

勞動部產業新尖兵計畫

人工智慧金融應用與實務培訓班



課程模組： AI 金融科技課程 - 金融大數據分析

3. 每日交易資訊圖表繪製

葉建華 (Yeh, Jian-hua)

tdi.jhyeh@tdi.edu.tw
au4290@gmail.com

講次內容

- matplotlib 模組介紹
- matplotlib 繪圖功能函數
- 各類型圖表製作
- 專題：繪製那斯達克收盤價走勢圖表

matplotlib 模組介紹

- matplotlib 是第三方模組
 - `pip install matplotlib` Anaconda 又幫你裝好了!
- 好用的數據繪圖工具
 - `import matplotlib.pyplot as plt`

數據繪圖基礎概念

- matplotlib 可用的函數很多
 - 最重要的兩個： plot(), show()
- 最基本的例子

```
import matplotlib.pyplot as plt
xs = [1, 5, 10]      # 三個點依序的x座標
ys = [7, 1, 16]      # 三個點依序的y座標
plt.plot(xs, ys)      # 基本折線圖繪製
plt.show()            # 顯示繪製的圖形
```

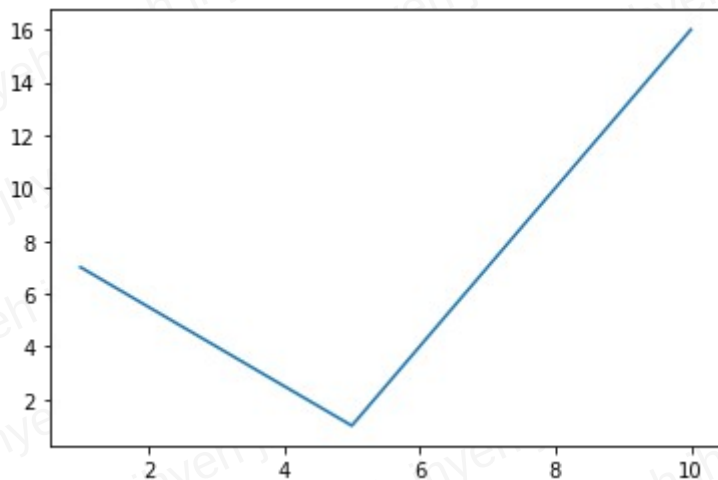
數據繪圖基礎概念

- matplotlib 可用的函數很多
 - 最重要的兩個： plot(), show()

是不是簡單到爆炸了！

- 最基本的例子

```
import matplotlib.pyplot as plt  
  
xs = [1, 5, 10]      # 三個點依序的x座標  
ys = [7, 1, 16]      # 三個點依序的y座標  
plt.plot(xs, ys)      # 基本折線圖繪製  
plt.show()            # 顯示繪製的圖形
```



講次內容

- matplotlib 模組介紹
- matplotlib 繪圖功能函數
- 各類型圖表製作
- 專題：繪製那斯達克收盤價走勢圖表

matplotlib 功能

- 數據繪製
- 圖表標題
- 圖表屬性調整：線段寬度、樣式、顏色等
- 訂定圖例

顯示標題

- 對說明清楚的圖表來說，標題很重要！！
 - 圖表標題： `title()`
 - 軸標題： `xlabel()`, `ylabel()`
 - `fontsize` 屬性設定字體大小
 - 目前無法顯示中文

顯示標題

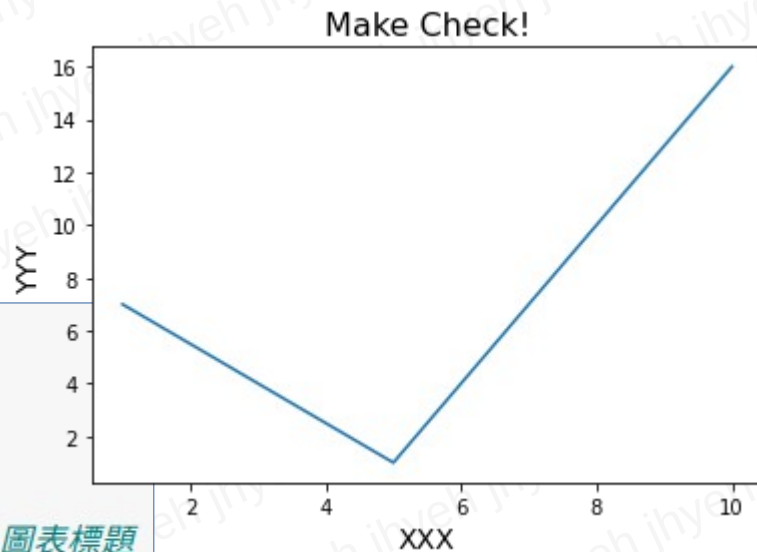
- 對說明清楚的圖表來說，標題很重要！！

- 圖表標題： `title()`
- 軸標題： `xlabel()`, `ylabel()`

- fonts

- 目前

```
import matplotlib.pyplot as plt
xs = [1, 5, 10]      # 三個點依序的x座標
ys = [7, 1, 16]      # 三個點依序的y座標
plt.plot(xs, ys)      # 基本折線圖繪製
plt.title('Make Check!', fontsize=16) # 圖表標題
plt.xlabel('XXX', fontsize=14)        # X軸標題
plt.ylabel('YYY', fontsize=14)        # Y軸標題
plt.show()                        # 顯示繪製的圖形
```



