PYTHON
数据可视化-2

编译环境：

PyCharm 2019.3 (Community Edition)

Build #PC-193.5233.109, built on November 28, 2019

Runtime version: 11.0.4+10-b520.11 amd64

VM: OpenJDK 64-Bit Server VM by JetBrains s.r.o

Windows 10 10.0

GC: ParNew, ConcurrentMarkSweep

Memory: 1963M

Cores: 8

Registry:

Non-Bundled Plugins:

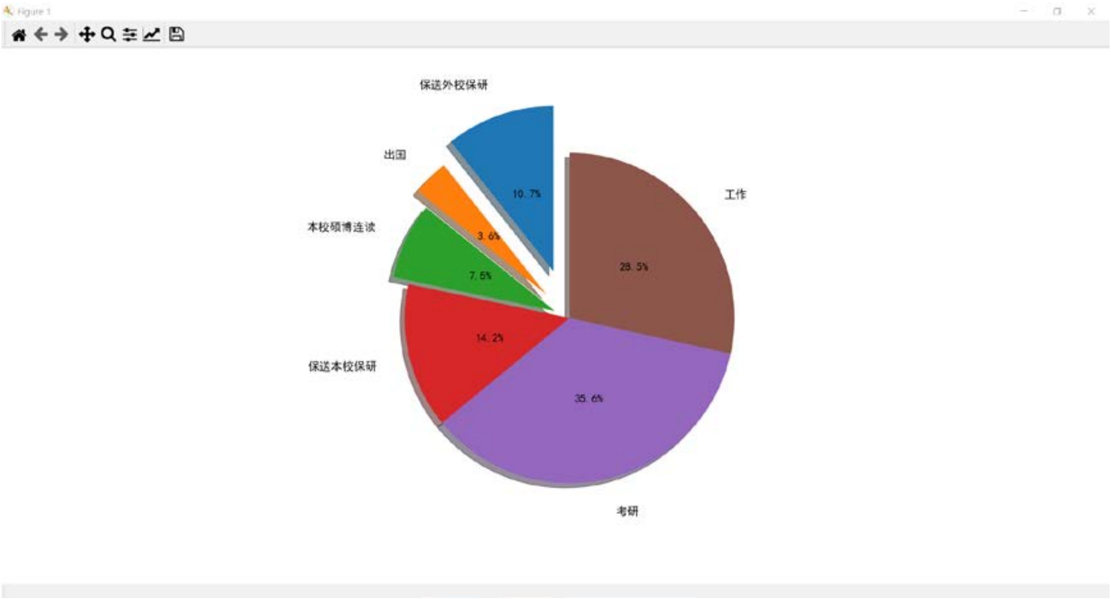
python 版本：3.7（Anaconda3）

作业 8：

Q31.py:

```
import matplotlib.pyplot as plt
import matplotlib
from matplotlib import font_manager as fm

matplotlib.rcParams['font.sans-serif'] = ['SimHei']
label_list = ["保送外校保研", "出国", "本校硕博连读", "保送本校保研", "考研", "工作"]
size = [300, 100, 210, 400, 1000, 800]
explode = [0.3, 0.2, 0.1, 0, 0, 0]
patches, texts, autotexts = plt.pie(size, explode=explode, labels=label_list, labeldistance=1.2,
                                    autopct="%1.1f%%", shadow=True,
startangle=90, pctdistance=0.5)
proptease = fm.FontProperties()
proptease.set_size('x-large')
plt.setp(texts, fontproperties=proptease)
plt.setp(autotexts, fontproperties=proptease)
plt.show()
```

运行结果：



作业 9：

Dataset.csv:

