

106-2: EE4052

通識課程：

計算機程式設計

之旅

Computer Programming

Unit 10: 繪圖功能與文字

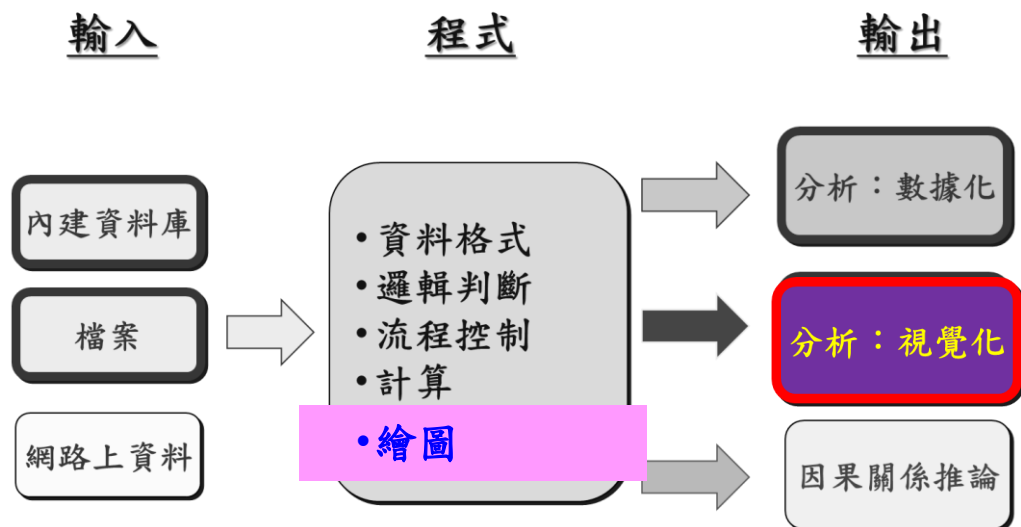
連 豐 力

臺大電機系

Feb 2018 - Jun 2018

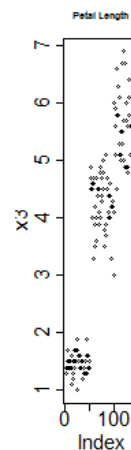
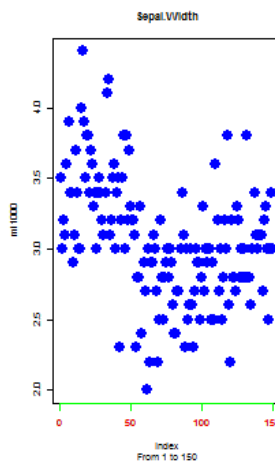
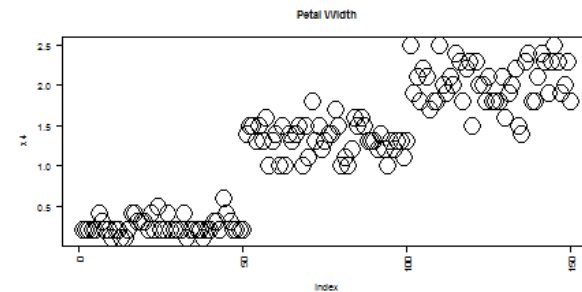
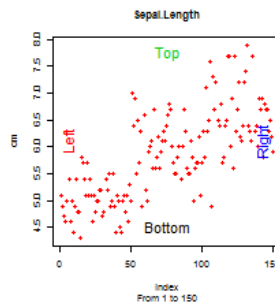
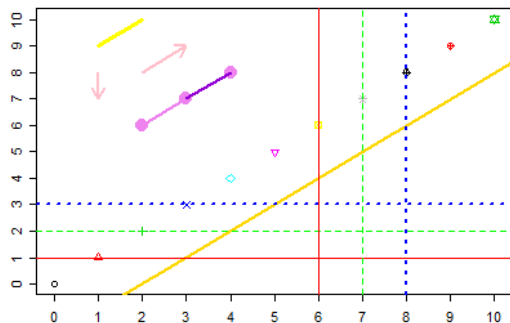
課程主題進度

- **U01:** 課程介紹：討論主題，作業，報告，進行方式
- **U02:** 主題，案例，程式，演算法，資源
- **U03:** 設定軟體 R 與 Rstudio
- **U04:** 數據處理與繪圖指令功能
- **U05:** 資料類別與基本運算
- **U06:** 邏輯判斷與流程控制
- **U07:** 函數：計算與排序
- **U08:** 多維度資料格式
- **U09:** 檔案資料輸入與輸出
- **U10:** 繪圖功能與文字
- **U11:** 多重繪圖與顏色
- **U12:** 資料間的相關性
- **U13:** 探索性資料分析
- **U14:** 資料連結分析
- **U15:** 影像與動畫

