

105-1: EE4052
計算機程式設計
Computer Programming

Unit 10: 多重繪圖與顏色

連 豐 力

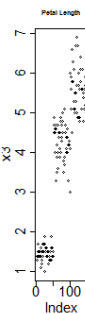
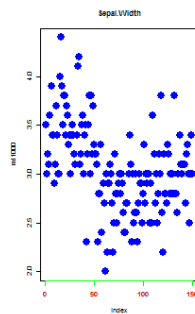
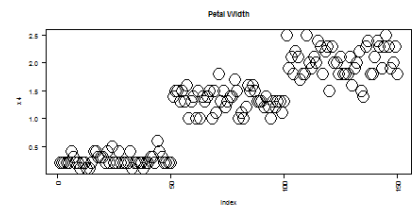
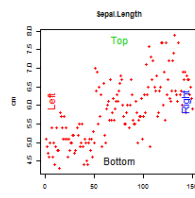
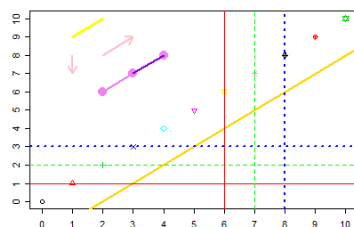
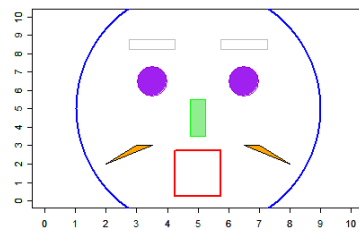
臺大電機系

Sep 2016 - Jan 2017

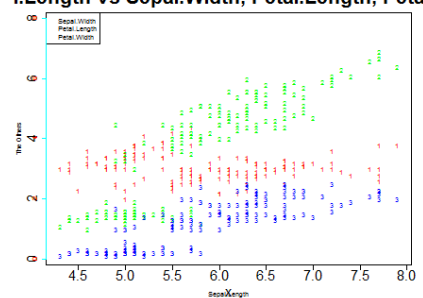
Unit 10: 多重繪圖與顏色

計算機程式設計 - 2016F
Chap 10: 多重繪圖與顏色
Feng-Li Lian @ NTU-EE

- 一頁之中，至少放了多張的圖
- 每張圖的長寬或大小，不一樣
- 主標題，軸標題，字體要有變化
- 數據點的顏色要有變化
- 數據點的形式要有變化
- 不同數據加上不同註解



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



- 繪圖視窗之設定
- 常用的圖形參數
- 座標軸及邊界
- 加入文字
- 多張圖形
- 多張圖形之位置安排
- 加入圖形元件
- 顏色
- 數學函數繪圖

多張圖形

多張圖形 – mfrow, mfcoll

計算機程式設計 – 2016F
Chap 10: 多重繪圖與顏色
Feng-Li Lian @ NTU-EE

- `mfrow()` # 依照橫列 (by row) 順序畫出
- `mfcoll()` # 依照直行 (by column) 順序畫出
- `cex.axis:` # 座標軸數字，文字及符號相對於內定值之縮放比
- `cex.lab:` # 座標軸標記文字及符號相對於內定值之縮放比
- `cex.main:` # 主標題 (上標題) 文字及符號相對於內定值之縮放比
- `cex.sub:` # 副標題 (下標題) 文字及符號相對於內定值之縮放比

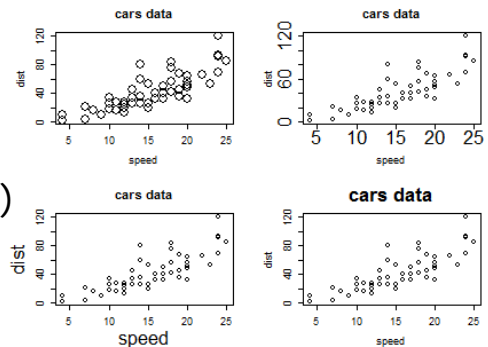
- 5

多張圖形 – mfrow, mfcoll

計算機程式設計 – 2016F
Chap 10: 多重繪圖與顏色
Feng-Li Lian @ NTU-EE

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

- `plot(cars, main = "cars data", cex = 2)`
- `plot(cars, main = "cars data", cex.axis = 2)`
- `plot(cars, main = "cars data", cex.lab = 2)`
- `plot (cars, main = "cars data", cex.main = 2)`



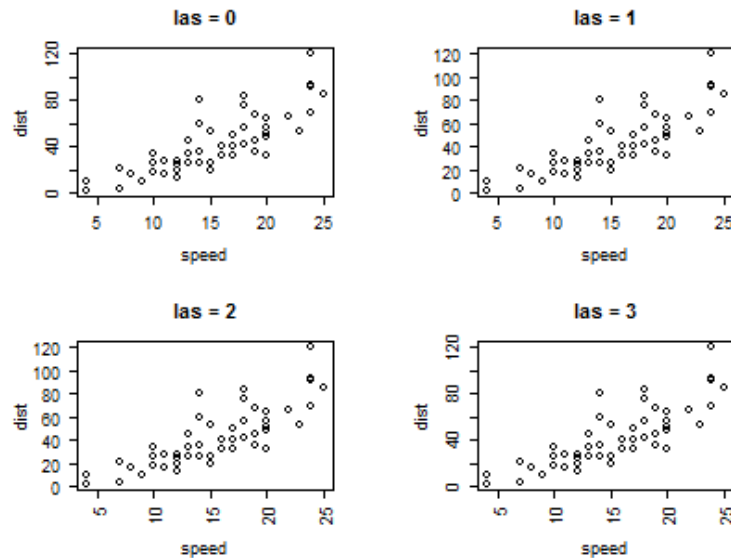
- `par(old.par)`

- 6

- `las`: # 座標軸數字，文字之展現方式
- `las = 0` # 0: always parallel to the axis [default]
- `las = 1` # 1: always horizontal
- `las = 2` # 2: always perpendicular to the axis
- `las = 3` # 3: always vertical

- `windows(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mfrow = c(2, 2), mex = 0.8, mar = c(5, 5, 4, 4) + 0.1)`
- `plot(cars, main = "las = 0", las = 0)`
0: always parallel to the axis [default]
- `plot(cars, main = "las = 1", las = 1)`
1: always horizontal
- `plot(cars, main = "las = 2", las = 2)`
2: always perpendicular to the axis
- `plot(cars, main = "las = 3", las = 3)`
3: always vertical
- `par(old.par)`

- `plot(cars, main = "las = 0", las = 0)` # 0: always parallel to the axis [default]
- `plot(cars, main = "las = 1", las = 1)` # 1: always horizontal
- `plot(cars, main = "las = 2", las = 2)` # 2: always perpendicular to the axis
- `plot(cars, main = "las = 3", las = 3)` # 3: always vertical



- 9

- `type:` # 點跟點之間的展現方式
- `type = "p"` # "p" for points
- `type = "l"` # "l" for lines
- `type = "b"` # "b" for both
- `type = "o"` # "o" for both 'overplotted'
- `type = "h"` # "h" for histogram-like vertical lines
- `type = "n"` # "n" for no plotting

