

BOKEH - 輕鬆以網頁呈現視覺化圖表



簡報閱讀



範例與作業



問題討論



學習心得(完成)



重要知識點



- 學習並了解 Bokeh 如何應用
- 初步了解 BOKEH 互動式 GUIDE LINE
- 完成今日課程後你應該可以了解
 - # 使用Bokeh 將數據轉換為視覺化

Bokeh

Bokeh 為一個 Python 函式庫，提供了各式各樣的視覺化必須的輔助函式，同時也將網頁前端的技術細節包裝成一個個的 Python 函式與參數供我們呼叫，讓我們不再需要編輯 HTML 與 JavaScript 便能製作網頁前端視覺化。

Bokeh 相關套件安裝

相關套件安裝

- NumPy, Jinja2, Six, Requests, Tornado
- PyYaml, DateUtil, pandas, bokeh, panel

需額外注意套件版本：

- Jinja2 >=2.7
- numpy >=1.7.1
- packaging >=16.8
- pillow >=4.0
- python-dateutil >=2.1
- PyYAML >=3.10
- six >=1.5.2
- tornado >=4.3

Bokeh 程式的基本運作

要注意的是bokeh會預設連BokehJS cdn，但連線有時不是很穩定，這時可多加“INLINE”環境變數設定，讓BokehJS驅動於local python env。



`bokeh.io.output_notebook(INLINE)`

Bokeh可以在Jupyter呈現開發也可以跳轉出html檔，可自由設定，預設是跳轉html檔(`output_file()`)。

若要更改預設，必須加上`bokeh.io.reset_output()`重設環境預設。

`output_notebook()` # jupyter呈現
`output_file()` # html呈現

fig物件可設定tools參數，圖表會自帶縮放、重整、儲存等功能。

載入套件函數

- `from bokeh.plotting import figure`
- `from bokeh.plotting import output_file`
- `from bokeh.plotting import show`
- `from bokeh.models import widgets`
- `from bokeh.io import output_notebook`

讓網頁直接輸出在NOTEBOOK

- `output_notebook()`

設定資料與輸出檔案

- `output_file("out.html")`

利用 Bokeh 繪製圖表

創建一個空畫板 (物件p)



在畫板上繪製一條座標(1,5)到(5,1)的直線

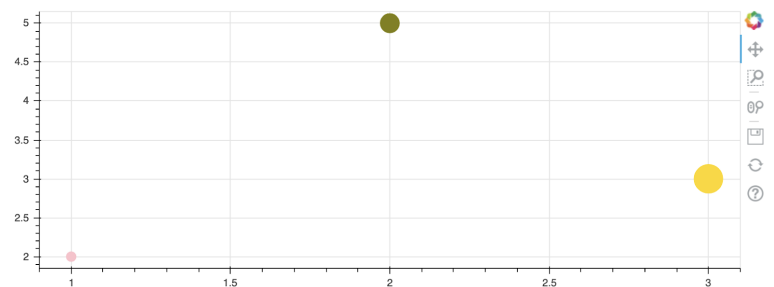
- `p.line([1,2,3,4,5], [5,4,3,2,1])`

開啟產生的 HTML 檔 (HTML + JavaScript , 自動生成)

- `show(p)`

Bokeh 也提供了一些預先建置好的圖表供我們使用，例如用 `circle` 繪製點圖：

```
from bokeh.plotting import figure, output_file,
show
p = figure(width=800,height=300)
p.circle([1,2,3],[2,5,3], size=[10,20,30], color=
["pink","olive","gold"])
show(p)
```