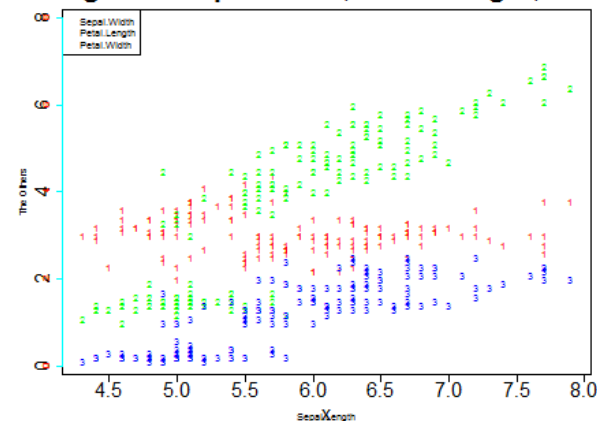


Unit 10: 繪圖功能與文字

- 一張圖的長寬或大小，主標題，軸標題，字體等特性的設定
- 數據點的顏色要有變化
- 數據點的形式要有變化
- 數據加上不同註解



I.Length vs Sepal.Width, Petal.Length, Petal



作業

HW08：繪圖功能與文字

On 5/15, 2018

- 請使用 iris 的第一組數據 (Sepal.Length)，繪製一張圖，符合下面的要求：
 - 邊界文字縮放比：`mex = 0.8`
 - 下左上右四個邊界之預留距離：`mar = c(6, 6, 3, 3)`
 - X 軸（橫）之範圍：`(-10, 160)`
 - Y 軸（縱）之範圍：`(4, 8)`
 - X 軸（橫）之名稱：`Plant ID`
 - Y 軸（縱）之名稱：`Length of Sepal`
 - 整個圖形的之主標題：`IRIS Data`
 - 繪製顏色：`blue`
 - 點的型式：`2` 號，三角形
 - 文字及符號相對於內定值之縮放比：`cex = 2`
 - 在下面座標軸 (`side=1`)，座標線為綠色，座標數字為紅色
 - 在左上角加上標記：`Plant ID vs Sepal.Length`
 - 在 `(120, 4)` 的地方，放入文字：`Plot the IRIS data, Sepal.Length`
 - 在 `(120, 4.5)` 的地方，放入數學符號：`y = x^2 + 5`

HW08：繪圖功能與文字

On 5/15, 2018

- 繳交下面檔案，檔案名稱：`HW08_學號_關鍵字.xxx`
 - 主要指定檔案：`HW08_B01921001_PlotFigure.R`
 - 報告檔案：`HW08_B01921001_PlotFigure.pdf`
 - 或者是：`R Markdown` 等整合式的檔案，`.Rmd` 與 `.pdf/.html`
- 繳交方式與期限：
 - E-mail 上面兩個檔案到：ntucp2018s@gmail.com
 - E-mail 主旨：`HW08_B01921001_PlotFigure`
(就是，作業編號_您的學號_關鍵字)
 - 繳交期限：**5/20 (Sun), 2018, 11pm 以前**
- 學習方式：請至下面網址輸入此次的學習方式所花的時間：
 - <https://goo.gl/k7tKLk>
 - https://docs.google.com/forms/d/e/1FAIpQLSdAZ_b-FUtvnNr_14rYQNYejMhDESy6jJ9ESh5XsjFI-DXMIw/viewform?c=0&w=1

- 繪圖視窗之設定
 - 常用的圖形參數
 - 座標軸及邊界
 - 加入圖形元件
 - 加入文字
-
- 多張圖形
 - 多張圖形之位置安排
 - 一張圖多筆數據
 - 顏色

Unit 10

Unit 11

繪圖視窗之設定

繪圖視窗

- `plot()`
- `windows()`
- `curve()`

`windows(width, height, pointsize, record, rescale, xpinch, ypinch, bg, canvas, gamma, xpos, ypos, buffered, title, restoreConsole, clickToConfirm, fillOddEven, family, antialias)`

`plot(x, y, ...)`

`curve(expr, from = NULL, to = NULL, n = 101, add = FALSE, type = "l", xname = "x", xlab = xname, ylab = NULL, log = NULL, xlim = NULL, ...)`

繪圖視窗

- # in Mac OS
- quartz()

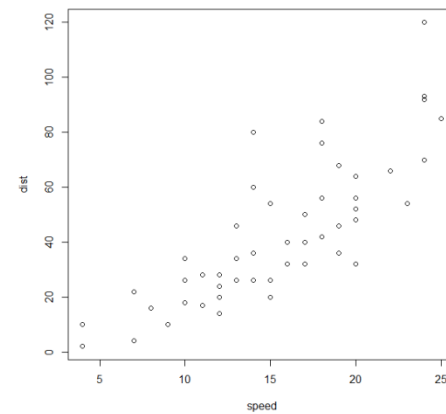
`quartz(title, width, height, pointsize, family, antialias, type, file = NULL, bg, canvas, dpi)`

- `windows(.)` --> `quartz(.)`
- `win.graph(.)` --> `dev.new(.)`

- # in UNIX
- X11()

`X11(width, height, pointsize, bg, gamma, xpos, ypos, title)`

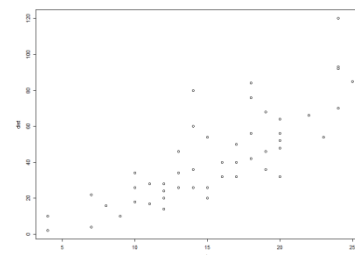
- `data(cars)`
- `summary(cars)`
- `cars`
- `head(cars)`
- `tail(cars)`
- `str(cars)`
- `plot (cars)`



繪圖視窗之設定

- `windows(width = 4.5, height = 3.3, pointsize = 8)`

pointsize: 文字或符號字體大小



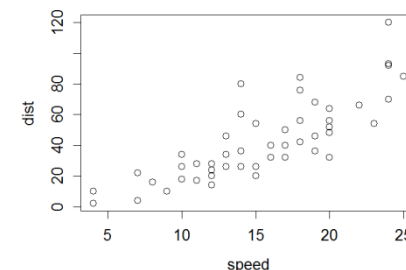
- `windows(width = 8, height = 6, pointsize = 20)`

- `old.par <- par(mex = 0.8, mar = c(5, 5, 4, 2) + 0.1)`

mex: 邊界文字的縮放比

mar: 下 左 上 右 四個邊界之預留距離

下: side=1, 左: side=2, 上: side=3, 右: side=4



- `plot(cars)`

- `par(old.par)` # reset to previous settings

繪圖視窗之設定

- `windows(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 5, 4, 2) + 0.1)`
- `plot(cars)`
- `par(old.par)` # reset to previous settings

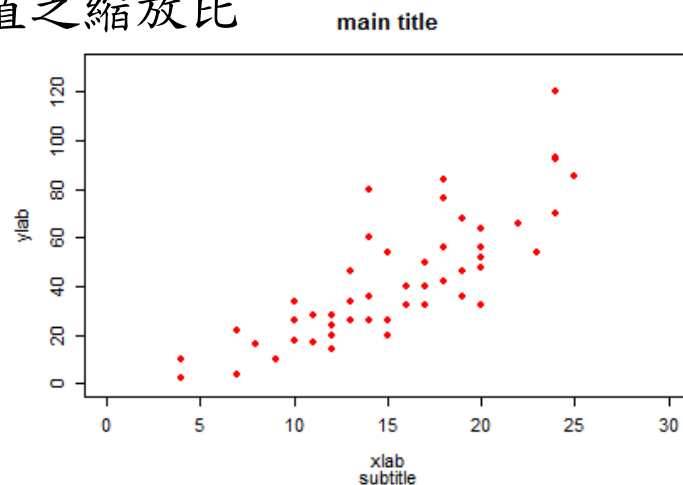
- `win.graph(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 5, 4, 2) + 0.1)`
- `plot(cars)`
- `par(old.par)` # reset to previous settings

