圁

問題討論

學習心得(完成)

簡報閱讀

D19 使用 Matplotlib 繪製各種常用圖表

範例與作業





前期回顧

import numpy as np

y = np.sin(x * np.pi / 180.0)

- 其他圖型:密度圖、等高線圖
- 1 # 載入需要的... import matplotlib.pyplot as plt

準備數據 ... 假設我要畫一個sin波 從0~180度 x = np.arange(0,180)

```
10 # 開始畫圖
11
12  # 設定要畫的的x,y數據list....
14 plt.plot(x,y)
16 # 設定圖的範圍,不設的話,系統會自行決定
17 plt.xlim(-30,390)
18 plt.ylim(-1.5,1.5)
    # 照需要寫入x 軸和y軸的 label 以及title
21
     plt.xlabel("x-axis")
    plt.ylabel("y-axis")
     plt.title("The Title")
    # 在這個指令之前,都還在做畫圖的動作
    # 這個指令算是 "秀圖"
     plt.show()
製作繪圖板 Subplots
matplotlib 的圖像都位於 Figure 物件,所以我們可以設定一個畫板,並在這畫板中配置子版
plt.subplot(a, b, c)
```

a:代表X軸的分割

b:代表y軸的分割

c:代表子版的編號數

subplot(2,1,1)

subplot(2,1,2)

subplot(1,2,1) subplot(1,2,2) subplot(2,2,1)subplot(2,2,2)subplot(2,2,3) subplot(2,2,4)

We can plot different things in the same figure using the subplot function.

#載入 numpy, 提供亂數資料與數學式,

import matplotlib.pyplot as plt

x = np.arange(0, 3 * np.pi, 0.1)

import numpy as np

載入 matplotlib

y_sin = np.sin(x)

10 plt.subplot(2, 1, 1)

 $y_{cos} = np.cos(x)$

9 # 設定雙格畫板大小

plt.plot(x, y_sin)

12 plt.title("Sine")

14 # 設定雙格畫板大小

15 plt.subplot(2, 1, 2)

2 from sklearn import datasets

9 # 設定圖形的大小(寬,高)

import matplotlib.pyplot as plt

digits = datasets.load_digits()

10 fig = plt.figure(figsize=(4, 2))

15 # 把前 8 個手寫數字顯示在子圖形

3 # 載入 matplotlib

6 # 載入 `digits`

12 # 調整子圖形

16 for i in range(8):

11

13

```
16 plt.plot(x, y_cos)
plt.title("Cosine")
18
19 plt.show()
                                          Sine
              0.5
              0.0
             -0.5
           -1.0 ____
                                      4 Cosine 6
                           2
                                                            8
                                                                      10
              1.0 r
              0.5
              0.0
             -0.5
1 # 從 `sklearn` 載入 `datasets`
```

```
# 在 2 x 4 網格中第 i + 1 個位置繪製子圖形,並且關掉座標軸刻度
    ax = fig.add_subplot(2, 4, i + 1, xticks = [], yticks = [])
   # 顯示圖形,色彩選擇灰階
19
     ax.imshow(digits.images[i], cmap = plt.cm.binary)
    # 在左下角標示目標值
21
     ax.text(0, 7, str(digits.target[i]))
23
24 # 顯示圖形
25 plt.show()
             4 5 6 7
製作繪圖板 Axes
  ● Axes 功用與 Subplot 類似,若是在圖形上需要顯示資料的子集,即可使用
  ● 但是可以將圖放置在圖中的任何位置。因此,如果要在較大的圖中放置較小的圖,則可以使用
    軸。
```

plt.text(0.6,0.6, 'axes([0.1,0.1,.8,.8])',ha='center',va='center',size=20,alpha=.5)

11 plt.text(0.5,0.5, 'axes([0.2,0.2,.3,.3])',ha='center',va='center',size=16,alpha=.5)

axes([0.2,0.2,.3,.3])

fig.subplots_adjust(left=0, right=1, bottom=0, top=1, hspace=0.05, wspace=0.05)

1 #導入必要的模組

2 import numpy as np

6 # 創建一個3d坐標系

3 import matplotlib.pyplot as plt

from mpl_toolkits.mplot3d import Axes3D

14 x = np.linspace(0, 1, 100) # linspace創建等差陣列

15 y = np.cos(x * 2 * np.pi) / 2 + 0.5

Y axis label

Figure

#import matplotlib.pyplot as plt

4 plt.axes([0.1,0.1,.8,.8])

5 plt.xticks([]), plt.yticks([])

plt.axes([0.2,0.2,.3,.3])

plt.xticks([]), plt.yticks([])

3 #決定最外框

#決定內框

13 plt.show()

12

製作 3D 繪圖板 • 3D 圖形在資料分析、資料建模、圖形和影像處理等領域中都有著廣泛的應用 • 主要把想要觀察的重點與場景實現兩種交互

axes([0.1,0.1,.8,.8])

7 fig = plt.figure() 8 ax = Axes3D(fig) 9 #直接查詢參數與設定 10 #help(plt.plot) #help(np.random.sample) 12 13 # 利用x軸和y軸繪製sin曲線

16 # 通過zdir = 'z' 將資料繪製在z軸, zs = 0.5 則是將資料繪製在z = 0.5的地方

ax.plot(x, y, zs = 0.5, zdir = 'z', color = 'black', label = 'curve in (x, y)')