座標軸刻度與樣式

- 如何設定座標軸刻度和樣式
 - 刻度: `xticks()`, `yticks()`
 - 樣式: `tick_param()`

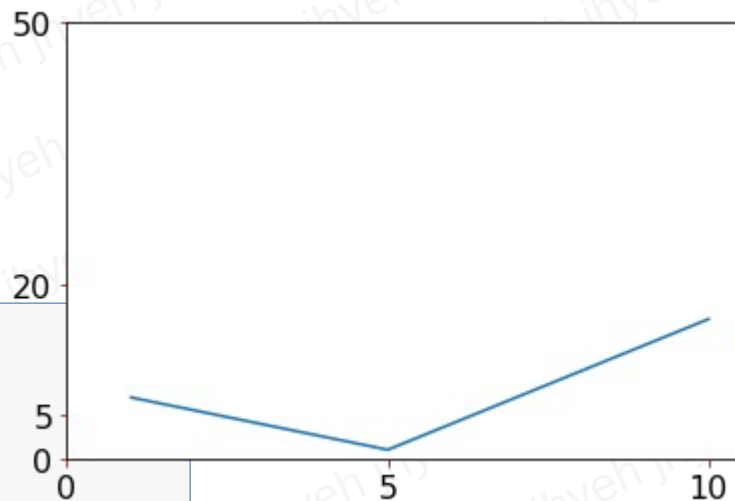
座標軸刻度與樣式

- 如何設定座標軸刻度和樣式

- 刻度: `xticks()`, `yticks()`
- 樣式: `tick_param()`

```
import matplotlib.pyplot as plt

xs = [1, 5, 10]          # 三個點依序的x座標
ys = [7, 1, 16]          # 三個點依序的y座標
plt.plot(xs, ys)         # 基本折線圖繪製
plt.xticks([0, 5, 10])   # x軸刻度
plt.yticks([0, 5, 20, 50]) # y軸刻度, 三級跳!
# 兩軸的標記數字大小設定為16點字, 刻度線改成紅色
plt.tick_params(axis='both', labelsize=16, color='red')
plt.show()               # 顯示繪製的圖形
```



顯示多組數據

- 多組數據？！
 - 直接在 `plt.plot()` 的參數依序輸入即可
 - x 序列，y 序列一對一對的輸入哦！

顯示多組數據

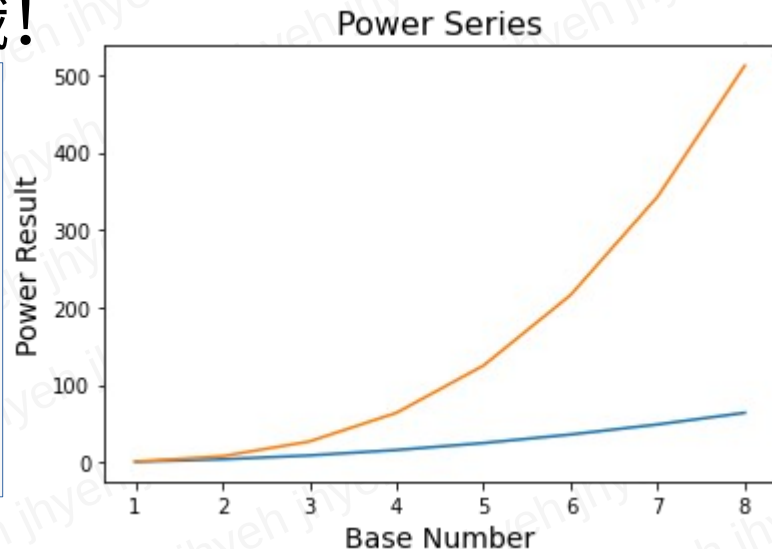
- 多組數據？！

- 直接在 `plt.plot()` 的參數依序輸入即可

- x 序列, y 序列一對一對的輸入哦！

```
import matplotlib.pyplot as plt

xs = [1, 2, 3, 4, 5, 6, 7, 8]
y1s = [1, 4, 9, 16, 25, 36, 49, 64]
y2s = [1, 8, 27, 64, 125, 216, 343, 512]
plt.plot(xs, y1s, xs, y2s) # 畫線: (x, y1s)一組, (x, y2s)一組
plt.title('Power Series', fontsize=16) # 圖表標題
plt.xlabel('Base Number', fontsize=14) # X軸標題
plt.ylabel('Power Result', fontsize=14) # Y軸標題
plt.show() # 顯示繪製的圖形
```



