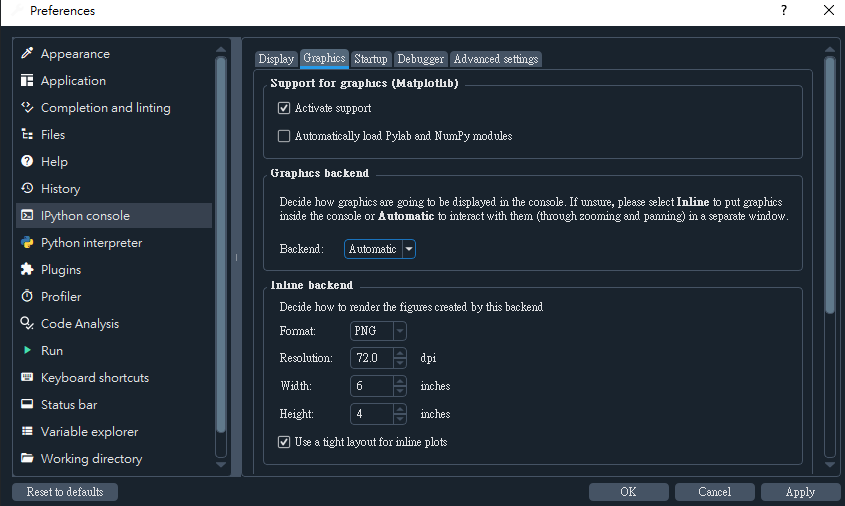
使用Matplotlib套件，在Python快速畫出分析圖表

1. 解決Figures now render in the Plots pane by default. 問題

只需要在Spyder裡面設定即可：Tools > Preferences > IPython console > Graphics 中的 Graphics backedn 改成 Automatic 就可以直接讓圖另外大大的顯示了。



1. Anaconda(裡面包含了我們之後會常用到的工具，包含Python3、Spyder編輯器等，也會自動安裝Matplotlib套件)，安裝好Anaconda就可以開啟Spyder編輯器開始寫程式了。
2. 如果寫完程式還無法顯示圖表時，有可能是Matplotlib套件未安裝，或是沒安裝好，此時可以使用pip的方式來安裝Matplotlib套件。

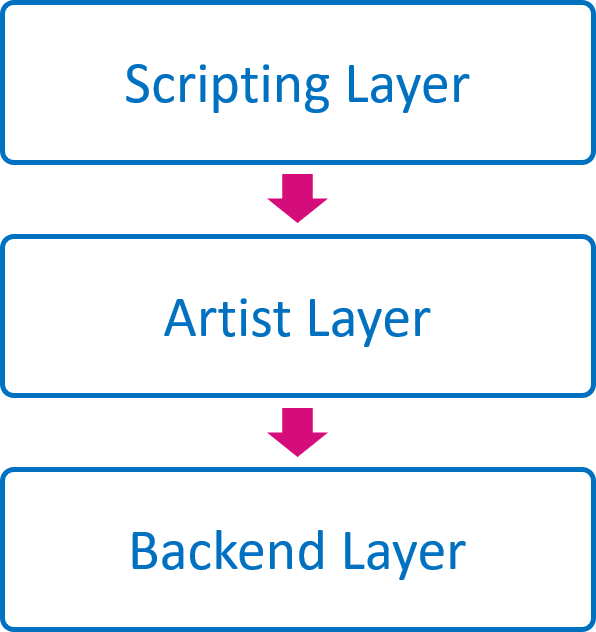
首先進入到命令提示字元cmd （按下右鍵以系統管理員身分執行）

1. 建議執行python -m pip install -U pip setuptools 進行升級，這樣之後下載套件就會是最新的版本。
2. 接著鍵入python -m pip install matplotlib進行自動的安裝，系統會自動下載安裝包。
3. 安裝完成後，可以用python -m pip list查看本機的安裝的所有模塊，確保Matplotlib已經安裝成功。
4. 接下來記得將spyder編輯器重新啟動，即可顯示圖表。

https://www.lccnet.com.tw/lccnet/article/details/2238

**1.Matplotlib Architecture**

Matplotlib的架構主要分三個部分，分別是**Backend Layer**、**Artist Layer**及**Scripting Layer**，每一個Layer都有其負責的功能如下：

