

SciPy 科學計算函式庫－繪圖篇

蔡尚融

2018-06-04

畫線段圖 (plot())

- 使用 Matplotlib 中 pyplot 模組。

```
import matplotlib.pyplot as plt
```

- 用 plot() 畫 $y = \frac{\cos(4\pi x)}{1+x}$, x 在 $[0, 1]$ 區間。

```
>>> import numpy as np
```

```
>>> import matplotlib.pyplot as plt
```

```
>>> ax = np.linspace(0, 1.0, 101)
```

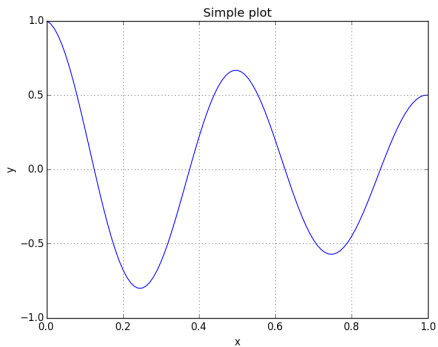
```
>>> vy = np.cos(4 * np.pi * ax) / (1 + ax)
```

```
>>> plt.plot(ax, vy)
```

```
>>> plt.show()
```

- 加入格點。

```
>>> plt.grid()
```



■ 加入標籤。

```
>>> plt.title('Simple plot')
>>> plt.xlabel('x')
>>> plt.ylabel('y')
>>> plt.grid()
```

■ 多線繪製

```
plt.hold(True)
plt.plot(...)
plt.plot(...)
```

■ 線條顏色

```
color = '顏色名稱'
'red', 'blue', 'black', ...
```

■ 線條樣式

```
linestyle = '樣式'
'dashed', 'solid', ...
```

■ 線條粗細

```
linewidth = 2
```

- 描繪點標記

```
marker = '樣式'
```

- 描繪點顏色

```
markeredgecolor = '顏色名稱'
```

- 描繪點大小

```
markeredgewidth = 2
```

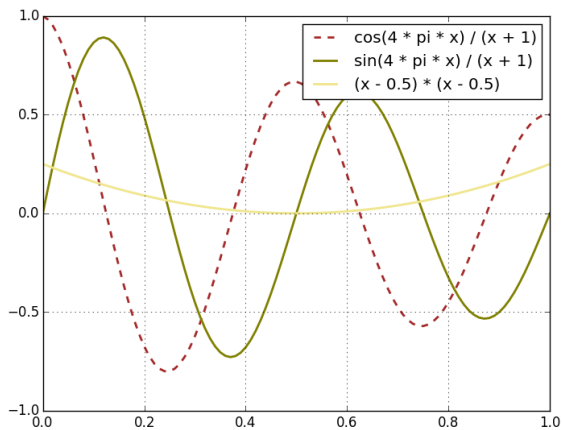
- 線條標籤

```
label = '名稱'
```

並加入

```
plt.legend()
```

隨堂練習：



```
#!/usr/bin/env python

import numpy as np
import matplotlib.pyplot as plt

ax = np.linspace(0, 1.0, 101)
f1 = np.cos(4 * np.pi * ax) / (1 + ax)
f2 = np.sin(4 * np.pi * ax) / (1 + ax)
f3 = (ax - 0.5) * (ax - 0.5)

plt.plot(ax, f1, color = 'brown', label = 'cos(4 * pi * x) / (x + 1)', \
         linewidth = 2, linestyle = 'dashed')
plt.plot(ax, f2, color = 'olive', label = 'sin(4 * pi * x) / (x + 1)', \
         linewidth = 2, linestyle = 'solid')
plt.plot(ax, f3, color = 'khaki', label = '(x - 0.5) * (x - 0.5)', \
         linewidth = 2)

plt.grid()
plt.legend()
plt.show()
```

等高線圖 (Contour)

考慮向量值函數

$$F(x, y) = \begin{bmatrix} f_1(x, y) \\ f_2(x, y) \end{bmatrix} = \begin{bmatrix} 4x^2 - y^2 \\ 4xy^2 - x - 1 \end{bmatrix}$$