線條色彩與樣式

顏色	使用字元
紅色	r
綠色	g
藍色	b
黃色	y
黑色	k
白色	w

線條	使用字元	樣式
實線 (預設)	-	
虛線	--	
虛點線	-.	
點線	:	

資料點樣式	使用字元
點標記	.
像素標記	,
圓標記	o
三角標記	^
倒三角標記	v
方形標記	s
五角標記	p
星星標記	*
加號標記	+

線條色彩與樣式

- 熟悉一下！

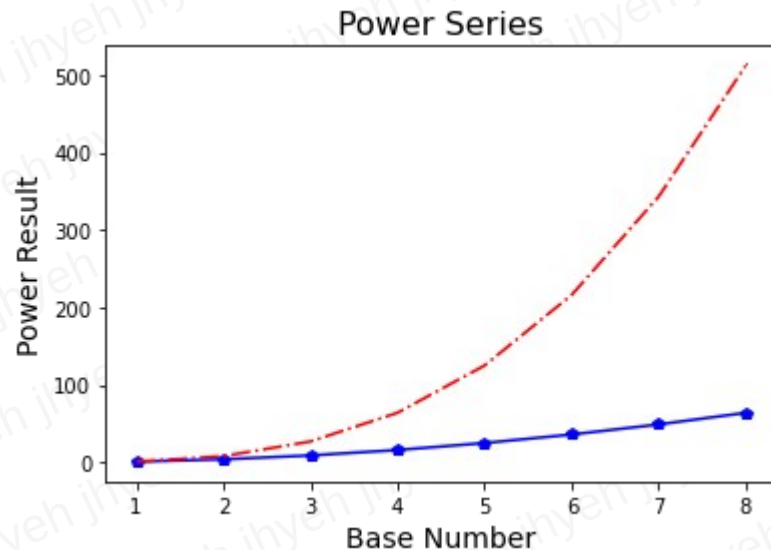
- 綠色點 - 實線 - 五角標記： `g-p`
- 紅色點 - 虛線 - 倒三角標記： `r--v`
- 黃色點 - 虛點線 - 像素標記： `y-.,`
- 緊接在 `plot()` 序列後方

線條色彩與樣式

- 熟悉一下！

```
import matplotlib.pyplot as plt

xs = [1, 2, 3, 4, 5, 6, 7, 8]
yls = [1, 4, 9, 16, 25, 36, 49, 64]
y2s = [1, 8, 27, 64, 125, 216, 343, 512]
# 畫線: (x, yls)一組, (x, y2s)一組
plt.plot(xs, yls, 'b-p', xs, y2s, 'r-.,')
plt.title('Power Series', fontsize=16) # 圖表標題
plt.xlabel('Base Number', fontsize=14) # X軸標題
plt.ylabel('Power Result', fontsize=14) # Y軸標題
plt.show() # 顯示繪製的圖形
```



– 緊接在 plot() 序列後方

設定圖例

- 用 legend() 設定圖例
 - 比較複雜，要針對每條線個別 plot()
 - 個別 plot() 要設定 label 屬性
 - 呼叫 legend()

圖例位置參數

- legend() 透過 loc 參數設定圖例位置

位置	參數數字	參數字串	位置	參數數字	參數字串
最佳位置	0	'best'	中左方	6	'center left'
右上方	1	'upper right'	中右方	7	'center right'
左上方	2	'upper left'	中下方	8	'lower center'
左下方	3	'lower left'	中上方	9	'upper center'
右下方	4	'lower right'	中間	10	'center'
右方	5	'right'			

設定圖例

- 用 legend() 設定圖例

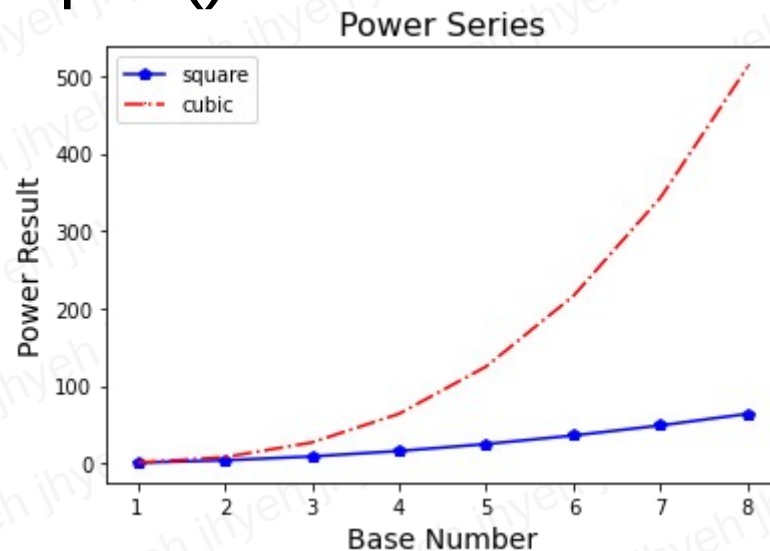
```
import matplotlib.pyplot as plt

xs = [1, 2, 3, 4, 5, 6, 7, 8]
y1s = [1, 4, 9, 16, 25, 36, 49, 64]
y2s = [1, 8, 27, 64, 125, 216, 343, 512]

# legend()要分別畫線: (x, y1s)一組, (x, y2s)一組
plt.plot(xs, y1s, 'b-p', label='square')
plt.plot(xs, y2s, 'r-.', label='cubic')
plt.legend()

plt.title('Power Series', fontsize=16) # 圖表標題
plt.xlabel('Base Number', fontsize=14) # X軸標題
plt.ylabel('Power Result', fontsize=14) # Y軸標題
plt.show() # 顯示繪製的圖形
```

