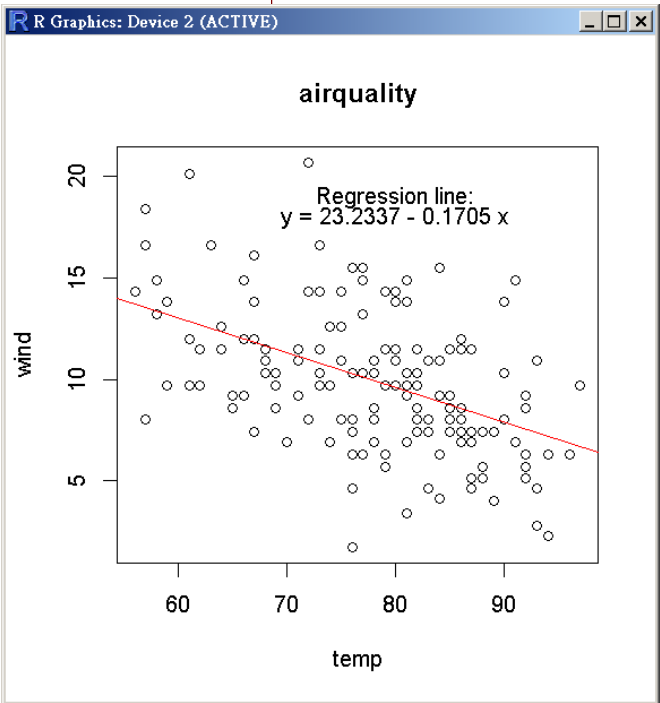
Test of all predictors

$$H_0: \beta_1 = \ldots \beta_{p-1} = 0 \qquad F = \frac{(\mathrm{SYY} - \mathrm{RSS})/(p-1)}{\mathrm{RSS}/(n-p)}$$

lm: Fit A Linear Model

```
> my_model <- lm(wind ~ temp)
> my_model

Call:
lm(formula = wind ~ temp)

Coefficients:
(Intercept)         temp
    23.2337      -0.1705

> summary(my_model)

Call:
lm(formula = wind ~ temp)

Residuals:
    Min      1Q  Median      3Q     Max
-8.5784 -2.4489 -0.2261  1.9853  9.7398

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 23.23369    2.11239  10.999  < 2e-16 ***
temp        -0.17046    0.02693  -6.331 2.64e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.142 on 151 degrees of freedom
Multiple R-squared: 0.2098,     Adjusted R-squared: 0.2045
F-statistic: 40.08 on 1 and 151 DF,  p-value: 2.642e-09
```

```
> plot(wind ~ temp, main = "airquality")
> abline(my_model, col = "red")
> text(80, 19, "Regression line:")
> text(80, 18, "y = 23.2337 - 0.1705 x")
```



Test of all Predictors and ANOVA Table

The ANOVA Table for Regression

Source	SS (*Sum of Squares, the numerator of the variance*)	DF (*the denominator*)	MS (*Mean Square, the variance*)	F
Regression (*or Model*)	$SSR=\sum_{i=1}^{n}((\hat{\beta}_0+\hat{\beta}_1 x_i)-\bar{y})^2$	*2-1=1*	$MSR=\frac{SSR}{1}$	$F=\frac{MSR}{MSE}$
Error	$SSE=\sum_{i=1}^{n}(y_i-(\hat{\beta}_0+\hat{\beta}_1 x_i))^2$	*n-2*	$MSE=\frac{SSE}{n-2}$	
Total	$TSS=\sum_{i=1}^{n}(y_i-\bar{y})^2$	*n-1*		

$$SST=\sum(y_i-\bar{y})^2$$

$$SSE=\sum(y_i-\hat{y}_i)^2$$

$$SSR=\sum(\hat{y}_i-\bar{y})^2$$

```
> my_aov <- aov(my_model)
> summary(my_aov)
             Df  Sum Sq Mean Sq F value    Pr(>F)
temp          1  395.71  395.71  40.080 2.642e-09 ***
Residuals   151 1490.84    9.87
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.
```

```
> n <- length(wind)
> e <- y - yhat
> SSE <- sum(e^2) ; SSE
[1] 1490.844
> MSE <- SSE/(n-2) ; MSE
[1] 9.873137
> SSR <- beta1.hat*Sxy ; SSR
[1] 395.7101
> MSR <- SSR/1 ; MSR
[1] 395.7101
> SST <- SSR + SSE ; SST
[1] 1886.554
> Syy
[1] 1886.554
> F0 <- MSR/MSE; F0
[1] 40.07947
```

F Test for Multiple Regression Relation: Test of all Predictors and ANOVA Table

H_0 : $\beta_1 = \beta_2 = \cdots = \beta_{p-1} = 0$

H_a : not all $\beta_k, (k = 1, \cdots, p-1)$ equal zero

$SST = \sum(y_i - \bar{y})^2$

$SSE = \sum(y_i - \hat{y}_i)^2$

$SSR = \sum(\hat{y}_i - \bar{y})^2$

Source of Variation	SS	df	MS
Regression	$SSR = \mathbf{b}'\mathbf{X}'\mathbf{Y} - \left(\frac{1}{n}\right)\mathbf{Y}'\mathbf{J}\mathbf{Y}$	$p-1$	$MSR = \frac{SSR}{p-1}$
Error	$SSE = \mathbf{Y}'\mathbf{Y} - \mathbf{b}'\mathbf{X}'\mathbf{Y}$	$n-p$	$MSE = \frac{SSE}{n-p}$
Total	$SSTO = \mathbf{Y}'\mathbf{Y} - \left(\frac{1}{n}\right)\mathbf{Y}'\mathbf{J}\mathbf{Y}$	$n-1$	

The test statistic: $F^* = \frac{MSR}{MSE}$

The decision rule to control the Type I error at α:

If $F^* > F_{(1-\alpha; p-1, n-p)}$, reject H_0.

判定系數 Coefficient of Determination $R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$

參數估計之信賴區間

$100(1-\alpha)\%$ confident interval on the intercept β_0.

$$E(\hat{\beta}_0) = \beta_0 \qquad se(\hat{\beta}_0) = \sqrt{MS_E(1/n + \bar{x}^2/S_{xx})}$$

$$\hat{\beta}_0 - t_{\alpha/2,n-1} se(\hat{\beta}_0) \leq \beta_0 \leq \hat{\beta}_0 + t_{\alpha/2,n-1} se(\hat{\beta}_0)$$

$100(1-\alpha)\%$ confident interval on the slope β_1.

$$E(\hat{\beta}_1) = \beta_1 \qquad se(\hat{\beta}_1) = \sqrt{MS_E/S_{xx}}$$

$$\hat{\beta}_1 - t_{\alpha/2,n-1} se(\hat{\beta}_1) \leq \beta_1 \leq \hat{\beta}_1 + t_{\alpha/2,n-1} se(\hat{\beta}_1)$$

```
> alpha <- 0.05
> se.beta0 <- sqrt(MSE*(1/n+xbar^2/Sxx)) ; se.beta0
[1] 2.112395
> tstar <- qt(alpha/2, n-1)* se.beta0
> CI.beta0 <- beta0.hat + c(tstar, - tstar) ; CI.beta0
[1] 19.0600210 27.407355
```

```
> confint(my_model)
                 2.5 %    97.5 %
(Intercept) 19.0600210 27.407355
Temp        -0.2236649 -0.117264
```

```
> se.beta1 <- sqrt(MSE/Sxx) ; se.beta1
[1] 0.02692606
> tstar <- qt(alpha/2, n-1)* se.beta1
> CI.beta1 <- beta1.hat + c(tstar, - tstar); CI.beta1
[1] -0.2236649 -0.117264
```

多重迴歸分析
Multiple regression model

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots \beta_p x_p + \epsilon$$

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_{p-1} X_{i,p-1} + \varepsilon_i$$

ε_i are independent $\underline{N(0, \sigma^2)}$, $i = 1, \cdots, n.$

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

$$\mathbf{Y} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} 1 & X_{11} & X_{12} & \cdots & X_{1,p-1} \\ 1 & X_{21} & X_{22} & \cdots & X_{2,p-1} \\ \vdots & \vdots & & & \\ 1 & X_{n1} & X_{n2} & \cdots & X_{n,p-1} \end{bmatrix} \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{p-1} \end{bmatrix}, \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

$$\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$$

$$R_a^2 = 1 - \frac{SSE/(n-p)}{SST/(n-1)} = 1 - \left(\frac{n-1}{n-p}\right)\frac{SSE}{SST}$$

Generic Functions

```
> my_model <- lm(wind ~ temp)
> summary(my_model)
```

- **`summary`**: produces parameter estimates and standard errors from **`lm`**, and ANOVA tables from **`aov`**.
- **`plot`**: produces diagnostic plots for model checking, including residuals against fitted values, influence tests, etc.
- **`update`**: is used to modify the last model fit; it saves both typing effort and computing time.
- **`predict`**: uses information from the fitted model to produce smooth functions for plotting a line through the scatterplot of your data.
- **`fitted`**: gives the fitted values, predicted by the model for the values of the explanatory variables included.
- **`resid`**: gives the residuals.

Extract Information from Model Objects

方法一: by functions

```
> my_model <- lm(wind ~ temp)
> summary(my_model)

Call:
lm(formula = wind ~ temp)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.5784 -2.4489 -0.2261  1.9853  9.7398 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 23.23369    2.11239  10.999  < 2e-16 ***
temp        -0.17046    0.02693  -6.331 2.64e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.142 on 151 degrees of freedom
Multiple R-squared: 0.2098,     Adjusted R-squared: 0.2045 
F-statistic: 40.08 on 1 and 151 DF,  p-value: 2.642e-09
```

```
> coef(my_model)
(Intercept)        temp 
 23.2336881  -0.1704644 

> vcov(my_model)
            (Intercept)          temp
(Intercept)  4.46221130 -0.0564656925
temp        -0.05646569  0.0007250127
```

Extract Information from Model Objects

```
> summary(my_model)[[1]]  # my.model formula
lm(formula = wind ~ temp)
> summary(my_model)[[2]]  # attributes of the objects
wind ~ temp
attr(,"variables")
list(wind, temp)
attr(,"factors")
     temp
wind    0
temp    1
attr(,"term.labels")
[1] "temp"
attr(,"order")
[1] 1
attr(,"intercept")
[1] 1
attr(,"response")
[1] 1
attr(,".Environment")
<environment: R_GlobalEnv>
attr(,"predvars")
list(wind, temp)
attr(,"dataClasses")
     wind      temp
"numeric" "numeric"
```

