



視覺化套件

Bokeh & folium

Bokeh

Bokeh介紹

- ◆ 在資料視覺化方面，matplotlib、Seaborn套件幾乎可以滿足大多多的繪圖條件，但這些套件只能做出靜態圖表，若想要讓使用者和圖表能直接互動，那就要使用bokeh套件
- ◆ 使用bokeh製作出來的圖表，可以使用滑鼠拖曳或縮放

The Bokeh logo features a stylized camera aperture icon composed of eight colorful segments (green, teal, blue, purple, pink, orange, yellow, and light green) arranged in a circular pattern. The word "bokeh" is written in a large, black, sans-serif font, with the aperture icon replacing the letter "o".

bokeh

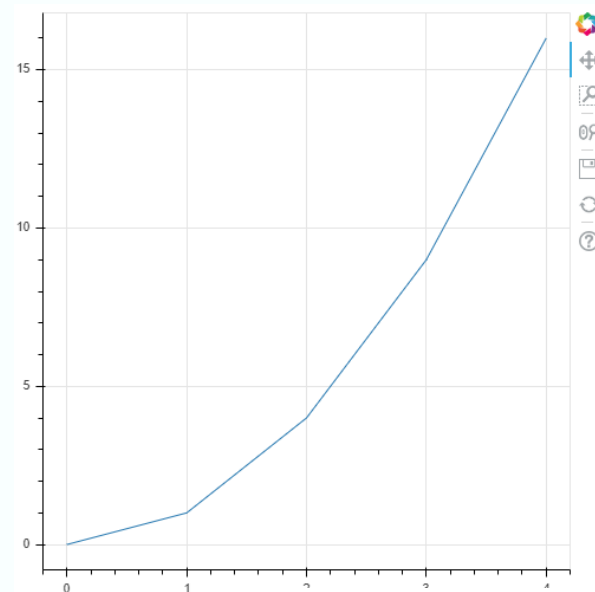
折線圖

```
from bokeh.plotting import figure, show

# 設定資料
x = [0, 1, 2, 3, 4]
y = [0, 1, 4, 9, 16]

# 設定繪圖板寬高
p = figure(plot_width=500, plot_height=500)

# 繪製折線圖
p.line(x, y)
show(p)
```