別 plot()



圖表存檔

- 使用 `savefig()` 函數
 - 必須放在 `show()` 之前
 - 自動判別副檔名
 - dpi 屬性設定解析度
 - 預設值 72 ，代表螢幕解析度

圖表存檔

- 使用 `savefig()` 函數

```
import matplotlib.pyplot as plt

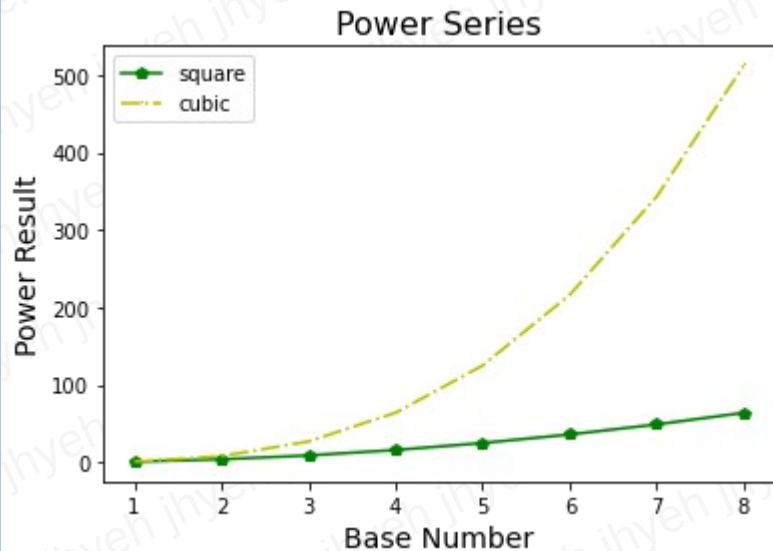
xs = [1, 2, 3, 4, 5, 6, 7, 8]
y1s = [1, 4, 9, 16, 25, 36, 49, 64]
y2s = [1, 8, 27, 64, 125, 216, 343, 512]

# legend()要分別畫線: (x, y1s)一組, (x, y2s)一組
plt.plot(xs, y1s, 'g-p', label='square')
plt.plot(xs, y2s, 'y-..', label='cubic')
plt.legend()

plt.title('Power Series', fontsize=16) # 圖表標題
plt.xlabel('Base Number', fontsize=14) # X軸標題
plt.ylabel('Power Result', fontsize=14) # Y軸標題

# 存成PNG檔案, 300dpi列印級解析度
plt.savefig('power_series.png', dpi=300)

plt.show() # 顯示繪製的圖形
```



講次內容

- matplotlib 模組介紹
- matplotlib 繪圖功能函數
- 各類型圖表製作
- 專題：繪製那斯達克收盤價走勢圖表

常見圖表類型

- matplotlib 常見的圖表類型
 - 柱狀圖 Bar Chart，另有類似的直方圖 Histogram
 - 折線圖 Line Chart，你已經試過了
 - 餅狀圖 Pie Chart，用面積比例表示
 - 散佈圖 Scatter Chart
 - 甚至可以畫 3D 立體圖表！

繪製柱狀圖

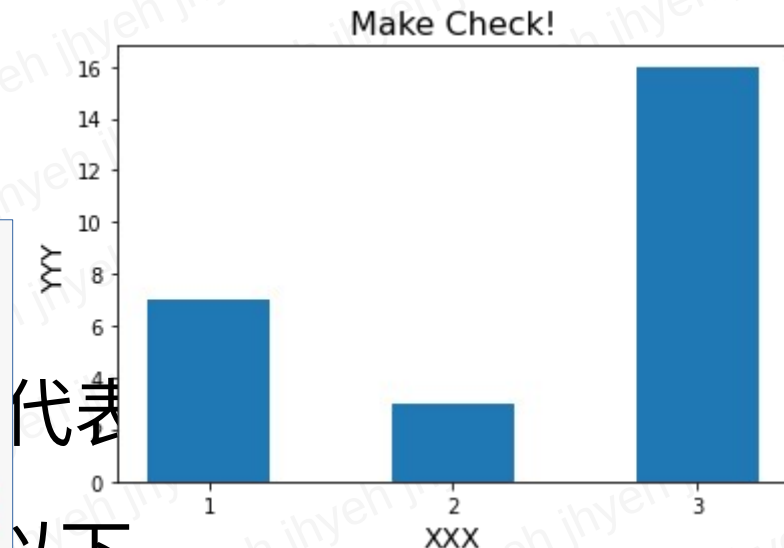
- 使用 `bar()` 函數
 - `bar(xs, ys, 寬度)`
 - 寬度設定為 0~1 之間的數字，代表寬度百分比
 - N 組數列寬度應設定在 $1/N$ 或以下