方法二: with list subscripts

```
> length(summary(my_model))
[1] 11
> names(summary(my_model))
 [1] "call"   "terms"   "residuals" "coefficients"
 [5] "aliased"  "sigma" "df"   "r.squared"
 [9] "adj.r.squared" "fstatistic"    "cov.unscaled"
> summary(my_model)$sigma
[1] 3.142155
> summary(my_model)[[6]]
[1] 3.142155
> length(summary(my_model)[[1]])
[1] 2
> length(summary(my_model)[[2]])
[1] 3
> length(summary(my_model)[[3]])
[1] 153
```

Extract Information from Model Objects

方法二: with list subscripts

```
> summary(my_model)[[3]]  # residuals for data points
          1           2           3           4           5           6 
-4.41257055 -2.96024835  1.98068054 -1.16489276  0.61232059  2.91696501 
...
145         146         147         148         149         150 
-1.93071279  0.87393162 -1.17164167  4.10557168 -4.40117723  3.09207386 
        151         152         153 
 3.85114498 -2.27839058 -0.14210611 

> summary(my_model)[[4]]  # parameters table
              Estimate Std. Error   t value     Pr(>|t|)
(Intercept) 23.2336881 2.11239468 10.998744 4.901351e-21
temp        -0.1704644 0.02692606 -6.330835 2.641597e-09

> summary(my_model)[[4]][[1]]  # intercept
[1] 23.23369

> summary(my_model)[[4]][[2]]  # slope,.... summary(my.model)[[4]][[28]]
[1] -0.1704644
```

```
> str(summary(my_model)[[4]])
 num [1:2, 1:4] 23.2337 -0.1705  2.1124  0.0269 10.9987 ...
 - attr(*, "dimnames")=List of 2
  ..$ : chr [1:2] "(Intercept)" "temp"
  ..$ : chr [1:4] "Estimate" "Std. Error" "t value" "Pr(>|t|)"
```

Extract Information from Model Objects

方法二: with list subscripts

```
> summary(my_model)[[5]]  # whether the fit should be returned.
(Intercept)        temp 
      FALSE       FALSE 
> summary(my_model)[[6]]  # residual standard error
[1] 3.142155
> summary(my_model)[[7]]  # the number of rows in the summary.lm table.
[1]   2 151   2
> summary(my_model)[[8]]  # r square, the fraction of the total variation
   in the response variable that is explained by the my.model.
[1] 0.2097529
> summary(my_model)[[9]]  # adjusted r square
[1] 0.2045195
> summary(my_model)[[10]]  # F ratio information
    value     numdf     dendf 
 40.07947   1.00000 151.00000 
> summary(my_model)[[11]]  # correlation matrix of the parameter estimates.
             (Intercept)          temp
(Intercept)  0.451954754 -5.719124e-03
temp        -0.005719124  7.343286e-05
```

Extract Information from Model Objects

方法三: using $

```
> my_model <- lm(wind ~ temp)
> names(my_model)
 [1] "coefficients"  "residuals"     "effects"       "rank"
 [5] "fitted.values" "assign"        "qr"            "df.residual"
 [9] "xlevels"       "call"          "terms"         "model"

> my_model$coefficients
> my_model$fitted.values
> my_model$residuals
```

依此類推...

```
> summary.aov(my_model)
> summary.aov(my_model)[[1]][[1]]
> summary.aov(my_model)[[1]][[5]]
```

Extract Information from Model Objects

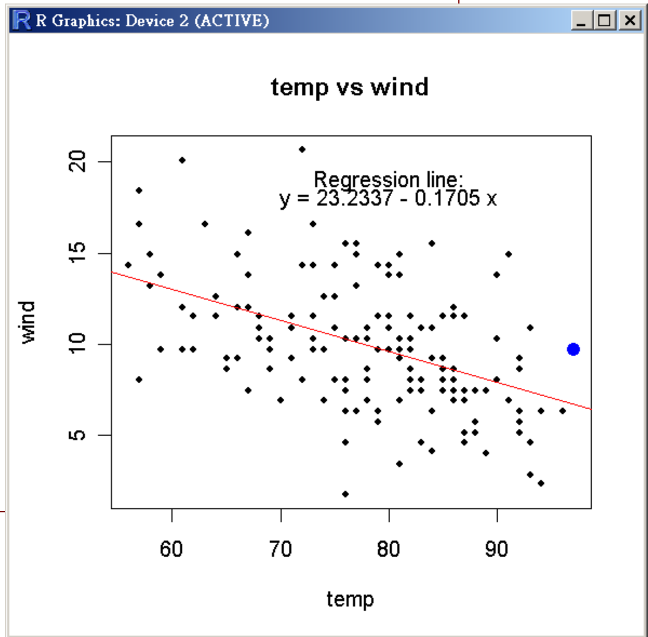
方法四: using ["names"]

```
> (iris_aov <- aov(iris[,1] ~ iris[,5]))
Call:
   aov(formula = iris[, 1] ~ iris[, 5])

Terms:
                iris[, 5] Residuals
Sum of Squares   63.21213  38.95620
Deg. of Freedom         2       147

Residual standard error: 0.5147894
Estimated effects may be unbalanced
> (iris_sum_aov <- summary(iris_aov))
             Df Sum Sq Mean Sq F value Pr(>F)
iris[, 5]     2  63.21  31.606   119.3 <2e-16 ***
Residuals   147  38.96   0.265
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> (iris_sum_aov2 <- unlist(iris_sum_aov))
         Df1          Df2      Sum Sq1      Sum Sq2     Mean Sq1     Mean Sq2     F value1
2.000000e+00 1.470000e+02 6.321213e+01 3.895620e+01 3.160607e+01 2.650082e-01 1.192645e+02
    F value2      Pr(>F)1      Pr(>F)2
          NA 1.669669e-31           NA
> names(iris_sum_aov2)
 [1] "Df1"      "Df2"      "Sum Sq1"  "Sum Sq2"  "Mean Sq1" "Mean Sq2" "F value1"
 [8] "F value2" "Pr(>F)1"  "Pr(>F)2"
> iris_sum_aov2["Pr(>F)1"]
     Pr(>F)1
1.669669e-31
```

使用子集合做分析

- Investigate how much a influence point affected the parameter estimates and their standard error.
- Repeat the statistical modeling but leave out the point in question, using subset.

```
> new_model <- update(my_model, subset = (temp != max(temp)))
> summary(new_model)

Call:
lm(formula = wind ~ temp, subset = (temp != max(temp)))

Residuals:
    Min      1Q  Median      3Q     Max 
-8.5663 -2.3871 -0.2027  1.9662  9.7344 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  23.5529     2.1382  11.015  < 2e-16 ***
temp         -0.1748     0.0273  -6.403 1.85e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.143 on 150 degrees of freedom
Multiple R-squared: 0.2147,     Adjusted R-squared: 0.2094 
F-statistic:    41 on 1 and 150 DF,  p-value: 1.847e-09
```

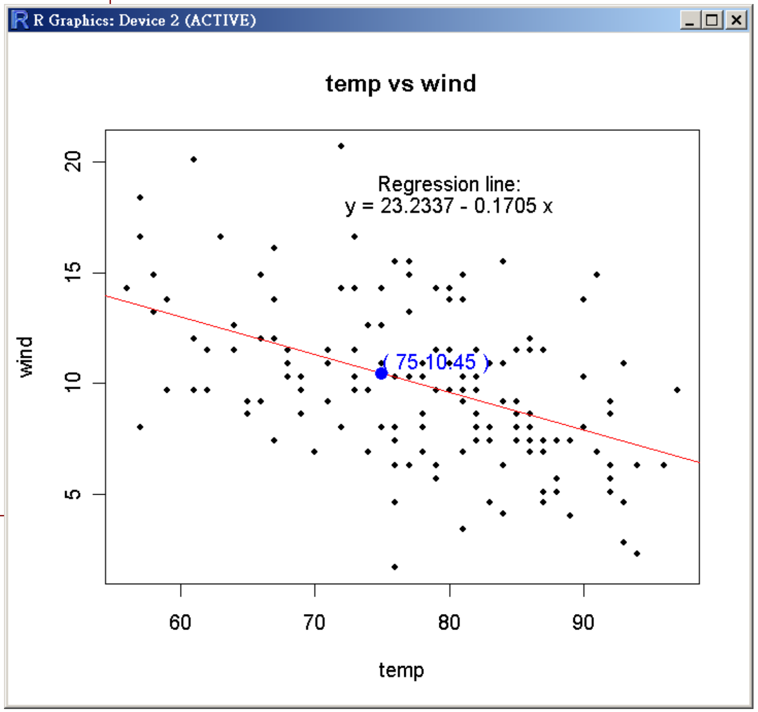


課堂練習：

- 將要刪除的點在二維散佈圖上標出來。
- 更新二維散佈圖及Regression Fit。

預測 (Prediction)

```
> summary(wind)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.700   7.400   9.700   9.958  11.500  20.700
> summary(temp)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  56.00   72.00   79.00   77.88   85.00   97.00

> predict(my_model, list(temp = 75))
[1] 10.44886
> predict(my_model, list(temp = c(66, 80,100)))
        1         2         3
11.983035  9.596533  6.187244
```



課堂練習:

- 將predict出來的值在二維散佈圖上標出來。

```
new_data <- data.frame(X1 = 160, X2 = 2)
predict(mylm, newdata = new_data, interval = "prediction")
```

- After fitting a model to data we need to investigate how well the model describes the data to see if there are any systematic trends in the goodness of fit.

- We hope that $\varepsilon \sim N(0,\sigma^2 I)$, but
  - Errors may be heterogeneous (unequal variance).
  - Errors may be correlated.
  - Errors may not be normally distributed. (less serious, the βhat's will tend to normality due to the power of the central limit theorem. With larger datasets, normality of the data is not much of a problem.

"Essentially, all models are wrong, but some are useful"
https://en.wikipedia.org/wiki/All_models_are_wrong

Box married Joan Fisher,
the second of R.A. Fisher (1890-1962) five daughters.

