1. **问题描述**
2. 对右图进行修改

1. 请更换图形的风格

2. 请将x轴的数据改为-10到10

3. 请自行构造一个y值的函数

4. 将直方图上的数字，位置改到柱形图的内部垂直居中的位置

1. 对成绩数据data1402.csv进行分段统计：每5分作为一个分数段，展示出每个分数段的人数直方图。
2. 自行创建出10个学生的3个学期排名数据，并通过直方图进行对比展示。
3. 线图
4. 把这个图像做一些调整，要求出现5个完整的波峰。
5. 调大cos波形的幅度
6. 调大sin波形的频率
7. 用线图展示北京空气质量数据

展示10-15年PM指数月平均数据的变化情况，一幅图中有6条曲线，每年1条曲线。

1. **实验环境**

Microsoft Windows 10 版本18363

PyCharm 2020.2.1 (Community Edition)

Python 3.8(Scrapy 2.4.0 + numpy 1.19.4 + pandas 1.1.4 + matplotlib 3.3.3)

1. **实验步骤及结果**
2. 对右图进行修改

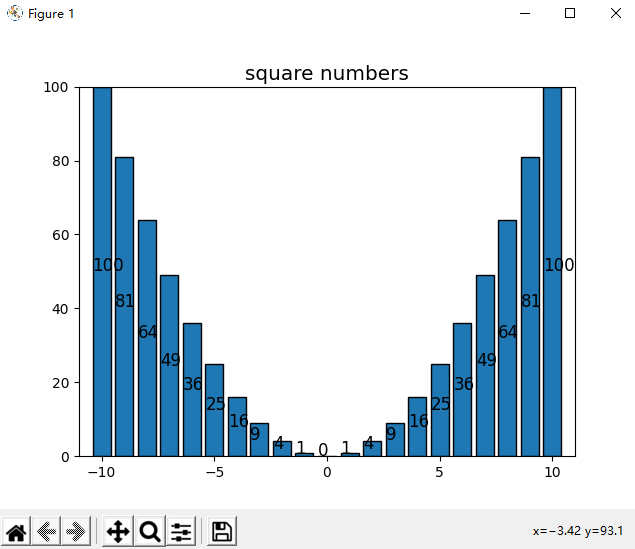
1. 请更换图形的风格

2. 请将x轴的数据改为-10到10

3. 请自行构造一个y值的函数

4. 将直方图上的数字，位置改到柱形图的内部垂直居中的位置

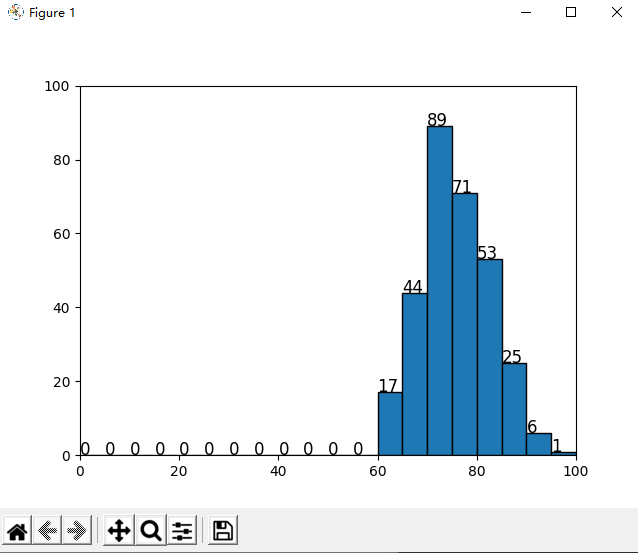
from matplotlib import pyplot as plt  
import numpy as np  
  
fig, ax = plt.subplots()  
plt.style.use('classic')  
plt.title("square numbers")  
  
ax.set\_xlim(-11, 11)  
ax.set\_ylim(0, 100)  
  
x = np.array(range(-10, 11))  
y = x \* x  
rect1 = plt.bar(x, y)  
  
for r in rect1:  
 ax.text(r.get\_x(), r.get\_height() / 2, r.get\_height())  
  
plt.show()



如图使用classic风格，x轴数据为[-10, 10]的整数，构造的函数为y=x2，显示位置并将其将数值改到了柱形图内部垂直居中的位置。

1. 对成绩数据data1402.csv进行分段统计：每5分作为一个分数段，展示出每个分数段的人数直方图。

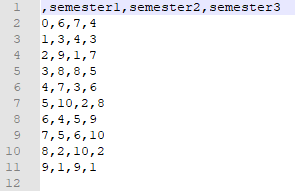
from matplotlib import pyplot as plt  
import numpy as np  
import pandas as pd  
  
df = pd.read\_csv("./data1402.csv", encoding='utf-8', dtype=str)  
df = pd.DataFrame(df, columns=['score'], dtype=np.float)  
  
section = np.array(range(0, 105, 5))  
  
result = pd.cut(df['score'], section)  
count = pd.value\_counts(result, sort=False)  
  
fig, ax = plt.subplots()  
plt.style.use('classic')  
  
ax.set\_xlim(0, 100)  
  
rect1 = plt.bar(np.arange(2.5, 100, 5), count, width=5)  
  
for r in rect1:  
 ax.text(r.get\_x(), r.get\_height(), r.get\_height())  
  
plt.show()