令 $z = \|F(x, y)\|_2$ ，繪製以 z 為高度的等高線圖。

載入模組：

```
import numpy as np
import matplotlib.pyplot as plt
```


以函式 `np.meshgrid()` 產生網格：

```
delta = 0.025
x = np.arange(-2.0, 2.0 + delta, delta)
y = np.arange(-2.0, 2.0 + delta, delta)
X, Y = np.meshgrid(x, y)
```

格點上計算高度 $z = \|F(x, y)\|_2$ ：

```
Z = np.zeros(X.shape)
(rs, cs) = X.shape
for ii in range(rs):
    for ij in range(cs):
        V = np.array([X[ii][ij], Y[ii][ij]])
        Z[ii][ij] = np.linalg.norm(formfunction(V))
```

畫等高線圖的指令與函式：

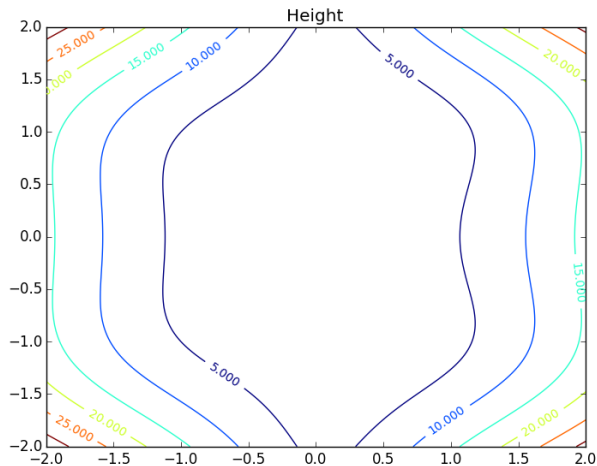
指令與函式：

- `matplotlib.figure()` 產生頂層的抽象繪圖物件。
- `matplotlib.pyplot.contour()` 繪製等高線圖。
- `matplotlib.pyplot.clabel()` 設定等高線上的標籤。

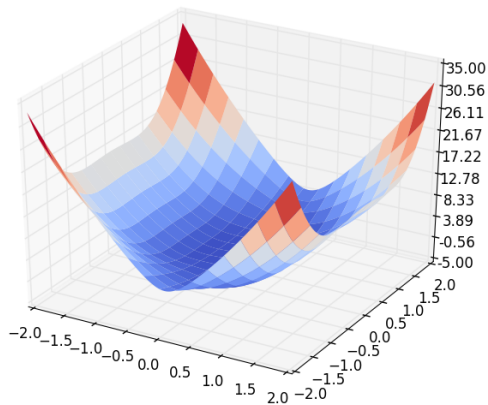
剩餘的部份：

```
plt.figure()  
CS = plt.contour(X, Y, Z)  
plt.clabel(CS, inline=1, fontsize=10)  
plt.title('Height')  
plt.show()
```

$F(X)$ 的等高線圖



$F(X)$ 的曲面圖



加入：

```
from mpl_toolkits.mplot3d import Axes3D
from matplotlib import cm
from matplotlib.ticker import LinearLocator, FormatStrFormatter
```

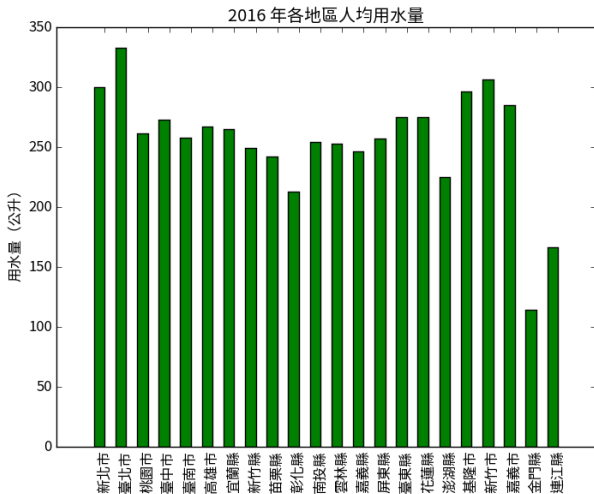
修改：

```
fig = plt.figure()
ax = fig.gca(projection='3d')
surf = ax.plot_surface(X, Y, Z, cmap = cm.coolwarm, linewidth = 0)

ax.set_zlim(-5.0, 35.0)
ax.zaxis.set_major_locator(LinearLocator(10))
ax.zaxis.set_major_formatter(FormatStrFormatter('%.02f'))

plt.show()
```

