

# R commands(11.09.13)

(Jah Tsai since 2011.01.21 jah\_tsai@hotmail.com)

- ➡ 參考來源：R Help & R 軟體 應用統計方法(修訂版) 陳景祥  
統計計算與模擬 余清祥(NCCU Stat.) & 多變量分析 洪英超(NCCU Stat.)
- ➡ 使用時機：已對 R 指令有基本了解，但對於不常用的指令還未完全熟記，或遇到問題時想知道 R 有哪些指令可用，所以整理此表以供簡單查詢，詳細用法請見參考資料。

指令	用法簡介
➡ 基本操作	
source("路徑")	載入外部程式
sink("路徑")+sink()	紀錄程式執行結果
demo(套件)	展示套件的示範功能
example(函數)	執行範例程式
library(套件)	載入套件
require(套件)	在自訂函數中使用以載入套件
detach(package:套件)	卸載以載入的套件
install.packages(套件)	安裝套件
data(資料檔)	載入資料檔
head(資料檔,n);tail(資料檔,n)	查看前(後)n 列資料
attach(資料檔)	使用後可直接使用變數名稱操作
with(資料檔,{程式...})	與 attach 同，但效力限制在{}範圍
update.packages(checkBuilt=T,ask=F)	安裝新版 R 後，用以更新套件
getwd()	查詢目前工作目錄
setwd("路徑")	改變工作目錄
objects()	列出目前用過的所有變數，指定物件則列出可呼叫名稱
rm(物件)	清除物件，(list=ls())清除所有
.First=function{}	設定啟動 R 前所執行的程式
.Last=function{}	設定結束 R 前所執行的程式
● 基本運算	
%%	整除
%%	餘數計算
&	可用於向量的 and
	可用於向量的 or
seq( , ,by)	產生以 by 為遞增值的數列
length()	計算元素個數
sum()	元素相加
cumsum()	累積相加
prod()	元素相乘
cumprod()	累積相乘

<b>➡ 變數與資料</b>	
<b>● 向量變數</b>	
<code>x[-i]</code>	刪除第 i 個元素，其後遞補
<code>matrix(x,r,c,byrow)</code>	將 x 轉成 rxc 矩陣，(byrow=T)以列排序
<code>names()</code>	查詢或建立向量的元素名稱
<b>● 陣列變數</b>	
<code>array(x,dim=c())</code>	依照 dim 維度建立 array 變數
<code>dimnames()[[k]]</code>	第 k 維度的名稱
<code>aperm(A,perm=c())</code>	依照 perm 順序轉換 A 矩陣
<b>● 矩陣變數</b>	
<code>t()</code>	轉置矩陣
<code>diag()</code>	對角線函數
<code>det()</code>	矩陣行列式值
<code>solve()</code>	反矩陣
<code>solve(A,b)</code>	計算 $Ax=b$ 的 x 解
<code>eigen()</code>	計算矩陣特徵值和特徵向量
<code>outer(x,y,"*")</code>	將 x 和 y 依照 "*" 的方法運算
<code>chol(A)</code>	將正定對稱方陣 A 執行 Cholesky 上三角分解
<code>qr(A);qr.Q(qr(A));qr.R(qr(A))</code>	執行 QR 分解
<code>svd(A)</code>	執行 SVD 分解
<b>● 因子變數</b>	
<code>factor(x,levels=c())</code>	依照 levels 排列建立 factor 變數
<code>levels()</code>	查詢或設定分類資料
<code>ordered(x,levels=c())</code>	依照 levels 由小到大順序建立有序 factor 變數
<b>● 串列變數</b>	
<code>list(name1=value1,...)</code>	建立 list 變數
<code>unlist()</code>	將 list 變數轉成 table 變數
<code>attributes()</code>	查看個元素的名稱
<b>● 資料框架變數</b>	
<code>data.frame()</code>	建立資料框架變數
<b>● 時間數列變數</b>	
<code>ts(x,start,end,frequency)</code>	建立時間數列變數
<b>● 與變數有關的工具函數</b>	
<code>attr(物件,"屬性名稱")</code>	查詢個別的屬性內容
<code>class(物件);str()</code>	查詢物件的類別;結構
<code>abbreviate(,minlength=n)</code>	將變數名稱縮寫，最小長度為 n
<code>which(,arr.ind)</code>	找出符合條件的元素，(arr.ind=T)則傳回指標
<code>which.max();which.min()</code>	找出向量第一個找到的最大值;最小值
<code>car::which.names()</code>	找出符合指定元素名稱的指標
<code>match(x1,x2,nomatch=T)</code>	從 x1 中找出 x2 的值