1. 自行创建出10个学生的3个学期排名数据，并通过直方图进行对比展示。

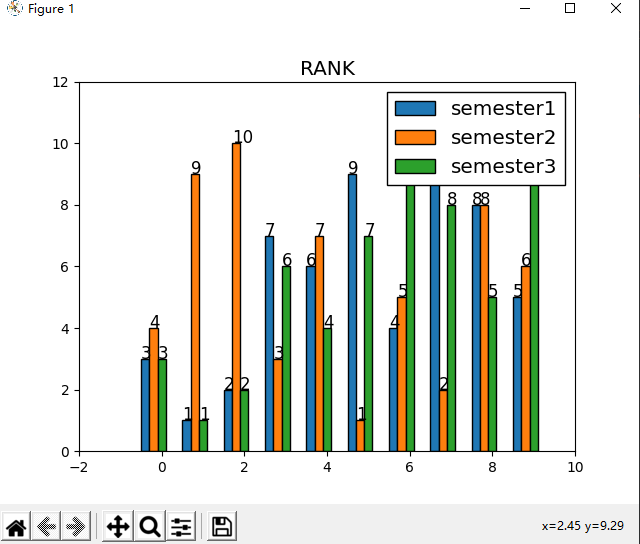
import random  
  
semester1 = np.arange(1, 11)  
semester2 = np.arange(1, 11)  
semester3 = np.arange(1, 11)  
  
random.shuffle(semester1)  
random.shuffle(semester2)  
random.shuffle(semester3)  
  
df = pd.DataFrame({'semester1':semester1, 'semester2':semester2, 'semester3':semester3})  
print(df)  
  
df.to\_csv("data1403.csv", encoding="utf-8")

使用如上代码创建出随机的排名数据。



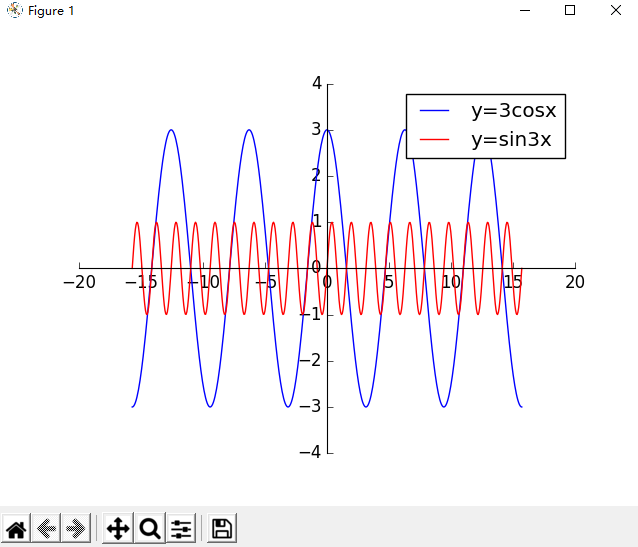
df = pd.read\_csv("./data1403.csv", encoding='utf-8', dtype=str)  
df = pd.DataFrame(df, columns=['semester1', 'semester2', 'semester3'], dtype=np.int)  
  
df['total'] = (df['semester1'] + df['semester2'] + df['semester3']) / 3  
df = df.sort\_values('total')  
  
fig, ax = plt.subplots()  
plt.style.use('classic')  
plt.title('RANK')  
  
width = 0.2  
x = np.array(range(0, 10))  
rect1 = ax.bar(x-2\*width, df['semester1'], width=width, label='semester1')  
rect2 = ax.bar(x-width, df['semester2'], width=width, label='semester2')  
rect3 = ax.bar(x, df['semester3'], width=width, label='semester3')  
  
for r in rect1:  
 ax.text(r.get\_x(), r.get\_height(), r.get\_height())  
for r in rect2:  
 ax.text(r.get\_x(), r.get\_height(), r.get\_height())  
for r in rect3:  
 ax.text(r.get\_x(), r.get\_height(), r.get\_height())  
  
plt.legend(ncol=1)  
plt.show()