長條圖 (Bar chart) 範例



```
#!/usr/bin/env python
import csv
import numpy as np
import matplotlib.pyplot as plt
from matplotlib import rcParams

rcParams['font.sans-serif'] = ['Source Han Sans TW']

area = []
water = []

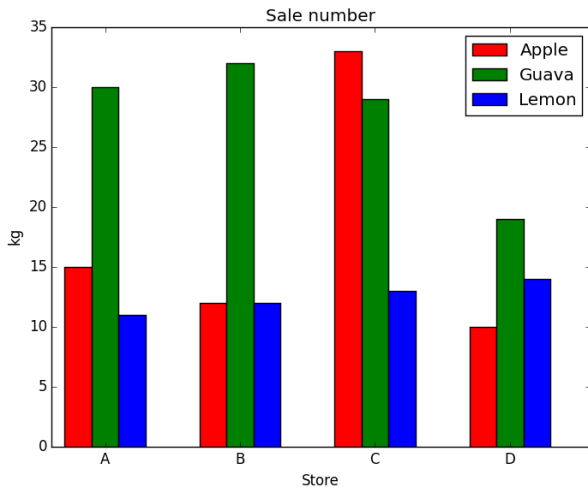
fp = open('自來水生活用水量統計.csv', 'r')
for row in csv.reader(fp):
    if row[0] == '2016-12-31T00:00:00':
        area.append(row[1])
        water.append((float(row[4])))
fp.close()
```

```
index = np.arange(len(area))
bar_width = 0.5

rects1 = plt.bar(1 + index + bar_width * 1.0, water, bar_width,
                 color = 'g', label = '總量')

plt.xlabel('地區')
plt.ylabel('用水量')
plt.title('2016 年各地區人均用水量')
plt.xticks(1.5 + index + bar_width * 1.5, area,
           rotation = 'vertical')
plt.show()
```


長條圖 (Bar chart) 範例二：分組



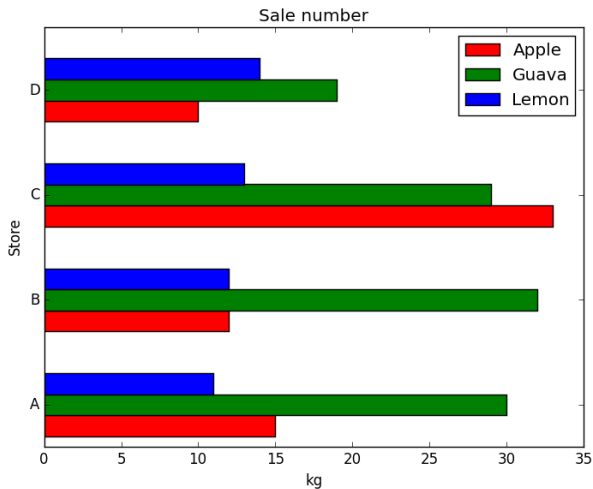
```
#!/usr/bin/env python
import numpy as np
import matplotlib.pyplot as plt

stores = 4
apple = [15, 12, 33, 10]
guava = [30, 32, 29, 19]
lemon = [11, 12, 13, 14]

index = np.arange(stores)
bar_width = 0.2
```

```
rects1 = plt.bar(index + bar_width * 0.5, apple, bar_width,  
    color = 'r', label = 'Apple')  
rects2 = plt.bar(index + bar_width * 1.5, guava, bar_width,  
    color = 'g', label = 'Guava')  
rects2 = plt.bar(index + bar_width * 2.5, lemon, bar_width,  
    color = 'b', label = 'Lemon')  
  
plt.xlabel('Store')  
plt.ylabel('kg')  
plt.title('Sale number')  
plt.xticks(index + bar_width * 2, ('A', 'B', 'C', 'D'))  
plt.legend()  
plt.show()
```