● 資料輸入與輸出	
<code>file.choose()</code>	以視窗選取來替代路徑，搭配輸入函數
<code>list.files(,full.name=T)</code>	將路徑底下所有檔案和目錄存成文字向量
<code>scan(file,what,sep,skip,nlines)</code>	資料輸入
<code>write()</code>	資料輸出(輸出矩陣前先轉置)
<code>read.table(header,na.strings,skip)</code>	輸入多變數資料檔
<code>read.csv()</code>	輸入逗號分隔的 CSV 格式檔案
<code>read.csv2()</code>	輸入分號分隔的 CSV 格式檔案
<code>read.delim()</code>	輸入 TAB 鍵分隔的檔案
<code>read.fwf()</code>	輸入固定寬度格式檔案
<code>write.table()</code>	輸出多變數資料檔
<code>write.csv()</code>	輸出逗號分隔的 CSV 格式檔案
<code>write.csv2()</code>	輸出分號分隔的 CSV 格式檔案
<code>save()</code>	將部分物件儲存為 Rdata 格式
<code>save.image()</code>	將所有物件儲存為 Rdata 格式
<code>load()</code>	載入 Rdata 檔內的所有物件
● 存取其他軟體的資料檔	
<code>gdata::read.xls</code>	載入 xls 檔
<code>xlsx::read.xlsx</code>	載入 xlsx 檔
<code>xlsReadWrite::write.xls</code>	輸出 xls 檔
<code>xlsx::write.xlsx</code>	輸出 xlsx 檔
<code>foreign::write.foreign()</code>	輸出資料給 SAS、SPSS、Stata
<code>foreign::data.restore()</code>	讀入 S-Plus 以 data.dump 輸出的檔案
<code>foreign::read.S()</code>	讀入 S-Plus 二進為資料檔
<code>foreign::read.xport()</code>	讀入 SAS Transport 資料檔
<code>foreign::read.ssd()</code>	讀入 SAS .ssd 與 .sas7bdat 資料檔
<code>foreign::read.spss()</code>	讀入 SPSS 以 save 或 export 輸出的資料檔
<code>foreign::read.mtp()</code>	讀入 Minitab Portable 格式檔(.mtp)
● 資料轉換	
<code>%in%</code>	判斷是否在某個集合內
<code>cut(,break)</code>	根據 break 的切割點將資料轉換為 factor 變數
<code>car::recode(,recodes)</code>	以 recodes 規則將資料重新編排(lo 代表 min、hi 代表 max)
● 資料排序	
<code>sort(,decreasing=F)</code>	將向量元素依照大小順序排列
<code>rank()</code>	傳回向量元素的排序等級
<code>order(,dec=F)</code>	傳回排序後元素在原向量的指標
<code>rev()</code>	將整個向量倒過來排
● 資料切割	
<code>split(x,f)</code>	將 x 元素各對應的分類變數 f 分割資料
<code>subset(,subset,select)</code>	以 subset 的邏輯判斷和 select 之選擇來篩選變數

● 資料合併	
<code>union()</code>	聯集兩個變數
<code>rbind()</code>	逐列合併
<code>cbind()</code>	逐行合併
<code>merge()</code>	合併擁有一列以上相同的兩資料框架
● 資料轉換	
<code>as.</code>	改變變數或元素的型態
<code>stack()</code> 、 <code>unstack()</code>	將 <code>data.frame</code> 各直行推疊在一起( <code>unstack</code> 作還原動作)
<code>reshape()</code>	做長型資料和寬型資料之間的轉換
<code>melt()</code>	將寬型資料轉換為長型資料
<code>cast()</code>	將 <code>melt</code> 處理後的變數彙整計算
<code>edit()</code>	以試算表介面編輯變數，僅供下次使用
<code>fix()</code>	以試算表介面編輯變數，改變後固定
<code>View()</code>	以試算表介面查看變數
<code>view(x,n,last,random)</code>	查看變數的 <code>n</code> 個元素
<code>expand.grid()</code>	將各變數交叉相乘排列
<code>gtools::permutations(n,r,v,set,rep)</code>	列出 <code>v</code> 向量前 <code>n</code> 個元素選 <code>r</code> 個排列的所有可能
<code>format()</code>	將資料轉成相同格式

## ➡ 常用函數與程式技巧

### ● 數學函數

<code>exp()</code> ; <code>log(x,base)</code>	指數;對數函數
<code>sin()</code> ; <code>cos()</code> ; <code>tan()</code> ; <code>asin()</code> ...	三角函數;反三角函數
<code>abs()</code>	絕對值函數
<code>sqrt()</code>	開根號函數
<code>sign()</code>	判斷正負號
<code>ceiling(x)</code>	大於等於 x 的最小整數
<code>floor(x)</code>	小於等於 x 的最大整數
<code>round(x,n)</code>	將 x 四捨五入到第 n 位
<code>trunc(x)</code>	傳回 x 的整數部分
<code>signif(x,k)</code>	顯示 x 的 k 為有效位數
<code>max()</code> ; <code>min()</code>	傳回元素最大值;最小值
<code>pmax(x,y,...)</code> ; <code>pmin()</code>	傳回指定向量各分量的最大值;最小值
<code>choose(n,k)</code> ; <code>factorial(x)</code>	$C_k^n$ ; $x!$
<code>integrate(f, lower, upper)</code>	設定上下限對 f 函數積分, 範圍可以是 Inf
<code>cubature::adaptIntegrate(f, lowerLimit, upperLimit, tol)</code>	多變數積分

### ● apply 系列函數

<code>apply(x,MARGIN,f,...)</code>	以 f 函數將矩陣或資料框架逐列、逐行(MA=1、2)計算
<code>tapply(x,INDEX,f,...)</code>	以 f 函數將向量依 INDEX 分類運算(...可設定 f 變數)
<code>sapply(x,f,...)</code> 、 <code>lapply(x,f,...)</code>	用於向量、list、資料框架, lapply 只會回傳 list