— curve in (x, y) point in (x, z)

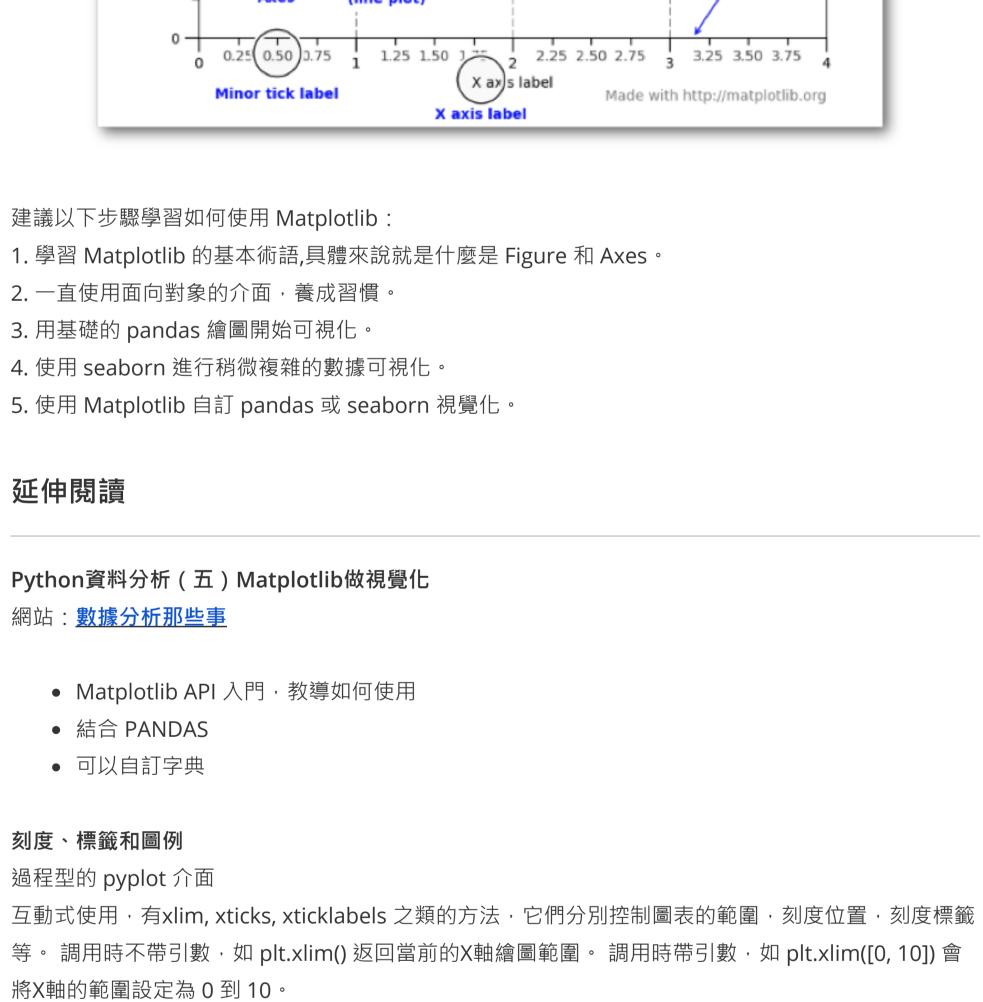
1.0

0.8

• 一種是可以操縱場景從而能夠從不同的角度觀察模型

• 一種是擁有添加與操作修改模型物件的能力

0.6 Z 0.2 0.2 知識點回顧 Anaconiy of a figure Re sign Minor tick 0 0 Major tick label



°°°°°

In [108]: ax.plot(randn(1000).cumsum(), 'k', label='one') Out[108]: [<matplotlib.lines.Line2D at 0x11d814240>] In [109]: ax.plot(randn(1000).cumsum(), 'k--', label='two')

添加圖例的方式:添加 subplot 的時候傳入 label 引數

更為物件導向的原生 matplotlib API。

t[111]:	<pre><matplotlib.legend.legend 0x11d82d438="" at=""></matplotlib.legend.legend></pre>
	Figure 3
30 -	— one
	— one two • three
30 - 20 -	two
	two

In [107]: fig = plt.figure(); ax = fig.add_subplot(1, 1, 1)

In [110]: ax.plot(randn(1000).cumsum(), 'k.', label='three')

Out[109]: [<matplotlib.lines.Line2D at 0x11d81bcc0>]

Out[110]: [<matplotlib.lines.Line2D at 0x11d82d668>]

-20 -30 -40 200 400 600 1000 **☆ ← → +** Q ≅ 🖺 Matplotlib 客製化 matplotlib customization example fig, (ax0, ax1) = plt.subplots(nrows=1, ncols=2, sharey=True, figsize=(7, 4))fig.suptitle('2014 Sales Analysis', fontsize=14, fontweight='bold')

2014 Sales AnalysisRevenue Units

top_10.plot(kind='barh', y="Purchases", x="Name", ax=ax1)

ax1.axvline(x=avg, color='b', label='Average', linestyle='--', linewidth=1)

pbpython.com

fig.savefig('sales.png', transparent=False, dpi=80, bbox_inches="tight")

下一步:閱讀範例與完成作業

ax0.set_xlim([-10000, 140000])

top_10.plot(kind='barh', y="Sales",