- 10

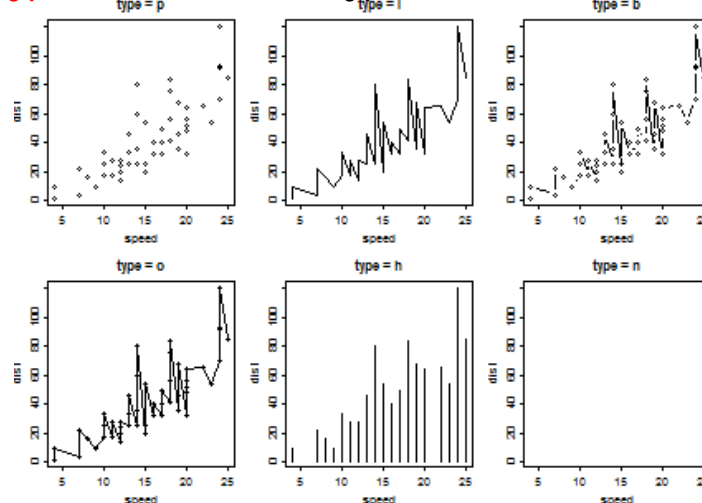
多張圖形 - type

- `windows(width = 4.5, height = 3.3, fontsize = 8)`
- `old.par <- par(mfrow = c(2, 3), mex = 0.6, mar = c(5, 4, 4, 2) + 0.1)`
- `plot(cars, type = "p", main = "type = p")` # "p" for points
- `plot(cars, type = "l", main = "type = l")` # "l" for lines
- `plot(cars, type = "b", main = "type = b")` # "b" for both
- `plot(cars, type = "o", main = "type = o")` # "o" for both 'overplotted'
- `plot(cars, type = "h", main = "type = h")` # "h" for histogram-like vertical lines
- `plot(cars, type = "n", main = "type = n")` # "n" for no plotting
- `par(old.par)`

- 11

多張圖形 - type

- `plot(cars, type = "p", main = "type = p")` # "p" for points
- `plot(cars, type = "l", main = "type = l")` # "l" for lines
- `plot(cars, type = "b", main = "type = b")` # "b" for both
- `plot(cars, type = "o", main = "type = o")` # "o" for both 'overplotted'
- `plot(cars, type = "h", main = "type = h")` # "h" for histogram-like vertical lines
- `plot(cars, type = "n", main = "type = n")` # "n" for no plotting



- 12

多張圖形之位置安排

13

繪圖 - 圖形位置安排

- `layout(M, widths, heights)`
- `M` 是圖形分佈的矩陣，
- `widths`、`heights` 各是設定 `M` 矩陣長、寬的比例，其基準點是左上角

`matrix(c(1, 2, 3, 4) , 2, 2, byrow = T)`

1	2
3	4

`widths = c(1,1), heights = c(1,1))`

`matrix(c(1, 2, 3, 4, 5, 6) , 3, 2, byrow = T)`

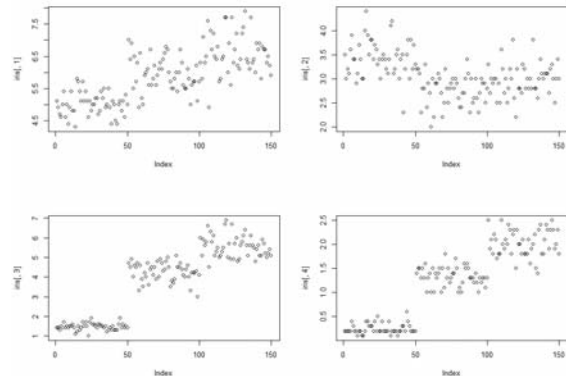
1	2
3	4
5	6

1	2
3	4

`widths = c(1,3), heights = c(1,2))`

繪圖 - 圖形位置安排

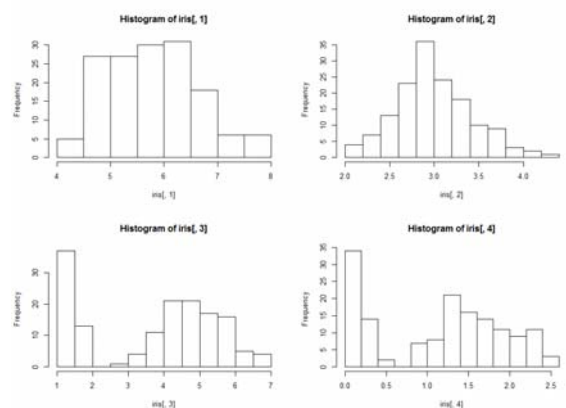
- `layout(M, widths, heights)`
- M是圖形分佈的矩陣，
- `widths`、`heights`各是設定M矩陣長、寬的比例，其基準點是左上角
- # 幾張圖繪製在一起
- `layout(matrix(c(1, 2, 3, 4), 2, 2, byrow = T), widths = c(1,1), heights = c(1,1))`
- `plot(iris[, 1])`
- `plot(iris[, 2])`
- `plot(iris[, 3])`
- `plot(iris[, 4])`



15

繪圖 - 圖形位置安排

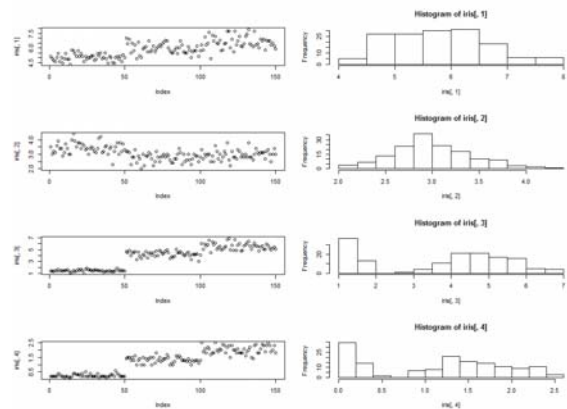
- `layout(M, widths, heights)`
- M是圖形分佈的矩陣，
- `widths`、`heights`各是設定M矩陣長、寬的比例，其基準點是左上角
- # 幾張圖繪製在一起
- `layout(matrix(c(1, 2, 3, 4), 2, 2, byrow = T), widths = c(1,1), heights = c(1,1))`
- `hist(iris[, 1])`
- `hist(iris[, 2])`
- `hist(iris[, 3])`
- `hist(iris[, 4])`



16

繪圖 - 圖形位置安排

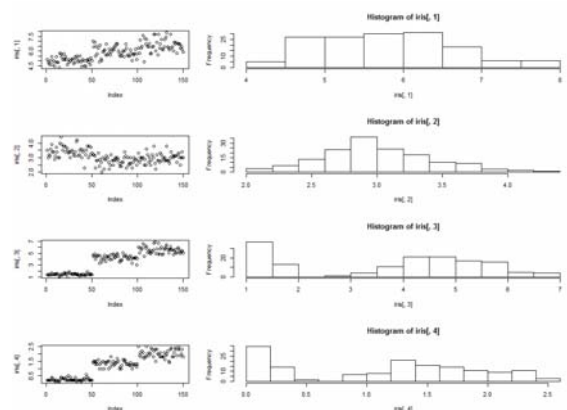
- `layout(M, widths, heights)`
- M是圖形分佈的矩陣，
- widths、heights各是設定M矩陣長、寬的比例，其基準點是左上角
- # 幾張圖繪製在一起
- `layout(matrix(c(1, 2, 3, 4, 5, 6, 7, 8), 4, 2, byrow = T), widths = c(1,1), heights = c(1,1,1,1))`
- `plot(iris[, 1])`
- `hist(iris[, 1])`
- `plot(iris[, 2])`
- `hist(iris[, 2])`
- `plot(iris[, 3])`
- `hist(iris[, 3])`
- `plot(iris[, 4])`
- `hist(iris[, 4])`



