

Python 数据分析与应用

实验四：Matplotlib 数据可视化

丁烨

dingye@dgut.edu.cn

计算机科学与技术学院

2023-11-15

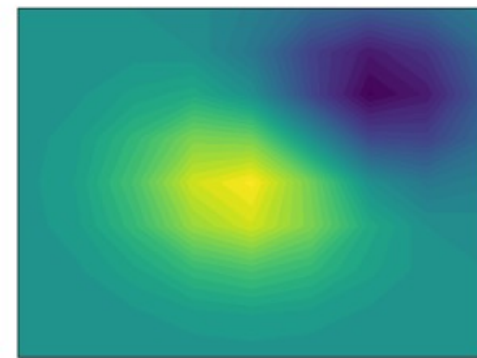
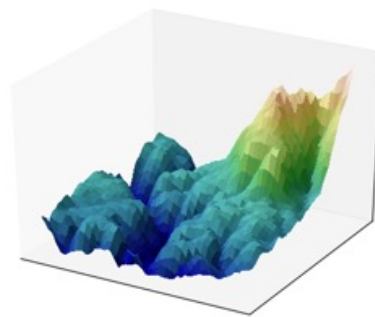
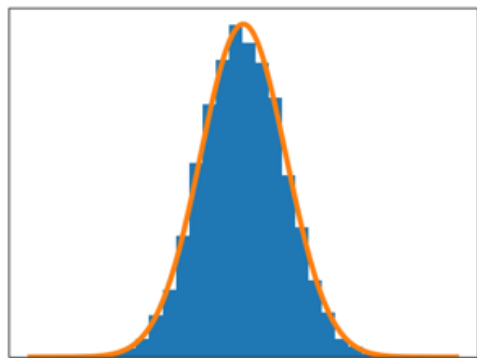
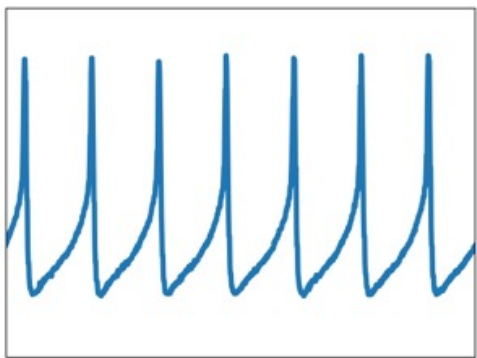


東莞理工學院
DONGGUAN UNIVERSITY OF TECHNOLOGY

Matplotlib 简介

概述

- ❖ Matplotlib 是 Python 的一个扩展程序库
- ❖ 官方网站: <https://matplotlib.org/>
- ❖ 源代码: <https://github.com/matplotlib/matplotlib>
- ❖ 提供了大量的绘图组件用于绘制图表, 图表可以保存成多种格式
- ❖ 包含一个互动的图表控制平台, 可以跨平台方便的修改图表样式