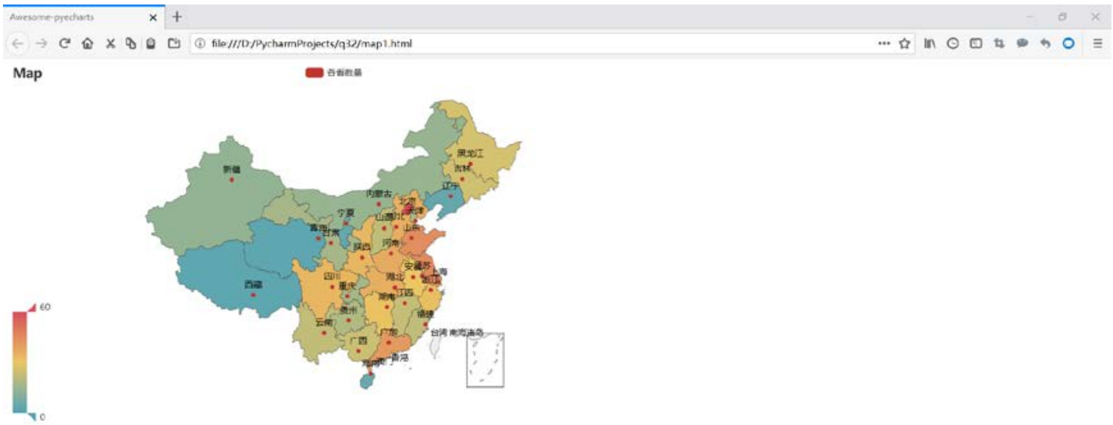
```
省/市,2017 年,2016 年,211&985 大学数量,公办本科大学数量
河南,86.3 万,82 万,1,37
广东,75.7 万,73.3 万,4,41
山东,58.3 万,60.2 万,3,44
四川,58.3 万,57.13 万,5,34
安徽,49.9 万,50.99 万,3,30
河北,43.6 万,42.31 万,0,36
贵州,41.2 万,37.38 万,1,19
湖南,41.1 万,40.16 万,3,31
广西,36.5 万,33 万,1,24
江西,36.5 万,36.06 万,1,23
湖北,36.2 万,36.14 万,1,36
江苏,33 万,36.04 万,11,45
陕西,31.9 万,32.8 万,1,34
山西,31.7 万,33.9 万,1,22
云南,29.3 万,28 万,1,22
浙江,29.1 万,30.74 万,1,32
```

```
甘肃,28.5 万,29.6 万,1,17
重庆,21.1 万,24.88 万,2,17
辽宁,20.8 万,21.82 万,4,4
内蒙古,19.8 万,20.11 万,1,14
福建,18.8 万,17.5 万,2,22
黑龙江,18.8 万,19.7 万,4,26
新疆,18.4 万,16.61 万,2,13
吉林,14.3 万,14.85 万,3,25
宁夏,6.9 万,6.9 万,1,4
北京,6 万,6.12 万,26,60
海南,5.7 万,6.04 万,1,5
天津,5.7 万,6 万,4,18
上海,5 万,5.1 万,9,31
青海,4.6 万,4.5 万,1,3
西藏,2.8 万,2.4 万,1,3
```

Q32.py:

```
from pyecharts import options as opts
from pyecharts.charts import Map
import pandas as pd

df = pd.read_csv('dataset.csv')


def map1() -> Map:
    c = (
        Map()
            .add("各省数量", [list(z) for z in zip(df['省/市'], df['公办本科大学数量'])], "china")
            .set_global_opts(
            title_opts=opts.TitleOpts(title="Map"),
            visualmap_opts=opts.VisualMapOpts(min_=0, max_=60))
            .set_series_opts(label_opts=opts.LabelOpts(is_show=True))
    )
    return c


map1().render("map1.html")
```