繪製柱狀圖

- 使用 bar() 函數

```
import matplotlib.pyplot as plt

xs = [1, 2, 3]           # 三個點依序的x座標
ys = [7, 3, 16]          # 三個點依序的y座標
plt.bar(xs, ys, 0.5)      # 繪製柱狀圖
plt.xticks([1, 2, 3])     # x軸刻度
plt.title('Make Check!', fontsize=16) # 圖表標題
plt.xlabel('XXX', fontsize=14) # X軸標題
plt.ylabel('YYY', fontsize=14) # Y軸標題
plt.show()                # 顯示繪製的圖形
```



繪製柱狀圖

- 兩組以上的數據？
 - 要計算資料柱的顯示位置，用寬度和 x 座標來計算
 - 寬度設定為 0~1 之間的數字，代表寬度百分比
 - N 組數列寬度應設定在 $1/N$ 或以下

繪製柱狀圖

- 兩組以上的數據？

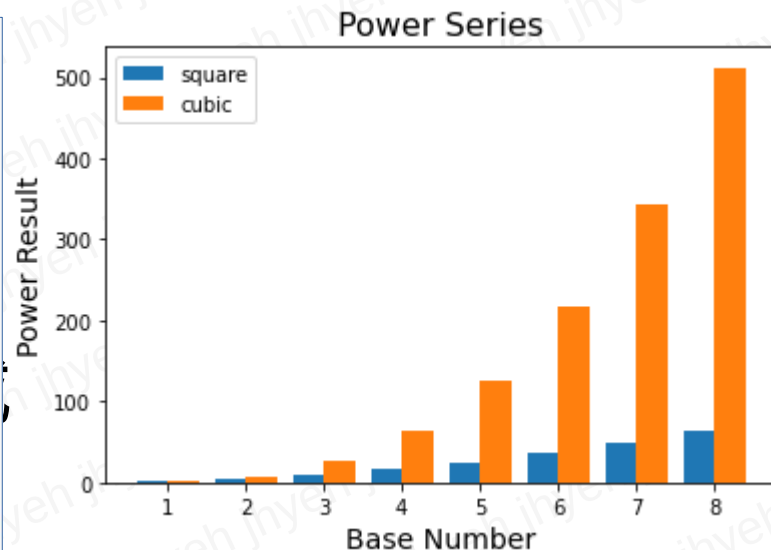
```
import matplotlib.pyplot as plt

xs = [1, 2, 3, 4, 5, 6, 7, 8]
y1s = [1, 4, 9, 16, 25, 36, 49, 64]
y2s = [1, 8, 27, 64, 125, 216, 343, 512]
# 柱寬比例，顯示兩個數列時，要設定0.5或以下才能正常顯示
width = 0.4

x1s = [x-width/2 for x in xs] # 調整第一數列的顯示座標
x2s = [x+width/2 for x in xs] # 調整第二數列的顯示座標
plt.bar(x1s, y1s, width, label='square')
plt.bar(x2s, y2s, width, label='cubic')
plt.legend(loc='best') # 自動選定最好的圖例位置

plt.title('Power Series', fontsize=16) # 圖表標題
plt.xlabel('Base Number', fontsize=14) # X軸標題
plt.ylabel('Power Result', fontsize=14) # Y軸標題

plt.show()
```



繪製柱狀圖

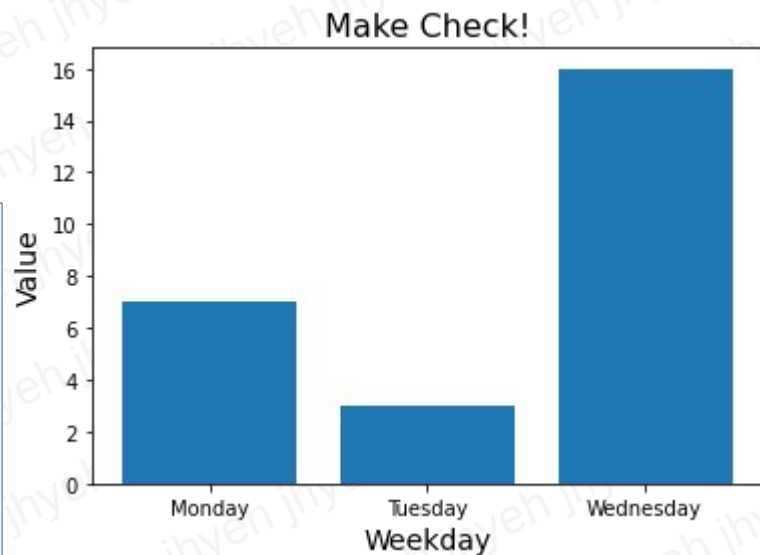
- X 軸想顯示類別而非座標？
 - 用 `xticks()` 設定文字刻度即可
 - `xticks(x 軸資料序列, 對應文字序列)`