George Box (1919-2013),
Professor Emeritus of Statistics,
University of Wisconsin-Madison

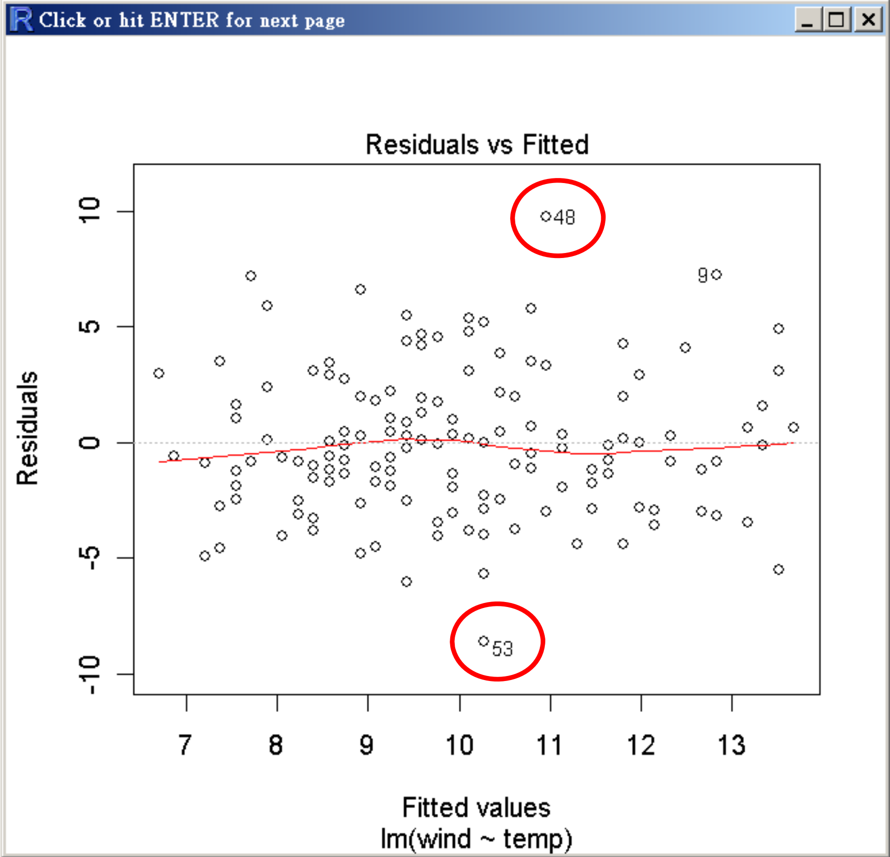
1. 殘差vs. 估計值: Residual Plots

```
> ?plot.lm
```

default

This plot should be
with no pattern of any sort.

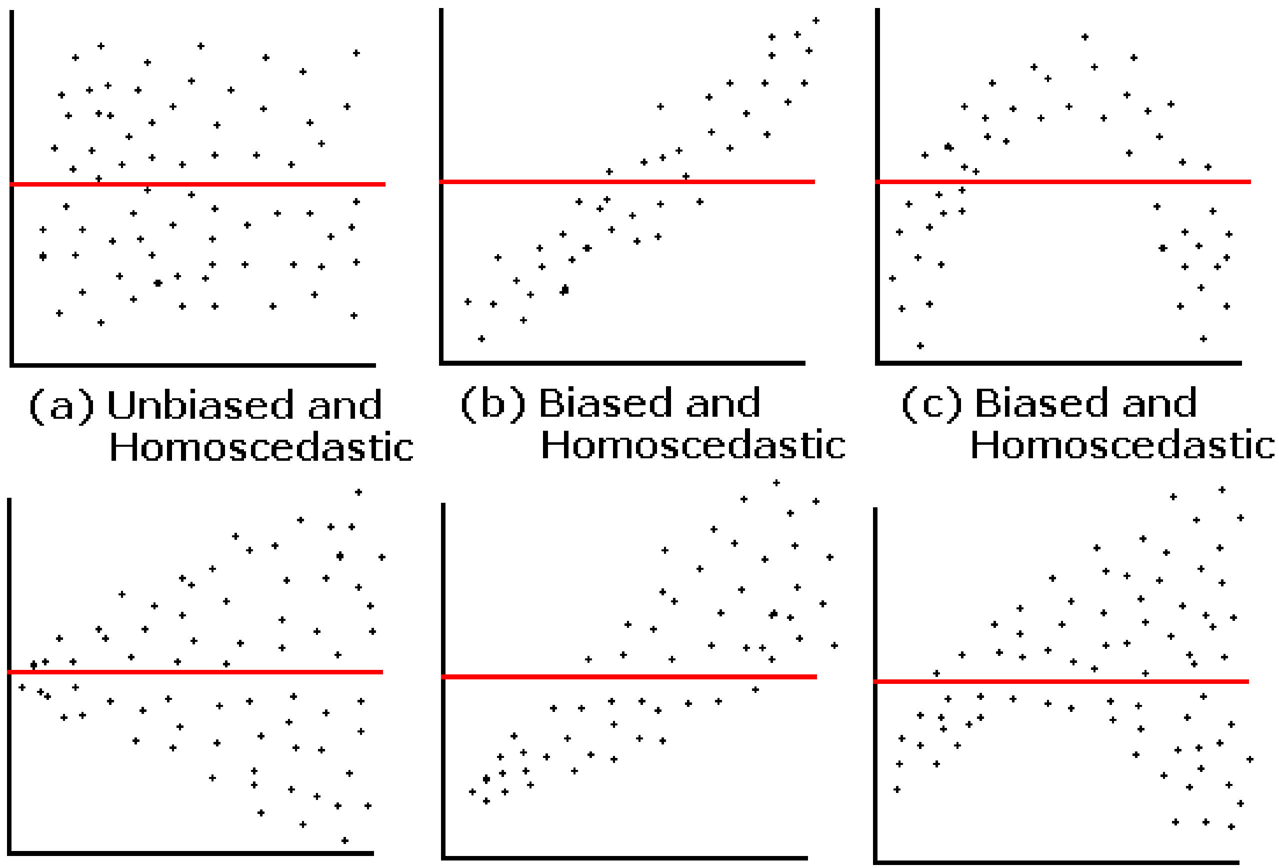
```
> wind <- airquality$Wind
> temp <- airquality$Temp
> my_model <- lm(wind ~ temp)
> plot(my_model, which = 1:6)
Waiting to confirm page change...
```




```
> plot(fitted(my_model), residuals(my_model), xlab = "Fitted values",
+  ylab = "Residuals")
> abline(h = 0, lty = 2)
```

課堂練習: 將Residuals大於±6的點標出來(顏色為紅色)。

殘差圖 Residual Plots

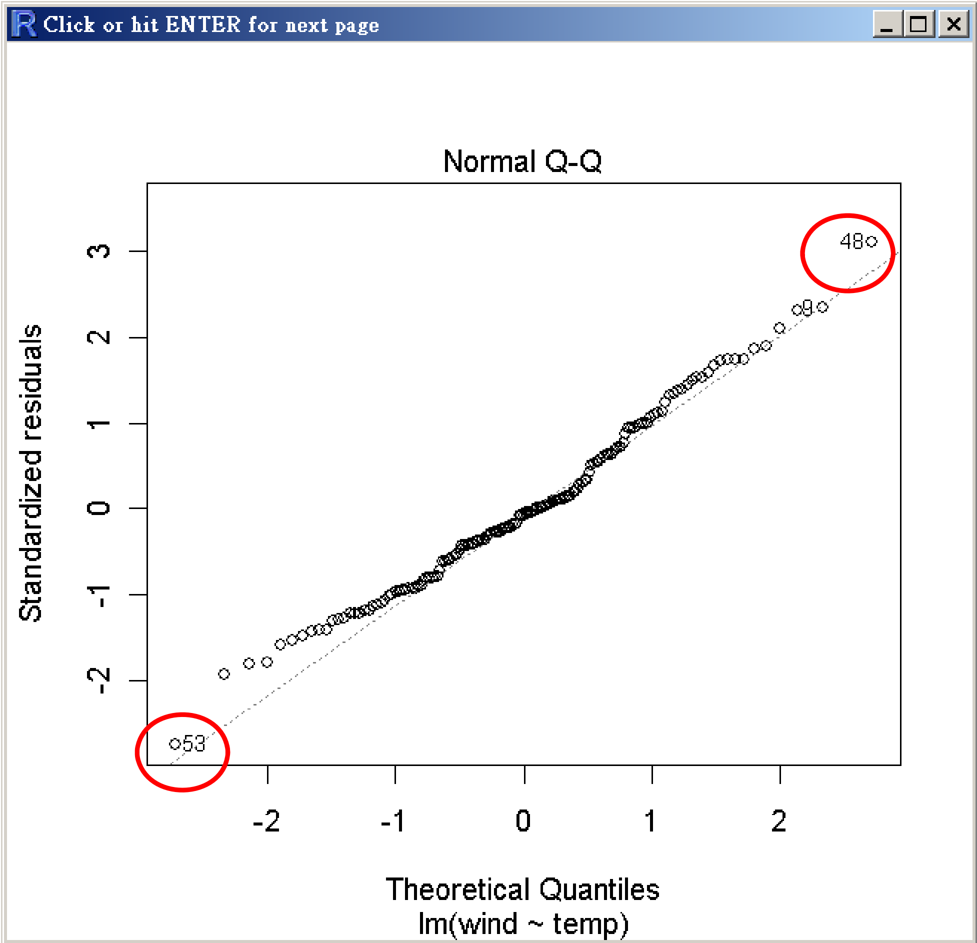


(a) Unbiased and Homoscedastic
(b) Biased and Homoscedastic
(c) Biased and Homoscedastic
(d) Unbiased and Heteroscedastic
(e) Biased and Heteroscedastic
(f) Biased and Heteroscedastic

https://www.r-bloggers.com/model-validation-interpreting-residual-plots/

2. 常態QQ圖 (Normal QQ-plot) (Normal Probability Plot)

default

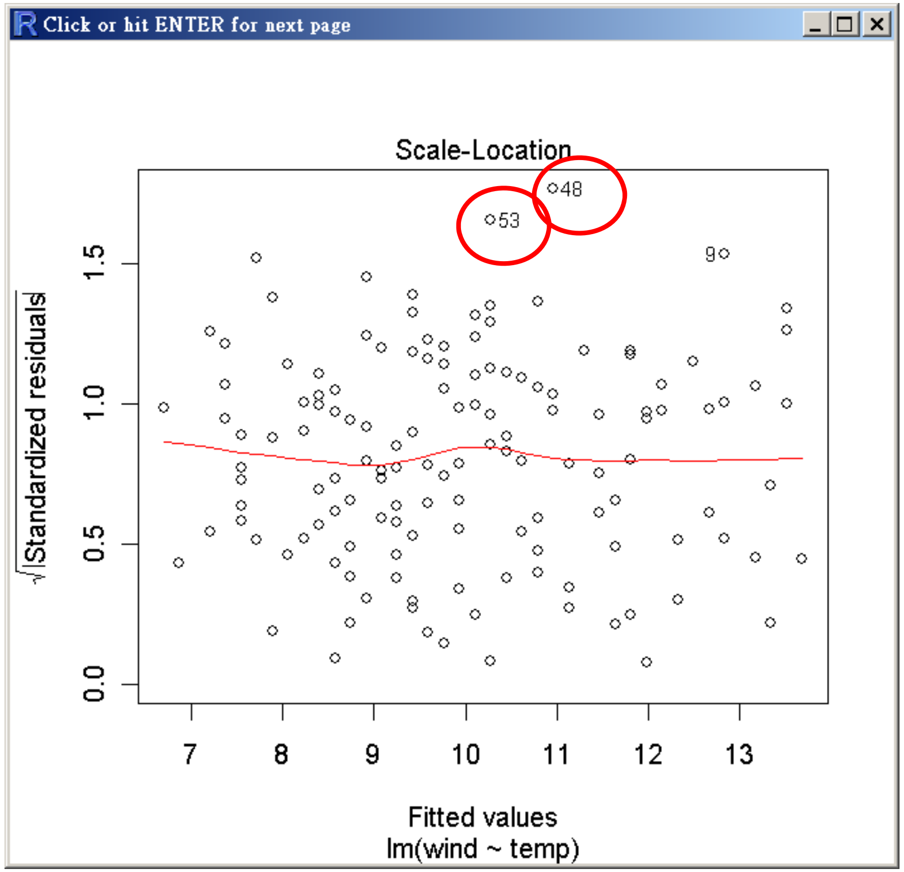



```
> qqnorm(residuals(my.model))
> qqline(residuals(my.model))
```