- ❖ 使用 pip 安装 Matplotlib:
- ❖ `pip3 install -U matplotlib`

- ❖ 如果安装不成功, 可尝试使用 apt 安装:
- ❖ `sudo apt install python3-matplotlib`

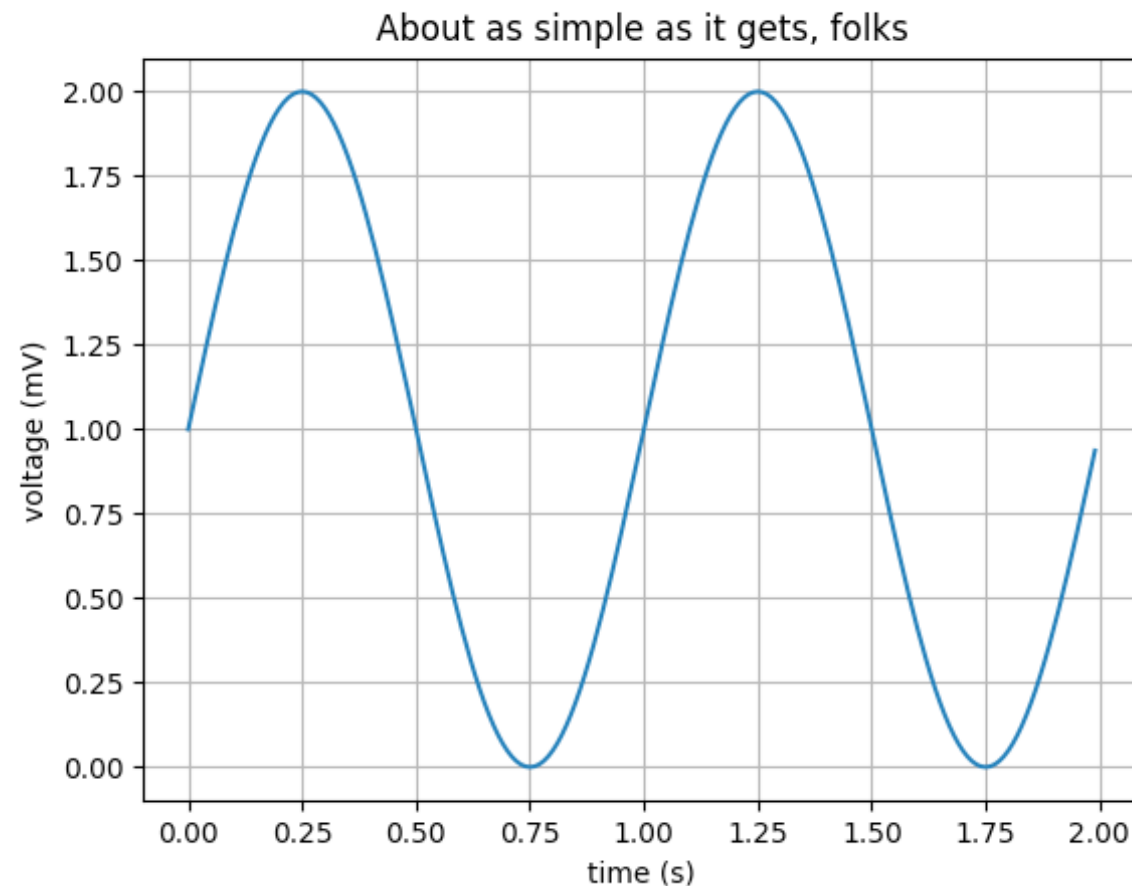
```
import matplotlib
import matplotlib.pyplot as plt
import numpy as np

t = np.arange(0.0, 2.0, 0.01)
s = 1 + np.sin(2 * np.pi * t)

fig, ax = plt.subplots()
ax.plot(t, s)

ax.set(xlabel='time (s)', ylabel='voltage (mV)',
       title='About as simple as it gets, folks')
ax.grid()

fig.savefig("test.png")
plt.show()
```



```
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.cbook as cbook

# Load a numpy record array from yahoo csv data with fields date, open, close,
# volume, adj_close from the mpl-data/example directory. The record array
# stores the date as an np.datetime64 with a day unit ('D') in the date column.

with cbook.get_sample_data('goog.npz') as datafile:
