中国石油大学（华东）
CHINA UNIVERSITY OF PETROLEUM

2020—2021 学年第 2 学期
《程序设计(Python)》课程报告

专业班级	计算 2001
姓　　名	孙铭俊
学　　号	2007010119
讲解成绩	
报告成绩	
评阅教师	张学辉

2021 年 6 月 13 日

基于全球温度变化的数据挖掘报告

一、数据描述

1.数据来源:

本项目数据集取自 https://www.kaggle.com/sevgisarac/temperature-change ，由世界粮农组织发布，且基于公开获得的 GISTEMP，即美国国家航空航天局戈达德空间研究所（NASA-GISS）分发的全球表面温度变化数据，其内容涵盖了 1961-2019 年间各个国家的地表温度变化。FAOSTAT_data_11-24-2020.csv 为相关国家标准码表格，与天气变化相关度较小，且数据量小，故不进行讨论。Environment_Temperature_change_E_All_Data_NOFLAG.csv 为温度变化数据表，数据量大，本报告将围绕其进行详细分析。

2.主要字段:

涵盖的国家/地区(Area 字段):190 个国家和 37 个其他领土实体。

月份(Months):1 到 12 月，4 个季度，气象年

测量单位(Unit):摄氏度℃

统计元素(Element):

(1).温度变化(Temperature change)

(2).标准差(Standard Deviation)

3.扩展知识:

本项目气象年(Meteorological year)是指以 1961-2019 年的月平均值为依据，从 1961-2019 年的资料中选取一年各月接近其平均值。

二、数据预处理

1.读取数据

```
import numpy as np
import pandas as pd
import seaborn as sns
from matplotlib import pyplot as plt

Etemp_path='../dataset/Environment_Temperature_change_E_All_Data_NOFLAG.csv'
Code_path='../dataset/FAOSTAT_data_11-24-2020.csv'
data = pd.read_csv(Etemp_path, encoding='latin-1')
data2 = pd.read_csv(Code_path)
data
```

:

	Area Code	Area	Months Code	Months	Element Code	Element	Unit	Y1961	Y1962	Y1963	...	Y2010	Y2011	Y2012	Y2013	Y2014	Y2015	Y2016	Y20
0	2	Afghanistan	7001	January	7271	Temperature change	°C	0.777	0.062	2.744	...	3.601	1.179	-0.583	1.233	1.755	1.943	3.416	1.2
1	2	Afghanistan	7001	January	6078	Standard Deviation	°C	1.950	1.950	1.950	...	1.950	1.950	1.950	1.950	1.950	1.950	1.950	1.9
2	2	Afghanistan	7002	February	7271	Temperature change	°C	-1.743	2.465	3.919	...	1.212	0.321	-3.201	1.494	-3.187	2.699	2.251	-0.3
3	2	Afghanistan	7002	February	6078	Standard Deviation	°C	2.597	2.597	2.597	...	2.597	2.597	2.597	2.597	2.597	2.597	2.597	2.5
4	2	Afghanistan	7003	March	7271	Temperature change	°C	0.516	1.336	0.403	...	3.390	0.748	-0.527	2.246	-0.076	-0.497	2.296	0.8
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
9651	5873	OECD	7018	Jun□Jul□Aug	6078	Standard Deviation	°C	0.247	0.247	0.247	...	0.247	0.247	0.247	0.247	0.247	0.247	0.247	0.2
9652	5873	OECD	7019	Sep□Oct□Nov	7271	Temperature change	°C	0.036	0.461	0.665	...	0.958	1.106	0.885	1.041	0.999	1.670	1.535	1.1
9653	5873	OECD	7019	Sep□Oct□Nov	6078	Standard Deviation	°C	0.378	0.378	0.378	...	0.378	0.378	0.378	0.378	0.378	0.378	0.378	0.3
9654	5873	OECD	7020	Meteorological year	7271	Temperature change	°C	0.165	-0.009	0.134	...	1.246	0.805	1.274	0.991	0.811	1.282	1.850	1.3
9655	5873	OECD	7020	Meteorological year	6078	Standard Deviation	°C	0.260	0.260	0.260	...	0.260	0.260	0.260	0.260	0.260	0.260	0.260	0.2