繪製柱狀圖

- X 軸想顯示類別而非座標？
 - 用 `xticks()` 設定文字刻度即可

```
import matplotlib.pyplot as plt

xs = [1, 2, 3]           # 三個點依序的x座標
ys = [7, 3, 16]          # 三個點依序的y座標
plt.bar(xs, ys, 0.8)     # 繪製柱狀圖
plt.title('Make Check!', fontsize=16) # 圖表標題
plt.xlabel('Weekday', fontsize=14)   # X軸標題
plt.ylabel('Value', fontsize=14)     # Y軸標題
plt.xticks(xs, ['Monday', 'Tuesday', 'Wednesday']) #設定x軸刻度
plt.show()                 # 顯示繪製的圖形
```



類似柱狀圖的直方圖

- 直方圖

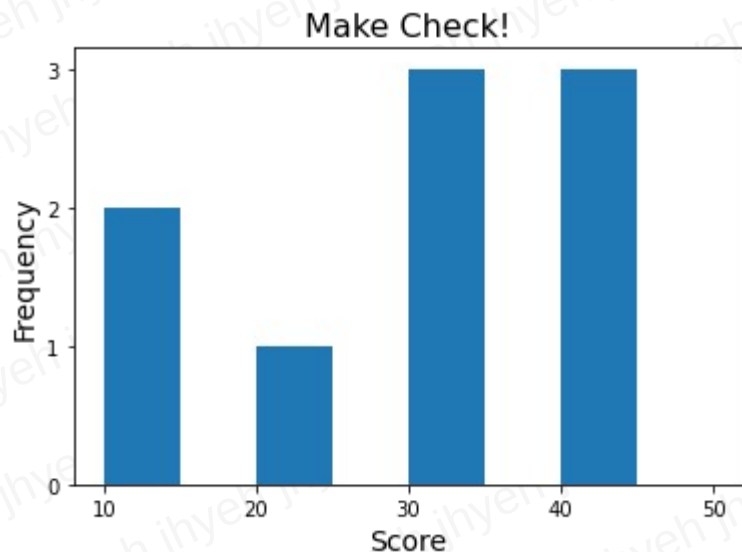
- 二維統計圖表
- 對數據分布情況的圖形表示
- 坐標分別是統計樣本和對應的統計量
 - 例如累計次數

類似柱狀圖的直方圖

- 直方圖

```
ys = [30, 20, 30, 10, 40, 40, 30, 10, 50] # 資料序列

plt.hist(ys, bins=4, width=5)                # 繪製直方圖，分成4群統計
plt.title('Make Check!', fontsize=16)        # 圖表標題
plt.xlabel('Score', fontsize=14)             # X軸標題
plt.ylabel('Frequency', fontsize=14)         # Y軸標題
#設定x軸刻度
plt.xticks([10, 20, 30, 40, 50])
plt.yticks([0, 1, 2, 3])
plt.show()                                   # 顯示繪製的圖形
```

