Assignment 02

12132834 曹喆

第一题：

1. 1

题目：

1. Significant earthquakes since 2150 B.C.

The Significant Earthquake Database contains information on destructive earthquakes from 2150 B.C. to the present. Select all columns and download the entire significant earthquake data file in `.tsv` format by clicking the `Download TSV File` button. Click the variable name for more information. Read the file (e.g., `earthquakes-2021-10-13_13-22-50_+0800.tsv`) as an object and name it `Sig_Eqs`.

1.1 [5 points] Compute the total number of deaths caused by earthquakes since 2150 B.C. in each country, and then print the top ten countries along with the total number of deaths.

代码：

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from collections import Counter
Sig_Eqs = pd.read_table('earthquakes-2021-10-13_20-13-06_+0800.tsv')
Sig_Eqs_A = np.array(Sig_Eqs[['Country','Deaths']])
C_D = []
row = len(Sig_Eqs_A)
k = 0
for i in range(row):
    if Sig_Eqs_A[i][1] > 0:
        C_D.append(Sig_Eqs_A[i])
C_D_dic = {}
for C_D_row in C_D:
    if C_D_row[0] in C_D_dic:
        C_D_dic[C_D_row[0]] += C_D_row[1]
    else:
        C_D_dic[C_D_row[0]] = C_D_row[1]
result = []
for (key,value) in C_D_dic.items():
    result.append([key,value])
result = sorted(result, key =(lambda x:x[1]), reverse = True)
print('top ten countries along with the total number of deaths')
for i in range(10):
    print(str(i+1)+'. country: '+str(result[i][0])+' death: '+str(result[i][1]))
```

输出结果：

输出为地震死亡总人数前十多的国家

```
top ten countries along with the total number of deaths
1. country: CHINA death: 2074900.0
2. country: TURKEY death: 1074769.0
3. country: IRAN death: 1011437.0
4. country: SYRIA death: 439224.0
5. country: ITALY death: 434863.0
6. country: HAITI death: 323472.0
7. country: AZERBAIJAN death: 317219.0
8. country: JAPAN death: 278138.0
9. country: ARMENIA death: 191890.0
10. country: PAKISTAN death: 148764.0
```

参考：

列表排序方法：(50条消息) python的sort()函数详解_robinson_的博客-CSDN博客_python sort

1.2

题目：

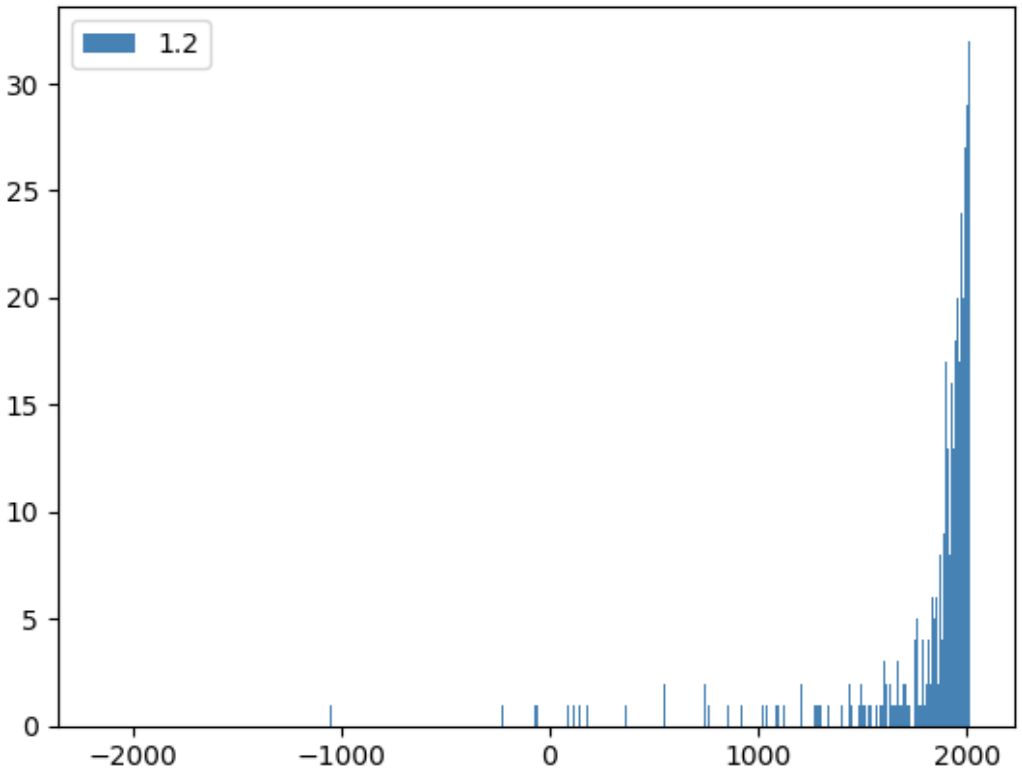
1.2 [10 points] Compute the total number of earthquakes with magnitude larger than `6.0` (use column `Mag` as the magnitude) worldwide each year, and then plot the time series. Do you observe any trend? Explain why or why not?

代码：

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from collections import Counter
Sig_Eqs = pd.read_table('earthquakes-2021-10-13_20-13-06_+0800.tsv')
Sig_Eqs_M = np.array(Sig_Eqs[['Year','Mag']])
num = len(Sig_Eqs_M)
Mag = []
for i in range(num):
    if(Sig_Eqs_M[i][1] > 6.0):
        Mag.append(int(Sig_Eqs_M[i][0]))
Mag_counter = Counter(Mag)
counter_mag_x = list(Mag_counter.keys())
counter_mag_y = list(Mag_counter.values())
plt.bar(counter_mag_x, counter_mag_y, ls='-', width=2, label='1.2', color='steelblue')
plt.legend()
plt.show()
```