- `windows(width, height, pointsize, record, rescale, xpinch, ypinch, bg, canvas, gamma, xpos, ypos, buffered, title, restoreConsole, clickToConfirm, fillOddEven, family, antialias)`
- `win.graph(width, height, pointsize)`
- `x11(width, height, pointsize, bg, gamma, xpos, ypos, title)`
- `X11(width, height, pointsize, bg, gamma, xpos, ypos, title)`
- `win.metafile(filename = "", width = 7, height = 7, pointsize = 12, family, restoreConsole = TRUE)`
- `win.print(width = 7, height = 7, pointsize = 12, printer = "", family, antialias, restoreConsole = TRUE)`

常用的圖形參數

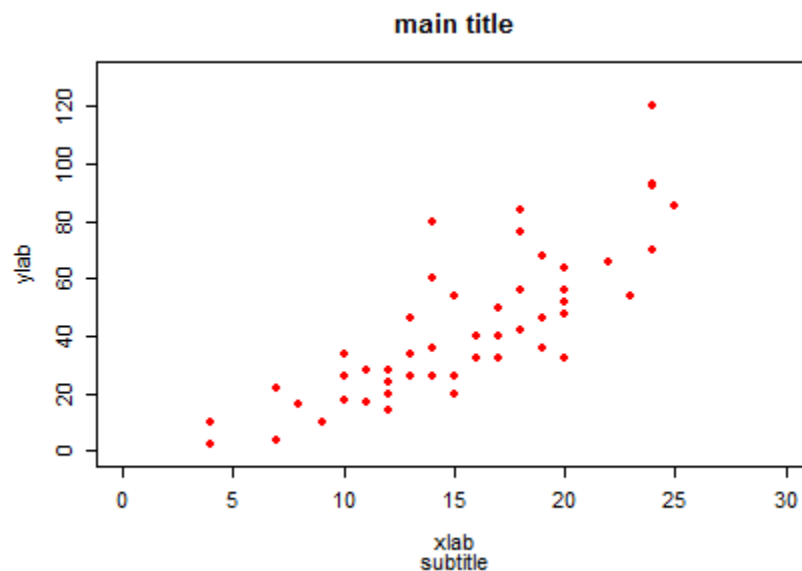
圖形參數

- **xlim:** # x 軸 (橫) 之範圍; **xlim[1]**: 左邊界, **xlim[2]**: 右邊界
- **ylim:** # y 軸 (縱) 之範圍; **ylim[1]**: 下邊界, **ylim[2]**: 上邊界
- **xlab:** # x 軸 (橫) 之名稱, 出現在橫軸下方
- **ylab:** # y 軸 (縱) 之名稱, 出現在橫軸左方
- **main:** # 整個圖形的之主標題, 出現在圖形的上方
- **sub:** # 副標題, 出現在橫軸下方
- **cex:** # 文字及符號相對於內定值之縮放比
- **pch:** # 點的型式
- **col:** # 繪圖的顏色



圖形參數

- `windows(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(7, 5, 4, 2) + 0.1)`
- `plot(cars, xlim = c(0, 30), ylim = c(0, 130), xlab = "xlab", ylab = "ylab", main = "main title", sub = "subtitle", cex = 0.8, pch = 16, col = "red")`
- `par(old.par)`
reset to previous settings



Graphical Parameters

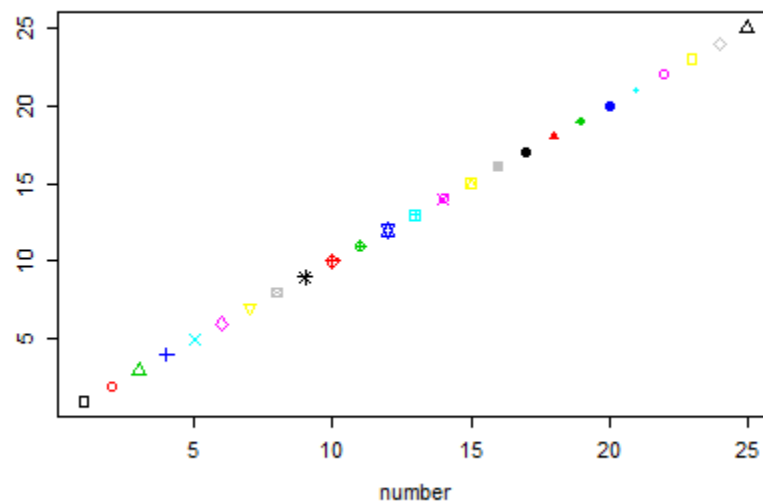
<http://www.statmethods.net/advgraphs/parameters.html>

R Graphical Parameters Cheat Sheet

<http://gastonsanchez.com/visually-enforced/resources/2015/09/22/R-cheat-sheet-graphical-parameters/>

圖形參數

- `windows(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 5, 4, 2) + 0.1)`
- `plot(1:25, pch = 0:25, col = 1:8, xlab = "number", ylab = "")`
- `par(old.par)`
reset to previous settings



Graphical Parameters

<http://www.statmethods.net/advgraphs/parameters.html>

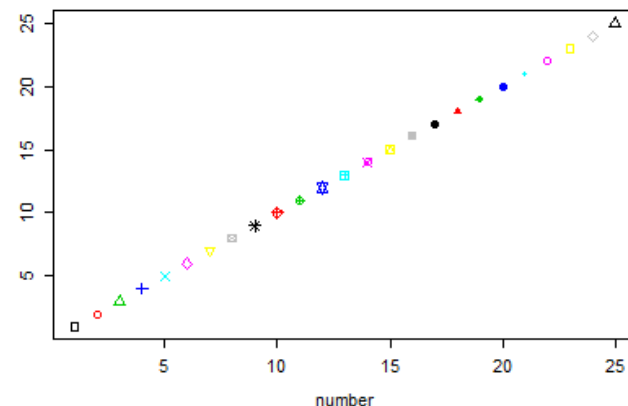
R Graphical Parameters Cheat Sheet

<http://gastonsanchez.com/visually-enforced/resources/2015/09/22/R-cheat-sheet-graphical-parameters/>