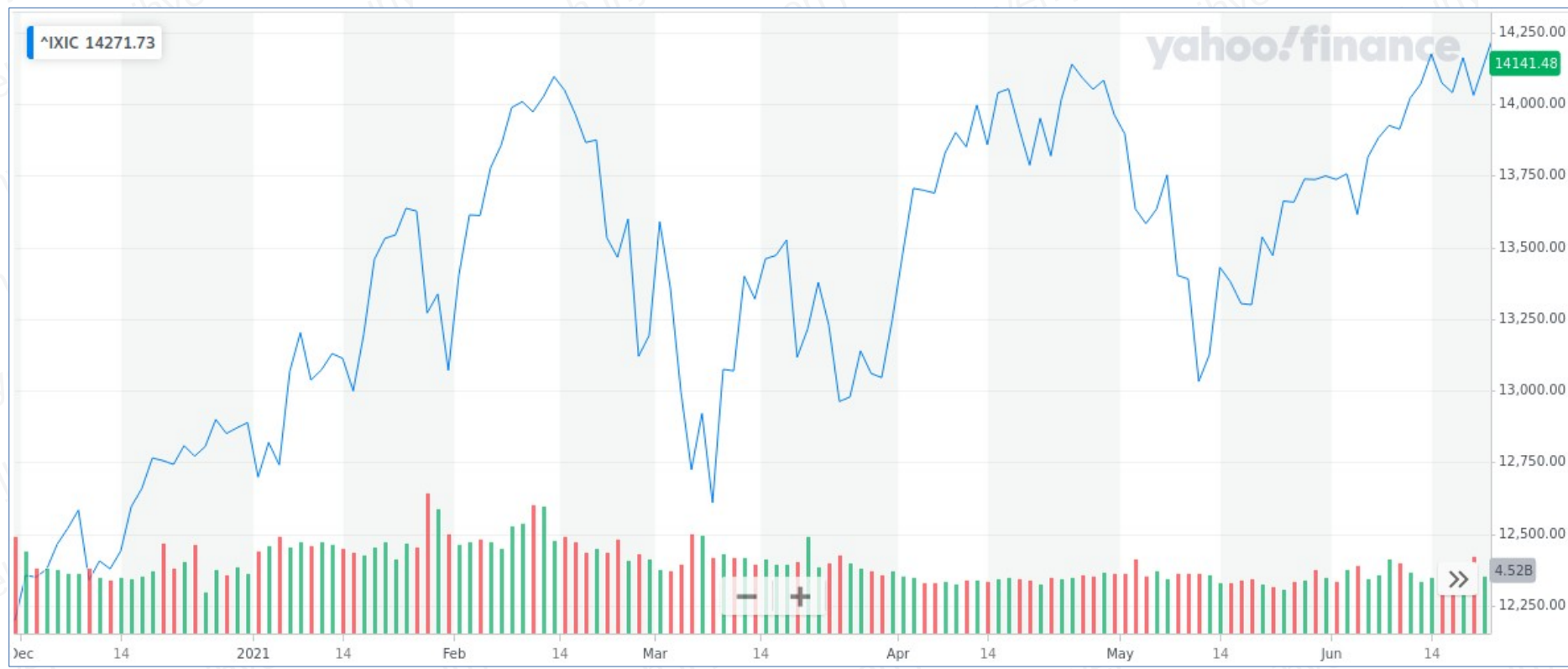


講次內容

- matplotlib 模組介紹
- matplotlib 繪圖功能函數
- 各類型圖表製作
- 專題：繪製那斯達克收盤價走勢圖表

繪製那斯達克收盤價走勢圖表

- 回想一下 Yahoo 財經很炫的圖表 ...



繪製那斯達克收盤價走勢圖表

- 我們來一個！簡易版的！
- 資料庫裡已經有 NASDAQ 收盤價格了
 - 資料期間： 2021-01-01 到 2021-06-18
- 其實很單純：從資料庫 SQLite 取出，製作資料序列，用 matplotlib 畫折線圖
- 怎麼做？

繪製那斯達克收盤價走勢圖表

- STEP1

- 從資料庫 SQLite 取出，製作資料序列
- 不就是 **SELECT-FROM-WHERE** 而已嗎？！

繪製那斯達克收盤價走勢圖表

- STEP1

- 從資料庫 SQLite 取出，製作資料序列
- 不就是 **SELECT-FROM-WHERE** 而已嗎？！

```
import sqlite3

conn = sqlite3.connect('mystock.db')
# 查詢NASDAQ交易資料的日期、收盤價格，用日期由小到大排序
sqlstr = "SELECT date, cl FROM stktrade WHERE id='^IXIC' ORDER BY date ASC;"
results = conn.execute(sqlstr)
dates = []                                # 交易日期序列
cls = []                                  # 收盤價序列
for rec in results:
    print(rec)
    dates.append(rec[0])
    cls.append(rec[1])
```

繪製那斯達克收盤價走勢圖表

• STEP1

- 從資料庫 SQLite 取出，製作資料序
- 不就是 **SELECT-FROM-WHERE** 而

```
import sqlite3

conn = sqlite3.connect('mystock.db')
# 查詢NASDAQ交易資料的日期、收盤價格，用日期由小到大排序
sqlstr = "SELECT date, cl FROM stktrade WHERE id='^IXIC' ORDER BY date ASC"
results = conn.execute(sqlstr)
dates = []
cls = []
# 交易日期序列
# 收盤價序列
for rec in results:
    print(rec)
    dates.append(rec[0])
    cls.append(rec[1])
```

('2021-01-04',	12698.450195)
('2021-01-05',	12818.959961)
('2021-01-06',	12740.790039)
('2021-01-07',	13067.480469)
('2021-01-08',	13201.980469)
('2021-01-11',	13036.429688)
('2021-01-12',	13072.429688)
('2021-01-13',	13128.950195)
('2021-01-14',	13112.639648)
('2021-01-15',	12998.5)
('2021-01-19',	13197.179688)
('2021-01-20',	13457.25)
('2021-01-21',	13530.910156)
('2021-01-22',	13543.05957)
('2021-01-25',	13635.990234)
('2021-01-26',	13626.05957)
('2021-01-27',	13270.599609)
('2021-01-28',	13337.160156)
('2021-01-29',	13070.69043)