运行结果：

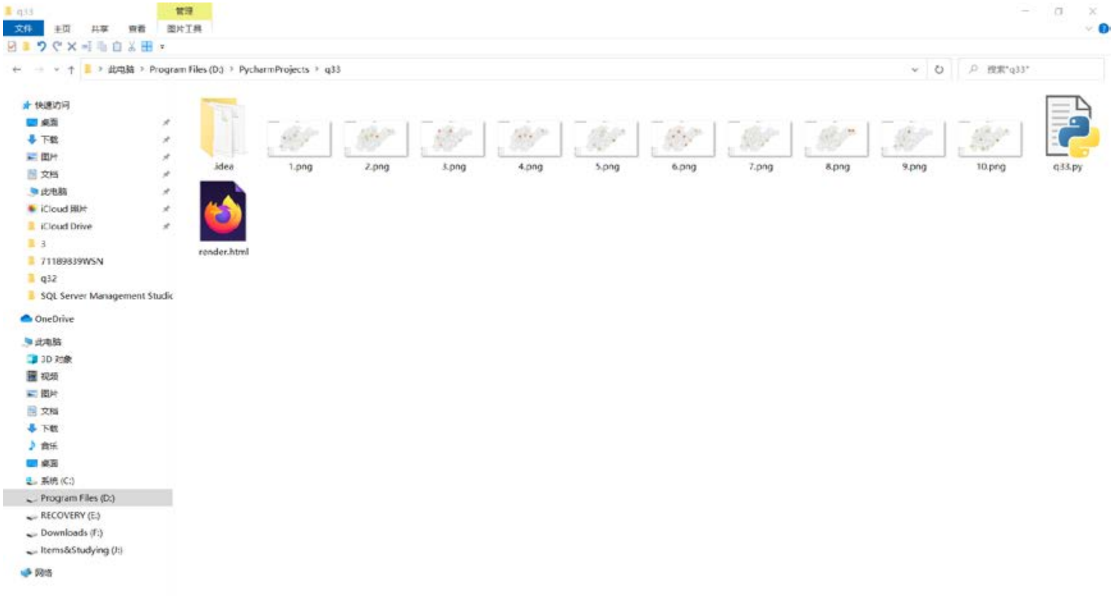


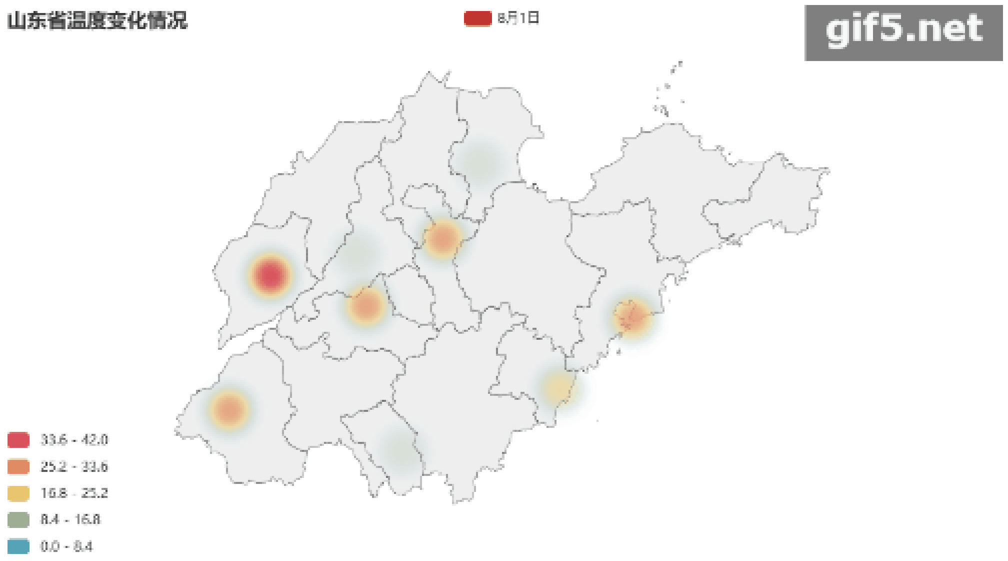
作业10：

Q33.py:

```
from pyecharts import options as opts
from pyecharts.charts import Geo
from pyecharts.globals import ChartType
from pyecharts.render import make_snapshot
from snapshot_phantomjs import snapshot
import random


class Data:
    cities = ['济南', '青岛', '淄博', '枣庄', '东营', '烟台', '潍坊', '济宁',
              '泰安', '威海', '日照', '滨州', '德州', '聊城', '临沂', '菏泽']

    @staticmethod
    def values(start: int = -10, end: int = 40) -> list:
        return [random.randint(start, end) for _ in range(16)]


def geo_shandong(title) -> Geo:
    c = (
        Geo()
            .add_schema(maptype="山东")
            .add(title, [list(z) for z in zip(Data.cities, Data.values())],
type_=ChartType.HEATMAP)
            .set_global_opts(
            title_opts=opts.TitleOpts(title="山东省温度变化情况"),
            visualmap_opts=opts.VisualMapOpts(max_=42, is_piecewise=True))
    )
    return c


for i in range(10):
    str_date = "8 月" + str(i + 1) + "日"
    make_snapshot(snapshot, geo_shandong(str_date).render(), str(i + 1) + ".png", pixel_ratio=1)
```