    price_data = np.load(datafile)['price_data'].view(np.recarray)
price_data = price_data[-250:] # get the most recent 250 trading days

delta1 = np.diff(price_data.adj_close) / price_data.adj_close[:-1]

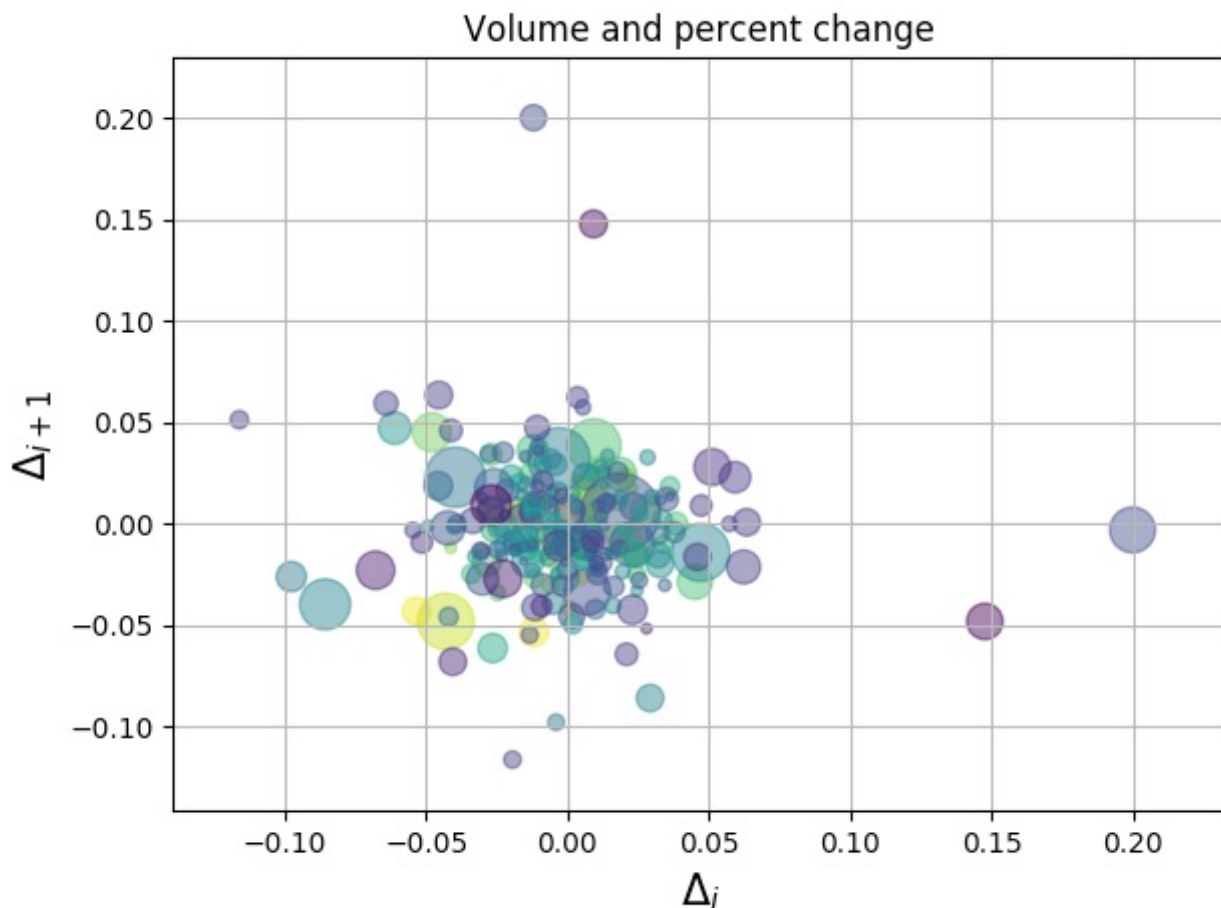
# Marker size in units of points^2
volume = (15 * price_data.volume[:-2] / price_data.volume[0])**2
close = 0.003 * price_data.close[:-2] / 0.003 * price_data.open[:-2]
```

```
fig, ax = plt.subplots()  
ax.scatter(delta1[:-1], delta1[1:], c=close, s=volume, alpha=0.5)
```

```
ax.set_xlabel(r'$\Delta_i$', fontsize=15)  
ax.set_ylabel(r'$\Delta_{i+1}$', fontsize=15)  
ax.set_title('Volume and percent change')
```

```
ax.grid(True)  
fig.tight_layout()
```

```
plt.show()
```