17

繪圖 - 圖形位置安排

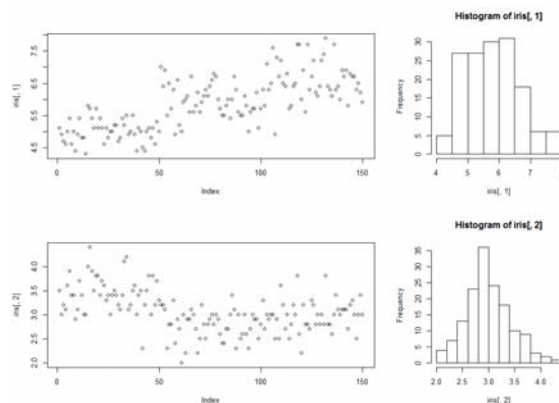
- `layout(M, widths, heights)`
- M是圖形分佈的矩陣，
- widths、heights各是設定M矩陣長、寬的比例，其基準點是左上角
- # 幾張圖繪製在一起
- `layout(matrix(c(1, 2, 3, 4, 5, 6, 7, 8), 4, 2, byrow = T), widths = c(1,2), heights = c(1,1,1,1))`
- `plot(iris[, 1])`
- `hist(iris[, 1])`
- `plot(iris[, 2])`
- `hist(iris[, 2])`
- `plot(iris[, 3])`
- `hist(iris[, 3])`
- `plot(iris[, 4])`
- `hist(iris[, 4])`



18

繪圖 - 圖形位置安排

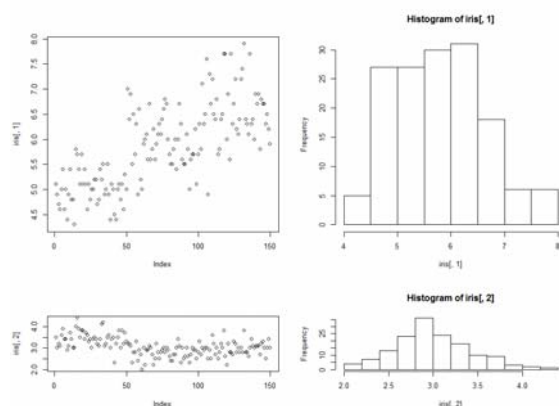
- `layout(M, widths, heights)`
- M是圖形分佈的矩陣，
- widths、heights各是設定M矩陣長、寬的比例，其基準點是左上角
- # 幾張圖繪製在一起
- `layout(matrix(c(1, 2, 3, 4), 2, 2, byrow = T), widths = c(2,1), heights = c(1,1))`
- `plot(iris[, 1])`
- `hist(iris[, 1])`
- `plot(iris[, 2])`
- `hist(iris[, 2])`



- 19

繪圖 - 圖形位置安排

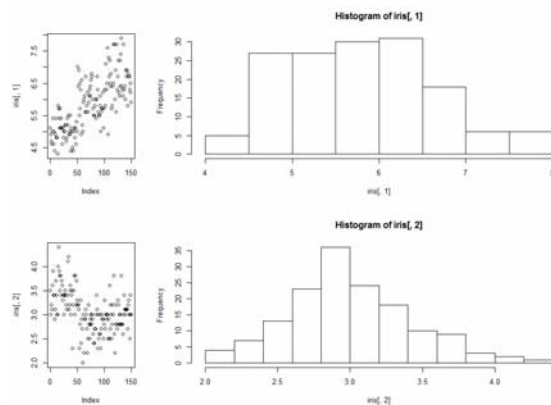
- `layout(M, widths, heights)`
- M是圖形分佈的矩陣，
- widths、heights各是設定M矩陣長、寬的比例，其基準點是左上角
- # 幾張圖繪製在一起
- `layout(matrix(c(1, 2, 3, 4), 2, 2, byrow = T), widths = c(1,1), heights = c(2,1))`
- `plot(iris[, 1])`
- `hist(iris[, 1])`
- `plot(iris[, 2])`
- `hist(iris[, 2])`



- 20

繪圖 - 圖形位置安排

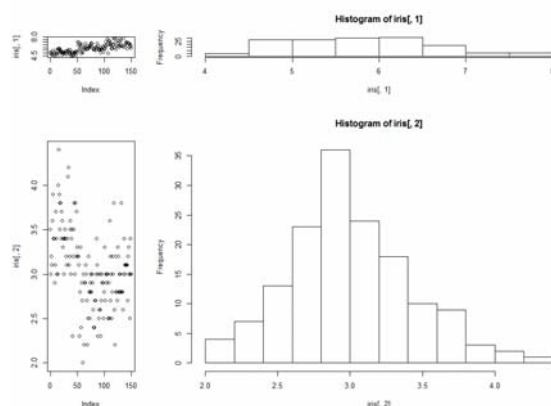
- `layout(M, widths, heights)`
- M是圖形分佈的矩陣，
- widths、heights各是設定M矩陣長、寬的比例，其基準點是左上角
- # 幾張圖繪製在一起
- `layout(matrix(c(1, 2, 3, 4), 2, 2, byrow = T), widths = c(1,3), heights = c(1,1))`
- `plot(iris[, 1])`
- `hist(iris[, 1])`
- `plot(iris[, 2])`
- `hist(iris[, 2])`



- 21

繪圖 - 圖形位置安排

- `layout(M, widths, heights)`
- M是圖形分佈的矩陣，
- widths、heights各是設定M矩陣長、寬的比例，其基準點是左上角
- # 幾張圖繪製在一起
- `layout(matrix(c(1, 2, 3, 4), 2, 2, byrow = T), widths = c(1,3), heights = c(1,3))`
- `plot(iris[, 1])`
- `hist(iris[, 1])`
- `plot(iris[, 2])`
- `hist(iris[, 2])`

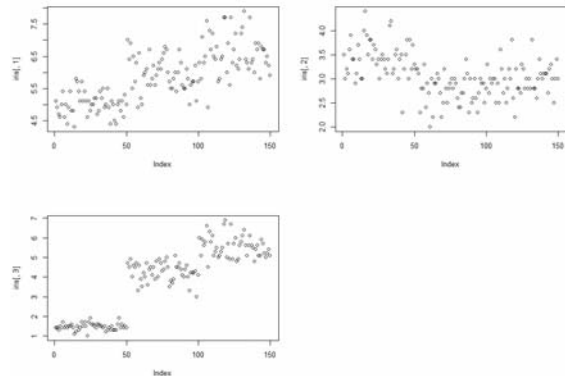


- 22

繪圖 - 圖形位置安排

計算機程式設計 - 2016F
Chap 10: 多重繪圖與顏色
Feng-Li Lian @ NTU-EE

- `layout(M, widths, heights)`
- M是圖形分佈的矩陣，
- widths、heights各是設定M矩陣長、寬的比例，其基準點是左上角
- # 幾張圖繪製在一起
- `layout(matrix(c(1, 2, 3, 0) , 2, 2, byrow = T), widths = c(1,1), heights = c(1,1))`
- `plot(iris[, 1])`
- `plot(iris[, 2])`
- `plot(iris[, 3])`

