**实验五 关系数据可视化**

**一、本次实验的目的**

通过该实验的实践，要求学生可以理解数据的关联性和分布性，掌握Matplotlib和pyecharts的使用过程，能够熟练绘制常见的适合展示关系数据的图表。

**二、实验相关知识**

关系数据主要包括数据关联性和数据分布性，多元变量之间的关系，可以使用散点图，散点矩阵图，或者气泡图；数据的分布，可以使用直方图、箱线图呈现。

数据关联性的可视化

1.散点图

散点图（Scatter）用于在水平轴和垂直轴上绘制数据点，它展示了因变量随自变量变化的趋势，反映的是一个变量受另一个变量的影响程度。散点图将序列显示为一组点，其中每个散点值都由该点在图表中的坐标位置表示。对于不同类别的点，则由图表中不同形状、大小或颜色的标记符表示。

2.散点矩阵图

散点图矩阵是一种显示多个变量之间关系的数据可视化工具，特别是当数据集包含三个或多个变量时，这种图表非常有用。这种图通常在探索性数据分析中使用，以便快速理解数据集中变量之间的关系。在散点图矩阵中，每行和每列都代表数据集中的一个变量，而矩阵中的每个小图（除了对角线）都是两个变量之间的散点图。

1. 气泡图

气泡图是一种数据可视化工具，它在传统的二维散点图的基础上增加了一个维度，使得我们能够同时观察三个变量之间的关系。这种图表通过点的大小来表示第三个数值变量的大小，从而提供了一种直观的方式来探索数据中的模式和趋势。

**数据分布性的可视化**

1.茎叶图

茎叶图是一个与直方图相类似的特殊工具，但又与直方图不同，茎叶图保留原始资料的资讯，直方图则失去原始资料的讯息。将茎叶图茎和叶逆时针方向旋转90度，实际上就是一个直方图，可以从中统计出次数，计算出各数据段的频率或百分比。从而可以看出分布是否与正态分布或单峰偏态分布逼近。

2.直方图

直方图描述的是一组数据的频次分布，是以矩形的长度表示每一组的频度或数量，宽度则表示各组的组距，因此其高度与宽度均有意义，利于展示大量数据集的统计结果，展示数据的分布。

3.密度图

密度图（Density Plot）是一种表示数值变量分布的图形。它通过核密度估计（Kernel Density Estimate）来展示变量的概率密度函数。

**实验内容：**

1. 散点图：展示数据之间的关联性。
2. 散点矩阵图：关系识别：快速识别多个变量间的潜在关系，比如线性、非线性或没有关系。

异常值检测：观察数据点集群之外的点来识别异常值。分布检查：在对角线的图表帮助理解各个变量的分布特征。群组比较：使用颜色或形状区分不同的群组，可以帮助比较不同群组的特征。

1. 气泡图：通过将数据点表示为不同大小和颜色的气泡来展示多维信息。它的作用与价值在于能够同时展示多个变量之间的关系，以及数据点的分布情况。
2. 茎叶图：是一种统计图表，用于展示数据的分布情况，同时保留原始数据点。这种图表在探索性数据分析中非常有用，尤其是当需要快速了解数据集的形状和分布时。
3. 直方图：一般用横轴表示数据类型，纵轴表示分布情况。
4. 密度图：它可用来对分布的细节变化进行可视化。密度图表现与数据值对应的边界或域对象的一种理论图形表示方法。一般用于呈现连续变量。
5. **实验步骤：**
6. 散点图的绘制

第一步：读取数据，这里通过pandas库中的read\_csv函数读取CSV文件中的数据，函数中需要给出CSV文件的路径。

pd.read\_csv("crimeRatesByState2005.csv")

第二步：绘图

import ggplot as gp

import pandas as pd

crime = pd.read\_csv("crimeRatesByState2005.csv")

plot = gp.ggplot(gp.aes(x='murder', y='burglary'), data=crime)

points = gp.geom\_point(color='red’)

print(plot + points)