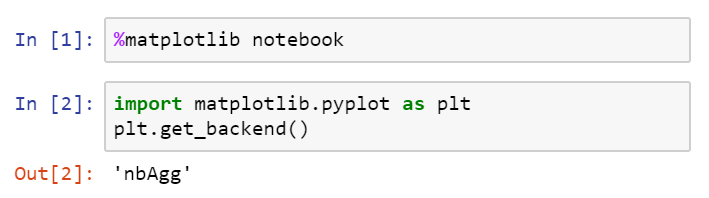


* **Backend Layer**

網路上有許多參考資料都提到matplotlib的Backend Layer，但對於初學者往往會對於一些術語及解釋感到困惑，我試著用比較簡單的方式說明。在使用matplotlib來繪圖或製作數據圖表時，由於每個人的開發環境或練習的介面不同，為了要能有效呈現其強大的繪圖功能，就必須有對應支援的backend來溝通，例如有些人是利用python shell來跟matplotlib進行互動，此時你可以輸入下列資料，你將得到此時用的backend是’TkAgg’來幫你完成你所要的功能。

>>> import matplotlib.pyplot as plt   
>>> plt.get\_backend()   
‘TkAgg’

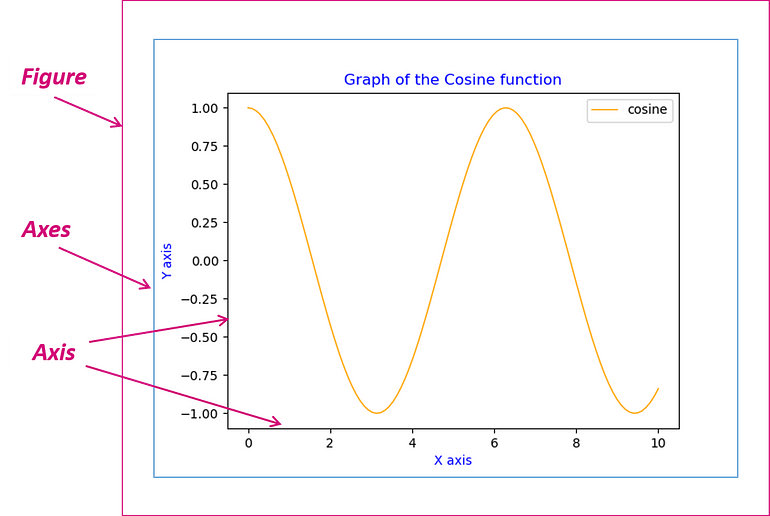
如果你是在[***Jupyter Notebook***](https://simplelearn.tw/jupyter-notebook-intro-and-tutorial/)當中使用inline的方式來進行互動，那你鍵入下列資料將得到此時的backend是’nbAgg’(如下圖)



當然因應不同使用者的環境及需求，所對應的後端(backend)也會不一樣，而為了支援這些不一樣的使用情況，matplotlib可以定義不同的輸出功能，並且由使用者前端的程式代碼(繪圖代碼)，交給後端來完成工作並製作漂亮或複雜的圖形。

* **Artist Layer**  
  Artist Layer有兩種型式：**基元(primitives)**和**容器(containers)**。基元主要是要處理我們在畫布上欲繪製的標準圖形對象，例如Line2D、Rectangle、Text、AxesImage等等，而容器則是要處理放置它們的位置，例如Figure, Subplot, and Axes。所以Artist Layer可以清楚知道圖形是如何由各子圖組成的，以及物體在給定的軸坐標系(Axes)中的相對位置。
* **Scripting Layer**  
  如果我們正在寫一個應用程式而來使用matplotlib，我們可能不會關心腳本層(Scripting Layer)，但這一層可以幫助我們簡化及加速與環境的互動，以方便快速建立圖形。前面所提Backend Layer著重於為工具包提供一個重要的接口，來呈現Artist Layer的基元(primitives)和容器(containers) ，但Scripting Layer卻是以用戶的角度來簡化與其他層一起工作的任務。而在本文章中的範例，所使用的腳本層就做pyplot。

上述說明了matplotlib就像個多層次的蛋糕一樣。有一個後端(Backend Layer)來處理實際的繪圖，而在他的上面則有一群藝術家(Artist Layer)描述著許多數據的排列方式，再根據腳本(Scripting Layer)實際上創建了這些藝術家並將它們一起編排起來表演。相信這樣的說明再配合後面的範例，將可讓初學者更了解其架構。



**2.Plot Anatomy & Workflow**

* **Plot Anatomy**  
  我們將一張圖的架構解析如下，其中圖形(Figure)包含並管理圖中的許多元素，而主要架構則決定將圖形與行為分開。
* **Workflow**用matplotlib串建新圖時，可參考下列基本步驟及範例：  
  Step 1 ：Prepare Data  
  Step 2 ：Create Plot  
  Step 3 ：Plot  
  Step 4 ：Customize Plot  
  Step 5 ：Save Plot  
  Step 6 ：Show Plot