3. 尺度-位置圖 (A Scale-Location Plot)

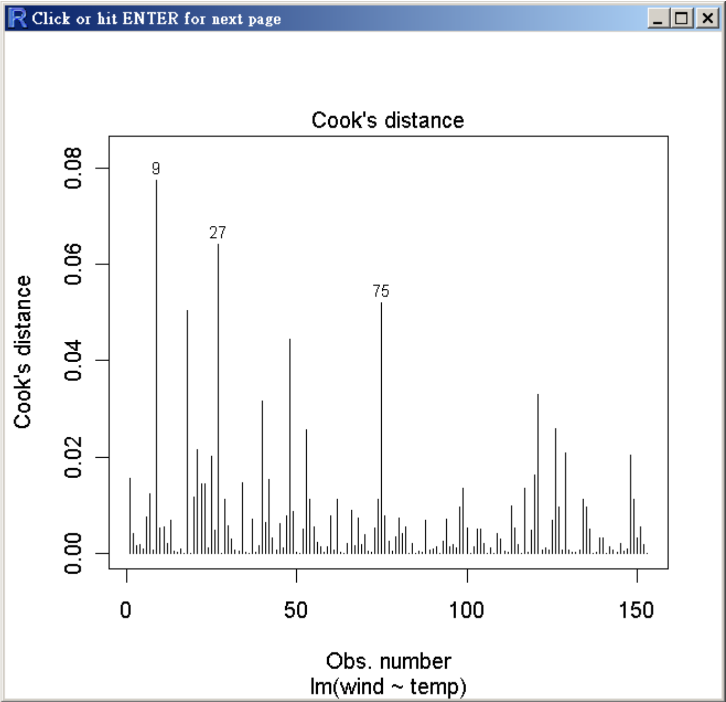
- A scale-loaction plot of sqrt(abs(residuals)) against fitted values.
- This is like a positive-valued version of the first graph; it is good for detecting non-constancy of variance (heteroscedasticity).

default



4. Plot of Cook's Distance vs Row Labels

- Cook's distance measures the effect of deleting a given observation.
- Cook's distance is a measure of the squared distance between the least square estimate based on all n points β and the estimate obtained by deleting the *i*th points $\beta_{(i)}$.
- Points with a Cook's distance of 1 or more are considered to be influential.



$$D_i = \frac{\sum_{j=1}^{n}(\hat{y}_j - \hat{y_{j(i)}})^2}{pMS_E}$$

課堂練習:

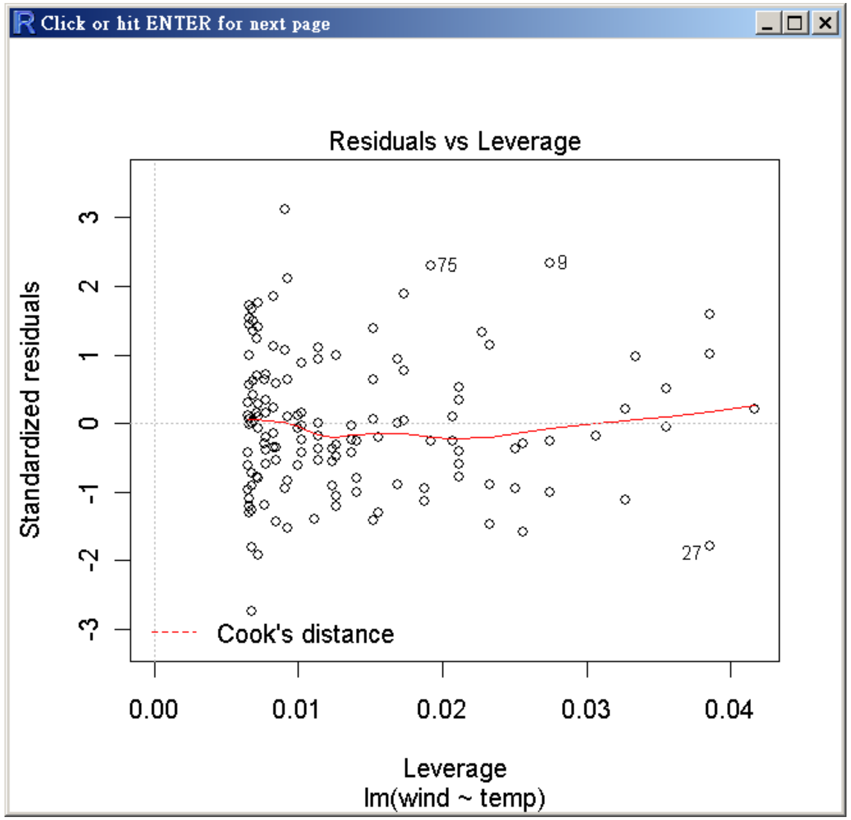
- 算出Cook's Distance。
- 畫出Cook's Distance vs. Row Labels的散佈圖。
- 標出前三大Cook's Distance值所在位置。

5. Plot of Residuals vs Leverages

選讀

- Outliers in the response variable are called outliers.
- Outliers with respect to the predictors are called leverage points.
- For the regression, it is the points that have large leverage are important.
- Points that have small leverage "do not count" in the regression – we could move them or remove them from the data and the regression line does not change very much.

default



$$\mathrm{Le}_i = \frac{(x_i - \bar{x})^2}{\sum_{j=1}^{n}(x_j - \bar{x})^2}$$

$$\hat{\beta}_1 = \sum_{i=1}^{n} \mathrm{Le}_i \frac{(y_i - \bar{y})}{(x_i - \bar{x})}$$

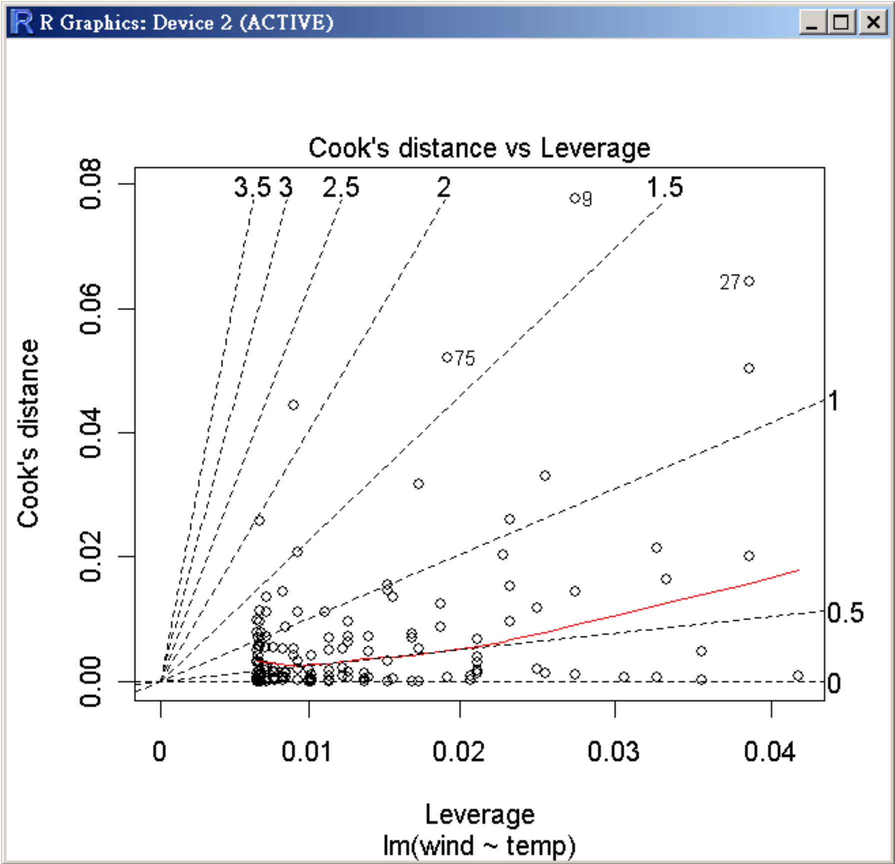
課堂練習：

- 算出Leverages 。
- 將Residuals標準化。
- 畫出Residuals標準化 vs. Leverages 的散佈圖。
- 標出前三大Leverages值所在位置。

6. Cook's Distance vs Leverage

選讀

- In the Cook's distance vs leverage/(1-leverage) plot, contours of standardized residuals that are equal in magnitude are lines through the origin.



