程设第九次作业

20377383 樊思涵

本次作业调用的库

```
import abc
import matplotlib.pyplot as plt
import numpy as np
import random
from wordcloud import WordCloud
import collections
import jieba
from PIL import Image
import imageio.v2 as imageio
from pathlib import Path
```

设计抽象类

```
class Plotter(metaclass=abc.ABCMeta):
    @abc.abstractmethod
    def plot(self,data,*args,**kwargs):
        pass

```

设计 Point 类

```
class Point(object):
    COUNT = 0 #记录一共生成了多少个点
    def __init__(self,x,y):
        self.X=x
        self.Y=y
        Point.COUNT+=1

```

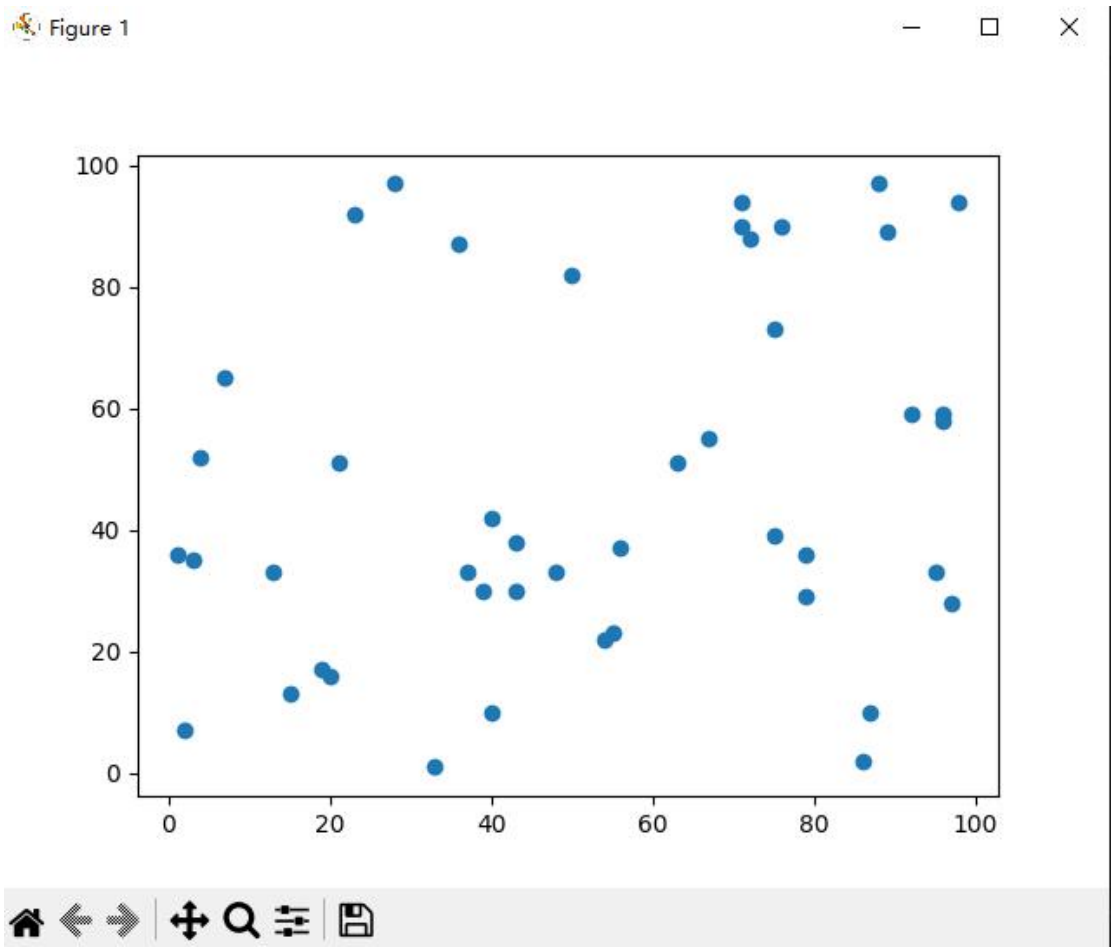
实现 PointPlotter 类

```
class PointPlotter(Plotter):
    def plot(self,data,*args,**kwargs):
        """
        :data:为[(x,y)...]型,每个元素为一个Point类的实例。
        """
        x_points=np.array([])
        y_points=np.array([])
        for point in data:
            #print(point.X)
            x_points = np.append(x_points,point.X) #注意必须接受np.append的返回值
            y_points = np.append(y_points,point.Y)
        print(x_points,y_points)
        plt.plot(x_points, y_points, 'o')
        plt.show()            #别忘了加括号!
```

在 main()函数中调用

```python
def main():
    #PointPlotter类
    PP=PointPlotter()
    #生成随机点
    data_point=[]
    for i in range(random.randint(2,100)):
        data_point.append(Point(random.randint(0,100),random.randint(0,100)))
    PP.plot(data_point)
    print(f'一共有{Point.COUNT}个点.')
```