座標軸及邊界

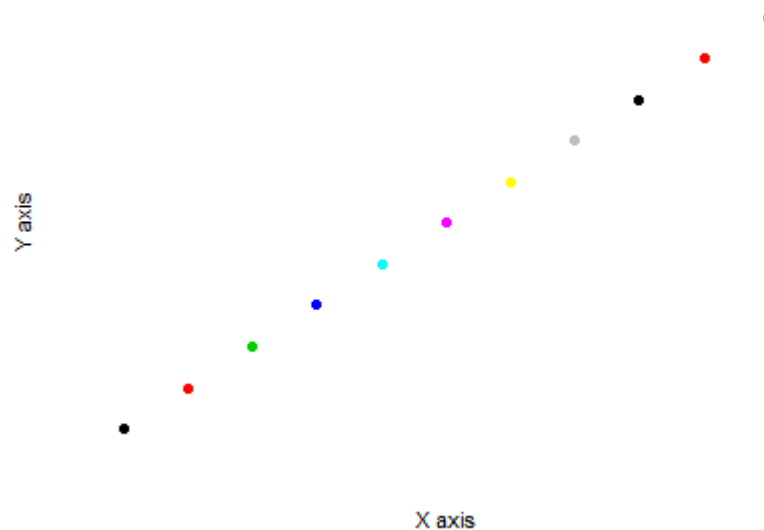
座標軸及邊界參數

- **axis:** # 設定座標軸之函數
- **axes:** # **axes = TRUE** : 要畫座標軸; **axes = FALSE** : 不要畫座標軸
- **xaxt:** # **xaxt = "n"** : 不要畫 x 軸
- **yaxt:** # **yaxt = "n"** : 不要畫 y 軸
- **col.axis:** # 座標軸之文字顏色
- **side:** # 繪圖區域邊界的編號
- **mex:** # 邊界文字之縮放比
- **mar:** # 下左上下右四個邊界之預留距離



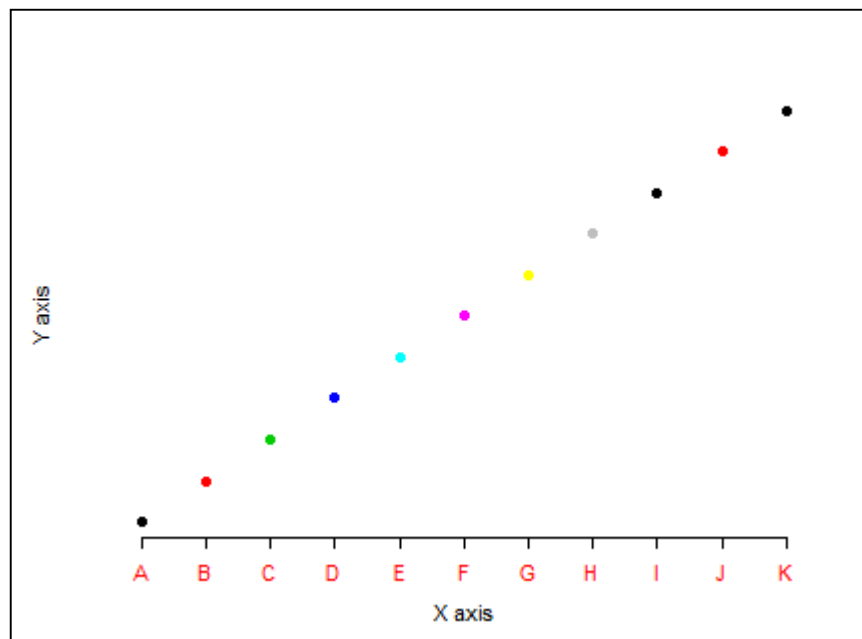
座標軸及邊界參數

- `windows(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 5, 4, 3) + 0.1)`
- `plot(0:10, 0:10, col = 1:10, pch = 16, xlab = "X axis", ylab = "Y axis", axes = FALSE)`



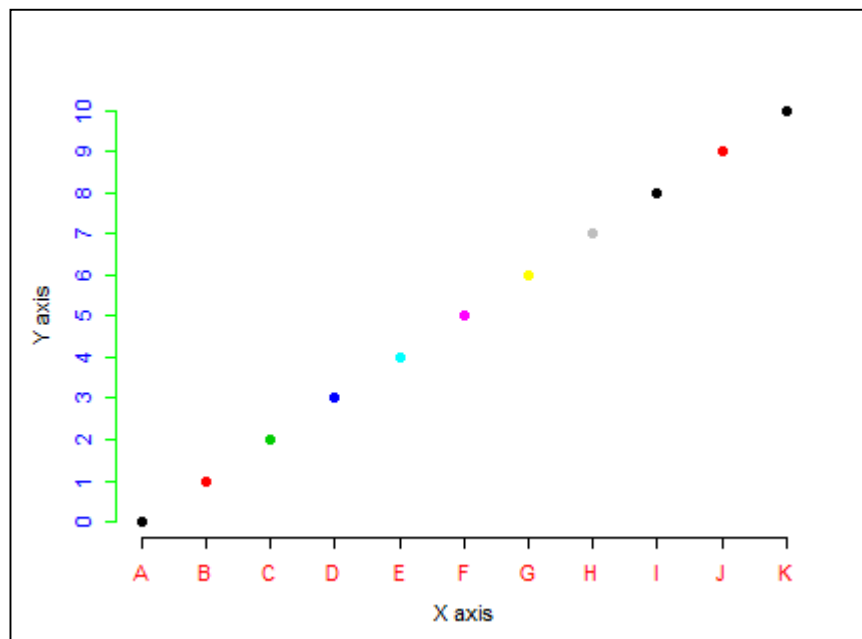
座標軸及邊界參數

- `plot(0:10, 0:10, col = 1:10, pch = 16, xlab = "X axis", ylab = "Y axis", axes = FALSE)`
- `axis(side = 1, col = "black", col.axis = "red", at = 0:10, label = LETTERS[1:11])`