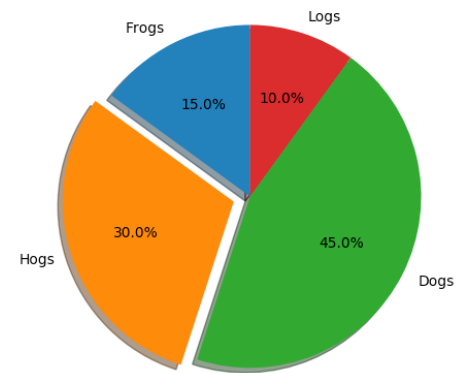


```
import matplotlib.pyplot as plt

# Pie chart, where the slices will be ordered and plotted counter-clockwise:
labels = 'Frogs', 'Hogs', 'Dogs', 'Logs'
sizes = [15, 30, 45, 10]
explode = (0, 0.1, 0, 0) # only "explode" the 2nd slice (i.e. 'Hogs')

fig1, ax1 = plt.subplots()
ax1.pie(sizes, explode=explode, labels=labels, autopct='%1.1f%%',
        shadow=True, startangle=90)
ax1.axis('equal') # Equal aspect ratio ensures that pie is drawn as a circle.

plt.show()
```



```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.ticker import MaxNLocator
from collections import namedtuple

n_groups = 5

means_men = (20, 35, 30, 35, 27)
std_men = (2, 3, 4, 1, 2)

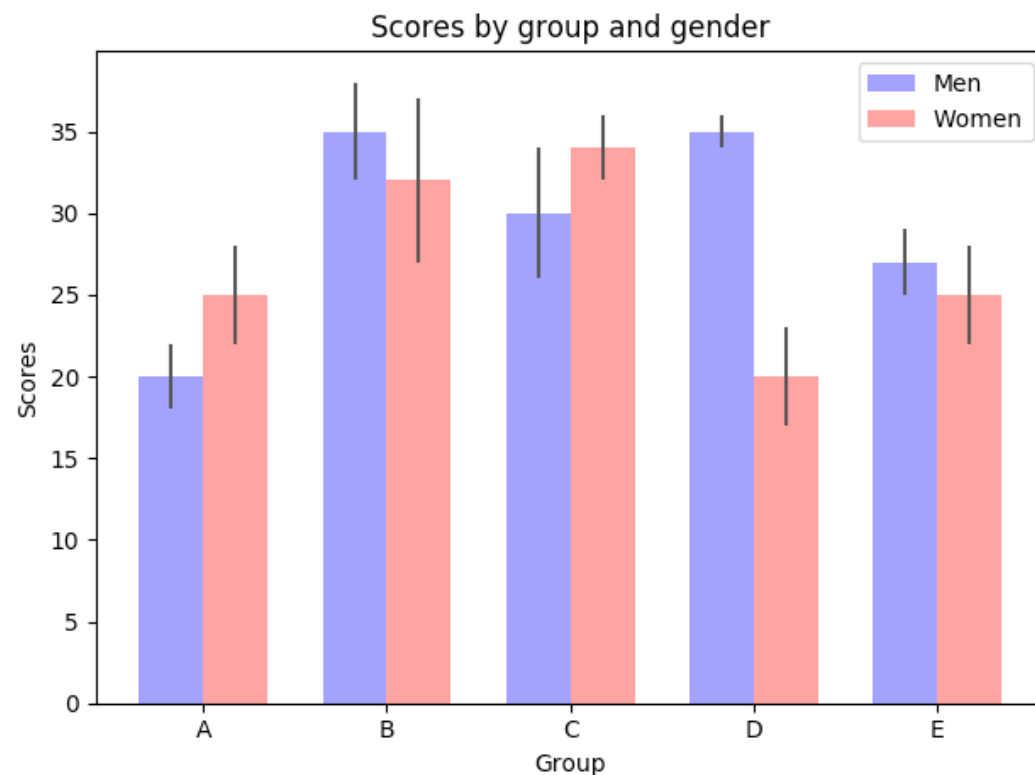
means_women = (25, 32, 34, 20, 25)
std_women = (3, 5, 2, 3, 3)

fig, ax = plt.subplots()

index = np.arange(n_groups)
bar_width = 0.35
opacity = 0.4
error_config = {'ecolor': '0.3'}
```



```
rects1 = ax.bar(index, means_men, bar_width,  
                alpha=opacity, color='b',  
                yerr=std_men, error_kw=error_config,  
                label='Men')  
  
rects2 = ax.bar(index + bar_width, means_women, bar_width,  
                alpha=opacity, color='r',  
                yerr=std_women, error_kw=error_config,  
                label='Women')  
  
ax.set_xlabel('Group')  
ax.set_ylabel('Scores')  
ax.set_title('Scores by group and gender')  
ax.set_xticks(index + bar_width / 2)  
ax.set_xticklabels(('A', 'B', 'C', 'D', 'E'))  
ax.legend()  
  
fig.tight_layout()  
plt.show()
```



```
import matplotlib
import numpy as np
import matplotlib.pyplot as plt

np.random.seed(19680801)

# example data
mu = 100 # mean of distribution
sigma = 15 # standard deviation of distribution
x = mu + sigma * np.random.randn(437)

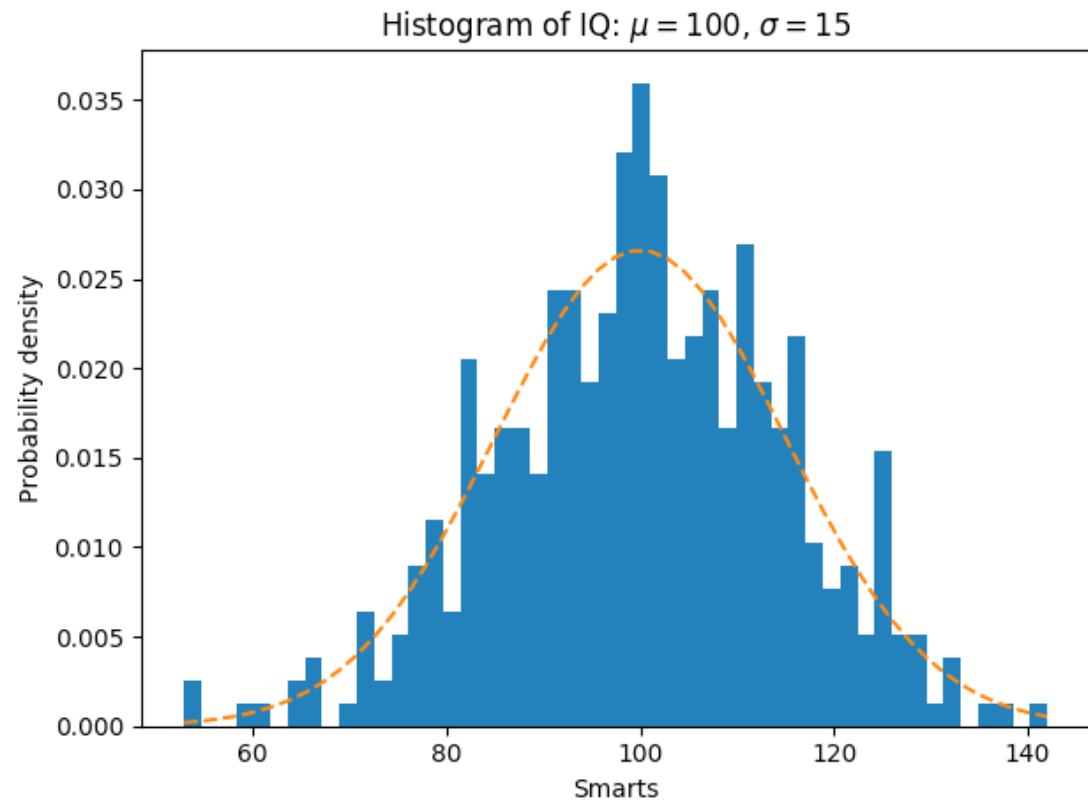
num_bins = 50

fig, ax = plt.subplots()

# the histogram of the data
n, bins, patches = ax.hist(x, num_bins, density=1)
```

```
# add a 'best fit' line
y = ((1 / (np.sqrt(2 * np.pi) * sigma)) *
      np.exp(-0.5 * (1 / sigma * (bins - mu))**2))
ax.plot(bins, y, '--')
ax.set_xlabel('Smarts')
ax.set_ylabel('Probability density')
ax.set_title(r'Histogram of IQ:  $\mu=100$ ,  $\sigma=15$ ' )

# Tweak spacing to prevent clipping of ylabel
fig.tight_layout()
plt.show()
```