如上代码绘图：



1. 线图
2. 把这个图像做一些调整，要求出现5个完整的波峰。
3. 调大cos波形的幅度
4. 调大sin波形的频率

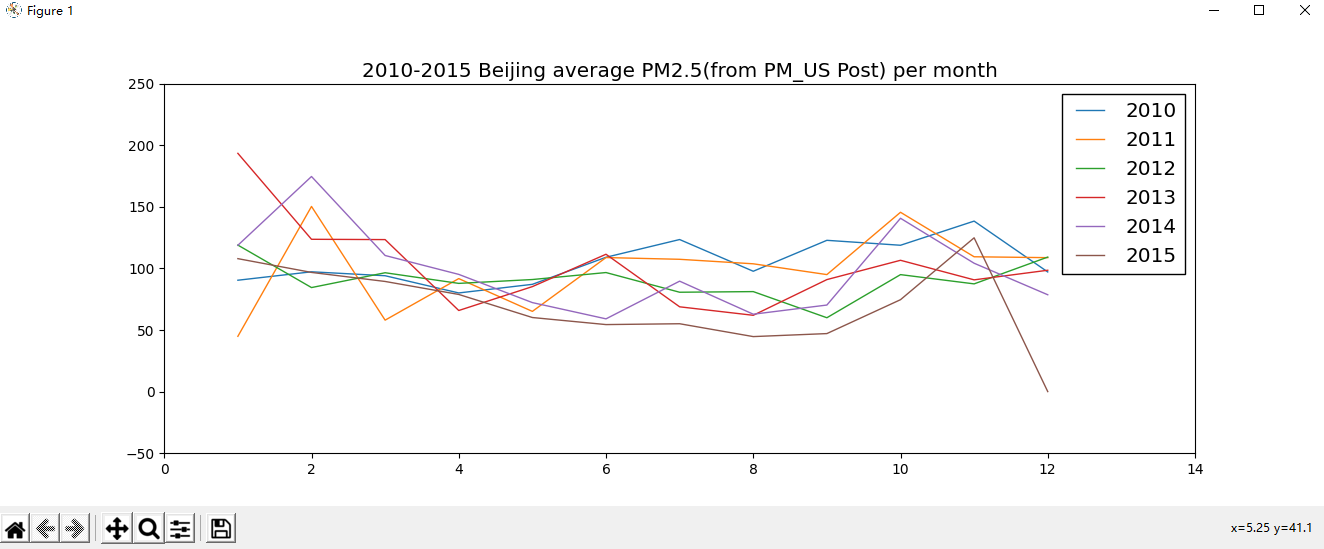
import numpy as np  
from matplotlib import pyplot as plt  
  
x = np.linspace(-5 \* np.pi, 5 \* np.pi, 500)  
y1 = 3 \* np.cos(x)  
y2 = np.sin(4\*x)  
  
fig, ax = plt.subplots()  
plt.style.use('classic')  
  
ax.spines["right"].set\_visible(False)  
ax.spines["top"].set\_visible(False)  
ax.spines['bottom'].set\_position(('data',0))  
ax.xaxis.set\_ticks\_position('bottom')  
ax.spines['left'].set\_position(('data',0))  
ax.yaxis.set\_ticks\_position('left')  
  
plt.plot(x, y1, color='blue', linestyle='-', label='y=3cosx')  
plt.plot(x, y2, color='red', linestyle='-', label='y=sin3x')  
  
plt.legend()  
plt.show()



1. 用线图展示北京空气质量数据

展示10-15年PM指数月平均数据的变化情况，一幅图中有6条曲线，每年1条曲线。

import numpy as np  
import pandas as pd  
from matplotlib import pyplot as plt  
  
orig\_df = pd.read\_csv("./BeijingPM20100101\_20151231.csv", encoding='utf-8', dtype=str)  
orig\_df = pd.DataFrame(orig\_df, columns=['year', 'month', 'PM\_US Post'])  
df = orig\_df.dropna(0, how='any')  
df['month'] = df['month'].astype(int)  
df['year'] = df['year'].astype(int)  
df['PM\_US Post'] = df['PM\_US Post'].astype(int)  
df.reset\_index(drop=True, inplace=True)  
num = len(df)  
  
section = np.arange(1, 13)  
record = 0  
fig, ax = plt.subplots()  
plt.style.use('classic')  
plt.title("2010-2015 Beijing average PM2.5(from PM\_US Post) per month")  
  
for nowyear in range(2010, 2016):  
 i = record  
 result = [0 for i in range(13)]  
 nowsum = 0  
 cntday = 0  
 nowmonth = 1  
 while i < num:  
 if df['month'][i] == nowmonth:  
 cntday = cntday + 1  
 nowsum = nowsum + df['PM\_US Post'][i]  
 else:  
 if df['year'][i] != nowyear:  
 record = i  
 result[nowmonth] = nowsum / cntday  
 break  
 result[nowmonth] = nowsum / cntday  
 cntday = 1  
 nowsum = df['PM\_US Post'][i]  
 nowmonth = df['month'][i]  
 i = i + 1  
  
 result = result[1:]  
  
 #  
 x = np.array(range(1, 13))  
  
 plt.plot(x, result, linestyle='-', label=str(nowyear))  
  
plt.legend()  
plt.show()



1. **心得与体会**

在本次实验中，通过几道题目的引导，逐步掌握了python中使用matplotlib并结合之前的数据分析处理工具numpy及pandas等对数据可视化的基本方法。非常感谢老师的实验设计！