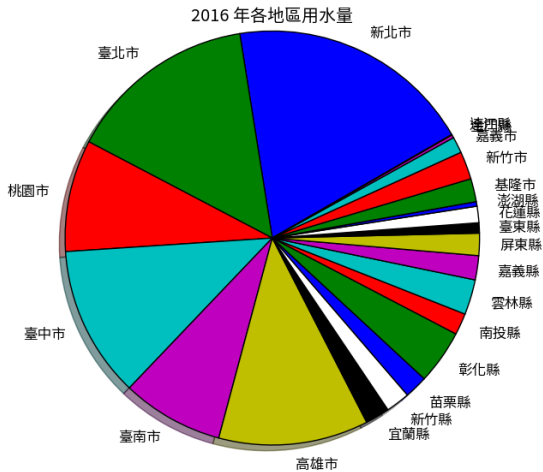
長條圖 (Bar chart) 範例三：水平



修改：

```
rects1 = plt.barh(index + bar_width * 0.5, apple, bar_width,  
    color = 'r', label = 'Apple')  
rects2 = plt.barh(index + bar_width * 1.5, guava, bar_width,  
    color = 'g', label = 'Guava')  
rects2 = plt.barh(index + bar_width * 2.5, lemon, bar_width,  
    color = 'b', label = 'Lemon')  
  
plt.xlabel('kg')  
plt.ylabel('Store')  
  
plt.yticks(index + bar_width * 2, ('A', 'B', 'C', 'D'))
```

圓餅圖 (Pie chart) 範例



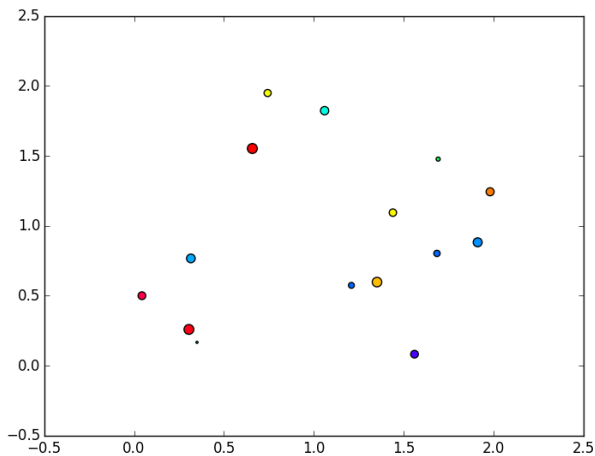
```
#!/usr/bin/env python import csv
import numpy as np
import matplotlib.pyplot as plt
from pylab import mpl

mpl.rcParams['font.sans-serif'] = ['Source Han Sans TW']
area = []
water = []

fp = open('自來水生活用水量統計.csv', 'r')
for row in csv.reader(fp):
    if row[0] == '2016-12-31T00:00:00':
        area.append(row[1])
        water.append((float(row[2])))
fp.close()
```

```
plt.pie(water, labels = area, shadow = True, startangle = 30)
plt.axis('equal')
plt.title('2016 年各地區用水量')
plt.show()
```


散布圖

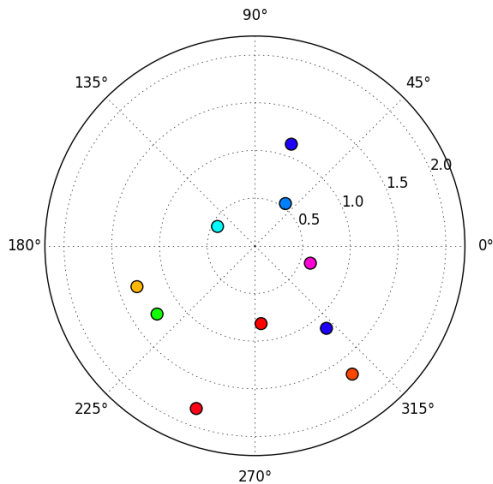


```
#!/usr/bin/env python
import numpy as np
import matplotlib.pyplot as plt

num = 15
vx = 2 * np.random.rand(num)
vy = 2 * np.random.rand(num)
colors = 2 * np.pi * np.random.rand(num)
scale = 100 * np.random.rand(num)

plt.scatter(vx, vy, c = colors, s = scale, cmap = 'hsv')
plt.show()
```