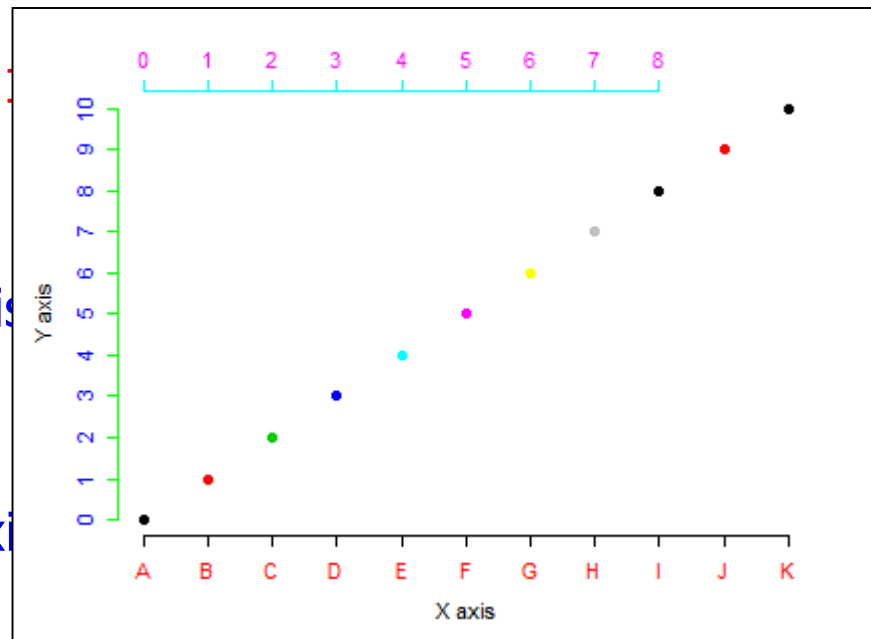
座標軸及邊界參數

- `plot(0:10, 0:10, col = 1:10, pch = 16, xlab = "X axis", ylab = "Y axis", axes = FALSE)`
- `axis(side = 1, col = "black", col.axis = "red", at = 0:10, label = LETTERS[1:11])`
- `axis(side = 2, col = "green", col.axis = "blue", at = seq(from = 0, to = 10, by = 1))`



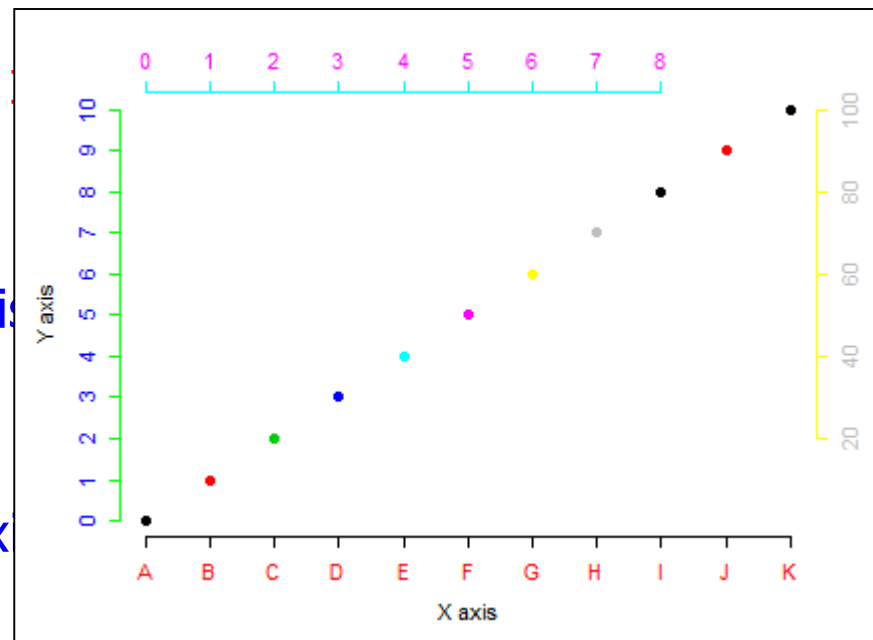
座標軸及邊界參數

- `plot(0:10, 0:10, col = 1:10, pch = 1, axes = FALSE)`
- `axis(side = 1, col = "black", col.axis = "black", at = LETTERS[1:11])`
- `axis(side = 2, col = "green", col.axis = "black", at = 0:10, by = 1))`
- `axis(side = 3, col = "cyan", col.axis = "magenta", at = seq(from = 0, to = 8, by = 1))`



座標軸及邊界參數

- `plot(0:10, 0:10, col = 1:10, pch = axes = FALSE)`
- `axis(side = 1, col = "black", col.axis = LETTERS[1:11])`
- `axis(side = 2, col = "green", col.axis = 10, by = 1))`
- `axis(side = 3, col = "cyan", col.axis = "magenta", at = seq(from = 0, to = 8, by = 1))`
- `axis(side = 4, col = "yellow", col.axis = "gray", at = seq(from = 2, to = 10, by = 2), label = c(20, 40, 60, 80, 100))`
- `par(old.par)`



加入圖形元件

加入圖形元件 - 點 線 框

- `points()` # 打點
- `lines()` # 畫線
- `abline()` # 畫 $y = b x + a$ 的直線
- `segments()` # 畫線段
- `arrows()` # 畫箭頭
- `box()` # 在原圖形最外圍加上框框
- `lty` # 直線的樣式
- `lwd` # 直線的寬度