9656 rows × 66 columns

图 1　运行结果截图

2.数据中可能存在的问题：

通过 plot()画图可得图 2，通过分析得出该数据集存在部分缺失值，

```
plt.plot(data.isnull())
plt.show()
```

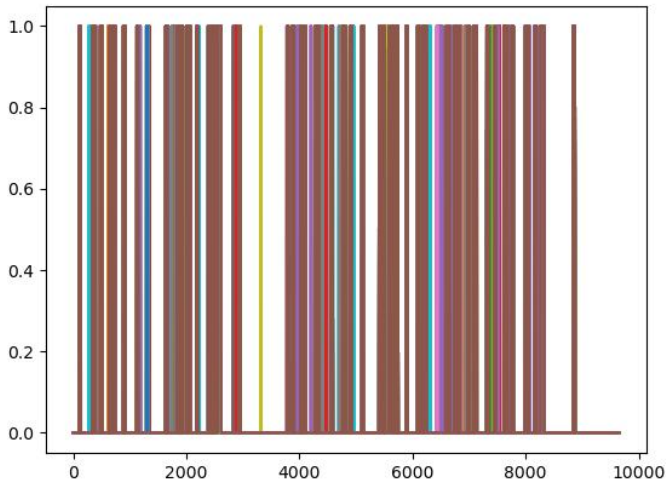

图 2　plot 画图结果

3.可能会造成的影响：

可能会影响结果准确性以及运行效率

4.处理缺失数据

#检验是否存在缺失值

```
print("处理前数据行列数:",data.shape)
data=data.dropna()
print("处理后数据行列数:",data.shape)
```

```
In [34]:
1  print("处理前数据行列数:", data.shape)
2  data=data.dropna()
3  print("处理后数据行列数:", data.shape)
处理前数据行列数: (9656, 66)
处理后数据行列数: (6760, 66)
```

5.提取主要字段

经分析，该数据集的 Element Code，Area Code，Months Code , Unit 气温变化无直接关系，仅仅是用于区分 Element,Area,Months 以及显现单位的，故去掉，仅保留影响气温的因变量，只取 1 到 12 月进行分析。

```
data.drop(columns=['Area Code', 'Months Code', 'Element Code', 'Unit'], axis=1, inplace=True)
TempC=data.loc[data.Months.isin(['January', 'February', 'March', 'April', 'May', 'June', 'July','August', 'September', 'October', 'November', 'December'])]
```

三、数据分析

1.总体分析-- 获取世界温度变化 (均值显示)

（1）排除大区如 Africa，Asia，world 的干扰

```
regions=TempC[TempC.Area.isin(['World', 'Africa',
        'Eastern Africa', 'Middle Africa', 'Northern Africa',
        'Southern Africa', 'Western Africa', 'Americas',
        'Northern America', 'Central America', 'Caribbean',
        'South America', 'Asia', 'Central Asia', 'Eastern Asia',
        'Southern Asia', 'South-Eastern Asia', 'Western Asia', 'Europe',
        'Eastern Europe', 'Northern Europe', 'Southern Europe',
        'Western Europe', 'Oceania', 'Australia and New Zealand',
        'Melanesia', 'Micronesia', 'Polynesia', 'European Union',
        'Least Developed Countries', 'Land Locked Developing Countries',
        'Small Island Developing States',
        'Low Income Food Deficit Countries',
        'Net Food Importing Developing Countries', 'Annex I countries',
        'Non-Annex I countries', 'OECD'])]
```

```
TempC=TempC[~TempC.Area.isin(['World', 'Africa',
        'Eastern Africa', 'Middle Africa', 'Northern Africa',
        'Southern Africa', 'Western Africa', 'Americas',
        'Northern America', 'Central America', 'Caribbean',
        'South America', 'Asia', 'Central Asia', 'Eastern Asia',
        'Southern Asia', 'South-Eastern Asia', 'Western Asia', 'Europe',
        'Eastern Europe', 'Northern Europe', 'Southern Europe',
        'Western Europe', 'Oceania', 'Australia and New Zealand',
        'Melanesia', 'Micronesia', 'Polynesia', 'European Union',
        'Least Developed Countries', 'Land Locked Developing Countries',
        'Small Island Developing States',
        'Low Income Food Deficit Countries',
        'Net Food Importing Developing Countries', 'Annex I countries',
        'Non-Annex I countries', 'OECD'])]
TempC
```

（2）运用 melt()函数获取另一种形式的 TempC

```
TempC=TempC.melt(id_vars=['Area','Months','Element'],var_name='Year',
value_name='TempC')
TempC['Year'] = TempC['Year'].str[1:].astype('str')
TempC
```

Out[102]:

	Area	Months	Element	Year	TempC
0	Afghanistan	January	Temperature change	1961	0.777
1	Afghanistan	January	Standard Deviation	1961	1.950
2	Afghanistan	February	Temperature change	1961	-1.743
3	Afghanistan	February	Standard Deviation	1961	2.597
4	Afghanistan	March	Temperature change	1961	0.516
...	...	...	...	...	...
229623	Zimbabwe	October	Standard Deviation	2019	0.727
229624	Zimbabwe	November	Temperature change	2019	2.448
229625	Zimbabwe	November	Standard Deviation	2019	0.861
229626	Zimbabwe	December	Temperature change	2019	2.083
229627	Zimbabwe	December	Standard Deviation	2019	0.804

229628 rows × 5 columns

图 3　运行结果截图

（3）通过世界温度变化绘制相关图像，绘制每个值的散点图

```
# 全世界温度变化平均值
AvgT=TempC.loc[TempC.Element=='Temperature
change'].groupby(['Year'],as_index=False).mean()
# 每个国家温度变化平均值
AvgTC=TempC.loc[TempC.Element=='Temperature
change'].groupby(['Area','Year'],as_index=False).mean()
plt.figure(figsize=(15,10))
plt.scatter(TempC['Year'].loc[TempC.Element=='Temperature
change'],TempC['TempC'].loc[TempC.Element=='Temperature change'])
plt.plot(AvgT.Year,AvgT.TempC,'r',label='Average')#平均温度折线
plt.axhline(y=0.0, color='k', linestyle='-')
plt.xlabel('Year')
plt.xticks(np.linspace(0,58,20),rotation=45)
plt.ylabel('Temperature change')
plt.legend()
plt.title('Temperature Change of the World')
plt.show()
```

结果见图 4

通过世界平均温度绘制相关图像，绘制每个国家的年份均温折线图

```
plt.figure(figsize=(15,10))
for i in AvgTC.Area.unique():

plt.plot(AvgTC.Year.loc[AvgTC.Area==str(i)],AvgTC.TempC.loc[AvgTC.Area==str
(i)],linewidth=0.5)

plt.plot(AvgT.Year,AvgT.TempC,'r',linewidth=2.0)
```

```
plt.axhline(y=0.0, color='k', linestyle='-')
plt.xlabel('Year')
plt.xticks(np.linspace(0,58,20),rotation=45)
plt.ylabel('Average Temperature change')
plt.title('Average Temperature Change of the World')
plt.show()
```

结果见图 5

通过折线图和散点图我们不难发现，尽管部分年份，部分地区的平均温度变化会有一定程度的下降，但总体上世界平均温度是在波动上升的

1.局部分析 -- 选取局部地区进行具体分析

选取中国为样本进行具体分析，并以年份为分界进行折线图的绘制

```
china_t=data.loc[data.Months.isin(['January', 'February', 'March', 'April', 'May', 'June', 'July','August', 'September', 'October', 'November', 'December'])]
chi=china_t.loc[TempC.Area=='China']
plt.figure(figsize=(15,10))
sns.lineplot(x=chi.Months.loc[chi.Element=='Temperature change'],y=chi.Y1961.loc[chi.Element=='Temperature change'], label='Y1961')
sns.lineplot(x=chi.Months.loc[chi.Element=='Temperature change'],y=chi.Y1971.loc[chi.Element=='Temperature change'], label='Y1971')
sns.lineplot(x=chi.Months.loc[chi.Element=='Temperature change'],y=chi.Y1981.loc[chi.Element=='Temperature change'], label='Y1981')
sns.lineplot(x=chi.Months.loc[chi.Element=='Temperature change'],y=chi.Y1991.loc[chi.Element=='Temperature change'], label='Y1991')
sns.lineplot(x=chi.Months.loc[chi.Element=='Temperature change'],y=chi.Y2001.loc[chi.Element=='Temperature change'], label='Y2001')
plt.xlabel('Months')
plt.ylabel('Temperature change (C)')
```

```
plt.title('Temperature Change in China')
plt.show()
```