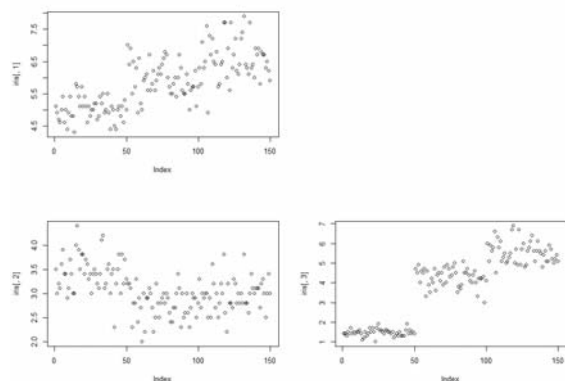


- 23

繪圖 - 圖形位置安排

計算機程式設計 - 2016F
Chap 10: 多重繪圖與顏色
Feng-Li Lian @ NTU-EE

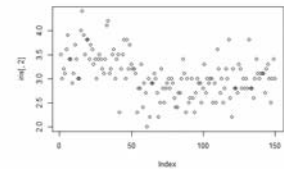
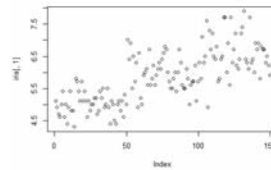
- `layout(M, widths, heights)`
- M是圖形分佈的矩陣，
- widths、heights各是設定M矩陣長、寬的比例，其基準點是左上角
- # 幾張圖繪製在一起
- `layout(matrix(c(1, 0, 2, 3) , 2, 2, byrow = T), widths = c(1,1), heights = c(1,1))`
- `plot(iris[, 1])`
- `plot(iris[, 2])`
- `plot(iris[, 3])`



- 24

繪圖 - 圖形位置安排

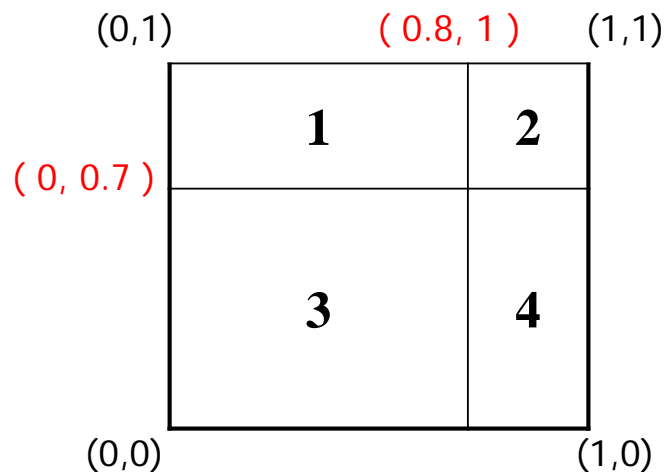
- `layout(M, widths, heights)`
- M是圖形分佈的矩陣，
- `widths`、`heights`各是設定M矩陣長、寬的比例，其基準點是左上角
- # 幾張圖繪製在一起
- `layout(matrix(c(1, 0, 0, 2), 2, 2, byrow = T), widths = c(1,1), heights = c(1,1))`
- `plot(iris[, 1])`
- `plot(iris[, 2])`



- 25

繪圖 - 圖形位置安排

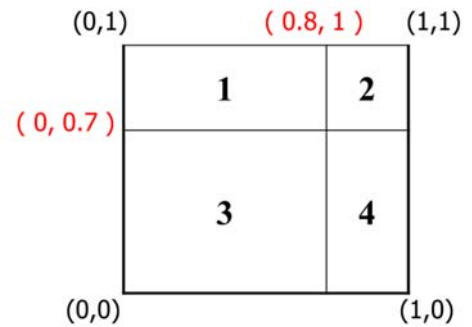
- `par(fig = c(x1, x2, y1, y2))`
- `par(fig = c(0, 0.8, 0.7, 1))`
圖1 的 左下角座標 $(x1,y1)$ 是 $(0, 0.7)$,
圖1 的 右上角座標 $(x2,y2)$ 是 $(0.8, 1)$)



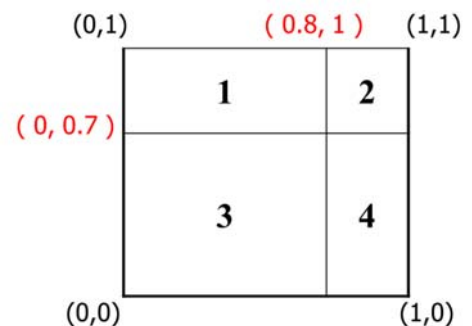
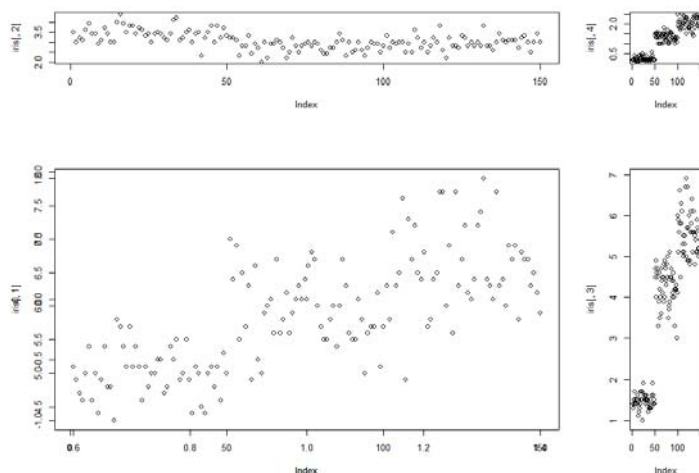
- 26

繪圖 - 圖形位置安排

- `par(fig = c(x1, x2, y1, y2))`
- `par(fig = c(0, 0.8, 0.7, 1))`
 圖1 的 左下角座標 (x1,y1) 是 (0, 0.7) ,
 圖1 的 右上角座標 (x2,y2) 是 (0.8, 1)
- # 幾張圖繪製在一起
- `par(fig=c(0, 0.8, 0, 0.7), new=TRUE)`
- `plot(iris[, 1])`
- `par(fig=c(0, 0.8, 0.7, 1), new=TRUE)`
- `plot(iris[, 2])`
- `par(fig=c(0.8, 1, 0, 0.7), new=TRUE)`
- `plot(iris[, 3])`
- `par(fig=c(0.8, 1, 0.7, 1), new=TRUE)`
- `plot(iris[, 4])`