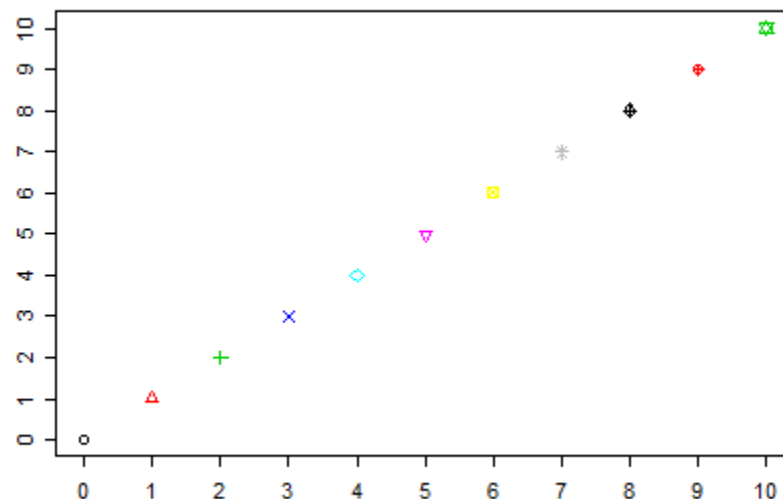
加入圖形元件 - 點線框

- `x <- c(2, 3, 4)`
- `y <- c(6, 7, 8)`

- `windows(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 4, 4, 2) + 0.1)`

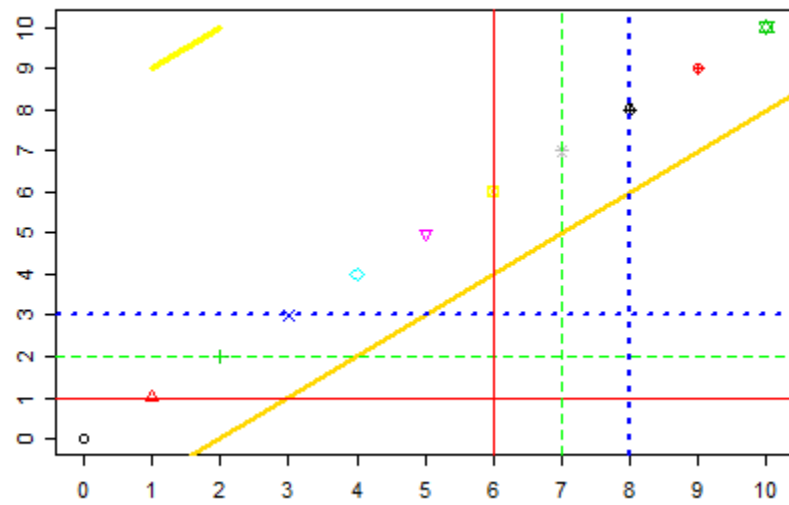
- `plot(0:10, 0:10, xlab = "", ylab = "", pch = 1:11, col = 1:8, axes = FALSE)`

- `axis(side = 1, at = 0:10)`
- `axis(side = 2, at = 0:10)`
- `box()`



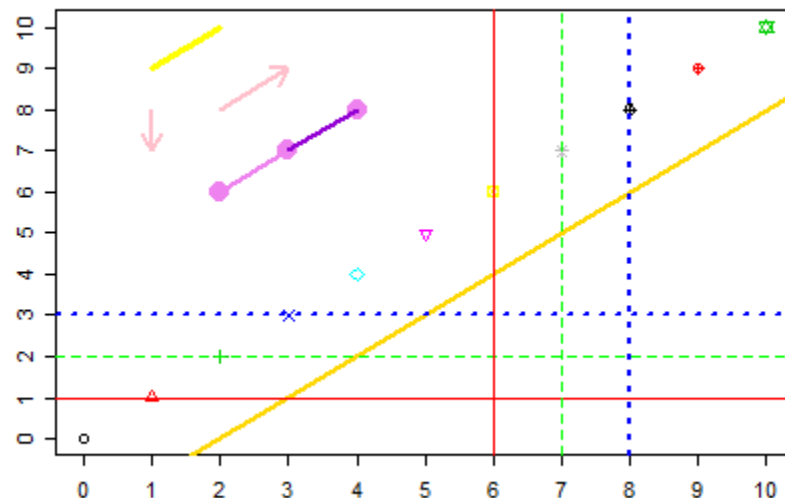
加入圖形元件 - 點 線 框

- `lines(c(1, 2), c(9, 10), col = "yellow", lwd = 3)`
- `abline(a = -2, b = 1, col = "gold", lwd = 2)`
- `# horizontal line`
- `abline(h = 1:3, lty = 1:3, lwd = c(1.0, 1.5, 2.0), col = c("red", "green", "blue")))`
- `# vertical lines`
- `abline(v = 6:8, lty = 1:3, lwd = c(1.0, 1.5, 2.0), col = c("red", "green", "blue")))`



加入圖形元件 - 點 線 框

- `points(x, y, col = "violet", pch = 16, cex = 2)`
- `segments(x[1], y[1], x[2], y[2], col = "violet", lwd = 2)`
- `segments(x[2], y[2], x[3], y[3], col = "dark violet", lwd = 2)`
- `arrows(1, 7, 1, 8, code = 1, length = 0.1, col = "pink", lwd = 2)`
- `arrows(2, 8, 3, 9, code = 2, length = 0.1, col = "pink", lwd = 2)`
- `par(old.par)`