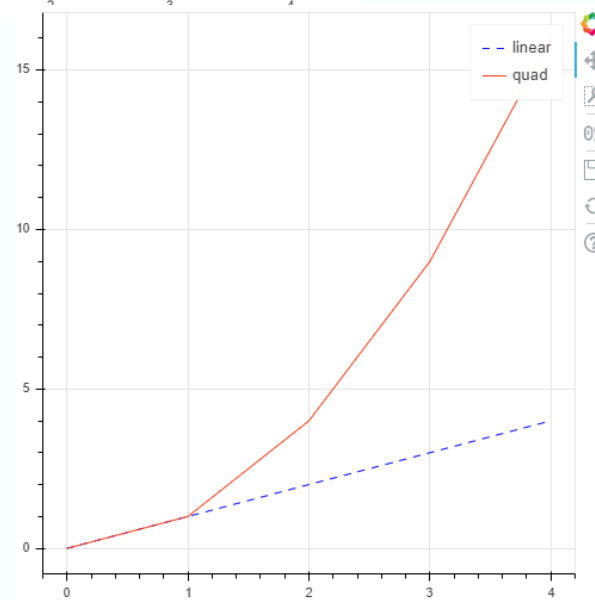
```
from bokeh.plotting import figure, show

# 設定資料
x = [0, 1, 2, 3, 4]
y = [0, 1, 4, 9, 16]

# 設定繪圖板寬高
p = figure(plot_width=500, plot_height=500)

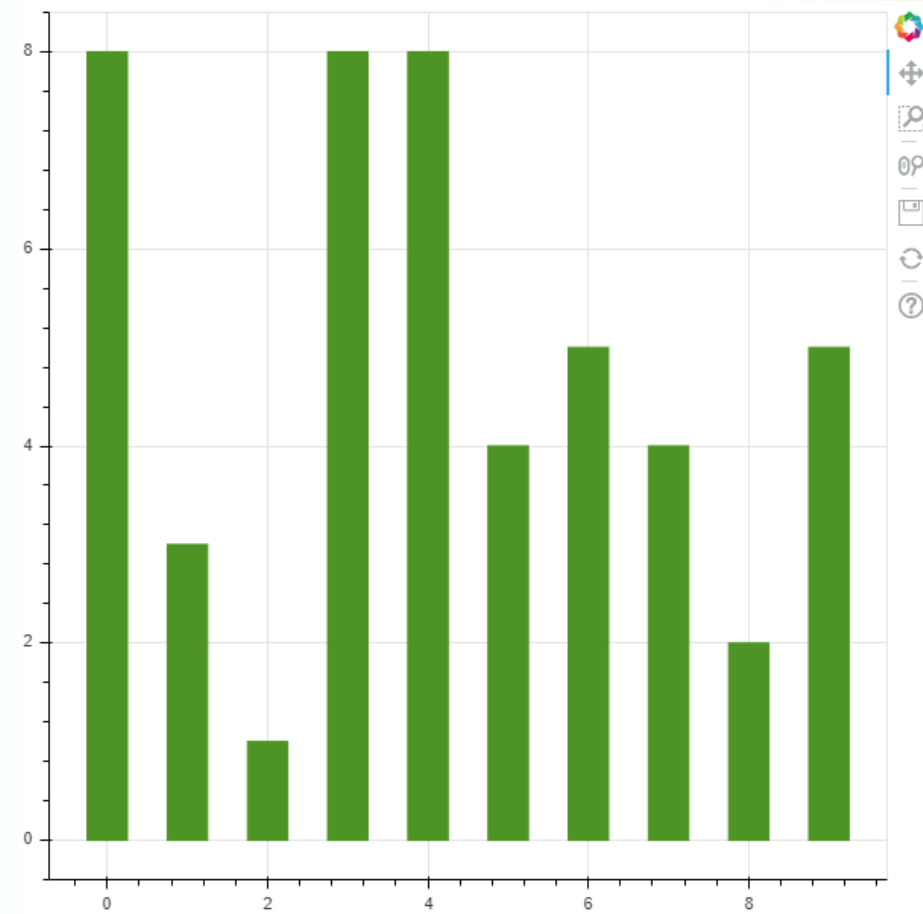
# 繪製折線圖
p.line(x, x, line_color='blue', legend_label='linear', line_dash='dashed')
p.line(x, y, line_color='#FF4310', legend_label='quad', line_dash='solid')

show(p)
```



長條圖

```
1 from bokeh.plotting import figure, show
2 import numpy as np
3
4 # 設定資料
5 x = np.arange(10)
6 y = np.random.randint(1,11,10)
7
8 # 設定畫布
9 p = figure()
10
11 # 繪製長條圖
12 p.vbar(x, top=y, width=0.5, bottom=0, color="#499312")
13
14 show(p)
```