繪圖 - 圖形位置安排



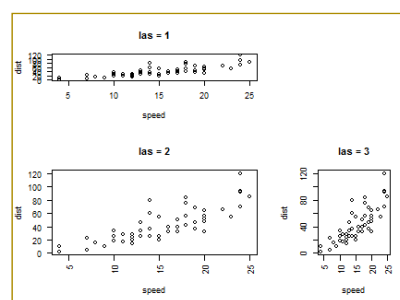
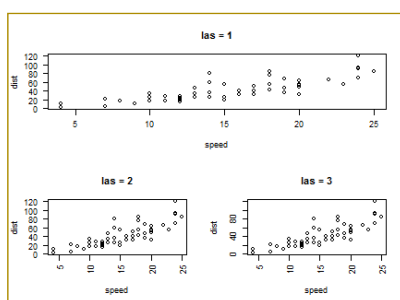
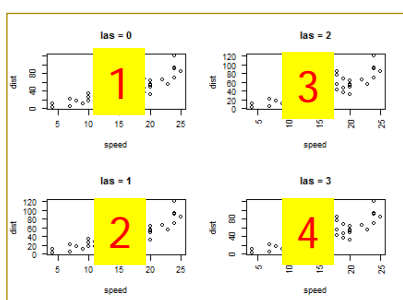
多張圖形 - layout

- `layout(matrix(1:4, nrow = 2))`

2x2 的圖形矩陣，依照指定的位置

- `layout(matrix(c(1, 2, 1, 3), nrow = 2, ncol = 2))`

- `layout(matrix(c(1, 2, 0, 3), nrow = 2, ncol = 2), width = c(2, 1), height = c(1, 1.5))`



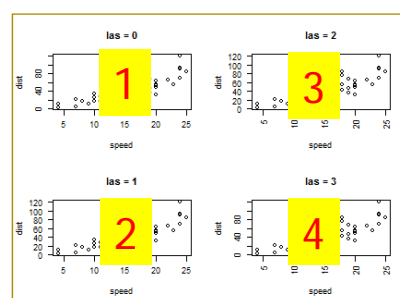
- 29

多張圖形 - layout

- `windows(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 4, 4, 2) + 0.1)`
- `layout(matrix(1:4, nrow = 2))`

- `plot(cars, las = 0, main = "las = 0")`
- `plot(cars, las = 1, main = "las = 1")`
- `plot(cars, las = 2, main = "las = 2")`
- `plot(cars, las = 3, main = "las = 3")`

- `layout(1)`
- `par(old.par)`



- 30

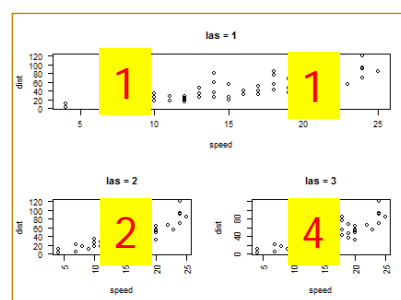
多張圖形 – layout

計算機程式設計 – 2016F
Chap 10: 多重繪圖與顏色
Feng-Li Lian @ NTU-EE

- `windows(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 4, 4, 2) + 0.1)`
- `layout(matrix(c(1, 2, 1, 3), nrow = 2, ncol = 2))`

- `plot(cars, las = 1, main = "las = 1")`
- `plot(cars, las = 2, main = "las = 2")`
- `plot(cars, las = 3, main = "las = 3")`

- `layout(1)`
- `par(old.par)`



- 31

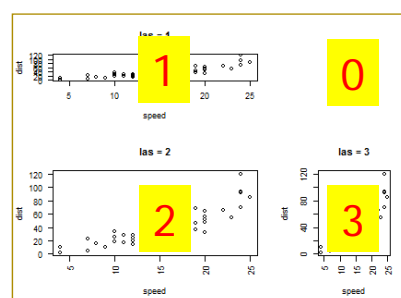
多張圖形 – layout

計算機程式設計 – 2016F
Chap 10: 多重繪圖與顏色
Feng-Li Lian @ NTU-EE

- `windows(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 4, 4, 2) + 0.1)`
- `layout(matrix(c(1, 2, 0, 3), nrow = 2, ncol = 2), width = c(2, 1), height = c(1, 1.5))`

- `plot(cars, las = 1, main = "las = 1")`
- `plot(cars, las = 2, main = "las = 2")`
- `plot(cars, las = 3, main = "las = 3")`

- `layout(1)`
- `par(old.par)`



- 32

加入圖形元件

33



加入圖形元件 - 點線框

計算機程式設計 - 2016F
Chap 10: 多重繪圖與顏色
Feng-Li Lian @ NTU-EE

- `points()` # 打點
- `lines()` # 畫線
- `abline()` # 畫 $y = ax + b$ 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

- 34

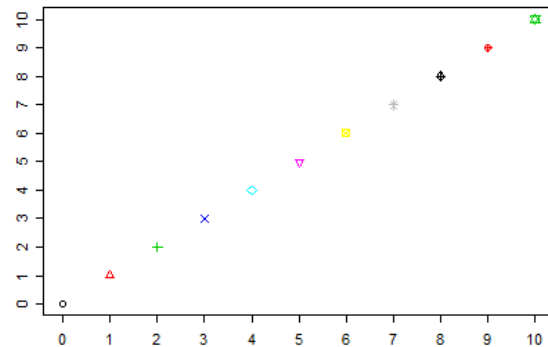
加入圖形元件 - 點線框

- `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)`

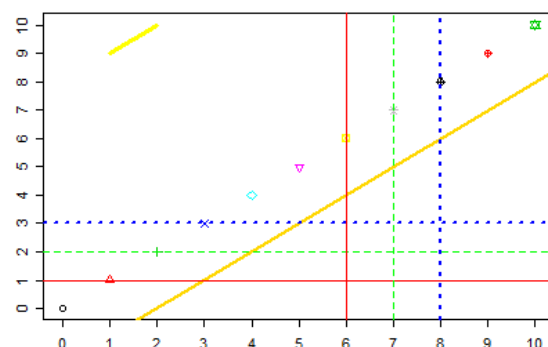
- `axis(side = 1, at = 0:10)`
- `axis(side = 2, at = 0:10)`
- `box()`



- 35

加入圖形元件 - 點線框

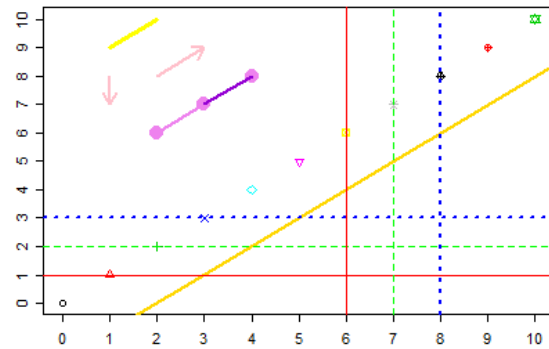
- `lines(c(1, 2), c(9, 10), col = "yellow", lwd = 3)`
- `abline(a = -2, b = 1, col = "gold", lwd = 2)`
- `# horizontal line`
- `abline(h = 1:3, lty = 1:3, lwd = c(1.0, 1.5, 2.0), col = c("red", "green", "blue"))`
- `# vertical lines`
- `abline(v = 6:8, lty = 1:3, lwd = c(1.0, 1.5, 2.0), col = c("red",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