極坐標圖表 (Polar chart) 範例

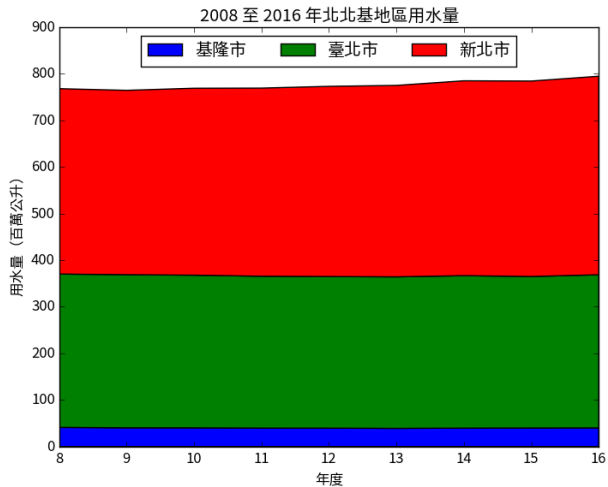


```
#!/usr/bin/env python
import numpy as np
import matplotlib.pyplot as plt

num = 10
radius = 2 * np.random.rand(num)
theta = 2 * np.pi * np.random.rand(num)
colors = 2 * np.pi * np.random.rand(num)

ax = plt.subplot(111, polar = True)
ax.scatter(theta, radius, c = colors, s = 100, cmap = 'hsv')
plt.show()
```

堆疊區域圖



```
#!/usr/bin/env python
import csv
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.patches import Rectangle
from pylab import mpl

mpl.rcParams['font.sans-serif'] = ['Source Han Sans TW']

year = range(8, 17)
water = np.zeros((3, 9))
```

```
fp = open('自來水生活用水量統計.csv', 'r')
for row in csv.reader(fp):
    area = row[1]
    if area == '基隆市':
        water[0][int(row[0][0:4]) - 2008] = float(row[2])
    elif area == '臺北市':
        water[1][int(row[0][0:4]) - 2008] = float(row[2])
    elif area == '臺北縣' or area == '新北市':
        water[2][int(row[0][0:4]) - 2008] = float(row[2])
fp.close()
```

```
area = ['基隆市', '臺北市', '新北市']
```

```
plt.title('2008 至 2016 年北北基地區用水量')
plt.xlabel('年度')
plt.ylabel('用水量 (百萬立方公尺) ')
plt.stackplot(year, water / 1000000)

p1 = Rectangle((0, 0), 1, 1, fc = 'blue')
p2 = Rectangle((0, 0), 1, 1, fc = 'green')
p3 = Rectangle((0, 0), 1, 1, fc = 'red')
plt.legend([p1, p2, p3], area)

plt.show()
```


影像

- 載入影像：`pyplot.imread()`。
原生只支援 png 檔案，回傳值型態是 `numpy.array`。灰階影像為 $m \times n$ 陣列，RGB 影像為 $m \times n \times 3$ 陣列，RGBA 影像為 $m \times n \times 4$ 陣列。
- 影像呈現：`pyplot.imshow()`。輸入的須下列格式：
 $m \times n$ 陣列（float 或 int 型態），或 $m \times n \times 3$ 陣列 RGB、 $m \times n \times 4$ 陣列 RGBA（float 或 uint8 型態），其中 RGB 或 RGBA 的 float 型態限制值在 0.0 到 1.0（含）之間。
- 儲存影像：`pyplot.imsave()`
儲存格式基本上支援 png、pdf、ps、eps、svg 等格式。