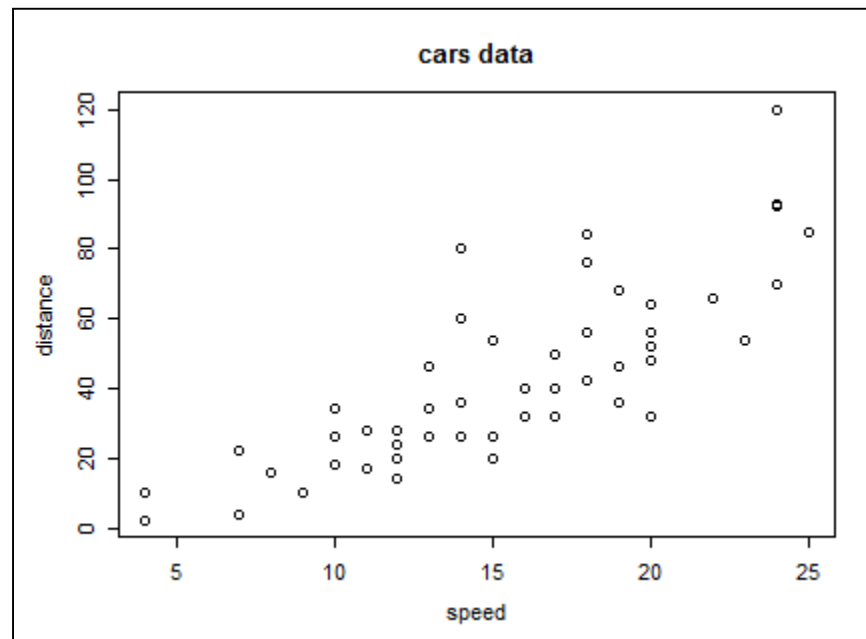
加入文字

加入文字

- **text:** # 加入文字
- **title:** # 加入註解
- **legend:** # 加入圖標，圖例，備註
- **mtext:** # 在邊界加入文字

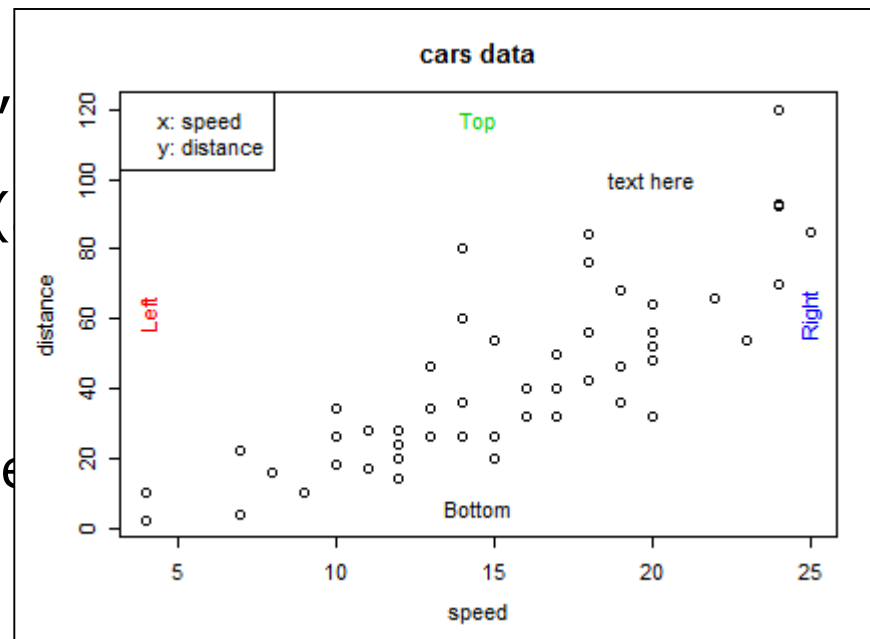
加入文字

- `windows(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 5, 4, 2) + 0.1)`
- `plot(cars, xlab = "", ylab = "")`
- `title(main = "cars data", xlab = "speed", ylab = "distance")`



加入文字

- `windows(width = 4.5, height = 3.3,`
- `old.par <- par(mex = 0.8, mar = c(`
- `plot(cars, xlab = "", ylab = "")`
- `title(main = "cars data", xlab = "spe`
- `text(20, 100, label = "text here")`
- `legend("topleft", legend = c("x: speed", "y: distance"))`
- `mtext(text = c("Bottom", "Left", "Top", "Right"), side = 1:4, col = 1:4,`
`line = -2)`
- `par(old.par)`



line = -2: 往內移動兩個線寬度

- legend 位置：

topleft,

top,

topright

left,

center,

right

bottomleft,

bottom,

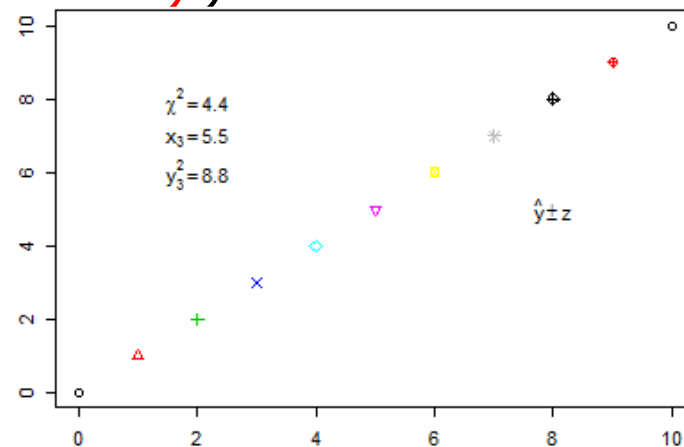
bottomright

加入文字 - 用滑鼠

- `windows(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 5, 4, 2) + 0.1)`
- `plot(cars, xlab = "", ylab = "")`
- `title(main = "cars data", xlab = "speed", ylab = "distance")`
- `text(20, 100, label = "text here")`
- `legend(locator(1), legend = c("x: speed", "y: distance"))`
- `mtext(text = c("Bottom", "Left", "Top", "Right"), side = 1:4, col = 1:4, line = -2)`
- `par(old.par)`

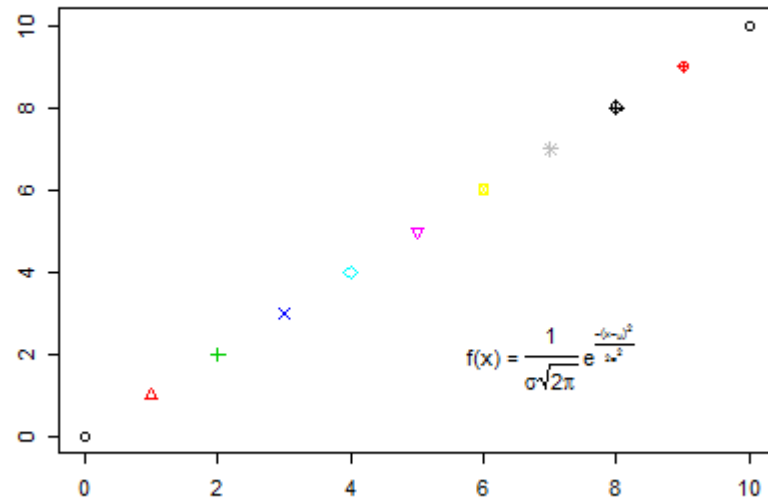
加入數學符號

- `windows(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 5, 4, 2) + 0.1)`
- `plot(0:10, 0:10, xlab = "", ylab = "", pch = 1:10, col = 1:10)`
- `text(2, 8, label = expression(chi^2 == "4.4"))`
- `text(2, 7, label = expression(x[3] == "5.5"))`
- `text(2, 6, label = expression(y[3]^2 == "8.8"))`
- `text(8, 5, label = expression(hat(y) %+-% z))`