- 36

加入圖形元件 - 點線框

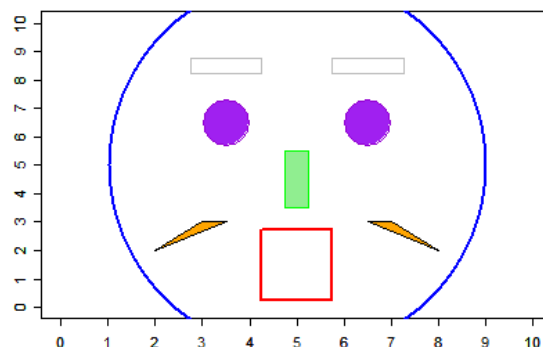
- `points(x, y, col = "violet", pch = 16, cex = 2)`
- `segments(x[1], y[1], x[2], y[2], col = "violet", lwd = 2)`
- `segments(x[2], y[2], x[3], y[3], col = "dark violet", lwd = 2)`
- `arrows(1, 7, 1, 8, code = 1, length = 0.1, col = "pink", lwd = 2)`
- `arrows(2, 8, 3, 9, code = 2, length = 0.1, col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



- 37

加入圖形元件 - 圓形 矩形 多邊形

- `symbols()` # 圓形 矩形
- `polygon()` # 多邊形



- 38

- `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, type = "n")`
- `axis(side = 1, at = 0:10)`
- `axis(side = 2, at = 0:10)`

- # 臉輪廓
- `symbols(5, 5, circles = 4, fg = "blue", lwd = 2, inches = FALSE, add = TRUE)`

- # 眼睛
- `symbols(3.5, 6.5, circles = 0.5, fg = "dark violet", bg = "purple", inches = FALSE, add = TRUE)`
- `symbols(6.5, 6.5, circles = 0.5, fg = "dark violet", bg = "purple", inches = FALSE, add = TRUE)`

- # 眉毛
- `symbols(3.5, 8.5, rectangles = matrix(c(1.5, 0.5), nrow = 1), fg = "gray", inches = FALSE, add = TRUE)`
- `symbols(6.5, 8.5, rectangles = matrix(c(1.5, 0.5), nrow = 1), fg = "gray", inches = FALSE, add = TRUE)`

加入圖形元件 - 圓形 矩形 多邊形

計算機程式設計 - 2016F
Chap 10: 多重繪圖與顏色
Feng-Li Lian @ NTU-EE

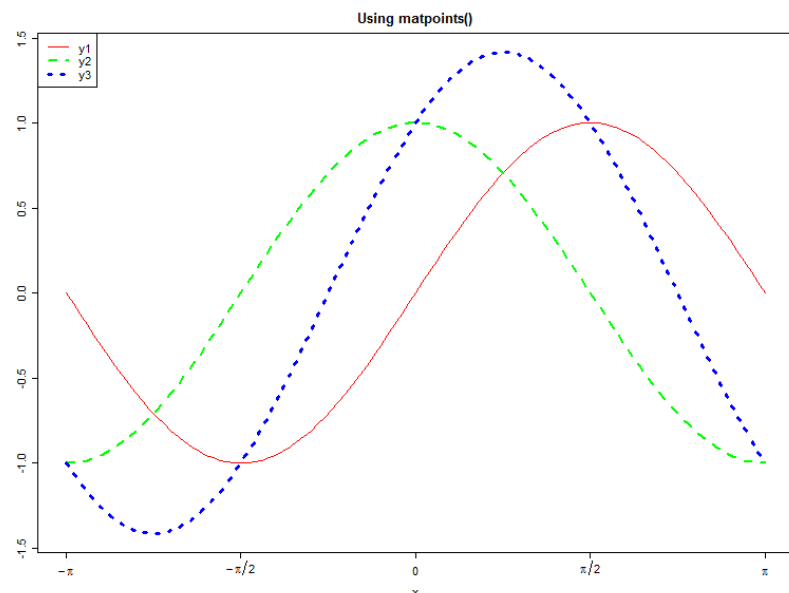
- # 鼻子
- `symbols(5, 4.5, rectangles = matrix(c(0.5, 2), nrow = 1), fg = "green", bg = "lightgreen", inches = FALSE, add = TRUE)`
- # 嘴巴
- `symbols(5, 1.5, squares = 1.5, fg = "red", lwd = 2, inches = FALSE, add = TRUE)`
- # 鬍子
- `polygon(c(3.5, 3, 2), c(3, 3, 2), col = "orange")`
- `polygon(c(6.5, 7, 8), c(3, 3, 2), col = "orange")`
- `par(old.par)`

- 41

呈現多筆數據於一張圖

計算機程式設計 - 2016F
Chap 10: 多重繪圖與顏色
Feng-Li Lian @ NTU-EE

- `matplot()`
- `matpoints()`
- `matlines()`



- 42

呈現多筆數據於一張圖

計算機程式設計 - 2016F
Chap 10: 多重繪圖與顏色
Feng-Li Lian @ NTU-EE

- `x <- seq(from = -pi, to = pi, length = 101)`
- `y1 <- sin(x)`
- `y2 <- cos(x)`
- `y3 <- sin(x) + cos(x)`

- `ylim <- range(y1, y2, y3)`

- `win.graph(width = 8, height = 6, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 4, 3, 1) + 0.1)`

- 43

呈現多筆數據於一張圖

計算機程式設計 - 2016F
Chap 10: 多重繪圖與顏色
Feng-Li Lian @ NTU-EE

- `plot(x, y1, xlim = range(x), ylim = ylim, type = "n", xaxt = "n", xlab = "x", ylab = "", main = "Using matpoints()")`

- `label <- expression(-pi, -pi / 2, 0, pi / 2, pi)`
- `axis(side = 1, at = c(-pi, -pi / 2, 0, pi / 2, pi), label = label)`

- `matpoints(x, cbind(y1, y2, y3), type = "l", col = c("red", "green", "blue"), lty = 1:3, lwd = 1:3)`

- `legend("topleft", legend = c("y1", "y2", "y3"), col = c("red", "green", "blue"), lty = 1:3, lwd = 1:3)`

- `par(old.par)`

- 44

- `win.graph(width = 8, height = 6, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 4, 3, 1) + 0.1)`

- `plot(x, y1, xlim = range(x), ylim = ylim, type = "n", xaxt = "n", xlab = "x", ylab = "", main = "Using matlines()")`

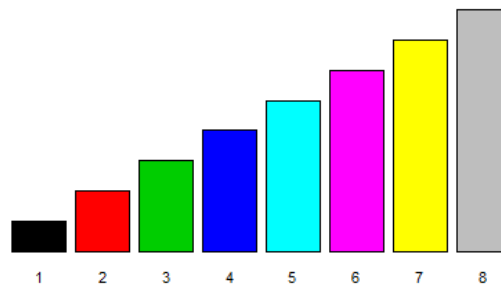
- `label <- expression(-pi, -pi / 2, 0, pi / 2, pi)`
- `axis(side = 1, at = c(-pi, -pi / 2, 0, pi / 2, pi), label = label)`
- `matlines(x, cbind(y1, y2, y3), col = c("red", "green", "blue"), lty = 1:3, lwd = 1:3)`
- `legend("topleft", legend = c("y1", "y2", "y3"), col = c("red", "green", "blue"), lty = 1:3, lwd = 1:3)`