### ● table 系列函數

<code>tabulate(x,nbins)</code>	彙整正整數次數或分類出現次數(nbins 設定項目個數)
<code>table(x1,x2,...,xk,dnn)</code>	x 為 factor、list、data frame, 彙整分類組合發生次數
<code>margin.table(X,margin)</code>	對 table X 逐列、逐行(MA=1、2)加總
<code>colSums()</code> ; <code>rowSums()</code>	計算逐行(逐列)總和
<code>colMeans()</code> ; <code>rowMeans()</code>	計算逐行(逐列)平均
<code>prop.table(X,margin)</code>	計算逐列、逐行(MA=1、2)邊際比例 不設定 MA 則計算佔所有格子的比例
<code>as.data.frame(X,responseName)</code>	將 table X 各分類數值以 frame 顯示, res 設定數值名稱
<code>xtabs(計數~分類 1+...,data=X,subset)</code>	指定 X 內分類來彙整指定計數, 用 subset 邏輯篩選
<code>ftable()</code>	依照其分類變數作成扁平 table
<code>xtable::xtable + print(xtable,type,file,append)</code>	將 X 彙整的數值輸出美觀表格, type="html"輸出 HTML 格式、type="latex"輸出 LaTeX 格式, 儲存在 file 路徑檔案

### ● 遺失值函數

<code>na.fail(x)</code>	偵測 x 是否有遺失值, 若有則出現錯誤訊息並停止
<code>na.omit()</code> ; <code>na.exclude()</code>	去除 NA 值
<code>is.na()</code>	判斷個元素是否為 NA
<code>na.rm=T</code>	在部分函數內使用, 計算時可刪除遺失值

● optimization 系列函數	
stats4::mle(minuslogl,start)	minus 為計算(-1)*log-likelihood 自訂函數，以 stat 設定參數起始 list 變數來計算最大概似估計值
optimize(f,interval,max=F)	計算一維函數 f 的最佳解
uniroot(f,interval)	計算一維函數 interval 內的根
polyroot(c(z0,z1,z2,...))	求多項式之實根、虛根 zk 為 k 次方係數
optim(par,f,lower,upper)	以 par 起始值計算多維函數最佳解
nlmind(par,f,lower,upper);nlm(f,p)	以 par 起始值計算多維函數最佳解
constrOptim(theta,f,ui,ci)	以 theta 初值計算多維函數在限制式(ui*x≥ci)下最佳解
DEoptim::DEoptim(fn,lower,upper)	Differential Evolution Optimization
● 文字與字串處理	
cat("字串或變數",file,sep)	在螢幕或外部檔案顯示一些計算結果或文字
sprintf("%",字串或變數)	依照%類型設定，精準控制計算輸出格式
print.data.frame(X,digits=k,quote=F,right=T)	印出 data frame 變數內容，至少印出 k 位數 quote 設定是否加雙引號，right 設定是否靠右
paste(...,sep,collapse)	將各字串變數組合，可將所有元素用 collapse 連結一起
rep(x,time,length.out,each)	重複 x，time=x 重複次數,each=x 內元素重複次數
gl(n,k,length,labels,ordered=F)	以 n 個分類個數重複 k 次來設定 labels 分類 factor 變數
substr(x,start,stop)	在 x 中取出從 start 到 end 的子字串
strtrim(x,width)	以 width 決定取出 x 的寬度
strsplit(x,split)	依 split 字元將 x 分開
sub("舊字串","新字串",x);gsub()	將 x 中所有找到的舊字串，用新字串代替
grep(pattern,x,ignore.case=T)	輸出 x 內指定字串指標,ignore.case 設定是否忽略大小寫
● 常見的 R 程式技巧	
proc.time();system.time()	測量程式碼執行時間
Sys.time()	讀取時間
assign("變數名稱",運算式)	自動建立變數，並將運算式值存入
tutor::eval.string("字串")	將字串指令執行
● 自訂函數	
名稱=function(參數,...){}	參數可以有預設值、也可以是其他已定義之函數 內部變數不影響外部(除非使用<<-) 可用...代表其他需要傳入的參數 函數可以呼叫函數本身成為遞迴
invisible()	用在函數回傳值則函數不顯示計算結果
stop(op)	停止函數計算;並輸出 Error: op
warning(op)	執行函數計算;最後輸出 Warning message: op
args()	顯示函數的參數與預設值
formals()	顯示函數個參數特質
body()	顯示函數的內容
"%anything%"=function(a,b){}	自訂二元運算子

## ● 流程控制

<code>ifelse(,T,F)</code>	用於二分類邏輯判斷、成立執行 T、否則執行 F
<code>if(){ + }else if(){ + }else{ }</code>	用於多分類的邏輯判斷
<code>switch(計算值,運算式 1, 運算式 2…)</code>	依照計算值(整數或文字)決定要執行的運算式
<code>for(k in seq){ }</code>	以 k 在範圍 seq 中逐步執行運算式
<code>while(){ }</code>	依照(…)的判斷來執行運算式
<code>repeat{ }</code>	重複執行運算式直到跳出(break)
<code>break</code>	終止並跳離迴圈
<code>next</code>	跳過其後的運算，直接執行下一次迴圈