1. 散点矩阵图

import pandas as pd  
import matplotlib.pyplot as plt  
import seaborn as sns  
#读取数据  
crime=pd.read\_csv("…/crimeRatesByState2005.csv")  
#去除掉state为United States的数据  
crime2 = crime[crime.state != "United States"]  
#去除掉state为District of Columbia的数据  
crime2 = crime2[crime2.state != "District of Columbia"]  
#删除state列  
crime2=crime2.drop(['state'],axis=1)  
#删除population列  
crime2=crime2.drop(['population'],axis=1)  
#绘制成对关系图  
g = sns.pairplot(crime2, diag\_kind="kde",kind="reg")  
plt.show()#显示图表

1. 气泡图

**（1）不**显示气泡的state名称

import matplotlib.pyplot as plt  
import pandas  as pd  
import  numpy as np  
#读取数据  
crime=pd.read\_csv("/home/ubuntu/crimeRatesByState2005.csv")  
print (list(crime.murder))#转化成列表  
#删除state为United States的数据  
crime2 = crime[crime.state != "United States"]  
#删除state为District of Columbia的数据  
crime2 = crime2[crime2.state != "District of Columbia"]  
z = list(crime2.population/10000)#取人口数据  
colors = np.random.rand(len(list(crime2.murder)))#根据谋杀率随机去颜色  
cm = plt.cm.get\_cmap('RdYlBu')#使用色谱RdYlBu  
plt.scatter(list(crime2.murder), list(crime2.burglary), s=z,c=z,cmap = cm, linewidth = 0.5, alpha     = 0.5)#绘制散点图  
plt.xlabel("murder")  
plt.ylabel("burglary")  
plt.show()

（2）显示气泡的state名称

功能：气泡图  
"""  
import matplotlib.pyplot as plt  
import pandas  as pd  
import  numpy as np  
# plt.rcParams['font.sans-serif'] = ['WenQuanYi Mico Hei'] # 解决中文不能正常显示的问题  
crime=pd.read\_csv("/home/ubuntu/crimeRatesByState2005.csv")  
print (list(crime.murder)) # murder是列名  
crime2 = crime[crime.state != "United States"] # 排除state为United States的这一行  
crime2 = crime2[crime2.state != "District of Columbia"] # 排除state为District of Columbia这一行  
z = list(crime2.population/10000)#取人口数作为气泡的大小的参考  
colors = np.random.rand(len(list(crime2.murder)))#根据谋杀率取颜色  
cm = plt.cm.get\_cmap('RdYlBu')#使用色谱RdYlBu  
fig,ax=plt.subplots(figsize=(10,5))#创建画布  
ax.scatter(list(crime2.murder), list(crime2.burglary), s=z,c=z,cmap=cm, linewidth=0.5, alpha=0.5)#绘制气泡图  
#ax.xlabel("murder")  
#ax.ylabel("burglary")  
# 图形不支持中文显示  
ax.set(xlim=(0,11),ylim=(200,1300),\  
       xlabel="Murder (per 10W population)",\  
       ylabel="Burglary (per 10W population)",\  
       title="Murder and burglary rates in the United States")#设置标签、标题的格式  
"""  
ax.set(xlim=(0,11),ylim=(200,1300),\  
       xlabel="murder",\  
       ylabel="burglary")  
"""  
# 特殊处理state，只显示部分地方  
state\_show = []  
#遍历各州的数据  
for sta in list(crime2.state):  
    print(sta)  
    if(sta == 'California'or  
    sta == 'Florida' or  
    sta == 'Texas' or  
    sta == 'New York' or  
    sta == 'Pennsylvania' or  
    sta == 'Louisiana' or  
    sta == 'Maryland'):  
        state\_show.append(sta)  
    else:  
        state\_show.append('')  
print(state\_show)  
# 显示所有气泡的state名称  
#for i,j,k in zip(crime2.murder,crime2.burglary,crime2.state):  
for i, j, k in zip(crime2.murder, crime2.burglary, state\_show):  
    plt.text(x=i-0.3,y=j-0.1,s=k,fontsize=7)  
ax.spines["top"].set\_visible(False)#不显示上边框  
ax.spines["left"].set\_visible(False)#不显示下边框  
ax.spines["right"].set\_visible(False)#不显示右边框  
plt.show()#显示图表