详细见图 6 进行不同月份温度的比较

```
chi=chi.melt(id_vars=['Area','Months','Element'],var_name='Year',
value_name='TempC')
chi['Year'] = chi['Year'].str[1:].astype('str')

# # chi.info()
plt.figure(figsize=(15,15))
plt.subplot(211)
for i in chi.Year.unique():
    plt.plot(chi.Months.loc[chi.Year==str(i)].loc[chi.Element=='Temperature
    change'],chi.TempC.loc[chi.Year==str(i)].loc[chi.Element=='Temperature
    change'],linewidth=0.5)
plt.plot(chi.Months.unique(),chi.loc[chi.Element=='Temperature
change'].groupby(['Months']).mean(),'r',linewidth=2.0,label='Average')
plt.xlabel('Months',)
plt.xticks(rotation=45)
plt.ylabel('Temperature change')
plt.title('Temperature Change in China')
plt.legend()

plt.subplot(212)
plt.plot(chi.Months.loc[chi.Year=='1961'].loc[chi.Element=='Standard
Deviation'],chi.TempC.loc[chi.Year=='1961'].loc[chi.Element=='Standard
Deviation'])
plt.xlabel('Year')
plt.xticks(rotation=45)
plt.ylabel('Standard Deviation')
```

```
plt.title('Standard Deviation of Temperature Change in China')

plt.subplots_adjust(hspace=0.3)
plt.show()
```

详细见图 7、8

得出结论：本次分析可以获得中国在 1961-2019 年间冬季温度变化发生了较大的变化，而本次数据记录的是地表温度，因此可以猜测可能是城市化或是能源结构的改善导致了冬季温度变化的差异。

四、数据可视化展示

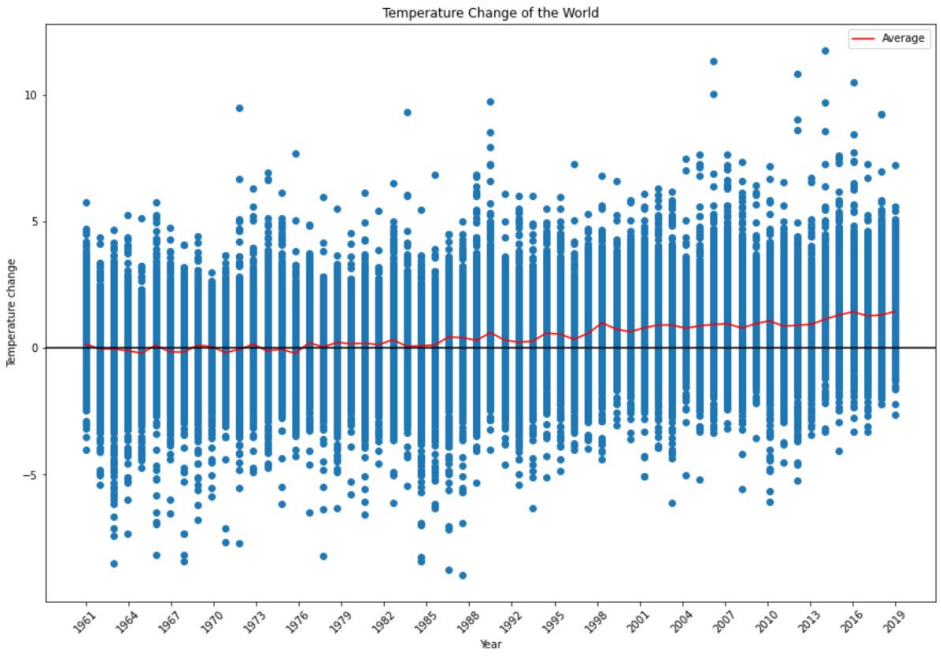

图 4　世界温度的散点图

通过图 4 可以看出离群点有向上的趋势，红色线是平均温度线，可以看出有波动上升的趋势.