输出结果：

输出每年6级以上的地震的次数



能够从输出的柱形图看出，每年的六级地震次数在逐年上升。随着科学技术的发展，越来越多的地震被测出并记载，在古代的地震很可能没有被记载。

参考：

绘制柱形图的方法：(50 条消息) Python 数据可视化之 12 种常用图表的绘制（一）——折线图/柱形图/条形图/散点图/气泡图/面积图_liuzuoping 的博客-CSDN 博客

1.3

题目：

1.3 [10 points] Write a function `CountEq_LargestEq` that returns both (1) the total number of earthquakes since 2150 B.C. in a given country AND (2) the date of the largest earthquake ever happened in this country. Apply `CountEq_LargestEq` to every country in the file, report your results in a descending order.

代码：

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from collections import Counter
Sig_Eqs_0 = pd.read_table('earthquakes-2021-10-13_20-13-06_+0800.tsv')
Sig_Eqs_3_0 = np.array(Sig_Eqs_0[['Country','Mag']])
eq_country_choose = []
for i_count in range(len(Sig_Eqs_3_0)):
    if(Sig_Eqs_3_0[i_count][1] > 0):
        eq_country_choose.append(Sig_Eqs_3_0[i_count][0])
country_counter = Counter(eq_country_choose)
country = []
for keys,values in country_counter.items():
    country.append(keys)
def CountEq_LargestEq(country):
    Sig_Eqs = pd.read_table('earthquakes-2021-10-13_20-13-06_+0800.tsv')
    Sig_Eqs_3 = np.array(Sig_Eqs[['Year','Mo','Dy','Country','Mag']])
    eq_country = []
    for j in range(len(Sig_Eqs_3)):
        eq_country.append(Sig_Eqs_3[j][3])
    eq_times_counter = Counter(eq_country)
    print('the total number of earthquakes since 2150 B.C. of '+country +' is '+str(eq_times_counter[country]))
    num = len(Sig_Eqs_3)
    Mag_3 = []
    for i in range(num):
        if(Sig_Eqs_3[i][4] > 0):
            Mag_3.append(Sig_Eqs_3[i])
    num_3 = len(Mag_3)
    country_eq = []
    for i_eq_countr in range(num_3):
        if(Mag_3[i_eq_countr][3] == country):
            country_eq.append(Mag_3[i_eq_countr])
    len_country_eq = len(country_eq)
    country_eq_mag = []
    for i_max in range(len_country_eq):
        country_eq_mag.append(int(country_eq[i_max][4]))
    max_mag = max(country_eq_mag)
    max_mag_index = country_eq_mag.index(max_mag)
    max_eq_date = str(country_eq[max_mag_index][0])+'-'+str(country_eq[max_mag_index][1])+'-'+str(country_eq[max_mag_index][2])
    print('the date of the largest earthquake ever happened in '+country+' is '+max_eq_date)

len_country = len(country)
for i_out_put in range(len_country):
    CountEq_LargestEq(country[i_out_put])
```

输出结果：

每个国家的发生地震的次数和该国家发生最大地震的日期