載入影像並另存新檔

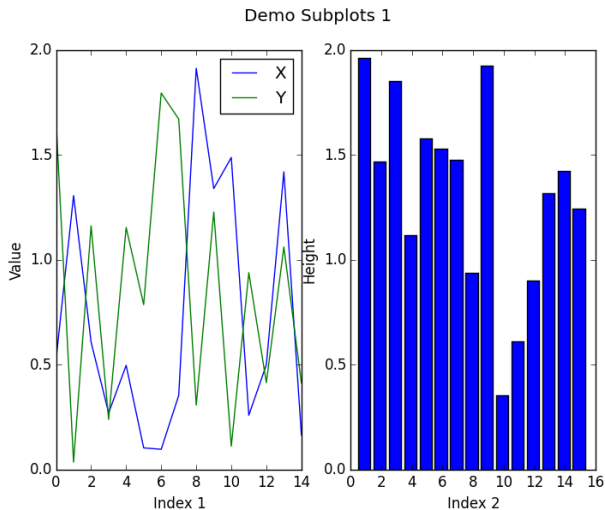
```
#!/usr/bin/env python
import matplotlib.pyplot as plt

image = plt.imread('demo_image.png')

plt.imshow(image)
plt.axis('off')
plt.show()

plt.imsave('output_image.jpg', image, format = 'jpg')
```

繪製多圖範例一



```
#!/usr/bin/env python
import numpy as np
import matplotlib.pyplot as plt

num = 15
vx = 2 * np.random.rand(num)
vy = 2 * np.random.rand(num)
index = np.arange(len(vx))
height = 2 * np.random.rand(num)

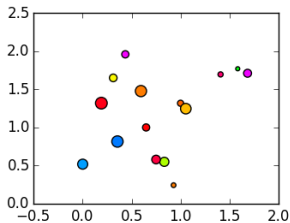
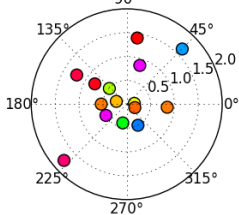
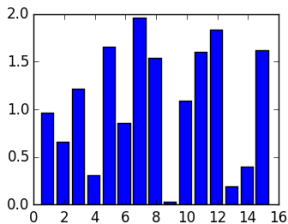
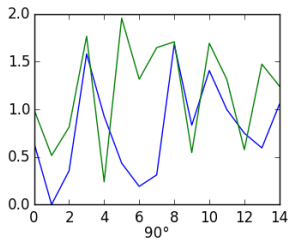
fig, ax = plt.subplots(nrows = 1, ncols = 2)
fig.suptitle('Demo Subplots 1', fontsize='large')
```

```
ax[0] = plt.subplot(121)
ax[0].plot(index, vx, 'b', label = 'X')
ax[0].plot(index, vy, 'g', label = 'Y')
ax[0] = plt.xlabel('Index 1')
ax[0] = plt.ylabel('Value')
ax[0] = plt.legend()
```

```
ax[1] = plt.subplot(122)
ax[1].bar(index + 0.5, radius)
ax[1] = plt.xlabel('Index 2')
ax[1] = plt.ylabel('Height')
```

```
plt.show()
```

繪製多圖範例二



```
#!/usr/bin/env python
import numpy as np
import matplotlib.pyplot as plt

num = 15
vx = 2 * np.random.rand(num)
vy = 2 * np.random.rand(num)
index = np.arange(len(vx))
height = 2 * np.random.rand(num)

fig, ax = plt.subplots(nrows = 2, ncols = 2)
ax[0][0] = plt.subplot(221)
ax[0][0].plot(index, vx, index, vy)
ax[0][1] = plt.subplot(222)
ax[0][1].bar(index + 0.5, height)
```

```
radius = 2 * np.random.rand(num)
theta = 2 * np.pi * np.random.rand(num)
colors = 2 * np.pi * np.random.rand(num)

ax[1][0] = plt.subplot(223, polar = True)
ax[1][0].scatter(theta, radius, c = colors, s = 100, cmap = 'hsv')

scale = 100 * np.random.rand(num)

ax[1][1] = plt.subplot(224)
ax[1][1].scatter(vx, vy, c = colors, s = scale, cmap = 'hsv')

plt.show()
```


匯出繪圖範例

用 `plt.savefig()` 匯出繪圖至檔案。

```
#!/usr/bin/env python import numpy as np
import matplotlib.pyplot as plt

num = 10
radius = 2 * np.random.rand(num)
theta = 2 * np.pi * np.random.rand(num)
colors = 2 * np.pi * np.random.rand(num)

ax = plt.subplot(111, polar = True)
ax.scatter(theta, radius, c = colors, s = 100, cmap = 'hsv')
plt.savefig('output_figure.png')
```