模型選取/變數選取

Swiss Fertility and Socioeconomic Indicators (1888) Data

```
> head(swiss)
             Fertility Agriculture Examination Education Catholic Infant.Mortality
Courtelary        80.2        17.0          15        12     9.96             22.2
Delemont          83.1        45.1           6         9    84.84             22.2
Franches-Mnt      92.5        39.7           5         5    93.40             20.2
Moutier           85.8        36.5          12         7    33.77             20.3
Neuveville        76.9        43.5          17        15     5.16             20.6
Porrentruy        76.1        35.3           9         7    90.57             26.6
```

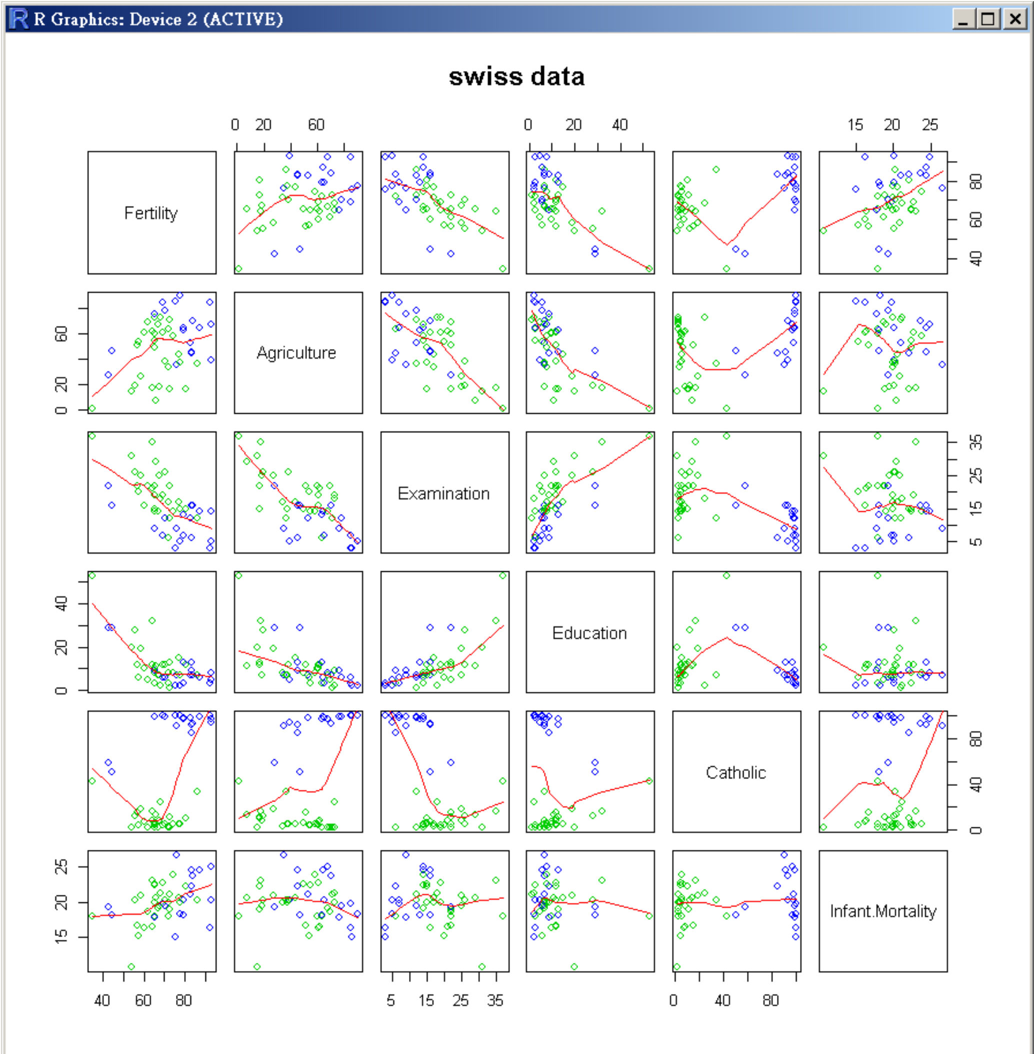
A data frame with 47 observations on 6 variables, each of which is in percent, i.e., in [0, 100].

[,1]	Fertility	Ig, 'common standardized fertility measure'
[,2]	Agriculture	% of males involved in agriculture as occupation
[,3]	Examination	% draftees receiving highest mark on army examination
[,4]	Education	% education beyond primary school for draftees.
[,5]	Catholic	% 'catholic' (as opposed to 'protestant').
[,6]	Infant.Mortality	live births who live less than 1 year.

All variables but 'Fertility' give proportions of the population.

散佈圖矩陣

```
pairs(swiss, panel  = panel.smooth, main = "swiss data", col = 3 + (swiss$Catholic > 50))
```



配適多重迴歸模型: lm

```
> summary(my_lm <- lm(Fertility ~ ., data = swiss))

Call:
lm(formula = Fertility ~ ., data = swiss)

Residuals:
     Min       1Q   Median       3Q      Max 
-15.2743  -5.2617   0.5032   4.1198  15.3213 

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)    
(Intercept)      66.91518   10.70604   6.250 1.91e-07 ***
Agriculture      -0.17211    0.07030  -2.448  0.01873 *  
Examination      -0.25801    0.25388  -1.016  0.31546    
Education        -0.87094    0.18303  -4.758 2.43e-05 ***
Catholic          0.10412    0.03526   2.953  0.00519 ** 
Infant.Mortality  1.07705    0.38172   2.822  0.00734 ** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.165 on 41 degrees of freedom
Multiple R-squared:  0.7067,	Adjusted R-squared:  0.671 
F-statistic: 19.76 on 5 and 41 DF,  p-value: 5.594e-10
```

step(): 逐步迴歸變數篩選

AIC (Akaike information criterion)常用來作為模型選取的準則。其值越小，代表模型的解釋能力越好(用的變數越少，或是誤差平方和越小)。

語法:

```
step(object, scope, scale = 0, direction = c("both", "backward", "forward"),
trace = 1, keep = NULL, steps = 1000, k = 2, ...)
```

```
> smy_lm <- step(my_lm)
Start:  AIC=190.69
Fertility ~ Agriculture + Examination + Education + Catholic +
    Infant.Mortality

                   Df Sum of Sq    RSS    AIC
- Examination       1     53.03 2158.1 189.86
<none>                          2105.0 190.69
- Agriculture       1    307.72 2412.8 195.10
- Infant.Mortality  1    408.75 2513.8 197.03
- Catholic          1    447.71 2552.8 197.75
- Education         1   1162.56 3267.6 209.36

Step:  AIC=189.86
Fertility ~ Agriculture + Education + Catholic + Infant.Mortality

                   Df Sum of Sq    RSS    AIC
<none>                          2158.1 189.86
- Agriculture       1    264.18 2422.2 193.29
- Infant.Mortality  1    409.81 2567.9 196.03
- Catholic          1    956.57 3114.6 205.10
- Education         1   2249.97 4408.0 221.43
```

$$AIC = \ln\left(\frac{ESS}{n}\right) + \frac{2p}{n}, \quad ESS = \sum_{i=1}^{n}(\hat{y}_i - y_i)^2$$

最後選取的模型

```
> summary(smy_lm)

Call:
lm(formula = Fertility ~ Agriculture + Education + Catholic +
    Infant.Mortality, data = swiss)

Residuals:
     Min       1Q   Median       3Q      Max
-14.6765  -6.0522   0.7514   3.1664  16.1422

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)      62.10131    9.60489   6.466 8.49e-08 ***
Agriculture      -0.15462    0.06819  -2.267  0.02857 *
Education        -0.98026    0.14814  -6.617 5.14e-08 ***
Catholic          0.12467    0.02889   4.315 9.50e-05 ***
Infant.Mortality  1.07844    0.38187   2.824  0.00722 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.168 on 42 degrees of freedom
Multiple R-squared:  0.6993,  Adjusted R-squared:  0.6707
F-statistic: 24.42 on 4 and 42 DF,  p-value: 1.717e-10
```

解釋變數是類別變數(categorical): using dummy variables

(Background) Johnson Filtration, Inc., provides maintenance service for water-filtration systems throughout southern Florida. Customers contact Johnson with requests for maintenance service on their water-filtration systems. To estimate the service time and the service cost, Johnson′ s managers want to predict the repair time necessary for each maintenance request.

```
> library(readxl)
> Johnson <- read_excel("data/Chap15/Johnson.xlsx")
> colnames(Johnson) <- c("MonthsSinceLastService", "RepairType", "RepairTime")
> print(Johnson)
# A tibble: 10 x 3
   MonthsSinceLastService RepairType RepairTime
                    <dbl>      <dbl>      <dbl>
 1                      2 Electrical        2.9
 2                      6 Mechanical        3
...
10                      6 Electrical        4.5
> str(Johnson)
tibble [10 x 3] (S3: tbl_df/tbl/data.frame)
 $ MonthsSinceLastService: num [1:10] 2 6 8 3 2 7 9 8 4 6
 $ RepairType            : chr [1:10] "Electrical" "Mechanical" "Electrical" "Mechanical" ...
 $ RepairTime            : num [1:10] 2.9 3 4.8 1.8 2.9 4.9 4.2 4.8 4.4 4.5
```

Service Call	Months Since Last Service	Type of Repair	Repair Time in Hours
1	2	Electrical	2.9
2	6	Mechanical	3.0
3	8	Electrical	4.8
4	3	Mechanical	1.8
5	2	Electrical	2.9
6	7	Electrical	4.9
7	9	Mechanical	4.2
8	8	Mechanical	4.8
9	4	Electrical	4.4
10	6	Electrical	4.5

Regression Model

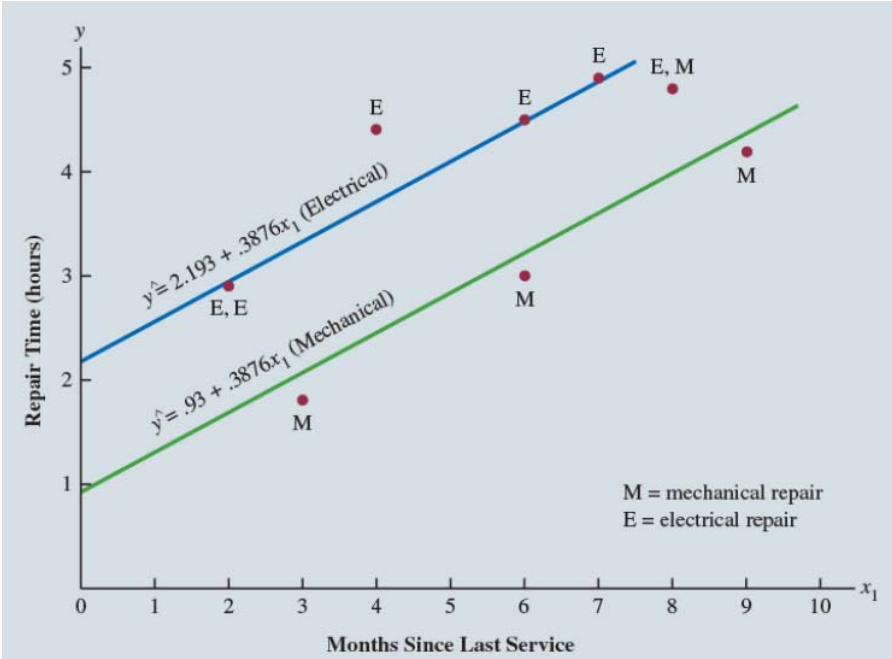
$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon \qquad x_2 = \begin{cases} \underline{\quad 0 \quad}, & \text{if the type of repair is mechanical} \\ \underline{\quad 1 \quad}, & \text{if the type of repair is electrical} \end{cases}$$

The multiple regression equation $E(y) = \beta_0 + \beta_1 x_1 + \beta_2 x_2$

$$E(y|\text{mechanical}) = \underline{\beta_0 + \beta_1 x_1 + \beta_2(0)} = \underline{\beta_0 + \beta_1 x_1}$$

$$E(y|\text{electrical}) = \underline{\beta_0 + \beta_1 x_1 + \beta_2(1)} = \underline{(\beta_0 + \beta_2) + \beta_1 x_1}$$

with $\beta_2 = 1.263$, we learn that, on average, electrical repairs require <u>1.263 hours longer</u> than mechanical repairs.