运行结果：



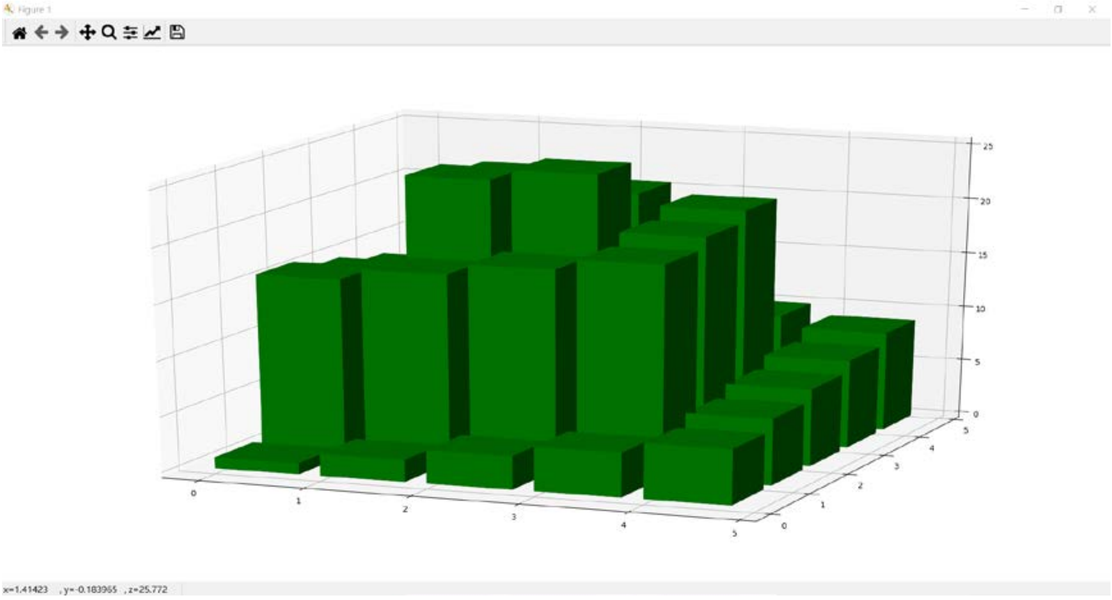


作业 pre11：

Q34.py:

```
from matplotlib import pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
import numpy as np

fig = plt.figure()
ax = Axes3D(fig)

z = np.linspace(0, 0, 25)
x = np.linspace(0, 4, 5)
x = np.hstack((x, np.linspace(4, 4, 4)))
x = np.hstack((x, np.linspace(3, 0, 4)))
x = np.hstack((x, np.linspace(0, 0, 3)))
x = np.hstack((x, np.linspace(1, 3, 3)))
x = np.hstack((x, np.linspace(3, 3, 2)))
x = np.hstack((x, np.linspace(2, 1, 2)))
x = np.hstack((x, np.linspace(1, 1, 1)))
x = np.hstack((x, np.linspace(2, 2, 1)))
y = np.linspace(0, 0, 5)
y = np.hstack((y, np.linspace(1, 4, 4)))
y = np.hstack((y, np.linspace(4, 4, 4)))
y = np.hstack((y, np.linspace(3, 1, 3)))
y = np.hstack((y, np.linspace(1, 1, 3)))
y = np.hstack((y, np.linspace(2, 3, 2)))
y = np.hstack((y, np.linspace(3, 3, 2)))
y = np.hstack((y, np.linspace(2, 2, 2)))
dz = np.linspace(1, 25, 25)
dx = np.linspace(0.8, 0.8, 25)
dy = np.linspace(0.8, 0.8, 25)
ax.bar3d(x, y, z, dx, dy, dz, color='g')
plt.show()
```