<b>➡ 繪圖功能</b>	
<b>● par 圖形設定函數</b>	
<code>mai=c(a,b,c,d)</code>	設定邊界值(inch)，順序為底左頂右
<code>mar</code>	與 mai 同，單位為文字行數
<code>mfc=c(a,b)</code>	以 a x b 矩陣將多張圖形畫在同一頁，依行排序
<code>mfrow=c(a,b)</code>	以 a x b 矩陣將多張圖形畫在同一頁，依列排序
<code>layout(M)</code>	設定比較不對稱的多圖佈局，依照 M 矩陣設計位置
<code>fig=(x1,x2,y1,y2)</code>	依照座標將圖擺在設定位置上
<b>● 圖形基本設定參數</b>	
<code>col=k</code> or "顏色名稱"	設定顏色(0 白,1 紅,2 黑,3 綠,4 藍,5 天藍,6 紫,7 黃,8 灰) colors() 可查詢顏色名稱 col.axis、col.lab、col.main、col.sub 改其他部分顏色
<code>lty=k</code>	設定線的種類(1 為實線、其他會各種虛線)
<code>pch=k</code> or "文字符號"	設定點的種類
<code>font=k</code>	1 為一般字體 2 為粗體字 3 斜體 4 粗斜體 font.axis...改其他部分的字體
<code>lwd=k</code>	設定 k 倍的線寬
<code>cex=k</code>	設定 k 倍的字型大小 cex.axis...改其他部分的字體
<b>● 高階繪圖函數</b>	
<code>plot(x,y);plot(xy)</code>	畫出 x,y 的散佈圖;畫出 xy(兩行的向量)的散佈圖
<code>plot(f)</code>	f 為 factor，畫出 f 的長條圖
<code>plot(f,y)</code>	以 f 為分類畫出 y 數值的盒狀圖
<code>plot(~x1+...+xk)</code>	畫出 k 個變數配對的矩陣散佈圖
<code>plot(y~x1+...+xk)</code>	逐一畫出 y 對各變數的散佈圖(以 ENTER 切換)
<code>curve()</code>	畫出給定的函數曲線
<code>pairs(X)</code>	X 為 data-frame、畫出 X 內行向量兩兩配對散佈圖
<code>coplot(x~y z)</code>	以 z 為分類下的 x~y 散佈圖
<code>qqnorm();qqline()</code>	畫出常態機率圖(Normal Probability Plot); 畫最佳斜線
<code>qqplot(x,y)</code>	畫出 x、y 兩組資料間的 QQ 圖
<code>hist(x)</code>	畫出直方圖(以 breaks 設定分隔點、nclass 設定群數)
<code>epicalc::dotplot()</code>	畫出點狀圖
<code>barplot()</code>	畫出長條圖
<code>boxplot()</code>	畫出合鬚圖
<code>pie(x,label)</code>	以 label 為分類畫出 x 數值的圓餅圖
<code>contour(x,y,z)</code>	畫出等高線圖
<code>image(x,y,z)</code>	畫出有色彩的等高線圖
<code>persp(x,y,z)</code>	畫出三度空間透視圖(以 theta、phi 設定角度)
<code>misc3d::parametric3d()</code>	畫出以兩個參數映射到 x、y、z 的 3D 圖形
<code>matplot(x,y)</code>	以 x 為橫軸，針對 y 的每一行畫圖



● 高階繪圖函數共用的輔助參數	
<code>add=T</code>	覆蓋在前一張圖形上
<code>axes=F</code>	不畫出座標軸
<code>log="x";log="y";log="xy"</code>	先將 x、y 的值取 log
<code>type=""</code>	p 圓點、l 線、b 實心圓點和線、o 空心圓點和線、s(S)階梯線、各點值在頂(底)端、h 垂直線、n 不畫圖
<code>xlab="" ;ylab=""</code>	設定軸的輔助說明文字
<code>xlim=c(a,b);ylim=c(a,b)</code>	將軸的範圍控制在(a,b)
<code>xaxt="n";yaxt="n"</code>	不畫出座標軸格線
<code>main="文字";sub="文字"</code>	加上主標題、次標題
● 附加圖形：低階繪圖函數	
<code>points(x,y)</code>	加上圖點
<code>lines(x,y)</code>	加上直線
<code>segments(x0,y0,x1,y1)</code>	以設定座標加上線段
<code>arrows(x0,y0,x1,y1,angle,code)</code>	加上箭頭線段(angle 設定角度、code 設定箭頭位置)
<code>text(x,y,labels)</code>	在指定位置寫出指定文字
<code>abline(a,b)</code>	加上直線( $y=ax+b$ )
<code>abline(h=y);abline(v=x)</code>	加上水平線或垂直線
<code>polygon()</code>	在指定位置畫出封閉多邊形、畫出陰影區域
<code>legend(x,y,legend)</code>	在指定位置畫出文字方塊
<code>title(main,sub)</code>	加上主標題或副標題
<code>axis(side,at,tick,labels)</code>	加上額外的座標軸
<code>locator(n,type="n")</code>	圖點座標位置查詢，最多點取 n 次
<code>identify(x,y,n,label)</code>	在指定點旁顯示其在原向量中的指標值，最多點取 n 次
<code>expression(plotmath)</code>	以 plotmath 撰寫規則畫出數學符號
● 繪圖設備	
<code>windows()</code>	圖行會顯示在視窗環境
<code>postscript("路徑")</code>	圖形輸出為 .ps 檔
<code>pdf("路徑")</code>	圖形輸出為 pdf 格式
<code>jpeg("路徑");png();bmp();tiff()</code>	輸出各類圖檔格式
<code>dev.off()</code>	終止畫圖指令