MLR in R using Categorical Variables

```
> Johnson$RepairType <- factor(Johnson$RepairType, levels = c("Mechanical", "Electrical"))
> Johnson_lm <- lm(RepairTime ~ MonthsSinceLastService + RepairType, data = Johnson)
> summary(Johnson_lm)
Call:
lm(formula = RepairTime ~ MonthsSinceLastService + RepairType,
    data = Johnson)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.49412 -0.24690 -0.06842 -0.00960  0.76858 

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)    
(Intercept)             0.93050    0.46697   1.993 0.086558 .  
MonthsSinceLastService  0.38762    0.06257   6.195 0.000447 ***
RepairType1             1.26269    0.31413   4.020 0.005062 ** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.459 on 7 degrees of freedom
Multiple R-squared:  0.8592,  Adjusted R-squared:  0.819 
F-statistic: 21.36 on 2 and 7 DF,  p-value: 0.001048
```

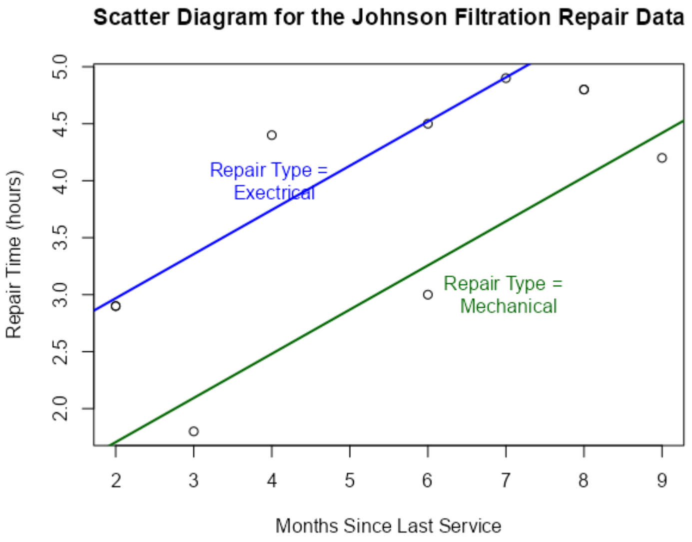
$$x_2 = \begin{cases} \underline{\quad 0 \quad}, & \text{mechanical} \\ \underline{\quad 1 \quad}, & \text{electrical} \end{cases}$$

```
> Johnson$RepairType
[1] Electrical Mechanical Electrical Mechanical
[5] Electrical Electrical Mechanical Mechanical
[9] Electrical Electrical
Levels: Mechanical Electrical
```

```
> anova(Johnson_lm)
Analysis of Variance Table
Response: RepairTime
                       Df Sum Sq Mean Sq F value   Pr(>F)   
MonthsSinceLastService  1 5.5960  5.5960  26.556 0.001319 **
RepairType              1 3.4049  3.4049  16.158 0.005062 **
Residuals               7 1.4751  0.2107                    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

MLR in R using Categorical Variables

```
attach(Johnson)
plot(MonthsSinceLastService, RepairTime,
     xlab = "Months Since Last Service",
     ylab = "Repair Time (hours)",
     main = "Scatter Diagram for the Johnson Filtration Repair Data")
abline(a = Johnson.lm$coefficients[1], b = Johnson.lm$coefficients[2], col = "darkgreen",
  lwd = 2)
text(7, 3, "Repair Type = \n Mechanical", col = "darkgreen")
abline(a = sum(Johnson.lm$coefficients[c(1, 3)]), b = Johnson.lm$coefficients[2],
  col = "blue",lwd = 2)
text(4, 4, "Repair Type = \n Exectrical", col = "blue")
```



$$x_1 = \begin{cases} 1, Taipei \\ 0, otherwise \end{cases} \quad x_2 = \begin{cases} 1, Tokyo \\ 0, otherwise \end{cases}$$

```
> cities <- sample(c("Taipei", "Seoul", "Tokyo"),
+    20, replace = T)
> factor(cities)
...
Levels: Seoul Taipei Tokyo
```

$$x_1 = \begin{cases} 1, Tokyo \\ 0, otherwise \end{cases} \quad x_2 = \begin{cases} 1, Seoul \\ 0, otherwise \end{cases}$$

```
> factor(cities, levels = c("Taipei", "Tokyo", "Seoul"))
...
Levels: Taipei Tokyo Seoul
```

羅吉斯迴歸

Dependent Variable is Binary: Logistic Regression

Regression Model $Y_i = E\{Y_i\} + \varepsilon_i$

$$Y_i = \begin{cases} 1, Event \\ 0, otherwise \end{cases}$$

Simple Logistic Regression Equation

$$E\{Y_i\} = \pi_i = \frac{\exp(\beta_0 + \beta_1 X_i)}{1 + \exp(\beta_0 + \beta_1 X_i)}$$

A formal statement of the simple logistic regression model: recall that when the response variable is binary, taking on the values 1 and 0 with probabilities π and $1 - \pi$, respectively, Y is a Bernoulli random variable with parameter $E\{Y\} = \pi$.

The fitted logistic response function

$$\hat{\pi} = \frac{\exp(b_0 + b_1 X)}{1 + \exp(b_0 + b_1 X)} \Rightarrow \ln\left(\frac{\hat{\pi}}{1 - \hat{\pi}}\right) = b_0 + b_1 X$$

Testing for Individual/Overall Significance

Multiple Logistic Regression Equation

$$E(y) = P(y = 1|x_1, x_2, \cdots, x_p)$$

$$= \frac{e^{\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_p x_p}}{1+e^{\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_p x_p}}$$

$H_0 : \beta_i = 0$ $\quad$ $H_a : \beta_i \neq 0$ $\quad$ χ^2 test

H_0 : $\underline{\beta_1 = \beta_2 = 0}$

H_a : One or both of the parameters is not equal to zero $\quad$ χ^2 test

Interpreting the Logistic Regression Equation

- The **odds (勝算)** in favor of an event occurring is defined as the probability the event will occur divided by the probability the event will not occur . In logistic regression the event of interest is always y = 1.

$$odds = \frac{P(y=1|x_1, x_2, \cdots, x_p)}{P(y=0|x_1, x_2, \cdots, x_p)} = \frac{P(y=1|x_1, x_2, \cdots, x_p)}{1 - P(y=1|x_1, x_2, \cdots, x_p)}$$

The odds ratio is the odds that $y = 1$ given that one of the independent variables has been increased by one unit (odd_{s1}) divided by the odds that $y = 1$ given no change in the values for the independent variables (odd_{s0}).

$$\text{Odds Ratio} = \frac{odd_{s1}}{odd_{s0}} \qquad \text{Odds ratio} = e^{\beta_i}$$

- The odds ratio measures the impact on the odds of a one-unit increase in only one of the independent variables.

Dependent Variable is Binary
範例: Logistic Regression

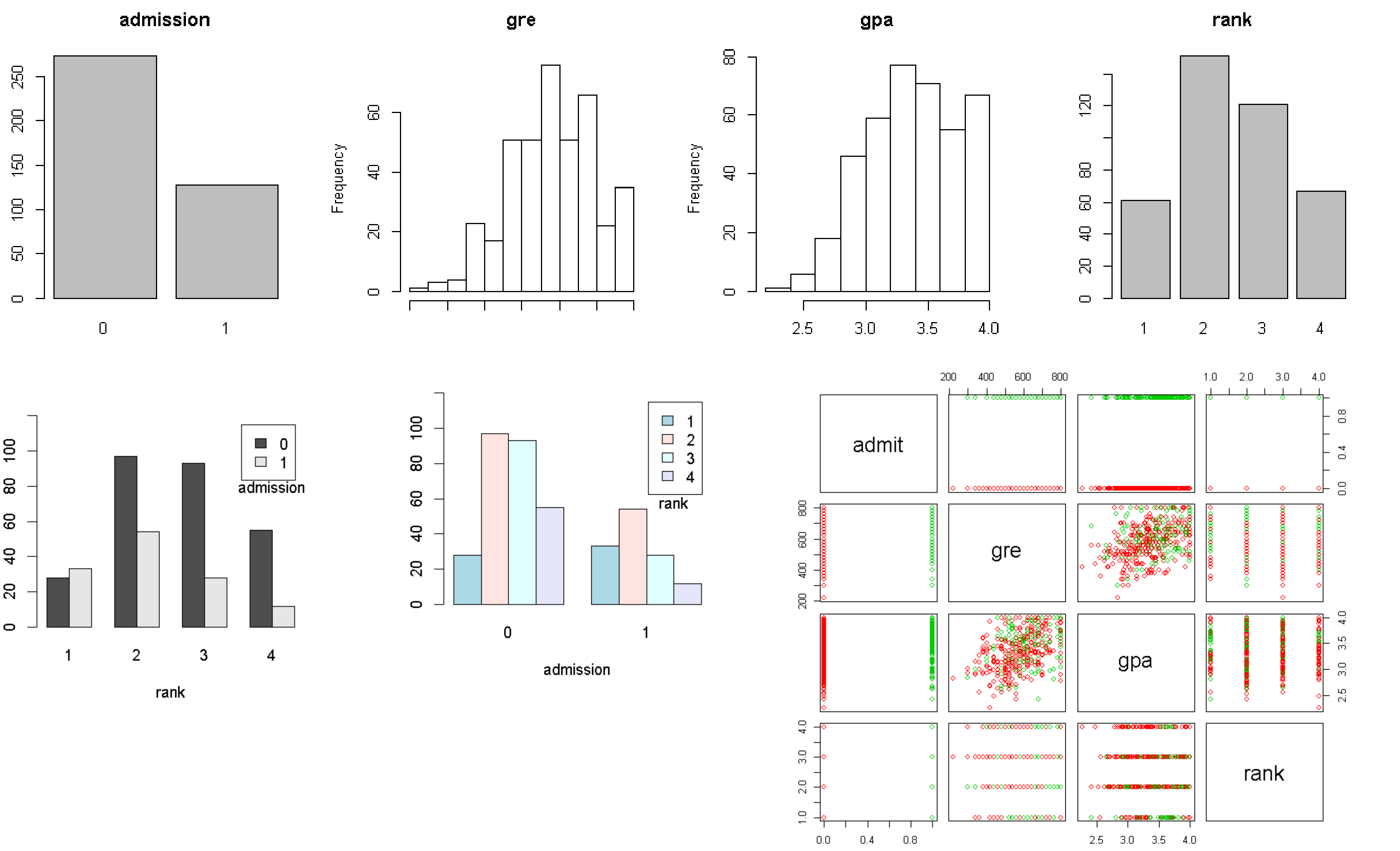
- A researcher is interested in how variables, such as **GRE** (Graduate Record Exam scores), **GPA** (grade point average) and prestige of the undergraduate institution (**rank**=1,2,3,4, 1 =highest prestige, 4 = the lowest), effect **admission** into graduate school.
- The response variable, admit/don't admit, is a binary variable.