散點圖

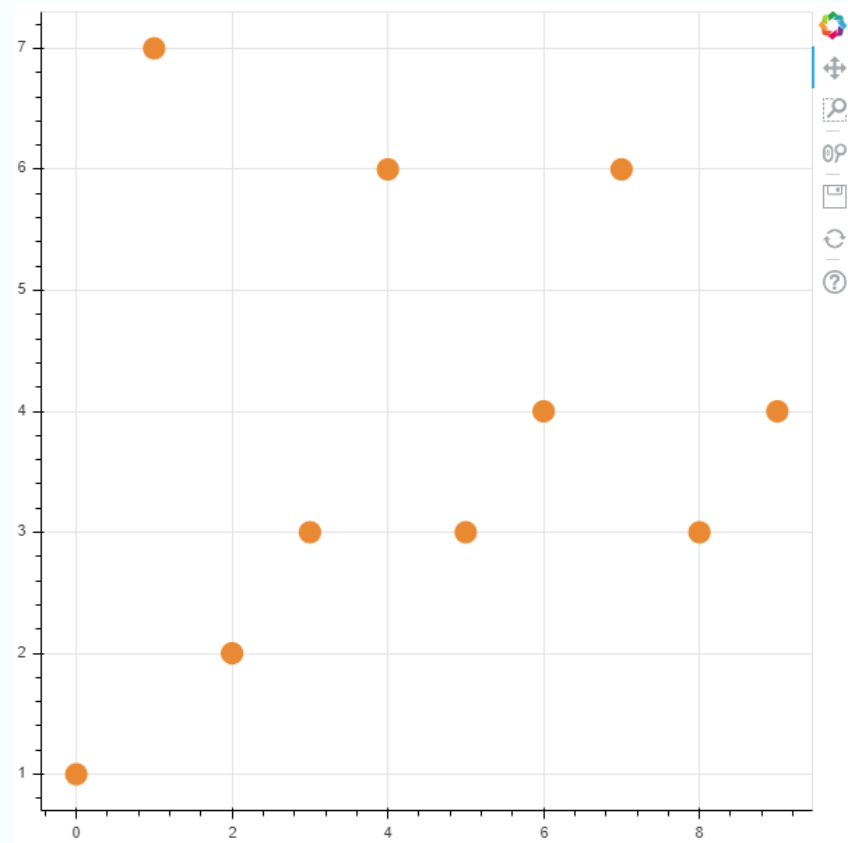
```
from bokeh.plotting import figure, show
import numpy as np

# 設定資料
x = np.arange(10)
y = np.random.randint(1,10,10)

# 設定繪圖
p = figure()

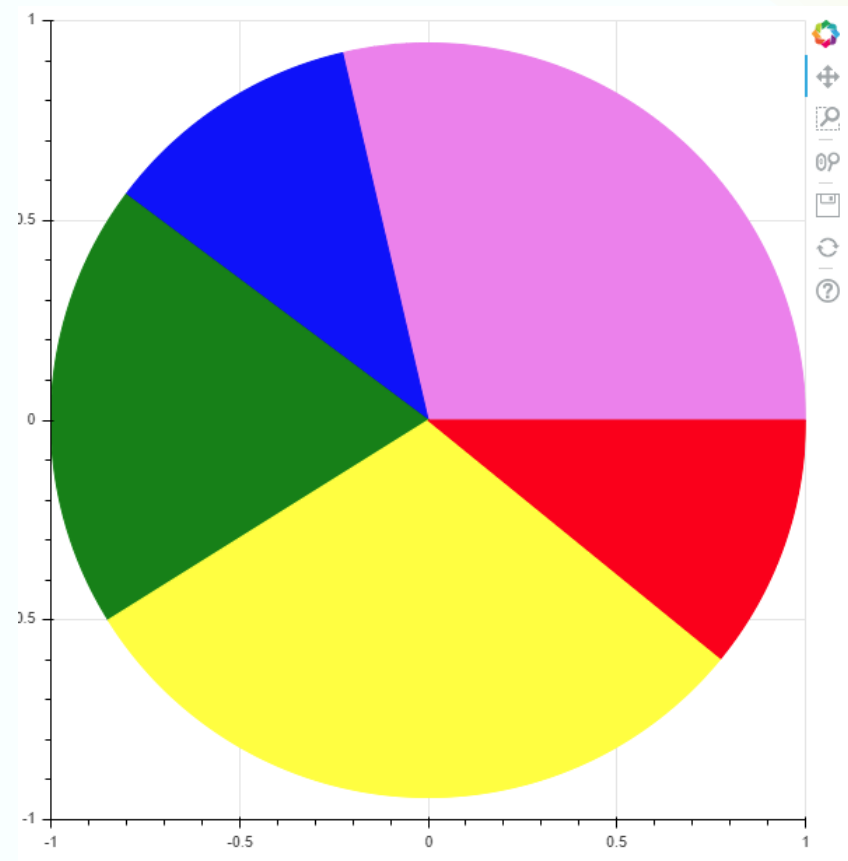
# 繪製散點圖(散點大小15)
p.circle(x, y, size=15, color="#ED8B21")

show(p)
```