Matplotlib 绘图练习

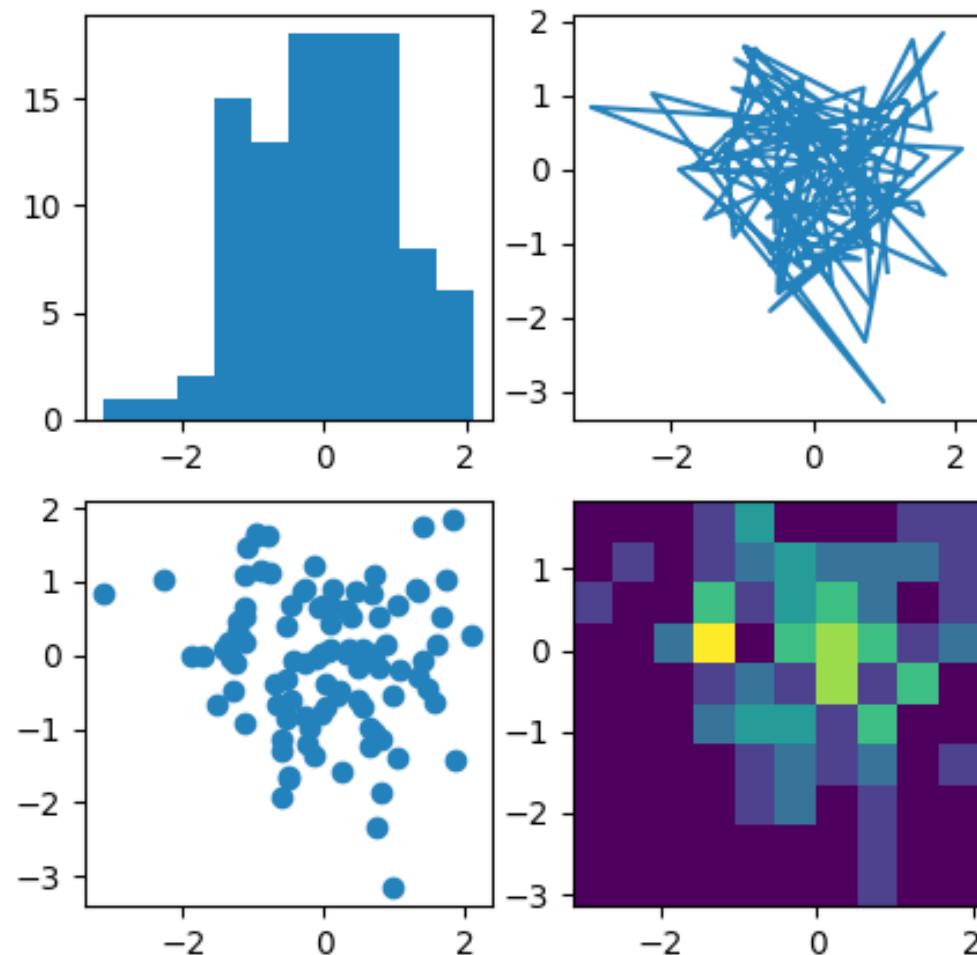
绘制子图

```
import matplotlib.pyplot as plt
import numpy as np
```