```
> # mydata <- read.csv("http://www.ats.ucla.edu/stat/data/binary.csv")
> mydata <- read.csv("binary.csv")
> dim(mydata)
[1] 400   4
> head(mydata)
  admit gre  gpa rank
1     0 380 3.61    3
2     1 660 3.67    3
3     1 800 4.00    1
4     1 640 3.19    4
5     0 520 2.93    4
6     1 760 3.00    2
> summary(mydata)
     admit              gre             gpa              rank
 Min.   :0.0000   Min.   :220.0   Min.   :2.260   Min.   :1.000
 1st Qu.:0.0000   1st Qu.:520.0   1st Qu.:3.130   1st Qu.:2.000
 Median :0.0000   Median :580.0   Median :3.395   Median :2.000
 Mean   :0.3175   Mean   :587.7   Mean   :3.390   Mean   :2.485
 3rd Qu.:1.0000   3rd Qu.:660.0   3rd Qu.:3.670   3rd Qu.:3.000
 Max.   :1.0000   Max.   :800.0   Max.   :4.000   Max.   :4.000
> sapply(mydata, sd)
      admit         gre         gpa        rank
  0.4660867 115.5165364   0.3805668   0.9444602
```

```
> xtabs(~ admit + rank, data = mydata)
     rank
admit  1  2  3  4
    0 28 97 93 55
    1 33 54 28 12
> mydata$rank <- factor(mydata$rank)
 [1] 3 3 1 4 4 2 1 2 3 2 4 1 1 2 1 3 4 3 2 1
...
[385] 2 1 2 2 2 2 2 2 3 2 3 2 3 2 2 3
Levels: 1 2 3 4
```

UCLA Statistical Consulting Group. http://www.ats.ucla.edu/stat/r/dae/logit.htm

Statistical Graphics



Modeling and Summary of Fit

```
> mylogit <- glm(admit ~ gre + gpa + rank, data = mydata, family = "binomial")
> summary(mylogit)
```

```
family = binomial(link = "logit")
```

```
Call:
glm(formula = admit ~ gre + gpa + rank, family = "binomial",
    data = mydata)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6268  -0.8662  -0.6388   1.1490   2.0790

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -3.989979   1.139951  -3.500 0.000465 ***
gre          0.002264   0.001094   2.070 0.038465 *
gpa          0.804038   0.331819   2.423 0.015388 *
rank2       -0.675443   0.316490  -2.134 0.032829 *
rank3       -1.340204   0.345306  -3.881 0.000104 ***
rank4       -1.551464   0.417832  -3.713 0.000205 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 499.98  on 399  degrees of freedom
Residual deviance: 458.52  on 394  degrees of freedom
AIC: 470.52

Number of Fisher Scoring iterations: 4
```

- For every one unit change in **gre**, the log odds of admission (versus non-admission) increases by 0.002.
- For a one unit increase in **gpa**, the log odds of being admitted to graduate school increases by 0.804.

Having attended an undergraduate institution with rank of 2, versus an institution with a rank of 1, changes the log odds of admission by -0.675.

UCLA Statistical Consulting Group. http://www.ats.ucla.edu/stat/r/dae/logit.htm

Wald Test for Model Coefficients

```
wald.test(Sigma, b, Terms = NULL, L = NULL, H0 = NULL, df = NULL, verbose = FALSE)
```

Sigma: the variance covariance matrix of the error terms, **b**: the coefficients, **Terms**: terms in the model are to be tested, in this case, terms 4, 5, and 6, are the three terms for the levels of rank.

```
> library(aod) #aod: Analysis of Overdispersed Data
> wald.test(b = coef(mylogit), Sigma = vcov(mylogit), Terms = 4:6)
Wald test:
----------

Chi-squared test:
X2 = 20.9, df = 3, P(> X2) = 0.00011
```

To contrast two terms, we multiply one of them by 1, and the other by -1. The other terms in the model are not involved in the test, so they are multiplied by 0.
Test the difference (subtraction) of the terms for **rank = 2** and **rank = 3** (i.e., the 4th and 5th terms in the model). **L = l**: base the test on the vector **l** (rather than using the Terms option).

```
> l <- cbind(0,0,0,1,-1,0)
> wald.test(b = coef(mylogit), Sigma = vcov(mylogit), L = l)
Wald test:
----------

Chi-squared test:
X2 = 5.5, df = 1, P(> X2) = 0.019
```

The difference between the coefficient for **rank = 2** and the coefficient for **rank = 3** is statistically significant.

UCLA Statistical Consulting Group. http://www.ats.ucla.edu/stat/r/dae/logit.htm

Interpret Coefficients as Odds-ratios

```
> exp(cbind(OR = coef(mylogit), confint(mylogit)))
Waiting for profiling to be done...
                   OR       2.5 %    97.5 %
(Intercept) 0.0185001 0.001889165 0.1665354
gre         1.0022670 1.000137602 1.0044457
gpa         2.2345448 1.173858216 4.3238349
rank2       0.5089310 0.272289674 0.9448343
rank3       0.2617923 0.131641717 0.5115181
rank4       0.2119375 0.090715546 0.4706961
```

- For a one unit increase in **gpa**, the odds of being admitted to graduate school (versus not being admitted) increase by a factor of 2.23.
- For more information on interpreting odds ratios see our FAQ page How do I interpret odds ratios in logistic regression? http://www.ats.ucla.edu/stat/mult_pkg/faq/general/odds_ratio.htm
- Note that while R produces it, the odds ratio for the intercept is not generally interpreted.

http://www.ats.ucla.edu/stat/r/dae/logit.htm

Predicted Probabilities

- Predicted probabilities can be computed for both categorical and continuous predictor variables.
- Want to calculate the predicted probability of admission at each value of **rank**, holding gre and **gpa** at their means.

```
> newdata1 <- with(mydata, data.frame(gre = mean(gre), gpa = mean(gpa), rank = factor(1:4)))
> newdata1
    gre    gpa rank
1 587.7 3.3899    1
2 587.7 3.3899    2
3 587.7 3.3899    3
4 587.7 3.3899    4
> newdata1$rankP <- predict(mylogit, newdata = newdata1, type = "response")
> newdata1
    gre    gpa rank     rankP
1 587.7 3.3899    1 0.5166016
2 587.7 3.3899    2 0.3522846
3 587.7 3.3899    3 0.2186120
4 587.7 3.3899    4 0.1846684
```

type = "link"

- The predicted probability of being accepted into a graduate program is 0.52 for students from the highest prestige undergraduate institutions (**rank = 1**), and 0.18 for students from the lowest ranked institutions (**rank = 4**), holding **gre** and **gpa** at their means.

http://www.ats.ucla.edu/stat/r/dae/logit.htm

Create a Table of Predicted Probabilities

```
> newdata2 <- with(mydata,
+   data.frame(gre = rep(seq(from = 200, to = 800, length.out = 100), 4),
+   gpa = mean(gpa), rank = factor(rep(1:4, each = 100))))
> dim(newdata2)
[1] 400   3
> head(newdata2)
       gre    gpa rank
1 200.0000 3.3899    1
2 206.0606 3.3899    1
3 212.1212 3.3899    1
4 218.1818 3.3899    1
5 224.2424 3.3899    1
6 230.3030 3.3899    1
>
> newdata3 <- cbind(newdata2, predict(mylogit, newdata = newdata2, type="link", se=TRUE))
> newdata3 <- within(newdata3, {
+   PredictedProb <- plogis(fit)
+   LL <- plogis(fit - (1.96 * se.fit))
+   UL <- plogis(fit + (1.96 * se.fit))
+ })
> head(newdata3)
       gre    gpa rank        fit    se.fit residual.scale        UL        LL PredictedProb
1 200.0000 3.3899    1 -0.8114870 0.5147714              1 0.5492064 0.1393812     0.3075737
2 206.0606 3.3899    1 -0.7977632 0.5090986              1 0.5498513 0.1423880     0.3105042
3 212.1212 3.3899    1 -0.7840394 0.5034491              1 0.5505074 0.1454429     0.3134499
4 218.1818 3.3899    1 -0.7703156 0.4978239              1 0.5511750 0.1485460     0.3164108
5 224.2424 3.3899    1 -0.7565919 0.4922237              1 0.5518545 0.1516973     0.3193867
6 230.3030 3.3899    1 -0.7428681 0.4866494              1 0.5525464 0.1548966     0.3223773
```

Create 100 values of **gre** between 200 and 800, at each value of **rank** (i.e., 1, 2, 3, and 4) and plot.

http://www.ats.ucla.edu/stat/r/dae/logit.htm

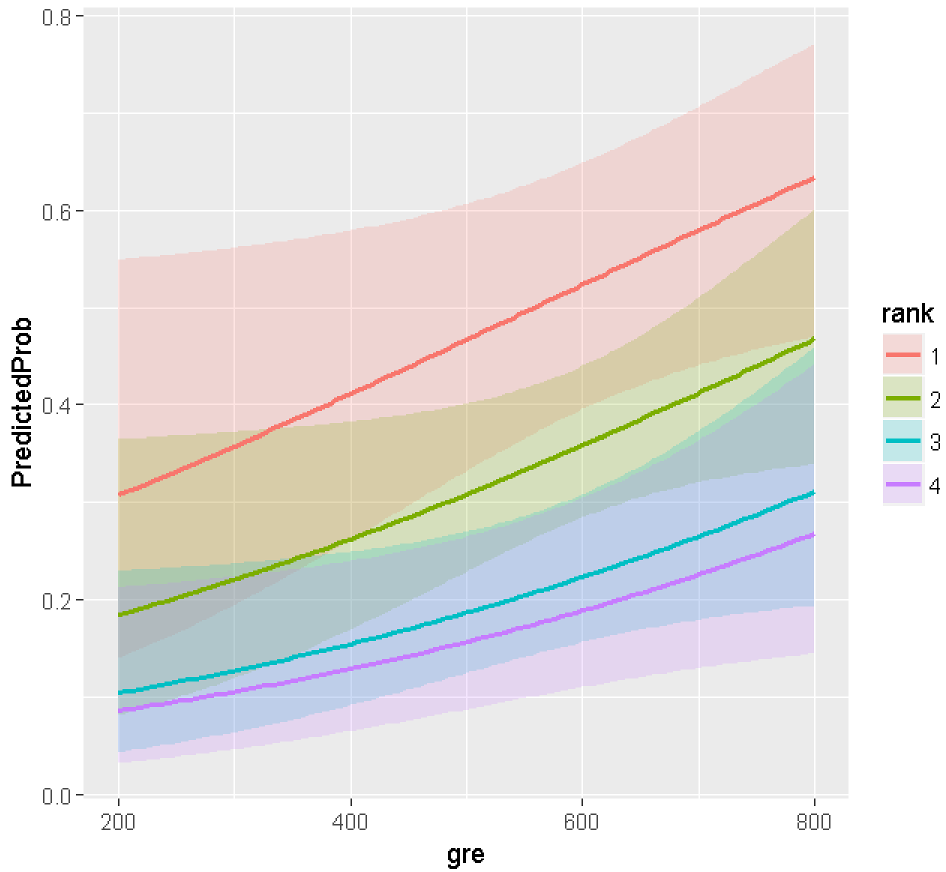
Plot the Predicted Probabilities

選讀