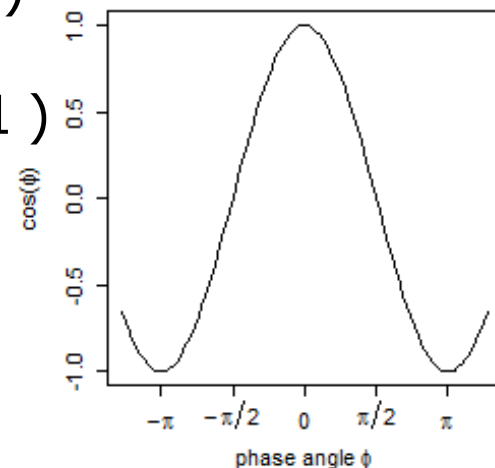
加入數學符號

- `label.eq <- expression(paste("f(x) = ", frac(1, sigma*sqrt(2 * pi)), " ", e^{frac(-(x - mu)^2, 2*sigma^2)}))`
- `text(7, 2, label = label.eq)`
- `par(old.par)`



加入數學符號

- `x <- seq(from = -4, to = 4, length = 101)`
- `win.graph(width = 2.8, height = 2.8, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 5, 3, 2) + 0.1)`
- `xlab <- expression(paste("phase angle ", phi))`
- `ylab <- expression(paste("cos(", phi, ")"))`
- `plot(x, cos(x), type = "l", xaxt = "n", xlab = xlab, ylab = ylab)`
- `label <- expression(-pi, -pi / 2, 0, pi / 2, pi)`
- `axis(side = 1, at = c(-pi, -pi / 2, 0, pi / 2, pi), label = label)`
- `par(old.par)`



數學符號 expression()

- ?plotmath
- demo(plotmath)

字型之圖形參數

- **font:** # 文字及符號之字型
- **font.axis:** # 座標軸數字，文字及符號之字型
- **font.lab:** # 座標軸標記之字型
- **font.mian:** # 主標題文字及符號之字型
- **font.sub:** # 副標題文字及符號之字型

- demo(Hershey)

Graphical Parameters Cheat Sheet

par() Graphical Parameters

Visual cheat sheet for some plot parameters in R. See `?par` for more information.

Symbol Styles

pch | Point Types

- 1
- △ 2
- +
- × 4
- ◇ 5
- ▽ 6
- ⊠ 7
- * 8
- ⊕ 9
- ⊗ 10
- ⊠ 11
- ⊠ 12
- ⊠ 13

lty | Line Types

- 1
- - - 2
- ⋯ 3
- · - · 4
- - - 5
- · - · 6

lwd | Line Width

- .1
- .25
- .5
- 1
- 3
- 6

Figures Arrangement

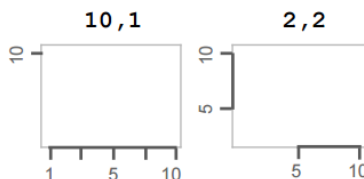
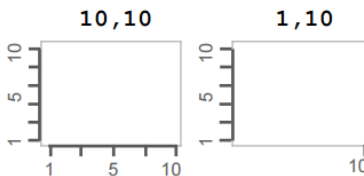
mflow | Multiple Figures by Row



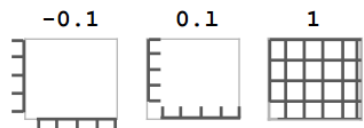
Also available `mfcoll` for multiple figures by column

Axes

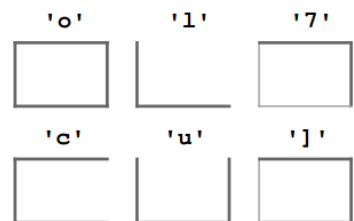
lab | Tick Placement



tck | Tick Length



bty | Box Type



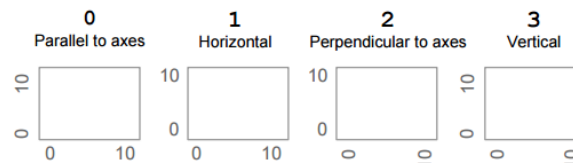
Text and Labels

family, font | Typeface and Font Style

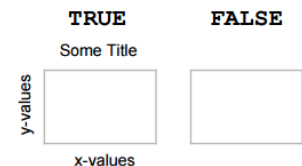
- | | | |
|-------------------------|--------------------------|-------------------------|
| family: mono
font: 1 | family: serif
font: 1 | family: sans
font: 1 |
| family: mono
font: 2 | family: serif
font: 2 | family: sans
font: 2 |
| family: mono
font: 3 | family: serif
font: 3 | family: sans
font: 3 |
| family: mono
font: 4 | family: serif
font: 4 | family: sans
font: 4 |

Also available: `font.main` (main title), `font.lab` (axis labels), `font.sub` (subtitle)

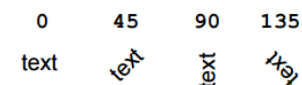
las | Label Orientation



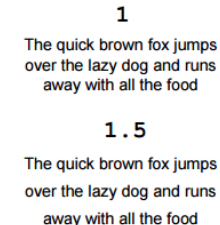
ann | Plot Annotation



srt | String Rotation



lheight | Line Height



Based on *Flowing Data's cheat sheet*