除了繪製點圖的 `circle` 函式以外，另外還有以下不同圖形的繪製函數：

`bokeh.charts.Bar` — 製作長條圖

`bokeh.charts.BoxPlot` — 製作盒鬚圖

`bokeh.charts.HeatMap` — 製作熱圖

`bokeh.charts.Donut` — 製作甜甜圈圖

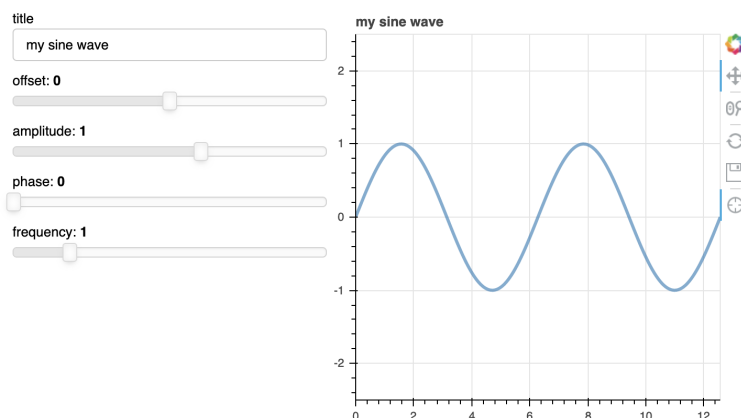
如軸線、形狀等等。用來打造各種元件。

- `bokeh.plotting` — 為我們處理掉一些基本細節（例如格點與軸線），但保留客製化的彈性。
- `bokeh.charts` — 直接使用各種完整圖表，例如長條圖、盒鬚圖等等。

網頁元件與互動圖表

我們先透過 `IFrame` 直接呼叫 bokeh 範例來看看

```
from IPython.display import IFrame
IFrame('https://demo.bokeh.org/sliders',
width=900, height=500)
```



可以看到 bokeh 的範例上，有許多互動功能右側更有一排小工具可供使用。

畫圖的API: `bokeh.plotting`

```
from bokeh.plotting import figure, output_file,
show
```

```
# 讓網頁直接輸出在NOTEBOOK
```

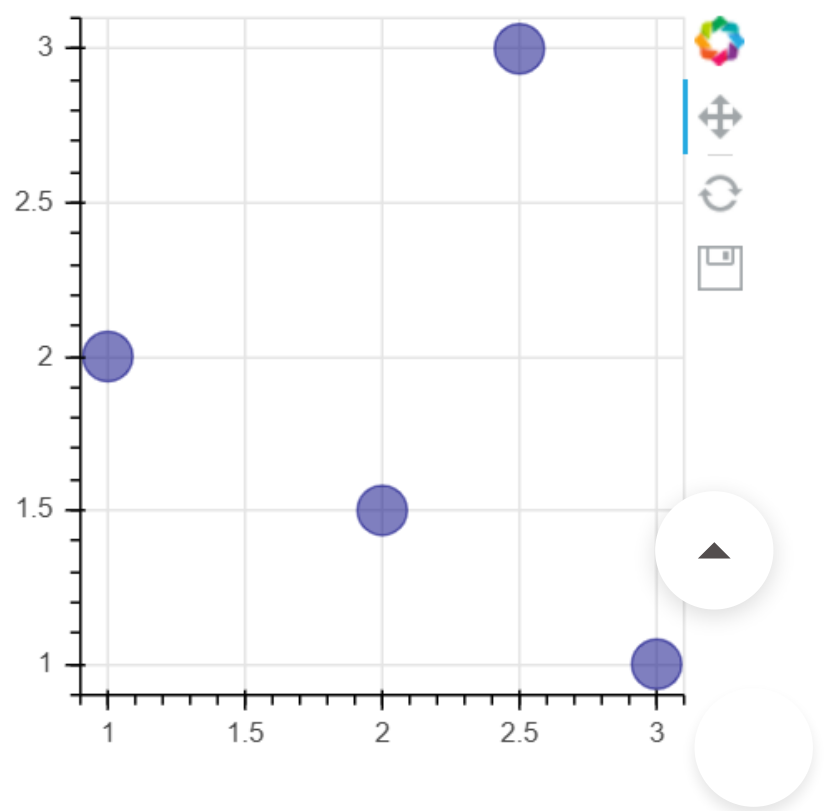
```
# create a Figure object
p = figure(plot_width=300, plot_height=300,
tools="pan,reset,save")
```

```
# 添加圓點的分佈圖
# p.circle (x/y 座標, size, color, and 透明度)
p.circle([1, 2.5, 3, 2], [2, 3, 1, 1.5], size=20,
color="navy", alpha=0.5)
```

```
# specify how to output the plot(s)
output_file("foo.html")
```

重點是讓用戶將他們希望顯示的視覺字形與其數據相關聯。

bokeh.plotting 界面中的主要類是 `figure()` 函數。這將創建一個 Figure 模型，該模型包括用於向繪圖添加不同種類的字形的方法。





9+



```
from bokeh.plotting import figure, output_file,  
show
```

```
output_file("toolbar.html")
```

```
# create a new plot with the toolbar below
```