➡ 機率統計	
● 機率分配	
<code>dnorm(x,mean,sd,log=F)</code>	計算常態分配在 x 的機率密度函數值
<code>pnorm(x,lower.tail=F,log.p=F)</code>	常態分配累積到 x 機率值(lower=F 則機率從右尾算起)
<code>qnorm(p, lower.tail=F,log.p=F)</code>	計算常態分配機率為 p 時的分為點
<code>rnorm(n,mean,sd)</code>	模擬常態分配 n 個隨機亂數
<code>dbinom(size,prob)</code>	Binomial
<code>dgeom(prod)</code>	Geometric
<code>dhyper(x,m,n,k)</code>	Hypergeometric( $C_x^m * C_{k-x}^n / C_k^{m+n}$ )
<code>dmultinom(size,prob)</code>	Multinomial
<code>dnbinom(size,prob)</code>	Negative Binomial
<code>dpois(lambda)</code>	Poisson
<code>dbeta(shape1,shape2,ncp)</code>	Beta
<code>dcauchy(location,scale)</code>	Cauchy
<code>dchisq(df,ncp)</code>	Chi-Square
<code>dexp(rate)</code>	Exponential
<code>df(df1,df2,ncp)</code>	F
<code>dgamma(shape,scale)</code>	Gamma( $\alpha, \beta$ )
<code>MCMCpack::dinvgamma(shape,rate)</code>	Inverse Gamma
<code>dlnorm(meanlog,sdlog)</code>	Log-Normal
<code>dlogis(location,scale)</code>	Logistic
<code>mvtnorm::dmvnorm(mean,sigma)</code>	Multivariate Normal
<code>dsignrank(n)</code>	Wilcoxon signed rank
<code>dt(df,ncp)</code>	T
<code>dunif(min,max)</code>	Uniform
<code>i+floor((j-i+1)*dunif(x))</code>	Discrete Uniform(i,j)
<code>dweibull(shape,scale)</code>	Weibull
<code>dwilcox(m,n)</code>	Wilcoxon
● 機率計算	
<code>gamma(x);beta(a,b)</code>	Gamma function;Beta function
<code>set.seed()</code>	設定隨機種子
<code>sample(x,size,re,pr)</code>	從 x 中依照 pr 機率取出 size 個樣本(re=F 不放回)
<code>ecdf();plot.ecdf()</code>	計算資料的經驗累積機率函數;畫出資料的經驗 CDF 圖形
<code>density(x);plot(density(x))</code>	依資料計算密度函數;畫出 pdf 圖形
<code>print();summary()</code>	彙整計算結果
<code>ConvergenceConcepts::</code>	提供函數以模擬的方式檢驗各種收斂狀況
<code>qAnalyst::rapidFitFun()</code>	對資料做各個分配的服從檢驗
<code>MASS::fitdistr(x,"分配名稱")</code>	做參數的估計
<code>qAnalyst::andersonDarlingFun(x,dis="",theta=c())</code>	檢定 x 是否與 theta 設定之參數下的 dis 相同分配

● 敘述統計	
<code>mean(x, trim)</code>	傳回 x 元素的算術平均, trim 決定去除資料比率
<code>asbio::G.mean(x)</code>	傳回 x 元素的幾何平均
<code>median(x)</code>	傳回 x 元素的中位數
<code>var(x);sd(x)</code>	傳回 x 元素的變異數、標準差
<code>range(x)</code>	傳回 x 的最大最小值
<code>pracma::mode()</code>	傳回 x 的眾數
<code>quantile(x, pr)</code>	計算第 pr*100 百分率下的值
<code>IQR()</code>	內四分位距
<code>fivenum()</code>	列出 min, Q1, median, Q3, max
<code>cor(X)</code>	X 為 data frame 時、傳回 X 的相關矩陣
<code>cor(x, y)</code>	計算 x、y 的相關係數
<code>cov(x, y)</code>	計算 x、y 的共變異數
<code>TSA::skewness();kurtosis()</code>	計算偏態;峰態係數
<code>mad()</code>	計算中位數絕對離差
<code>stem()</code>	莖葉圖
<code>stats::scale(x, center, scale)</code>	將 x 向量標準化
● 單樣本推論	
<code>asbio::one.sample.z(x, null.mu, sigma, test)</code>	單樣本 Z 檢定 (test="two.sided", "lower", "upper")
<code>asbio::power.z.test()</code>	計算單樣本 Z 檢定的檢定力
<code>t.test(x, mu, alternative, conf.level)</code>	t 檢定(alternative="two.sided", "less", "greater")
<code>power.t.test()</code>	計算 t 檢定的檢定力
<code>binom.test(x, n, p, alt, conf)</code>	母體比例 p 檢定(精確二項分配檢定)
<code>prop.test(x, n, p, alt, conf)</code>	母體比例 p 檢定(比例檢定)
● 雙樣本推論	
<code>t.test(x1, x2, alt, mu, paired=F, var.equal=T, conf)</code>	雙樣本 t 檢定 (pair 設定是否為配對, var.equal 設定變異數是否相等)
<code>var.test(x1, x2, alt)</code>	檢定兩母體變異數是否相等
<code>prop.test(c(x1, x2), c(n1, n2))</code>	雙樣本母體比例差異檢定
<code>cor.test(x1, x2, alt, method, conf)</code>	雙樣本相關係數檢 (meth="pearson", "kendall", "spearman")
● 常態性檢定	
<code>shapiro.test()</code>	Shapiro-Wilk 檢定
<code>nor.test::ad.test()</code>	Anderson-Darling 檢定
<code>nor.test::sf.test()</code>	Shapiro-Francia 檢定
<code>nor.test::cvm.test()</code>	Cramer-von Mises 檢定
<code>nor.test::pearson.test()</code>	Pearson chi-square 檢定
<code>nor.test::lillie.test()</code>	Lilliefors(Kolmogorov-Smirnov) 檢定
<code>tseries::jarque.bera.test()</code>	Jarque-Bera 檢定