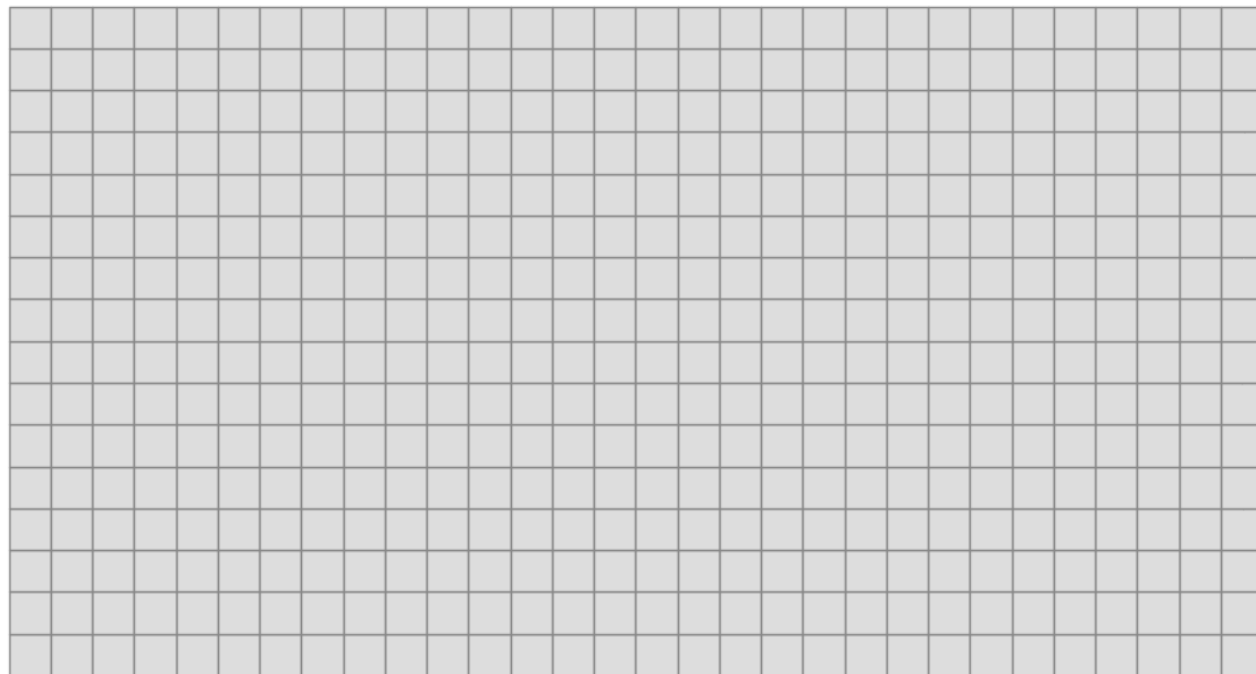
```
np.random.seed(19680801)
data = np.random.randn(2, 100)
```

```
fig, axs = plt.subplots(2, 2, figsize=(5, 5))
axs[0, 0].hist(data[0])
axs[1, 0].scatter(data[0], data[1])
axs[0, 1].plot(data[0], data[1])
axs[1, 1].hist2d(data[0], data[1])
```