圖餅圖

```
1  from bokeh.plotting import figure, show
2
3  # 設定資料
4  x = 0
5  y = 0
6  radius = 1
7
8  start_angle = [0, 1.8, 2.5, 3.7, 5.6]
9  end_angle = [1.8, 2.5, 3.7, 5.6, 0]
10 color = ["violet", "blue", "green", "yellow", "red"]
11
12 p = figure()
13
14 p.wedge(x, y, radius, start_angle, end_angle, color=color)
15
16 show(p)
```



儲存檔案

- ◆ 若想將bokeh產生的圖存檔，可以使用 `save()` 功能
- ◆ 使用前需要先匯入 `save`、`output_file`

```
from bokeh.plotting import figure, show, save, output_file
import numpy as np

# 設定資料
x = np.arange(10)
y = np.random.randint(1,10,10)

# 設定繪圖
p = figure()

# 繪製散點圖(散點大小15)
p.circle(x, y, size=15, color="#00ff00")

# 設定檔名
output_file("output.html")

# 儲存檔案
save(p)
show(p)
```


folium

安裝folium套件

- ◆ folium是基於Leaflet的JavaScript Library的Python地圖視覺化套件，讓使用者可以快速產生可互動的地圖
- ◆ 使用前需要先安裝folium套件，語法為：

`pip install folium`

```
Anaconda Prompt (Anaconda3)

(base) C:\Users\selp\>pip install folium
Collecting folium
  Downloading folium-0.12.1.post1-py2.py3-none-any.whl (95 kB)
    | 95 kB 681 kB/s
Requirement already satisfied: requests in c:\users\selp\anaconda3\lib\site-packages (from folium) (2.22.0)
Collecting branca>=0.3.0
  Downloading branca-0.5.0-py3-none-any.whl (24 kB)
Requirement already satisfied: Jinja2>=2.9 in c:\users\selp\anaconda3\lib\site-packages (from folium) (2.11.1)
Requirement already satisfied: numpy in c:\users\selp\anaconda3\lib\site-packages (from folium) (1.18.1)
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in c:\users\selp\anaconda3\lib\site-packages (from requests->folium) (3.0.4)
Requirement already satisfied: urllib3!=1.25.0,!1.25.1,<1.26,>=1.21.1 in c:\users\selp\anaconda3\lib\site-packages (from requests->folium) (1.25.8)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\selp\anaconda3\lib\site-packages (from requests->folium) (2019.11.28)
Requirement already satisfied: idna<2.9,>=2.5 in c:\users\selp\anaconda3\lib\site-packages (from requests->folium) (2.8)
Requirement already satisfied: MarkupSafe>=0.23 in c:\users\selp\anaconda3\lib\site-packages (from Jinja2>=2.9->folium) (1.1.1)
Installing collected packages: branca, folium
Successfully installed branca-0.5.0 folium-0.12.1.post1
```