● 卡方檢定	
<code>chisq.test(x,p,rescale.p=F)</code>	適合度檢定(檢定 x 是否服從 p 機率之分配)
<code>chisq.test(X,simulate.p.value=F)</code>	齊一性檢定、獨立性檢定(X 為儲存列聯表之矩陣)
<code>fisher.test()</code>	Fisher 精確獨立性檢定
● 無母數檢定	
<code>UsingR::simple.median.test(x,mde)</code>	單樣本中位數符號檢定
<code>lawstat::runs.test()</code>	隨機性檢定
<code>tseries::runs.test()</code>	二元資料隨機性檢定
<code>wilcox.test(x,y,mu,paired=F)</code>	兩樣本位置參數檢定(paired 設定是否獨立)
<code>ks.test(x,y,...)</code>	檢定是否為相同連續分配(y 可改成 p+分配, ...設定參數)
<code>kruskal.test(X); kruskal.test(x,g)</code>	多樣本位置參數檢定(X 為 list;x 向量,g 分類)
<code>BHH2::permtest()</code>	randomization test for small size samples
<code>coin::spearman_test(y~x)</code>	Spearman rank for test correlation
<code>psych::corr.test(x,y,use,method)</code>	Find and test the correlation
● 其他	
<code>boot::boot(data,stat,r)</code>	Bootstrap Resampling

➡ 迴歸分析	
<code>lm(...,data,subset,x)</code>	配置模型，x=T 回傳設計矩陣
<code>predict(lm,newdata, interval,level,se)</code>	用模型對新資料做預測
<code>anova(lm);aov(y~.)</code>	對模型做 ANOVA 表
● 羅吉斯迴歸	
<code>glm(...,data,family)</code>	Generalized Linear Models
<code>nnet::multinom(y~.)</code>	Fit Multinomial Log-linear Models
<code>MASS::polr(y~.)</code>	Ordered Logistic or Probit Regression
<code>Design::lrm(y~.)</code>	Logistic Regression Model
<code>epicalc::logistic.display(glm)</code>	彙整 glm 計算結果
<code>epicalc::mlogit.display(multinom)</code>	彙整 multinom 計算結果
<code>epicalc::ordinal.or.display(polr)</code>	彙整 polr 計算結果
● 結果擷取	
<code>deviance(lm)</code>	擷取 SSE
<code>residuals(lm);lm\$residuals</code>	擷取殘差
<code>fitted(lm);lm\$fitted.value</code>	擷取配適值
<code>coef(lm);lm\$coef</code>	擷取係數
<code>effects(lm)</code>	擷取正交效果估計值
<code>vcov(lm)</code>	擷取係數 Variance-Covariance 矩陣
<code>formula(lm)</code>	擷取原始模型
<code>model.matrix(lm);lm\$x</code>	擷取設計矩陣
● 模型檢測	
<code>lmtest::coeftest(lm)</code>	係數顯著性檢定
<code>car::durbinWatsonTest(lm)</code>	殘差獨立性檢定
<code>lmtest::dwtest(lm)</code>	殘差獨立性檢定
<code>car::ncvTest(lm)</code>	殘差齊一性檢定
<code>car::leveneTest(lm,data)</code>	殘差齊一性檢定
<code>lmtest::bptest(lm)</code>	殘差齊一性檢定
<code>Bartlett.test(y~.)</code>	殘差齊一性檢定
<code>cooks.distance(lm);car::cookd()</code>	Cook' s Distance
<code>dfbetas(lm)</code>	DBETAS
<code>dffits(lm)</code>	DFFITS
<code>lm.influence(lm)</code>	計算各類影響點的指標值
<code>hat(X);hatvalues(lm)</code>	Hat matrix 的對角線元素(X 為設計矩陣)
<code>car::outlier.test(lm)</code>	Bonferroni Outlier Test
<code>rstandard(lm)</code>	standardize residuals
<code>rstudent(lm)</code>	studentized residuals
<code>car::vif(lm)</code>	計算 VIF 值
<code>covratio(lm)</code>	Covariance ratio

● 變數篩選	
<code>step(lm,direction,k)</code>	逐步迴歸(AIC:k=2,BIC:k=log(樣本數))
<code>MASS::stepAIC()</code>	逐步迴歸
<code>leaps::leaps(X,y)</code>	All-Possible 選取(X 為自變數矩陣)
<code>leaps::regsubsets(X,y,nbest,nvmax,method)</code>	向前、向後、逐步、All-Possible
<code>update(lm, .~. +x1-x2...)</code>	新增或移除模型的解釋變數(第一個點為原始 y)
<code>add1()</code>	計算給定範圍內的單一變數加入模型的影響
<code>drop1()</code>	計算給定範圍內的單一變數離開模型的影響
<code>offset()</code>	強制加入係數為 1 的解釋變數項
<code>AIC(lm,k)</code>	計算 AIC、BIC
<code>extractAIC(lm,k)</code>	計算 AIC、BIC
<code>BIC()</code>	計算 BIC
<code>maxadjr()</code>	計算 Maximum Adjusted R-Square
● 繪圖	
<code>plot(lm)</code>	畫四種殘差分析圖
<code>pairs(data); car::scatterplotMatrix(data)</code>	畫出所有變數中兩兩配對的矩陣散佈圖
<code>faraway::Cpplot(leaps())</code>	Cp plot
<code>faraway::prplot(lm,i)</code>	Partial Residual Plot
<code>car::regLine(lm)</code>	附加畫上回歸線
<code>car::crPlot(lm,x)</code>	Component+Residual Plots(x 為自變數名稱)
<code>car::avPlots(lm)</code>	Added-Variable Plots
<code>car::influencePlot(lm)</code>	Regression Influence Plot
<code>car::leveragePlots(lm)</code>	Regression Leverage Plots
<code>car::ceresPlots(lm)</code>	Ceres Plots
<code>car::spreadLevelPlot(lm)</code>	Spread-Level Plots
● 進階指令	
<code>car::bcPower(x,lambda);yjPower()</code>	Box-Cox and Yeo-Johnson Power Transformations
<code>car::powerTransform(object)</code>	Finding Univariate or Multivariate Power Transformations
<code>car::boxTidwell(y)</code>	Box-Tidwell Transformations
<code>MASS::lm.ridge(y~.)</code>	Ridge Regression
<code>nls(y~.)</code>	Nonlinear Least Squares
<code>MASS::rlm(y~.)</code>	Robust Fitting of Linear Models
<code>MASS::lqs(y~.)</code>	Resistant Regression
<code>systemfit::systemfit(y~.)</code>	Linear Equation System Estimation