运行结果：



x=1.41423 , y=-0.183965 , z=25.772

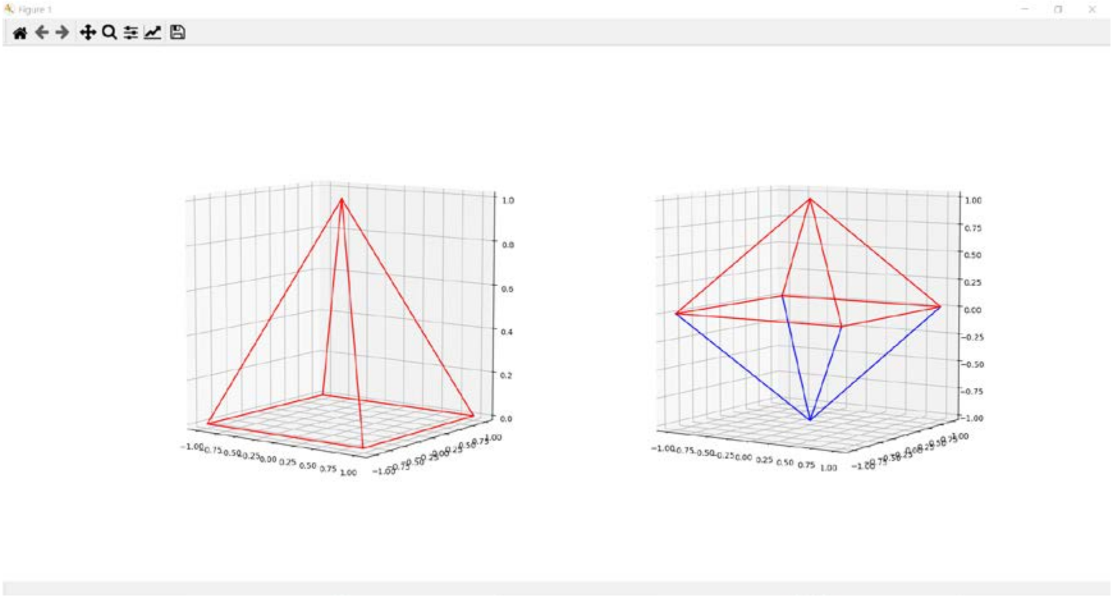
作业 11：

Q35.py:

```
from matplotlib import pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
import numpy as np

fig = plt.figure()
ax = fig.add_subplot(121, projection='3d')
z = np.array([1, 0])
x = np.array([0, -1])
y = np.array([0, -1])
ax.plot(x, y, z, color='r')
z = np.array([1, 0])
x = np.array([0, 1])
y = np.array([0, -1])
ax.plot(x, y, z, color='r')
z = np.array([1, 0])
x = np.array([0, -1])
y = np.array([0, 1])
ax.plot(x, y, z, color='r')
z = np.array([1, 0])
x = np.array([0, 1])
y = np.array([0, 1])
ax.plot(x, y, z, color='r')
z = np.array([0, 0, 0, 0, 0])
x = np.array([-1, -1, 1, 1, -1])
y = np.array([-1, 1, 1, -1, -1])
```

```
ax.plot(x, y, z, color='r')

ax = fig.add_subplot(122, projection='3d')
z = np.array([1, 0])
x = np.array([0, -1])
y = np.array([0, -1])
ax.plot(x, y, z, color='r')
z = np.array([1, 0])
x = np.array([0, 1])
y = np.array([0, -1])
ax.plot(x, y, z, color='r')
z = np.array([1, 0])
x = np.array([0, -1])
y = np.array([0, 1])
ax.plot(x, y, z, color='r')
z = np.array([1, 0])
x = np.array([0, 1])
y = np.array([0, 1])
ax.plot(x, y, z, color='r')
z = np.array([-1, 0])
x = np.array([0, -1])
y = np.array([0, -1])
ax.plot(x, y, z, color='b')
z = np.array([-1, 0])
x = np.array([0, 1])
y = np.array([0, -1])
ax.plot(x, y, z, color='b')
z = np.array([-1, 0])
x = np.array([0, -1])
y = np.array([0, 1])
ax.plot(x, y, z, color='b')
z = np.array([-1, 0])
x = np.array([0, 1])
y = np.array([0, 1])
ax.plot(x, y, z, color='b')
z = np.array([0, 0, 0, 0, 0])
x = np.array([-1, -1, 1, 1, -1])
y = np.array([-1, 1, 1, -1, -1])
ax.plot(x, y, z, color='r')

plt.show()
```