效果展示




```
[54. 13. 40. 75. 87. 15.  1.  7. 97. 76. 40. 50. 39. 96. 63. 48. 36. 20.
 67. 72.  4.  3. 28. 86. 43. 92. 71. 88. 71. 75. 89. 95. 55. 33. 23. 98.
 43. 79. 96. 21. 37. 56.  2. 79. 19.] [22. 33. 42. 39. 10. 13. 36. 65. 28. 90. 10. 82. 30. 59. 51. 33. 87. 16.
 55. 88. 52. 35. 97.  2. 30. 59. 94. 97. 90. 73. 89. 33. 23.  1. 92. 94.
 38. 29. 58. 51. 33. 37.  7. 36. 17.]
一共有45个点.
```

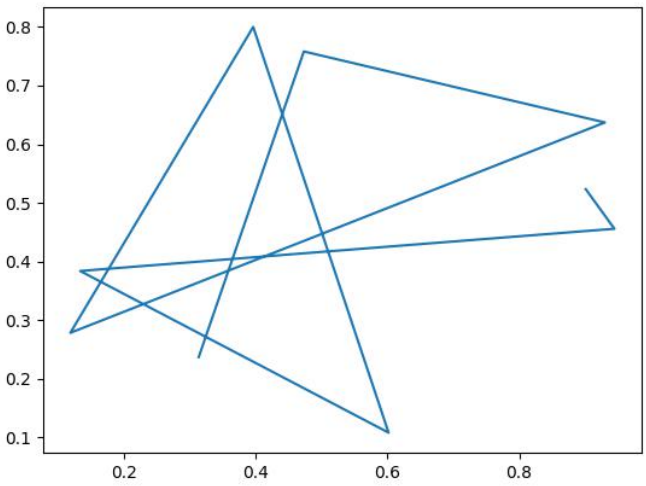
实现 ArrayPlotter 类

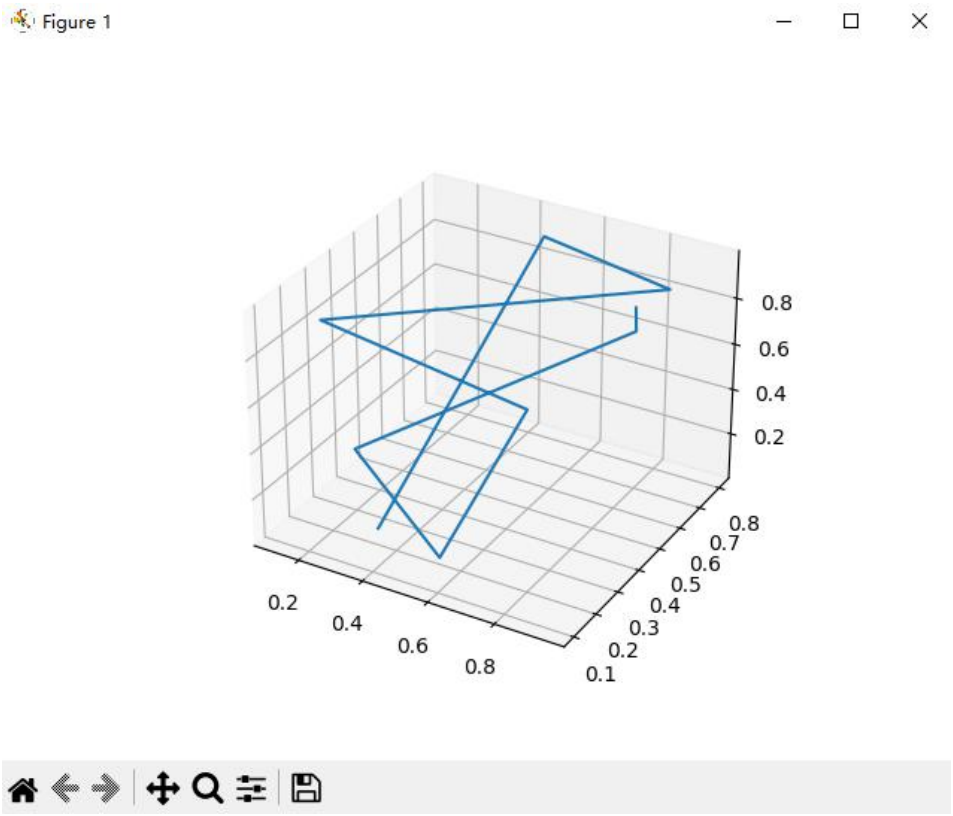
```
class ArrayPlotter(Plotter):
    def plot(self,data,*args,**kwargs):
        """
        :data:数据可能为[[x1,x2...],[y1,y2...]]或者[[x1,x2...],[y1,y2...],[z1,z2...]]
              二维:绘制平面轨迹曲线
              三维:绘制空间轨迹曲线
        """
        if len(data) == 2:
            x = data[0]
            y = data[1]
            plt.plot(x,y)
            plt.show()
        elif len(data) == 3:
            x = np.expand_dims(data[0],axis=0)
            y = np.expand_dims(data[1],axis=0)
            z = np.expand_dims(data[2],axis=0)
            fig=plt.figure()
            ax = fig.add_subplot(111,projection='3d')
            ax.plot_wireframe(x,y,z,rstride=10,cstride=10)
            plt.show()
        else:
            print('DataError!')
```