<b>➡ 實驗設計</b>	
<b>● ANOVA 基本分析模型</b>	
<code>interaction.plot(x,y,Y)</code>	因子變數 x,y 間的交互作用圖，Y 為觀察值
<code>anova(lm(模型))</code>	依照指定模型，計算 ANOVA Table
<code>aov(模型)</code>	作用同上，輸出資料格式不同
<code>model.matrix(模型)</code>	輸出設計矩陣
<code>model.tables(aov(模型), type="")</code>	<code>type="effects"</code> 輸出影響估計量； <code>"means"</code> 輸出平均數
<b>● ANOVA 殘差檢定函數</b>	
<code>lmtest::dwtest(模型)</code>	檢定殘差是否互相獨立
<code>car::durbinWatsonTest(lm(模型), method="normal")</code>	算出 Durbin-Watson 檢定值與 P-value
<code>bartlett.test(lm(模型)\$res,A)</code>	檢定殘差變異數是否有明顯差異
<code>ncvTest(lm(模型))</code>	檢定殘差變異數是否有明顯差異
<code>car::outlierTest(lm(模型))</code>	檢定資料是否含有極端值
<b>● 一因子模型</b>	
$Y \sim A$	一因子模型，Y 為觀察值向量，A 為因子變數
$Y \sim A + x$	一因子模型，x 為共變量變數
<code>oneway.test(Y~A, var.equal=F)</code>	當殘差變異數部為常數時，檢定一因子影響的顯著性
<code>kruskal.test(Y~A)</code>	不需要 Normal 分配假設下，檢定一因子影響的顯著性
<b>● 隨機集區設計</b>	
<code>B=as.factor(paste("B",1:n,sep=""))</code>	生成集區變數
$Y \sim E + B$	隨機集區設計模型，E 為因子變數，B 為集區變數
<b>● 拉丁方格設計</b>	
<code>crossdes::MOLS(dim,k)</code>	生成拉丁方格
<code>T=toupper(letters[as.vector(M)])</code>	生成 Treat. 變數(需再轉成 factor)，M 為拉丁方格矩陣
<code>B1=rep(paste("a",1:n,sep=""),e=n)</code>	生成 Block1 變數(需再轉成 factor)
<code>B2=rep(paste("b",1:n,sep=""),t=n)</code>	生成 Block2 變數(需再轉成 factor)
$Y \sim B1 + B2 + T$	拉丁方格設計模型
<b>● 二因子設計</b>	
<code>FA=rep(paste("A",1:n,sep=""),t=n)</code>	生成 FactorA 變數(需再轉成 factor)
<code>FB=rep(paste("B",1:n,sep=""),e=n)</code>	生成 FactorB 變數(需再轉成 factor)
$Y \sim FA + FB; Y \sim FA + FB + FA:FB$ or $Y \sim FA * FB$	二因子設計模型( $n=1; n>1$ )，FA,FB 代表兩因子變數
$Y \sim B \%in\% A; Y \sim A/B$	巢狀模型(Nested Model)，A 分類下細分出 B
$Y \sim A * B + Error(C)$	二因子裂區設計，A,B 分配機制由 C 決定
<b>● 兩水準因子設計</b>	
$Y \sim A * B * C$	兩水準因子設計模型
$Y \sim (A+B+C)^2$	兩水準因子設計模型，不含三因子交互作用項
$Y \sim A * B * C + Error(A/B/C)$	三因子巢狀裂區設計(Split-Plot)，影響區域由大而小
<code>BsMD::DanielPlot(lm(模型), half=T)</code>	畫出 Half-Normal 圖形
<code>BsMD::LenthPlot(lm(模型))</code>	畫出 Lenth 篩選圖形

● 多重比較	
TukeyHSD(aov( 模型))	依照模型，計算多重比較
plot(TukeyHSD(aov( 模型)))	畫出多重比較結果
multcomp::glht(aov( 模型), linfct=mcp(A=""))	計算多重比較 (A="Tukey"; "Dunnett")
multcomp::plot(cld(x))	畫出多重比較結果，x 為 glht 之 output
multcompView::multcompBoxplot( 模型, data=data.frame(Y,A))	畫出多重比較圖
laercio::LDuncan(aov( 模型), "A")	計算 Duncan 多重比較信賴區間
laercio::LTukey(aov( 模型), "A")	計算 Tukey 多重比較信賴區間
asbio::LSD.test(Y,A)	計算 LSD 多重比較信賴區間(A 可為任何 factor 變數)
asbio::BonferroniCI(Y,A)	計算 Bonferroni 多重比較信賴區間
asbio::TukeyCI(Y,A)	計算 Tukey 多重比較信賴區間
asbio::ScheffeCI(Y,A)	計算 Scheffe 多重比較信賴區間
asbio::Pairw.test(Y,A, method="Scheffe")	計算配對多重比較信賴區間