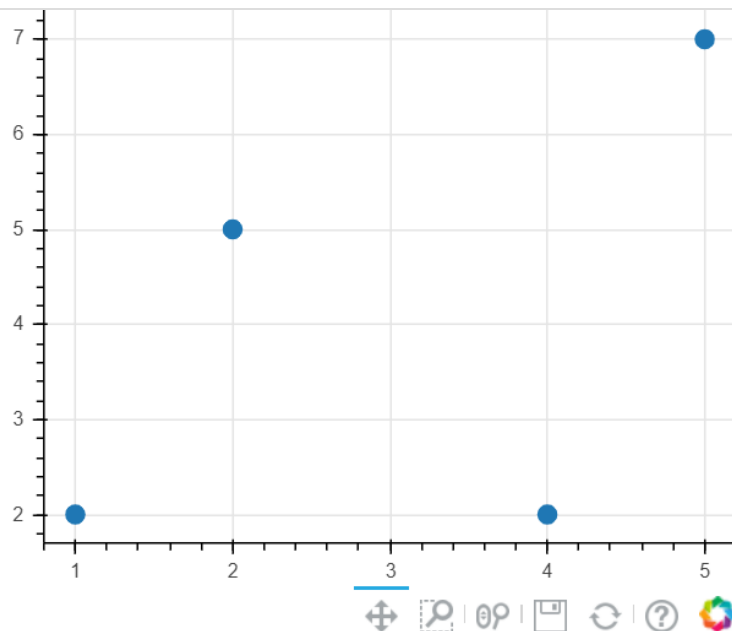
```
p = figure(plot_width=400, plot_height=400,  
           title=None, toolbar_location="below",  
           toolbar_sticky=False)
```

```
p.circle([1, 2, 3, 4, 5], [2, 5, 8, 2, 7], size=10)  
show(p)
```

Bokeh 附帶了許多交互式工具，可用於報告信息，更改繪圖參數（例如縮放級別或範圍範圍）或添加，編輯或刪除字形。

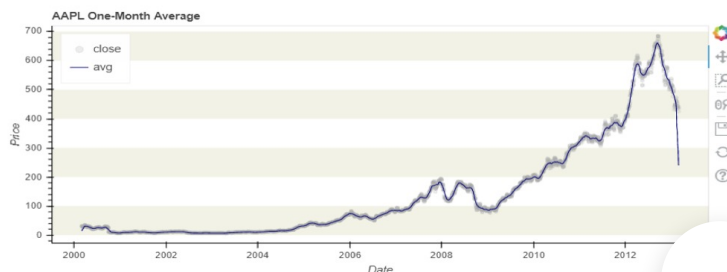
工具欄位置與默認軸發生衝突，在這種情況下，將 `toolbar_sticky` 選項設置為 `False` 將工具欄移動到繪製軸的區域之外。





知識點回顧

- Bokeh 以成為交互式數據可視化的庫而自豪。
- 不同於 Matplotlib 和 Seaborn 等 Python 可視化領域的流行同行，Bokeh 使用 HTML 和 JavaScript 渲染其圖形。這使其非常適合構建基於Web的儀表板和應用程序。但是，它是用於探索和理解數據或為項目或報告創建漂亮的自定義圖表的功能同樣強大的工具。



延伸閱讀


```
from bokeh.plotting import figure, output_file,
show
#套用內建的模型
from bokeh.themes import built_in_themes
#導入輸出的繪製套件
from bokeh.io import curdoc
#建立數據
x = [1, 2, 3, 4, 5]
y = [6, 7, 6, 4, 5]
#設定輸出
output_file("caliber.html")
curdoc().theme = 'caliber'
p = figure(title='caliber', plot_width=300,
plot_height=300)
p.line(x, y)
show(p)
```

Bokeh In Python

Python 中使用 Bokeh 進行數據視覺化，第一部分：入門

利用 Python 中 Bokeh 實現資料視覺化，第二部分：互動

Bokeh 的主要概念是，圖形一次只能建立一層。我們首先創建一個圖形，然後向該圖形添加稱為字形的元素。（對於使用 ggplot 的人來說，字形（shape）的思想與一次添加到一個“層”的圖形中的幾何（geom）思想相同。）字形可以根據所需的用途採用多種形狀：圓，線，補丁，條形，弧形等。讓我們通過製作帶有正方形和圓形的基本圖表來說明字形的想

重要知識點

Bokeh

Bokeh 相關套件安裝

Bokeh 程式的基本運作

網頁元件與互動圖表

上。最後，我們顯示我們的繪圖。

[下一步：閱讀範例與完成作業](#)