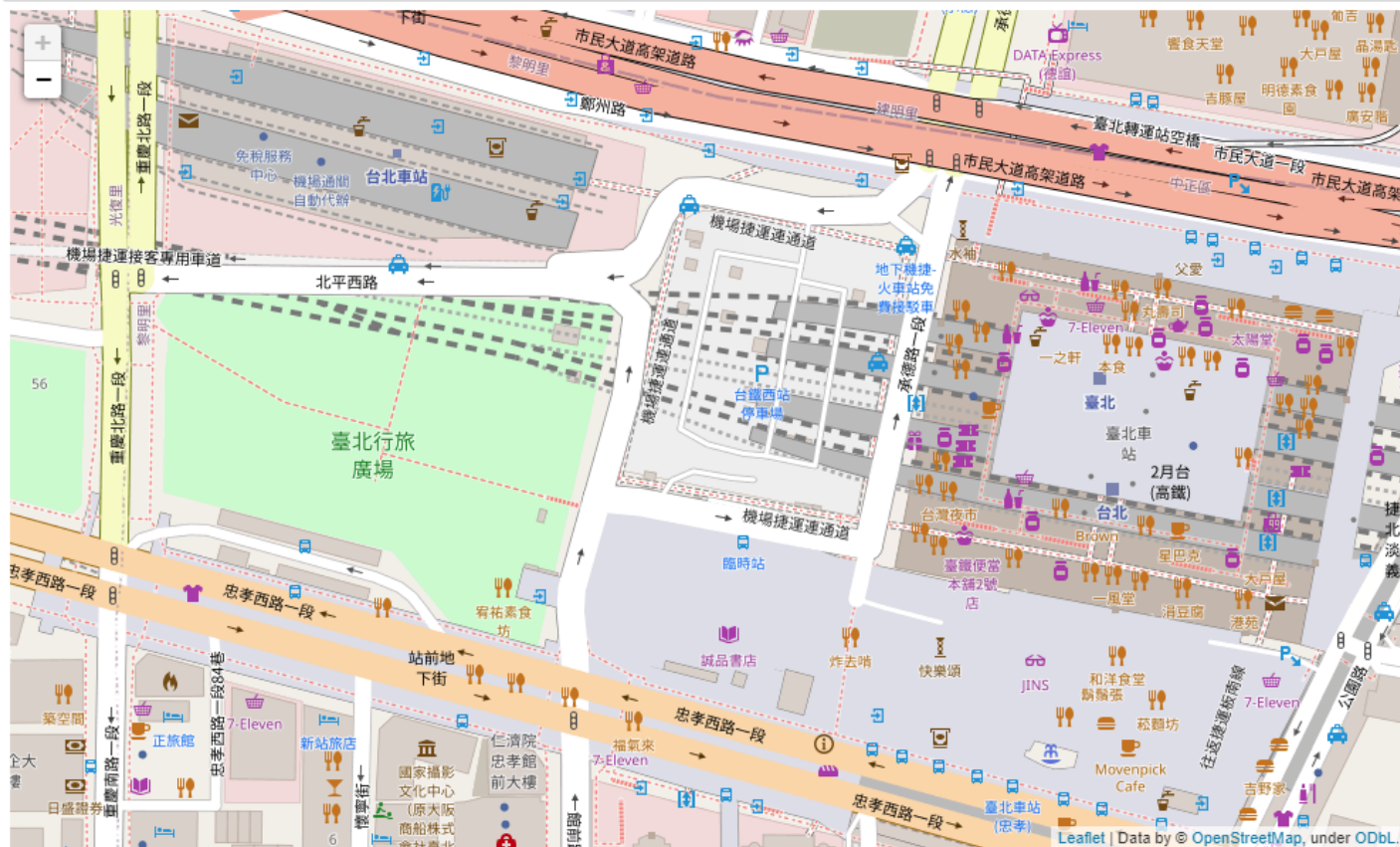
繪製地圖

◆ 在jupyter notebook裡輸入程式碼

```
In [1]: import numpy as np
import folium

fmap = folium.Map(location=[25.0477596,121.515526], zoom_start=20)
fmap
fmap.save('map.html') #若想將檔案儲存, 可以加上這句
```

Out [1]:



地圖標籤

◆ folium.Marker(location=[標籤經緯度], popup="提示字")

```
In [9]: import numpy as np
import folium

fmap = folium.Map(location=[25.0477596,121.515528], tiles="OpenStreetMap", zoom_start=17)
fmap.add_child(folium.Marker(location=[25.0462322,121.5162631], popup='新光三越站前店')) # 增加地圖標籤
fmap
```

Out [9]:



繪製多個地圖標籤

```
import numpy as np
import folium

list_station = [[35.69018, 139.70038, "新宿"], [35.67129, 139.70261, "原宿"], [35.65796, 139.70149, "渋谷"]]

fmap = folium.Map(location=[35.6761354, 139.7475033], tiles="OpenStreetMap", zoom_start=13)
for i in list_station:
    fmap.add_child(folium.Marker(location=[i[0], i[1]], popup=i[2]))
fmap
```