➡ 品質管制	
● qcc 套件	
cause.and.effect(cause=list(), effect="")	effect 為主要結果，list()列出所有可能原因
pareto.chart()	繪製柏拉圖
qcc.options(bg.margin,bg.figure)	設定管制圖的顏色
qcc(data,type)	繪製管制圖(type="xbar","R","S","xbar.one")
qcc(data,sizes,type)	繪製管制圖(type="p","np")
qcc(data,sizes=units,type="c")	繪製 c 管制圖
process.capability(xbar 物件,spec)	依照 spec 之規格上下限計算製程能力指標
oc.curves()	畫出 OC 曲線
qcc.groups(x,y)	將 x 資料依照 y 類別轉換成 qcc 可用的 data 矩陣
cusum()	繪製 CUSUM 管制圖
ewma(x,lambda)	繪製 EWMA 管制圖
● qAnalyst 套件	
paretoChart()	繪製柏拉圖
plot(spc(x,sg,type))	將 x 資料依照 sg 之類別繪製管制圖 (type="xbar","r","s",)
plot(spc(x,sg=2,type="i",...))	繪製單一觀察值 X 管制圖
plot(spc(x,sg=2,type="mr",...))	繪製 MR 管制圖

<b>➡ 多變量分析</b>	
<b>● Principal Component Analysis</b>	
<code>princomp(x)</code>	執行主成分分析
<code>stats::loadings(x)</code>	x 為分析結果變數，計算係數矩陣(B)
<code>biplot(x)</code>	x 為分析結果變數，畫出分析結果
<b>● Canonical Correlation Analysis</b>	
<code>stats::cancor(X,Y)</code>	依據 X,Y 矩陣計算相關係數
<code>CCP::p.asym(cor,N,p,q,tstat)</code>	檢定相關係數的顯著性
<b>● Classification</b>	
<code>rpart::rpart(Y~.,data,control)</code>	執行 Classification Tree 分類(結合 plot;text 畫圖)
<code>rpart::printcp(X)</code>	check out the complexity，X 為 rpart 物件
<code>rpart::prune.rpart(X,cp)</code>	以 cp 值修剪分類樹
<code>MASS::lda(Y~.,data)</code>	Linear discriminant analysis
<code>MASS::qda(Y~.,data)</code>	Quadratic Discriminant Analysis
<code>class::knn(X,X,Y,k,prob)</code>	k-Nearest Neighbour(X 為變數資料;Y 為分類)
<code>class::knn.cv(X,Y,k,prob)</code>	k-Nearest Neighbour Cross-Validatory Classification
<code>nnet::multinom(Y~.,data,maxit)</code>	Fit Multinomial Log-linear Models
<code>e1071::svm(X,Y)</code>	Support Vector Machines
<b>● Cluster</b>	
<code>MVA::hclust()</code>	Hierarchical Clustering
<code>stats::kmeans()</code>	K-Means Clustering
<code>cluster::agnes()</code>	Agglomerative Nesting
<code>cluster::diana()</code>	DIVISive ANALYSIS Clustering
<code>cluster::mona()</code>	MONothetic Analysis Clustering of Binary Variables
<code>cluster::pam()</code>	Partitioning Around Medoids
<code>cluster::silhouette()</code>	Compute or Extract Silhouette Information
<code>cluster::clara()</code>	Clustering Large Applications
<code>cluster::fanny()</code>	Fuzzy Analysis Clustering
<code>cluster::daisy()</code>	Dissimilarity Matrix Calculation
<code>som::som()</code>	Function to train a Self-Organizing Map
<code>klaR::shardsplot()</code>	Plotting Eight Direction Arranged Maps or Self-Organizing Maps
<b>● Multidimensional Scaling</b>	
<code>sweep(x, MARGIN, STATS)</code>	Sweep out Array Summaries
<code>stats::cmdscale(x,k,eig)</code>	Classical (Metric) Multidimensional Scaling
<code>stats::dist()</code>	Distance Matrix Computation
<code>MASS::isoMDS()</code>	Kruskal's Non-metric Multidimensional Scaling Description
<b>● Factor Analysis</b>	
<code>factanal(x,facrors)</code>	Factor Analysis

● Correspondence Analysis	
MASS::corresp(x,nf)	Simple Correspondence Analysis
MSSS::mca(x,nf,abbrev)	Multiple Correspondence Analysis
● Independent Component Analysis	
fastICA:: fastICA(x, n.comp)	FastICA algorithm

## ● 其他重要指令

<code>gtools::permute(x)</code>	將 x 向量元素隨機排列
<code>schoolmath::is.prim(x)</code>	判斷 x 值是否為質數
<code>unique(x)</code>	唯一化 x 的元素
<code>duplicated(x)</code>	判斷 x 內元素之前是否已經出現
<code>letters</code>	小寫字母所組成之向量
<code>toupper(x)</code>	將英文小寫字母轉成大寫
<code>date()</code>	查詢現在日期時間
<code>menu()</code>	執行選擇程序
<code>library(Rcmdr)</code>	叫出 R Commander 介面
<code>all(logical vector)</code>	向量存在一個以上的 F 則回傳 F，否則回傳 T
<code>rainbow();heat.colors()</code>	顏色抓取