在 main()函数中调用

```
#ArrayPlotter类
AP = ArrayPlotter()
n_points = random.randint(3,10) #生成随机数组
x=np.random.random(n_points)
y=np.random.random(n_points)
z=np.random.random(n_points)
AP.plot([x,y])
AP.plot([x,y,z])
print(f'绘制的数组长度为{n_points}')
```

结果展示





绘制的数组长度为9

实现 TextPlotter 类

```
class TextPlotter(Plotter):
    def plot(self,data,*args,**kwargs):
        """
        :data:输入数据为一段或多段文本
        """
        wordList_jieba = jieba.lcut(data)
        lis_clean=[]
        stopwords_filepath =  r'C:\Users\LF\Desktop\stopwords_list.txt'
        stopwords_file=[]
        with open(stopwords_filepath, encoding='UTF-8') as f:
            stopwords_file = f.readlines()
        stopwords = [word.strip() for word in stopwords_file]
        stopwords.extend(['图片','分享'])
        for i in wordList_jieba:
            if len(i)>1 and i not in stopwords:
                lis_clean.append(i)
        dic_counter = dict(collections.Counter(lis_clean))     #词频统计
        font = r'C:\Windows\Fonts\STKAITI.ttf'
        wc = WordCloud(font,max_words=50,background_color="white",width = 1500,height= 960,margin= 10)
        t = wc.fit_words(dic_counter)
        plt.imshow(wc, interpolation='bilinear')
        plt.axis('off')
        plt.show()
```

在 main()函数中调用

```
    #TextPlotter类
    TP = TextPlotter()
    filename = r'C:\Users\LF\Desktop\out.txt'
    with open(filename, encoding='UTF-8') as f:
        data = f.read()
    TP.plot(data)
```

结果展示

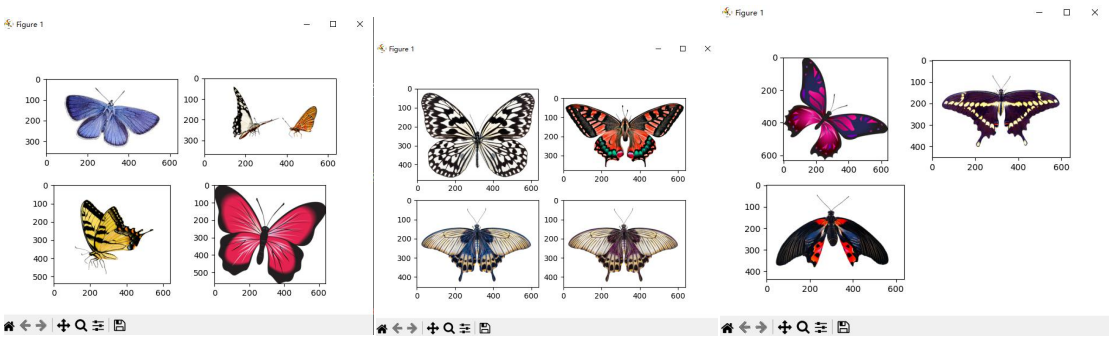
实现 ImagePlotter 类

```
class ImagePlotter(Plotter):
    def plot(self,data,*args,**kwargs):
        '''
        :data:输入数据为图片的路径或者图片内容（可以是多张图片）
        '''
        plt.ion()       #为了使循环能够正常进行
        for page in range(0,len(data),2*2):     #控制每一页生产的图片数量
            for i in range(2*2):                        #控制每张子图展示图片数量
                if page + i < len(data):
                    img = Image.open(data[page + i])
                    plt.subplot(2,2,i + 1)
                    plt.imshow(img)
                else:
                    continue
            plt.show()
            plt.pause(10)    #等待10秒后关闭当前页
            plt.close('all')
```

在 main()函数中调用

```
    #ImagePlotter类
    IP = ImagePlotter()
    tp='D:\\课程\\大三上\\现代程序设计\\现代程序设计技术第五次作业\\animals\\'
    data = []
    for i in range(1,12):
        data.append(tp+str(i)+'.png')
```

结果展示



设计 GifPlotter 类

```
104 class GifPlotter(Plotter):
105     def plot(self,data,*args,**kwargs):
106         """
107         :data:输入是图片文件夹地址
108         """
109         P_image = Path(data)
110         path_generator = P_image.rglob(r"*")                              #获得给定地址下的所有文件
111         image_list = list(filter(lambda x : '.png' in str(x),path_generator)) #返回后缀为'.jpg'的文件地址列表
112         frames = []
113         for image_name in image_list:
114             frames.append(imageio.imread(image_name))
115         duration = 0.5
116         imageio.mimsave('new.gif', frames, 'GIF', duration=duration)
```

在 main()函数中调用

```
153     #GifPlotter类
154     GP = GifPlotter()
155     data = 'D:\\课程\\大三上\\现代程序设计\\现代程序设计技术第五次作业\\animals'
156     GP.plot(data)
157
```

结果展示