```
the total number of earthquakes since 2150 B.C. of JORDAN is 5
the date of the largest earthquake ever happened in JORDAN is -2150.0-nan-nan
the total number of earthquakes since 2150 B.C. of TURKMENISTAN is 11
the date of the largest earthquake ever happened in TURKMENISTAN is 1895.0-7.0-8.0
the total number of earthquakes since 2150 B.C. of ISRAEL is 23
the date of the largest earthquake ever happened in ISRAEL is -31.0-9.0-2.0
the total number of earthquakes since 2150 B.C. of GREECE is 269
the date of the largest earthquake ever happened in GREECE is 365.0-7.0-21.0
the total number of earthquakes since 2150 B.C. of IRAN is 380
the date of the largest earthquake ever happened in IRAN is -400.0-nan-nan
the total number of earthquakes since 2150 B.C. of KYRGYZSTAN is 14
the date of the largest earthquake ever happened in KYRGYZSTAN is 1946.0-11.0-2.0
the total number of earthquakes since 2150 B.C. of CHINA is 610
the date of the largest earthquake ever happened in CHINA is 1303.0-9.0-17.0
the total number of earthquakes since 2150 B.C. of RUSSIA is 150
the date of the largest earthquake ever happened in RUSSIA is 1952.0-11.0-4.0
the total number of earthquakes since 2150 B.C. of PORTUGAL is 26
the date of the largest earthquake ever happened in PORTUGAL is -60.0-nan-nan
the total number of earthquakes since 2150 B.C. of ALBANIA is 56
the date of the largest earthquake ever happened in ALBANIA is 1893.0-6.0-14.0
the total number of earthquakes since 2150 B.C. of GEORGIA is 15
the date of the largest earthquake ever happened in GEORGIA is 1905.0-10.0-21.0
the total number of earthquakes since 2150 B.C. of SOUTH KOREA is 20
the date of the largest earthquake ever happened in SOUTH KOREA is 27.0-nan-nan
the total number of earthquakes since 2150 B.C. of TURKEY is 330

show more (open the raw output data in a text editor) ...

the date of the largest earthquake ever happened in MADAGASCAR is 2017.0-1.0-11.0
the total number of earthquakes since 2150 B.C. of ZAMBIA is 1
the date of the largest earthquake ever happened in ZAMBIA is 2017.0-2.0-24.0
the total number of earthquakes since 2150 B.C. of COMOROS is 1
the date of the largest earthquake ever happened in COMOROS is 2018.0-5.0-15.0
```

第二题：

题目：

2. Wind speed in Shenzhen during the past 10 years

In this problem set, we will examine how wind speed changes in Shenzhen during the past `10` years, we will take a look at the hourly weather data measured at the BaoAn International Airport. The data set is from NOAA Integrated Surface Dataset. Download the file 2281305.zip, where the number `2281305` is the site ID. Extract the zip file, you should see a file named `2281305.csv`. Save the `.csv` file to your `working directory`.

Read page `8-9` of the comprehensive user guide for the detailed format of the wind data. Explain how you filter the data in your report.

[10 points] Plot monthly averaged wind speed as a function of the observation time. Is there a trend in monthly averaged wind speed within the past `10` years?

首先，csv 文件的’WND’表示测量风的一些参数，通过查找用户手册，逗号隔开的五个参数分别代表：测风方向角、测风方向质量代码、测风类型代码、测风速度、测风速度质量代码，在 dataframe 中运用 split 函数将这五个参部分开，具体代码为：

```
wind_s = wind_d_s['WND'].str.split(',',expand = True)
```

通过逗号作为切割符号，将其分开成为五个参数。

代码：