1. 茎叶图

import  numpy as np  
import math  
from itertools import groupby  
import pandas  as pd  
#读取数据  
birth=pd.read\_csv("/home/ubuntu/birth-rate.csv")  
#去除掉空值  
birth.dropna(subset=[ '2008'], inplace= True)  
dirt={}  
data = list(round(birth ['2008'],1))#转换数据  
rangenum = []  
#先把数据排序，再根据值进行分布，进行取余取商运算  
for k,g in groupby(sorted(data),key = lambda x: int(x)):  
    lst = map(str,list(map(lambda y: divmod(int(y\*10),10)[1],list(g))))  
    dirt[k] = ' '.join(lst)  
    rangenum.append(k)  
num = list(range(rangenum[0],rangenum[-1],2))

for i in num:

a =‘’

for k in sorted(dirt.keys()):

if 0<=k-i <=1:

a = a +’ ’ + dirt[k]

elif k-i > 1:

break

print(str(i).rjust(5), ‘|’, a)  
 5.直方图

import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
# 解决中文不能正常显示的问题  
# plt.rcParams[ 'font.sans-serif'] = [ 'WenQuanYi Mico Hei']  
# plt.rcParams[ 'axes.unicode\_minus'] = False  
#plt.rcParams['font.sans-serif'] = ['SimHei']   
#读取数据  
titanic = pd.read\_csv( '/home/ubuntu/birth-rate.csv')  
#去除空值  
titanic.dropna(subset=[ '2008'], inplace= True)  
plt.style.use( 'ggplot')  
#绘制直方图  
plt.hist(titanic['2008'], bins = 10, color = 'steelblue', edgecolor = 'k', label = "histogram")  
plt.tick\_params(top= 'off', right= 'off')#右侧，顶侧不显示刻度  
plt.legend()#显示图例  
plt.show()#显示图表

1. 密度图

import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
import matplotlib.mlab as mlab  
*#解决中文乱码*  
# plt.rcParams[ 'font.sans-serif'] = [ 'WenQuanYi Mico Hei']  
# plt.rcParams[ 'axes.unicode\_minus'] = False  
#读取数据  
titanic = pd.read\_csv( '…/birth-rate.csv')  
#去除空值  
titanic.dropna(subset=[ '2008'], inplace= True)  
#使用kde密度图  
kde = mlab.GaussianKDE(titanic['2008'])  
#取最大值和最小值之间1000个均匀分割的数  
x2 = np.linspace(titanic['2008'].min(), titanic['2008'].max(), 1000)  
line2 = plt.plot(x2,kde(x2), 'g-', linewidth = 2)#绘制折线图  
plt.show()#显示图表

1. 密度图+直方图

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import matplotlib.mlab as mlab

plt.rcParams['font.sans-serif'] = ['SimHei'] # 解决中文不能正常显示的问题

titanic = pd.read\_csv('../data/birth-rate.csv')

titanic.dropna(subset=['2008'], inplace=True)

plt.style.use('ggplot')

plt.hist(titanic['2008'], bins=np.arange(titanic['2008'].min(),

titanic['2008'].max(), 3), normed=True, color= 'steelblue', edgecolor='k')

plt.title(' 2008出生直方图和密度图')

plt.xlabel('出生率')

plt.ylabel('频率')

kde = mlab.GaussianKDE(titanic['2008'])

x2 = np.linspace(titanic['2008'].min(), titanic['2008'].max(), 1000)

line2 = plt.plot(x2, kde(x2), 'g-', linewidth=2)

plt.tick\_params(top='off', right='off')

plt.show()

**四、实验任务：**

根据实验内容和实验步骤的所学知识，完成下面的任务

1.取数据表“crimeRatesByState2005”中 robbery和burglary两列数据绘制散点图，标题为robbery和burglary关联性散点图，并显示散点图图例。

2.利用“crimeRatesByState2005”中的数据，完成散点矩阵图的绘制。

3.利用气泡图可视化方式探索larceny\_theft 和motor\_vehicle\_theft 两种犯罪率是否与所在地区的人口数量有关系，分别绘制**不显示**、显示部分，显示全部气泡的state名称的三种气泡图。

4.分别将中国、日本从1960-2008年出生率变化绘制为茎叶图。

5.为1970年各国出生率绘制合理的直方图。

6.为2006年各国出生率绘制合理的密度图。

7.将2006年各国出生率绘制为直方图，并将任务6中的密度图与直方图结合起来。