- `par(old.par)`

- 45

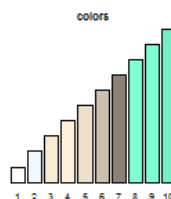
顏色

- `palette()`
- `windows(width = 4.5, height = 2.5, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(4, 2, 2, 2) + 0.1)`
- `barplot(1:8, col = palette(), names.arg = 1:8, yaxt = "n")`
- `par(old.par)`



- 47

- `n <- 10`
- `windows(width = 5, height = 3.3, pointsize = 8)`
- `old.par <- par(mfrow = c(2, 3), mex = 0.6, mar = c(5, 4, 4, 2) + 0.1)`
- `barplot(1:n, col = colors()[1:n], names.arg = 1:n, yaxt = "n", main = "colors")`



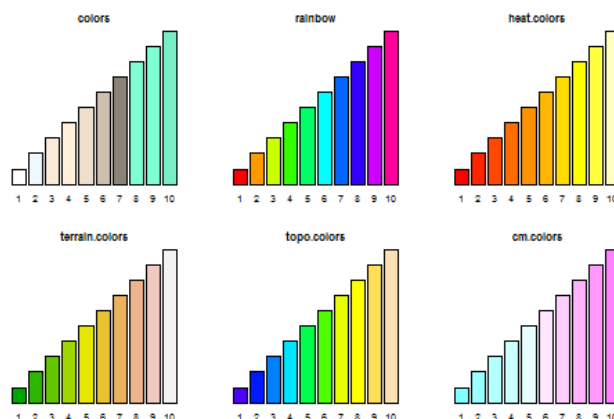
- 48

- `barplot(1:n, col = colors()[1:n], names.arg = 1:n, yaxt = "n", main = "colors")`
- `barplot(1:n, col = rainbow(n), names.arg = 1:n, yaxt = "n", main = "rainbow")`
- `barplot(1:n, col = heat.colors(n), names.arg = 1:n, yaxt = "n", main = "heat.colors")`
- `barplot(1:n, col = terrain.colors(n), names.arg = 1:n, yaxt = "n", main = "terrain.colors")`
- `barplot(1:n, col = topo.colors(n), names.arg = 1:n, yaxt = "n", main = "topo.colors")`
- `barplot(1:n, col = cm.colors(n), names.arg = 1:n, yaxt = "n", main = "cm.colors")`

cm: cyan-magenta

- `par(old.par)`

- 49



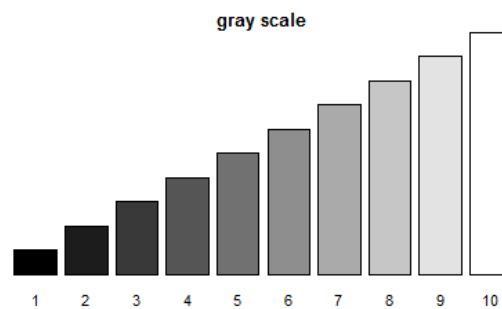
- 50

顏色 - 灰階圖

- `gray.scale <- seq(from = 0, to = 1, length = 10)`
- `windows(width = 4.5, height = 2.5, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(4, 2, 2, 2) + 0.1)`
- `barplot(1:10, col = gray(gray.scale), names.arg = 1:10, yaxt = "n", main = "gray scale")`

From 0 (black) -> 1 (white)

- `par(old.par)`

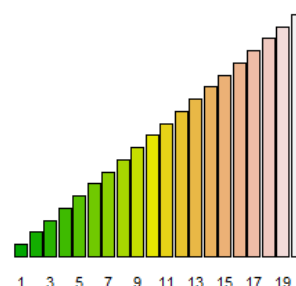


- 51

顏色 - 自行定義調色盤

- `palette(terrain.colors(20))` # redefine palette
- `palette()`
- `windows(width = 2.8, height = 2.5, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(4, 2, 2, 2) + 0.1)`
- `barplot(1:20, col = 1:20, names.arg = 1:20, yaxt = "n")`

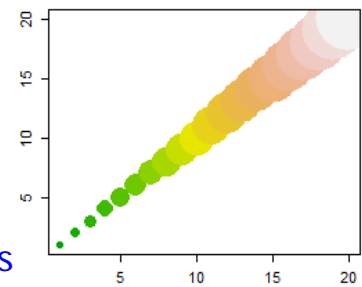
- `par(old.par)`



- 52

顏色 - 自行定義調色盤

- `palette(terrain.colors(20))` # redefine palette
- `palette()`
- `windows(width = 2.8, height = 2.5, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(4, 2, 2, 2) + 0.1)`
- `plot(1:20, pch = 16, cex = seq(from = 1, to = 10, length = 20), col = 1:20, xlab = "")`
- `par(old.par)`
- `palette("default")` # using default colors
- `palette()`



- 53

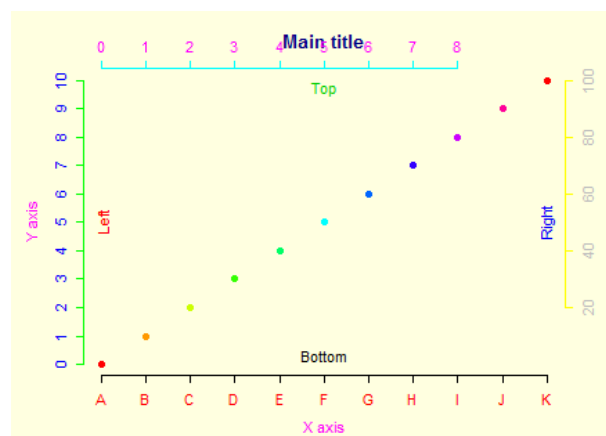
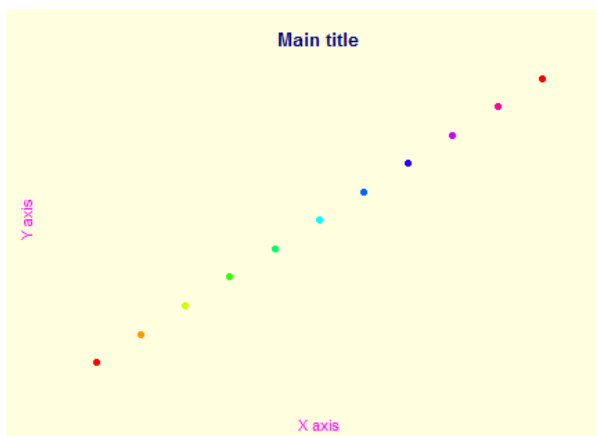
顏色有關的指令