繪製那斯達克收盤價走勢圖表

- STEP2

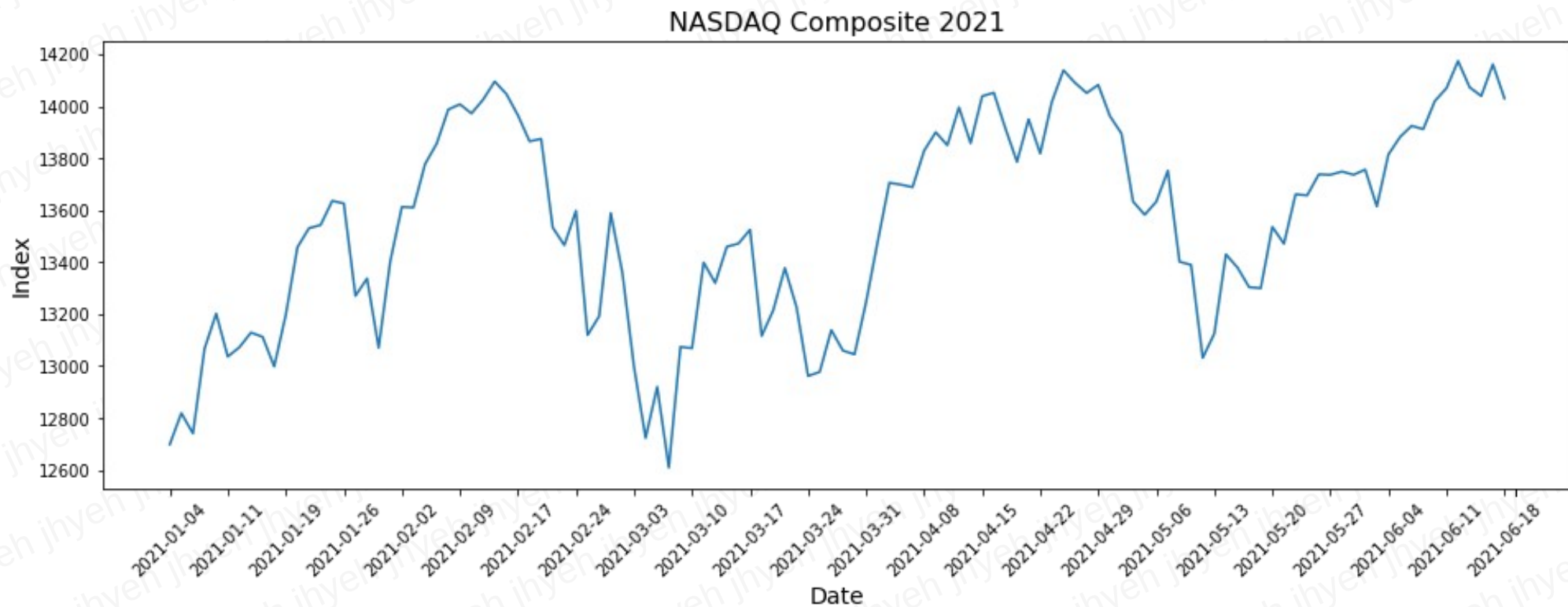
- 用 matplotlib 畫折線圖

```
import matplotlib.pyplot as plt

plt.figure(figsize=(16, 5)) # 設定圖表長寬比

# NASDAQ交易日期作為x座標，用串列生成式故意略過一些標籤不顯示
xs = [dates[i] if i%5==0 else '' for i in range(len(dates))]
ys = cls # NASDAQ收盤價作為y座標
plt.plot(dates, ys) # 基本折線圖繪製
plt.title('NASDAQ Composite 2021', fontsize=16) # 圖表標題
plt.xlabel('Date', fontsize=14) # X軸標題
plt.ylabel('Index', fontsize=14) # Y軸標題
plt.xticks(xs, rotation='45')
plt.show() # 顯示繪製的圖形
```

成果：那斯達克收盤價走勢圖



似乎不怎麼難做，對吧！！

其實程式邏輯只有這樣而已 ...

```
import sqlite3
import matplotlib.pyplot as plt

conn = sqlite3.connect('mystock.db')
# 查詢NASDAQ交易資料的日期、收盤價格，用日期由小到大排序
sqlstr = "SELECT date, cl FROM stktrade WHERE id='^IXIC' ORDER BY date ASC;"
results = conn.execute(sqlstr)
dates = [] # 交易日期序列
cls = [] # 收盤價序列
for rec in results:
    dates.append(rec[0])
    cls.append(rec[1])

plt.figure(figsize=(16, 5)) # 設定圖表長寬比

# NASDAQ交易日期作為x座標，用串列生成式故意略過一些標籤不顯示
xs = [dates[i] if i%5==0 else '' for i in range(len(dates))]
ys = cls # NASDAQ收盤價作為y座標
plt.plot(dates, ys) # 基本折線圖繪製
plt.title('NASDAQ Composite 2021', fontsize=16) # 圖表標題
plt.xlabel('Date', fontsize=14) # X軸標題
plt.ylabel('Index', fontsize=14) # Y軸標題
plt.xticks(xs, rotation='45')
plt.show() # 顯示繪製的圖形
```



這個講次中，你應該學到了 ...

- 了解 matplotlib 模組功能
- 繪製基本圖表：折線圖、柱狀圖、直方圖
- 配合資料庫存取繪製金融數據

那斯達克！！