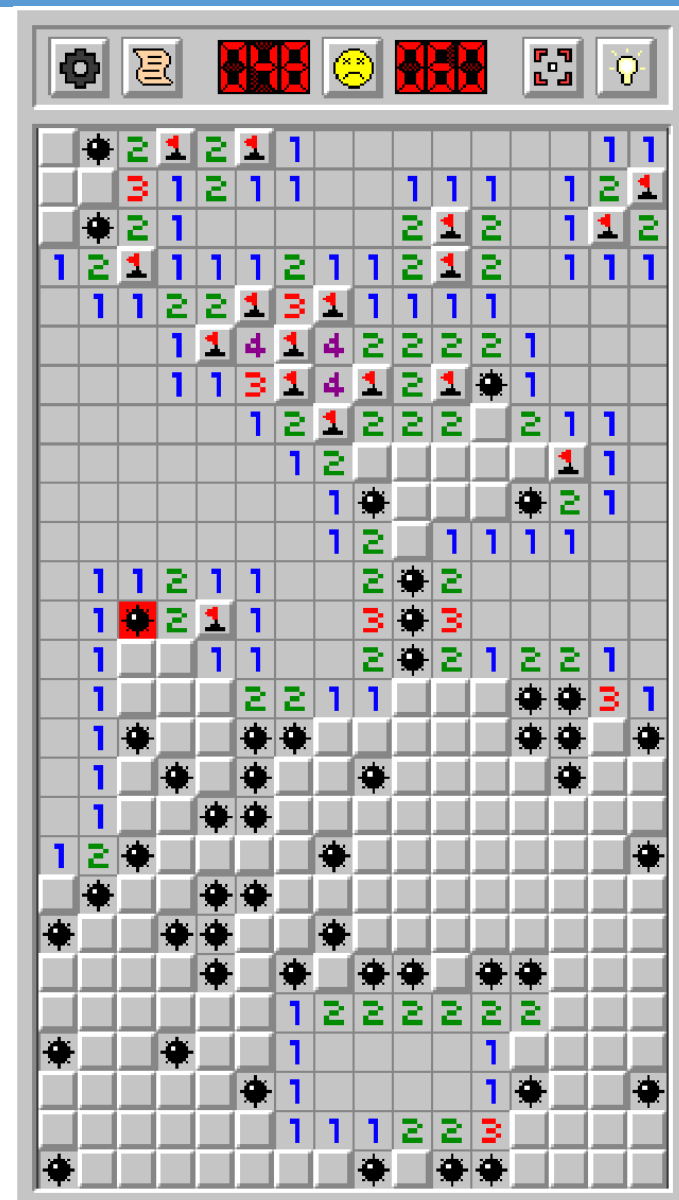
```
plt.show()
```



- ❖ 扫雷 (Minesweeper)
- ❖ 扫雷是一款单人的计算机游戏
- ❖ 此款游戏最有名的是 Windows 操作系统附带的版本
- ❖ 可在应用程序商店中免费下载



- ❖ 游戏开始时，玩家可看到空白方块排成的矩形，方块数可由玩家自行选择
- ❖ 如果玩家点开没有地雷的方块，会有一个数字显现其上，这个数字代表着邻近方块有多少颗地雷
- ❖ 玩家可在推测有地雷的方块上点鼠标右键，以放置旗帜来标明地雷的位置
- ❖ 游戏目标是找出所有没有地雷的方格，完成游戏
- ❖ 要是按了有地雷的方格，游戏失败
- ❖ 在数学上，扫雷的复杂度为 NP-complete