- `colors()` # 所有的內建顏色的指令
- `colors()[grep("pink", colors())]` # 所有與粉紅色有關的指令
- `show.colors()` # 展現各種內顏色的名稱與顏色
- `library(DAAG)` # Tools/Install Packages : DAAG
- `show.colors(type = "singles", order.cols = TRUE)` # single shade
- `show.colors(type = "shades", order.cols = TRUE)` # multiple shades
- `show.colors(type = "gray", order.cols = TRUE)` # gray shades

- 54

- `col` # 一般狀況之繪圖顏色設定
- `col.axis` # 座標軸的顏色：內定是黑色
- `col.lab` # 座標軸標記文字的顏色：內定是黑色
- `col.main` # 主標題（上標題）的顏色：內定是黑色
- `col.sub` # 副標題（下標題）的顏色：內定是黑色
- `bg` # 背景顏色：內定是透明

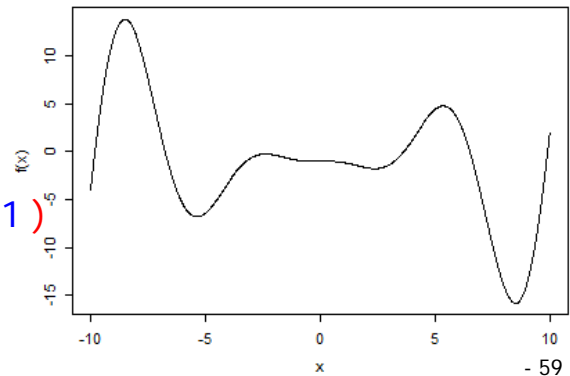
- `windows(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(bg = "lightyellow", col.main = "navy", col.lab = "magenta", mex = 0.8, mar = c(5, 5, 4, 3) + 0.1)`
- `plot(0:10, 0:10, pch = 16, col = rainbow(10), main = "Main title", xlab = "X axis", ylab = "Y axis", axes = FALSE)`



- # 設定四邊的文字符號
- `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))`
- `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))`
- `mtext(text = c("Bottom", "Left", "Top", "Right"), side = 1:4, col = 1:4, line = -2)`
- `par(old.par)`

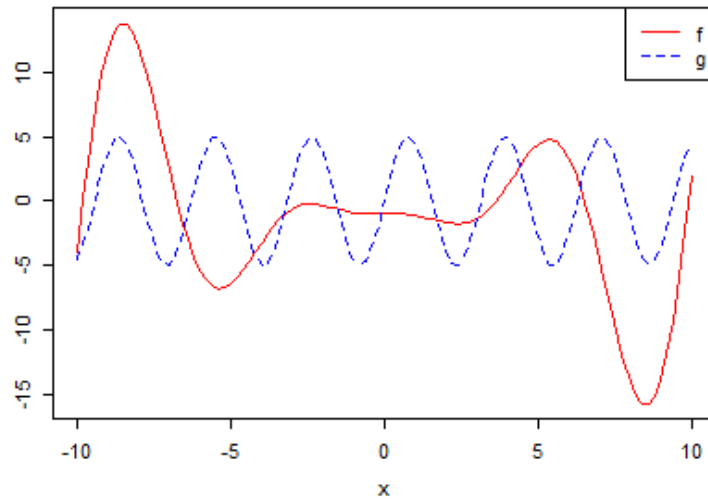
數學函數繪圖

- `curve()` # 數學函數繪圖
- `f <- function(x) 0.01 * x^3 * cos(x) - 0.2 * x^2 * sin(x) + 0.05 * x - 1`
- `win.graph(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 5, 4, 2) + 0.1)`
- `curve(f, from = -10, to = 10)`
- `curve(f, from = -10, to = 10, n = 1001)`
- `par(old.par)`



- `f <- function(x) 0.01 * x^3 * cos(x) - 0.2 * x^2 * sin(x) + 0.05 * x - 1`
- `g <- function(x) 10 * cos(x) * sin(x)`
- `win.graph(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 5, 4, 2) + 0.1)`
- `curve(f, from = -10, to = 10, lty = 1, col = "red", ylab = "")`
- `curve(g, add = TRUE, lty = 2, col = "blue")`
- `legend("topright", legend = c("f", "g"), lty = 1:2, col = c("red", "blue"))`
- `par(old.par)`

- `f <- function(x) 0.01 * x^3 * cos(x) - 0.2 * x^2 * sin(x) + 0.05 * x - 1`
- `g <- function(x) 10 * cos(x) * sin(x)`



- 61

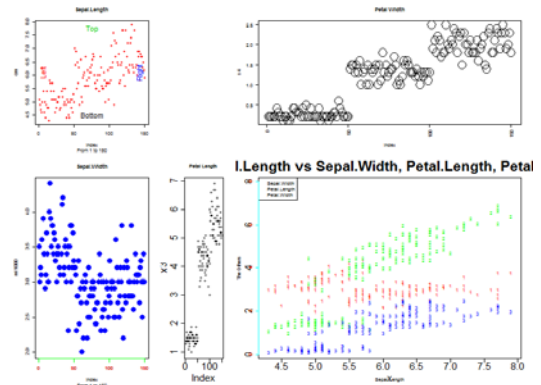
作業

HW08：繪圖功能與參數設定

計算機程式設計 - 2016F
Chap 10: 多重繪圖與顏色
Feng-Li Lian @ NTU-EE

On 12/6, 2016

- 右下角的圖是使用 iris 的數據，所繪製一組圖。
- 程式為：[HW08_PlotFigures.R](#)（請從課程網站下載到工作目錄）
- 此程式主要的功能為：
 - 一頁之中，擺設多張的圖
 - 每張圖的長寬或大小，要不一樣
 - 主標題，軸標題，字體要有變化
 - 數據點的顏色要有變化
 - 數據點的形式要有變化
 - 試著給不同數據加上註解
- 在本次作業中，
 - 請任意挑選**三個**，您覺得比要醜或不喜歡的地方，
 - 改變原始程式對應的參數數值或設定，
 - 在對應的下面加上註解，說明所改變的內容，
 - 然後，再重新執行一次，
 - 將所產生的新的圖，複製到報告之中。



- 63

HW08：繪圖功能與參數設定

計算機程式設計 - 2016F
Chap 10: 多重繪圖與顏色
Feng-Li Lian @ NTU-EE

On 12/6, 2016

- 繳交兩個檔案：
- 一個程式碼：檔案名稱：[HW08_B01921001_PlotFigures.R](#)
- 以及一個報告檔案，檔案名稱：[HW08_B01921001.pptx](#) 或者 [.pdf](#)
包含：
 - 一頁：學生基本資料，
 - 一頁：將有進行改變的哪幾行程式碼以及對應的註解解釋，複製到報告中，（所以要有幾行程式碼，以及**三行** # 開頭的註解）
 - 一頁：程式執行之後所產生的圖，圈出所改變的位置（所以要看到**三的圈**喔）
- 繳交方式與期限：
 - E-mail 檔案到：ntucp105f@gmail.com
 - E-mail 主旨：[HW08_B01921001](#)（就是，作業編號_您的學號）
 - 繳交期限：**12/7 (Wed), 2016, 11pm 以前**
 - **HW08**，每位繳交自己的程式碼，
程式碼中需要註解所加註的程式碼的意義或想法。

- 64