```
wind_d_s = wind[['DATE','WND']]
wind_y_d = wind_d_s['DATE'].str.split('-',expand = True)
wind_s = wind_d_s['WND'].str.split(',',expand = True)
wind_merge = pd.merge(wind_y_d, wind_s, left_index=True, right_index=True)
wind_merge.drop(index = (wind_merge.loc[(wind_merge[3] == '9999')].index), inplace=True)
wind_merge.drop(index = (wind_merge.loc[(wind_merge[4] != '1')].index), inplace=True)
wind_merge[3] = wind_merge[3].str[2].astype(int)
wind_afterwash = wind_merge[['0_x','1_x',3]]
test = wind_merge.groupby(['0_x','1_x'],as_index=False).mean()
X = []
Y = []
for i_out in range(len(test)):
    X.append(float(test['0_x'][i_out]+'.'+test['1_x'][i_out]) - 2000)
    Y.append(test[3][i_out])
    print(test['0_x'][i_out]+' - '+test['1_x'][i_out]+' 的平均风速为：  '+str(test[3][i_out]))
plt.bar(X,Y,width = 0.5)
```

输出结果：

从 2010 年到 2020 年每月的平均风速为：

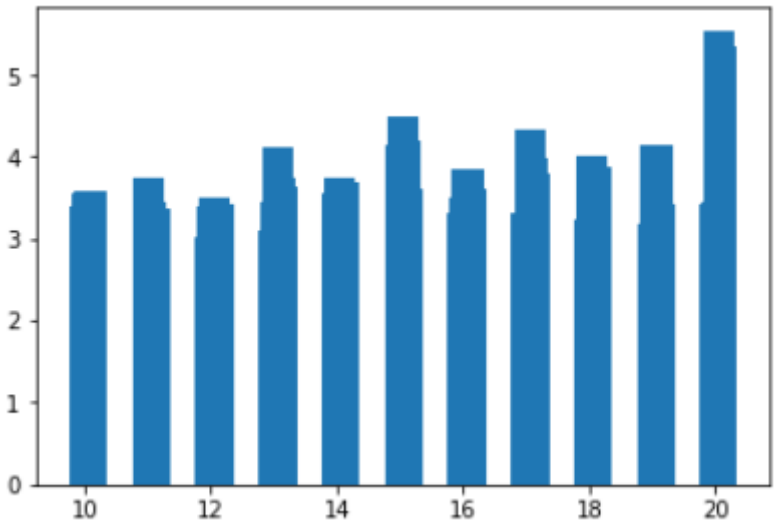
```
2010 - 01 的平均风速为:  2.756267409470752
2010 - 02 的平均风速为:  3.388059701492537
2010 - 03 的平均风速为:  3.3606998654104982
2010 - 04 的平均风速为:  3.191340782122905
2010 - 05 的平均风速为:  3.293640054127199
2010 - 06 的平均风速为:  3.5444444444444443
2010 - 07 的平均风速为:  3.5619946091644206
2010 - 08 的平均风速为:  2.5954301075268815
2010 - 09 的平均风速为:  2.5933147632311977
2010 - 10 的平均风速为:  3.58974358974359
2010 - 11 的平均风速为:  2.5195530726256985
2010 - 12 的平均风速为:  2.7671601615074026
2011 - 01 的平均风速为:  3.7462887989203777
2011 - 02 的平均风速为:  2.5529061102831596
2011 - 03 的平均风速为:  3.096075778078484
2011 - 04 的平均风速为:  2.8284518828451883
2011 - 05 的平均风速为:  2.945872801082544
2011 - 06 的平均风速为:  3.4407252440725244
2011 - 07 的平均风速为:  3.025537634408602
2011 - 08 的平均风速为:  2.8423180592991915
2011 - 09 的平均风速为:  3.1152777777777776
2011 - 10 的平均风速为:  2.873144399460189
2011 - 11 的平均风速为:  2.5597222222222222
2011 - 12 的平均风速为:  3.381081081081081
2012 - 01 的平均风速为:  3.012129380053908

show more (open the raw output data in a tex

2020 - 05 的平均风速为:  4.306970509383378
2020 - 06 的平均风速为:  5.54798331015299
2020 - 07 的平均风速为:  5.350806451612903
2020 - 08 的平均风速为:  3.6090534979423867
2020 - 09 的平均风速为:  3.0823970037453186