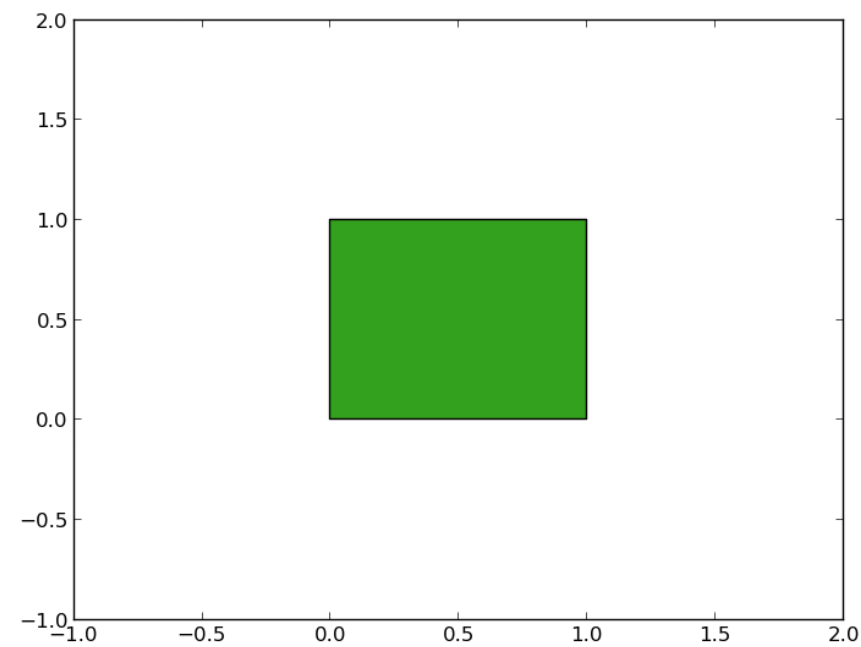



```
import numpy as np
import matplotlib.pyplot as plt

fig = plt.figure()
ax = fig.add_subplot(111, xlim=(-1, 2), ylim=(-1, 2))
polygon = plt.Polygon([[0, 0], [1, 0], [1, 1], [0, 1], [0, 0]])
ax.add_patch(polygon)

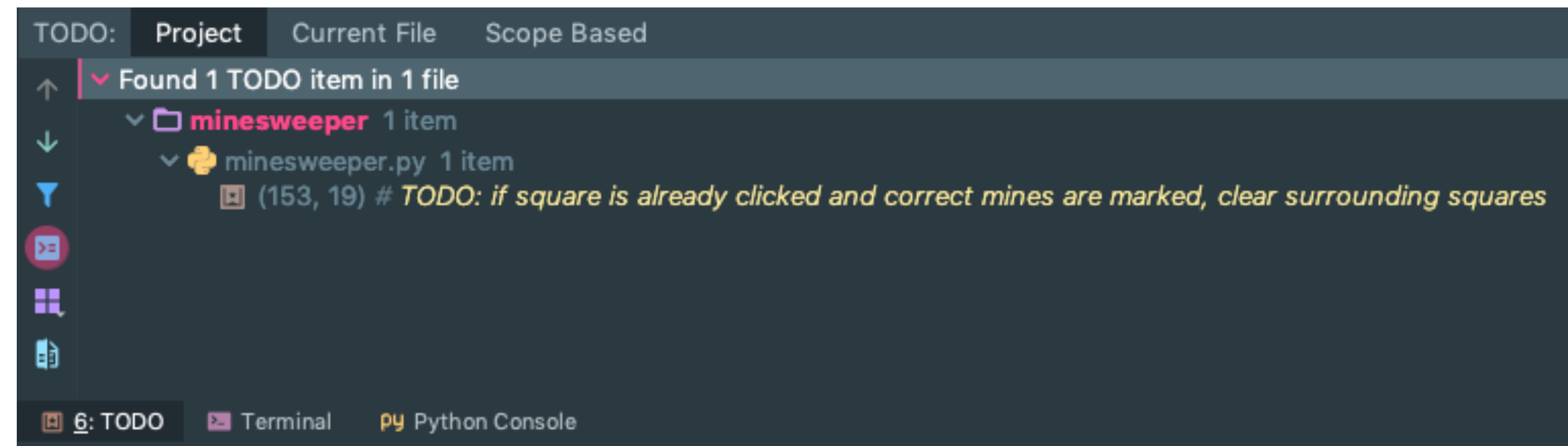
# Function to be called when mouse is clicked
def on_click(event):
    if polygon.contains_point((event.x, event.y)):
        polygon.set_facecolor(np.random.random(3))
        fig.canvas.draw()

# Connect the click function to the button press event
fig.canvas.mpl_connect('button_press_event', on_click)
plt.show()
```



❖ 源代码:

❖ <https://unicorn.org.cn/valency/src/minesweeper-0.5.7.png>



The screenshot shows an IDE interface with a TODO list. The list is titled "Found 1 TODO item in 1 file" and is organized as follows:

- Project
 - minesweeper 1 item
 - minesweeper.py 1 item
 - (153, 19) # *TODO: if square is already clicked and correct mines are marked, clear surrounding squares*

The IDE interface includes a sidebar with navigation icons (up, down, filter, search, etc.) and a bottom panel with tabs for "6: TODO", "Terminal", and "Python Console".

- ❖ 为扫雷游戏添加以下功能：
- ❖ 当玩家点击已经打开的方块时，如果方块显示的数字等于周围标记的地雷的数量，则一次性打开周围所有的方块



- ❖ 在作业系统中下载并完成本实验课对应实验报告
- ❖ <https://hw.dgut.edu.cn/>
- ❖ **注意：**所有标识为 * 的地方都需要填写
- ❖ **截止日期：**2023-11-22 23:59:59

课程名称：Python 数据分析与应用

学期：20

实验名称	Python 语言回顾			实验序号	
姓名	***	学号	***	班级	
实验地点	***	实验日期	***	指导老师	
教师评语	-			实验成绩	
				百分制	
同组同学	无				

四、 实验作业及分析

4.1 实验过程

1) *** 请将详细实验过程填写在此处 ***

4.2 实验结果

*** 请将实验结果截图填写在此处 ***

五、 实验总结

*** 请撰写一段 200 字左右的实验总结 ***

GOOD LUCK!