运行结果：

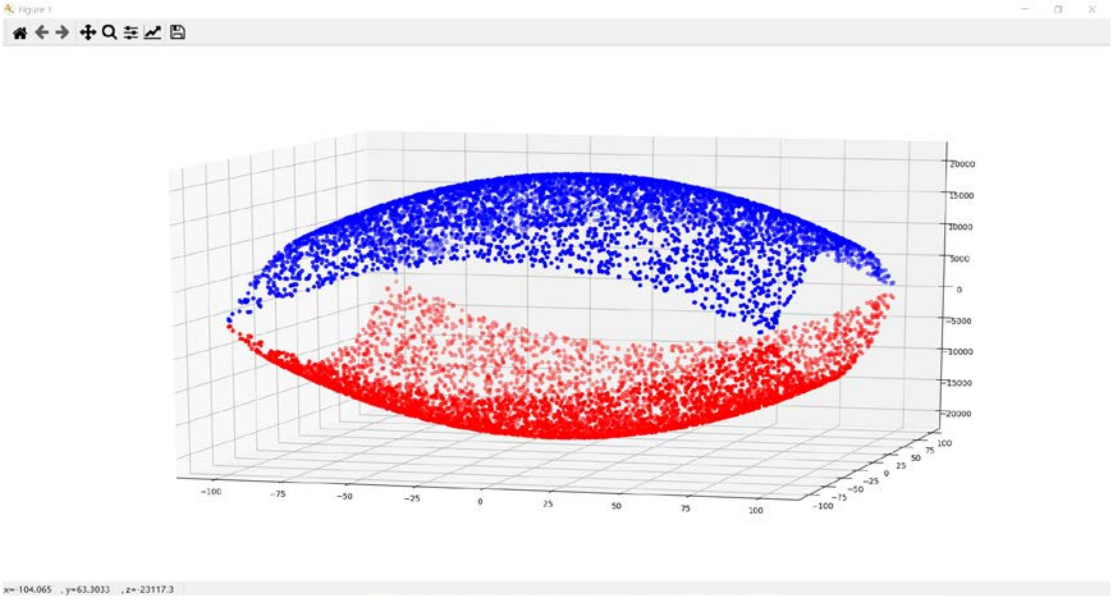


作业 12：

Q36.py:

```
from matplotlib import pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
import numpy as np

fig = plt.figure()
ax = Axes3D(fig)

x = np.random.random(5000)*200-100
y = np.random.random(5000)*200-100
z = x ** 2 + y ** 2 - 20000
ax.scatter(x, y, z, color='r')
ax.scatter(x, y, -z, color='b')
plt.show()
```

运行结果：

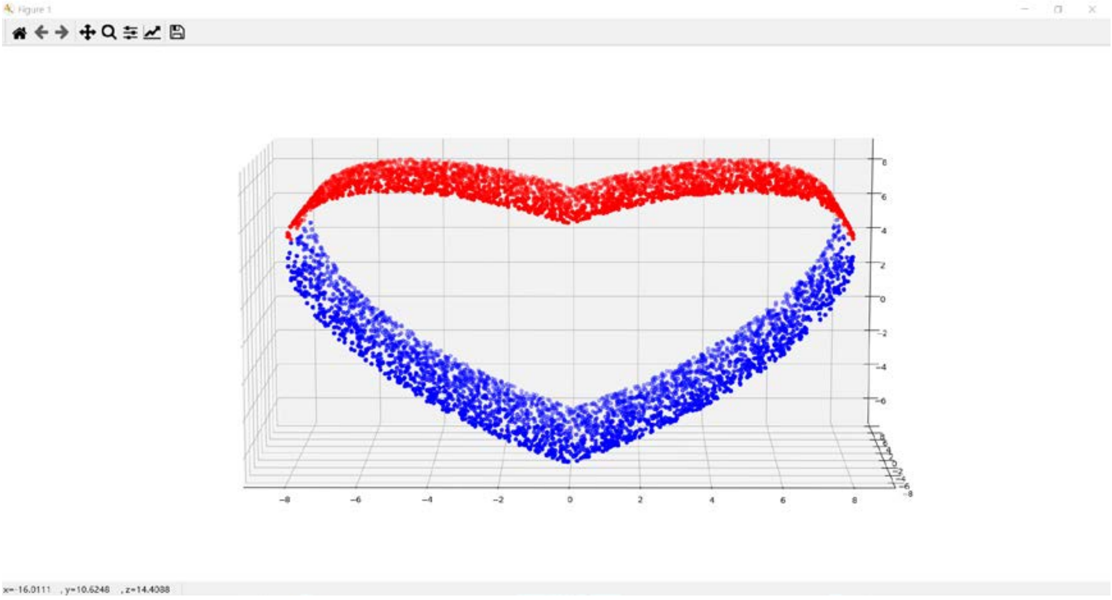


作业 13：画 3D 心形线

Q37.py:

```
from matplotlib import pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
import numpy as np

fig = plt.figure()
ax = Axes3D(fig)

x = np.random.random(3000) * 16 - 8
y = np.random.random(3000) * 16 - 8
z1 = 0.618 * np.abs(x) - 0.8 * np.sqrt(64 - x ** 2)
z2 = 0.618 * np.abs(x) + 0.8 * np.sqrt(64 - x ** 2)

ax.scatter(x, y, z1, color='b')
ax.scatter(x, y, z2, color='r')
plt.show()
```

运行结果：



x=-16.0111 , y=10.6248 , z=14.4088