<BarContainer object of 129 artists>
```

做成柱状图为：



可以看出：从 2010 年到 2019 年每年每月的平均风速都大致相同，但 2020 年的平均风速明显增加。

参考：

[(51 条消息) pandas 的字符串的分割之 str.split()_lyy 的博客-CSDN 博客_pandas split]

第三题：

题目：

3. Explore a data set

Browse the CASEarth, NOAA Land-Based Datasets and Products, or Advanced Global Atmospheric Gases Experiment (AGAGE) website. Search and download a data set you are interested in. You are also welcome to use data from your group in this problem set. But the data set should be in `csv`, `XLS`, or `XLSX` format, and have temporal information.

3.1 [5 points] Load the `csv`, `XLS`, or `XLSX` file, and clean possible data points with missing values or bad quality.

3.2 [5 points] Plot the time series of a certain variable.

3.3 [5 points] Conduct at least `5` simple statistical checks with the variable, and report your findings.

本题我选择的是中国水稻出口数据表。

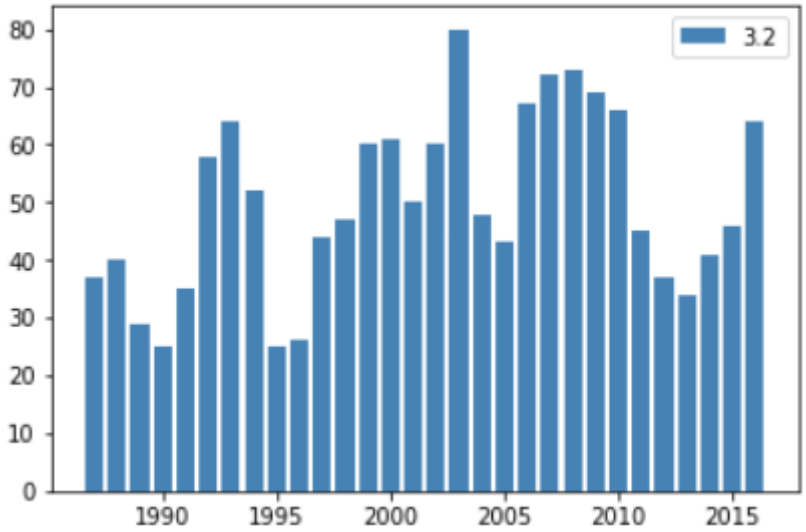
3.1：

本题要求清洗数据，在该表中每行出口水稻的具体数量确实的数据清除。代码为：

```
for i_clean in range(len(export)):
      if(export[i_clean][2] > 0):
            export_after_clean.append(export[i_clean]) #洗数据
```

3.2：

本题要求展示出一个确定的变量随时间变化的趋势，我选择的是列出每年出口水稻的总次数作为变量，最后每年的水稻出口次数的统计图为：



3.3：

本题要求对选定变量至少计算 5 次，我分别计算了 1987 年、1993 年、1995 年、2000 年、2003 年、2005 年、2008 年、2013 年、2015 年的中国出口水稻的次数，结果为：

```
37
64
25
61
80
43
73
34
46
```

可以看出：1987 年-1993 年水稻出口次数增多，而 1993 年-1995 年水稻出口次数减少、1995 年-2003 年水稻出口次数增多、2003 年-2005 年出口次数减少、2005 年-2008 年水稻出口次数增多、2008 年-2013 年出口次数减少、2013 年-2015 年水稻出口次数增多

全部代码为：

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from collections import Counter
export_all = pd.read_csv('C:\\Users\\Cao Zhe\\Desktop\\南科大课后作业\\环境编程\\PS2\\Rice_China_export_Quantity.csv',engine='python')
export = export_all[['Partner Countries','Year Code','Value']]
export = np.array(export)
export_after_clean = []
for i_clean in range(len(export)):
    if(export[i_clean][2] > 0):
        export_after_clean.append(export[i_clean]) #洗数据
count_country = []
for i_count in range(len(export_after_clean)):
    count_country.append(int(export_after_clean[i_count][1]))
year_count = Counter(count_country) #统计每年的出口次数
print(year_count)
X = []
Y = []
for key,value in year_count.items():
    X.append(key)
    Y.append(value)
plt.bar(X, Y, label='3.2', color='steelblue')
plt.legend()
plt.show()
print(year_count[1987])
print(year_count[1993])
print(year_count[1995])
print(year_count[2000])
print(year_count[2003])
print(year_count[2005])
print(year_count[2008])
print(year_count[2013])
print(year_count[2015])
```