```
> library(ggplot2)
> ggplot(newdata3, aes(x = gre, y = PredictedProb)) +
+   geom_ribbon(aes(ymin = LL, ymax = UL, fill = rank), alpha = 0.2) +
+   geom_line(aes(colour = rank), size=1)
```

Plot the predicted probabilities and 95% confidence intervals to understand and/or present the model.

http://www.ats.ucla.edu/stat/r/dae/logit.htm



If

```
Error in .Call.graphics(C_palette2, .Call(C_palette2, NULL)) :
  invalid graphics state
```

then use `def.off()`

Measure the Model Fits

選讀

```
> # the difference in deviance for the two models (i.e., the test statistic)
> with(mylogit, null.deviance - deviance)
[1] 41.45903
> with(mylogit, df.null - df.residual)
[1] 5
> #the p-value
> with(mylogit, pchisq(null.deviance - deviance, df.null - df.residual, lower.tail = FALSE))
[1] 7.578194e-08
> # the model's log likelihood
> logLik(mylogit)
'log Lik.' -229.2587 (df=6)
```

- One measure of model fit is the significance of the overall model: whether the model with predictors fits significantly better than a model with just an intercept (i.e., a null model).
- The test statistic is the difference between the residual deviance for the model with predictors and the null model.
- The chi-square of 41.46 with 5 degrees of freedom and an associated p-value of less than 0.001 tells us that our model as a whole fits significantly better than an empty model.

http://www.ats.ucla.edu/stat/r/dae/logit.htm

Analysis of Deviance Table

```
> # Testing for Significance: individuals
> anova(mylogit, test = "Chisq")
Analysis of Deviance Table

Model: binomial, link: logit

Response: admit

Terms added sequentially (first to last)

     Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                   399     499.98
gre   1  13.9204       398     486.06 0.0001907 ***
gpa   1   5.7122       397     480.34 0.0168478 *
rank  3  21.8265       394     458.52 7.088e-05 ***
---
Signif. codes:  0  '***'  0.001  '**'  0.01  '*'  0.05  '.'  0.1  ' '  1
```

```
> library(car) # Companion to Applied Regression
> Anova(mylogit)
```

Use **anova** function to give an analysis of deviance table, or the **drop1** function to try dropping each factor.

```
> drop1(mylogit, test = "Chisq")
Single term deletions

Model:
admit ~ gre + gpa + rank
       Df Deviance    AIC     LRT  Pr(>Chi)
<none>      458.52 470.52
gre     1   462.88 472.88  4.3578   0.03684 *
gpa     1   464.53 474.53  6.0143   0.01419 *
rank    3   480.34 486.34 21.8265 7.088e-05 ***
---
Signif. codes:  0  '***'  0.001  '**'  0.01  '*'  0.05  '.'  0.1  ' '  1
```

Things to Consider

選讀

- **Empty cells or small cells**: check the crosstab between categorical predictors and the outcome variable. If a cell has very few cases (a small cell), the model may become unstable or it might not run at all.

- **Separation or quasi-separation** (also called perfect prediction), a condition in which the outcome does not vary at some levels of the independent variables. See
http://www.ats.ucla.edu/stat/mult_pkg/faq/general/complete_separation_logit_models.htm

- **Sample size**: Both logit and probit models require more cases than OLS regression because they use maximum likelihood estimation techniques.

- **Pseudo-R-squared**: none of psuedo-R-squared measures can be interpreted exactly as R-squared in OLS regression is interpreted. See
http://www.ats.ucla.edu/stat/mult_pkg/faq/general/Psuedo_RSquareds.htm

- **Diagnostics**: The diagnostics for logistic regression are different from those for OLS regression. See Hosmer and Lemeshow (2000, Chapter 5).

http://www.ats.ucla.edu/stat/r/dae/logit.htm

共線性 (Collinearity)

- **What is the multicollinearity (collinearity)**
  - it is a statistical phenomenon in which two or more predictor variables in a multiple regression model are highly correlated.
  - one predictor can be linearly predicted from the others with a non-trivial degree of accuracy.
- **How problematic is multicollinearity?**
  - Moderate multicollinearity may not be problematic.
  - Severe multicollinearity can increase the variance of the coefficient estimates and make the estimates very sensitive to minor changes in the model:
    - the coefficient estimates are unstable (may be to switch signs) and difficult to interpret, or
    - parameter estimates may include substantial amounts of uncertainty,
    - forward or backward selection of variables could produce inconsistent results,
    - variance partitioning analyses may be unable to identify unique sources of variation.
    - the precision of fitted values within the range of the observations on the predictor variables is not eroded with the addition of correlated predictor variables into the regression model.
- The centered and scaled variables reduce the correlations between the first power and second power terms markedly.

變異數膨脹因子
The Variance Inflation Factors

$$X_j = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_k X_k + \varepsilon.$$

$$VIF_j = \frac{1}{1 - R_j^2}$$

- A VIF for a single explanatory variable is obtained using the R-squared value of the regression of that variable X_j against all other explanatory variables.

- A VIF measures how much the variance of the estimated regression coefficients are inflated as compared to when the predictor variables are not linearly related.

VIF	Status of predictors
VIF = 1	Not correlated
1 < VIF < 5	Moderately correlated
VIF > 5 to 10	Highly correlated

`vif` in R

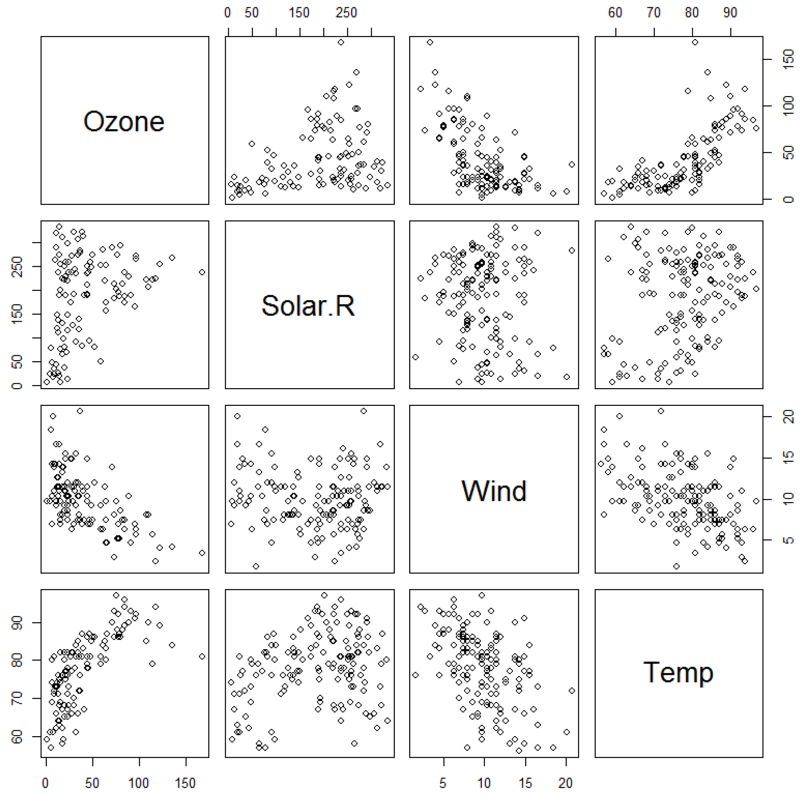
- R packages:
  `vif{faraway}, vif{HH}, vif{car}, VIF{fmsb}, vif{VIF}`
  - **`faraway`**: Functions and Datasets for Books by Julian Faraway
  - **`HH`**: Statistical Analysis and Data Display: Heiberger and Holland
  - **`car`**: Companion to Applied Regression
  - **`fmsb`**: Functions for Medical Statistics Book with some Demographic Data
  - **`VIF`**: A Fast Regression Algorithm For Large Data

```
> head(airquality)
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6
>
> model0 <- lm(Ozone ~ Wind + Temp + Solar.R, data = airquality)
```

An Example

```
> cor(airquality[,1:4], use = "pairwise")
             Ozone     Solar.R        Wind       Temp
Ozone    1.0000000  0.34834169 -0.60154653  0.6983603
Solar.R  0.3483417  1.00000000 -0.05679167  0.2758403
Wind    -0.6015465 -0.05679167  1.00000000 -0.4579879
Temp     0.6983603  0.27584027 -0.45798788  1.0000000
> pairs(airquality[,1:4])
```

```
> library(car)
> vif(model0)
    Wind     Temp  Solar.R
1.329070 1.431367 1.095253
```



The Stepwise VIF Selection

```
> summary(model0)
Call:
lm(formula = Ozone ~ Wind + Temp + Solar.R, data = airquality)

Residuals:
    Min      1Q  Median      3Q     Max 
-40.485 -14.219  -3.551  10.097  95.619 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -64.34208   23.05472  -2.791  0.00623 ** 
Wind         -3.33359    0.65441  -5.094 1.52e-06 ***
Temp          1.65209    0.25353   6.516 2.42e-09 ***
Solar.R       0.05982    0.02319   2.580  0.01124 *  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 21.18 on 107 degrees of freedom
  (42 observations deleted due to missingness)
Multiple R-squared:  0.6059,    Adjusted R-squared:  0.5948 
F-statistic: 54.83 on 3 and 107 DF,  p-value: < 2.2e-16
```

```
> library(fmsb)
> model1 <- lm(Wind ~ Temp + Solar.R, data = airquality)
> model2 <- lm(Temp ~ Wind + Solar.R, data = airquality)
> model3 <- lm(Solar.R ~ Wind + Temp, data = airquality)
> # checking multicolinearity for independent variables.
> VIF(model0)
[1] 2.537392
> sapply(list(model1, model2, model3), VIF)
[1] 1.267492 1.367450 1.089300
```