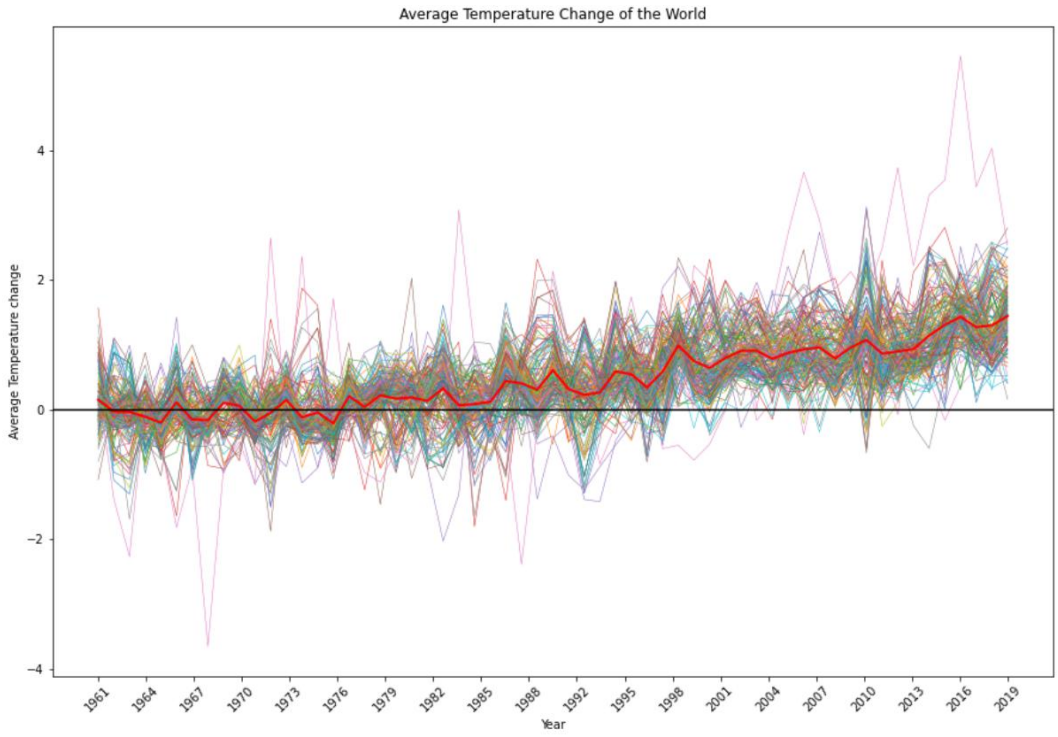

图 5　各个国家平均温度变化趋势折线图

图 5 体现了并非所有国家均是温度逐年上升，但是从总体大部分来看，温度均存在较大的上升趋势。

图 6　中国温度变化趋势折线图（1）

通过图 6 可以看出我国一二月温度随着年份改变差异较大，而 9，10 月则变化较小。

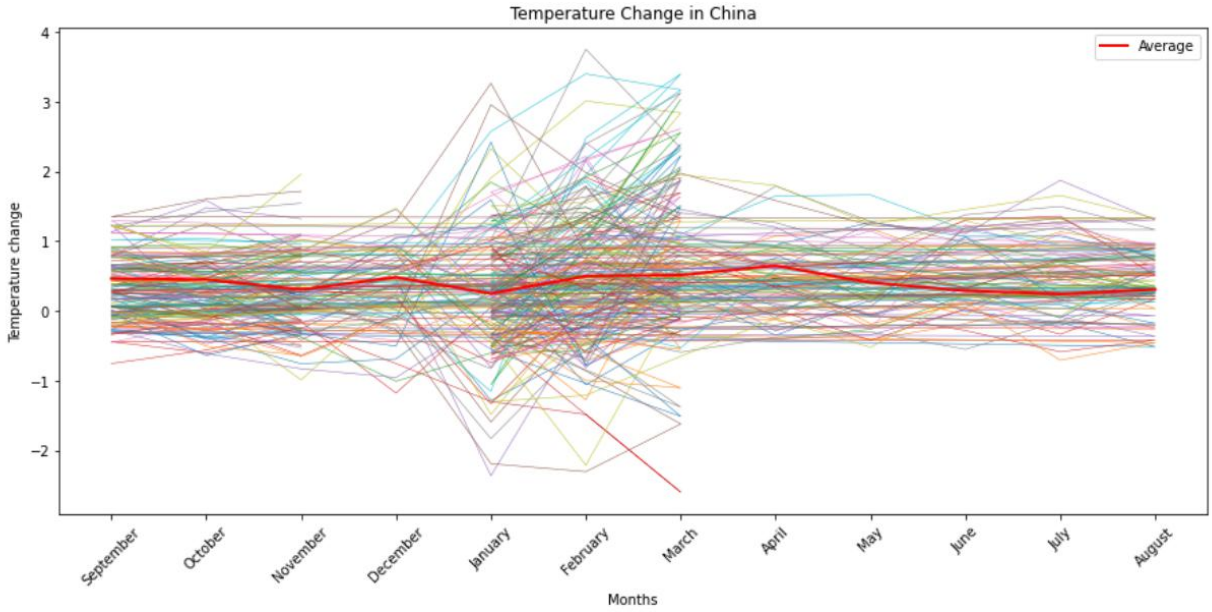

图 7　中国温度变化趋势折线图（2）

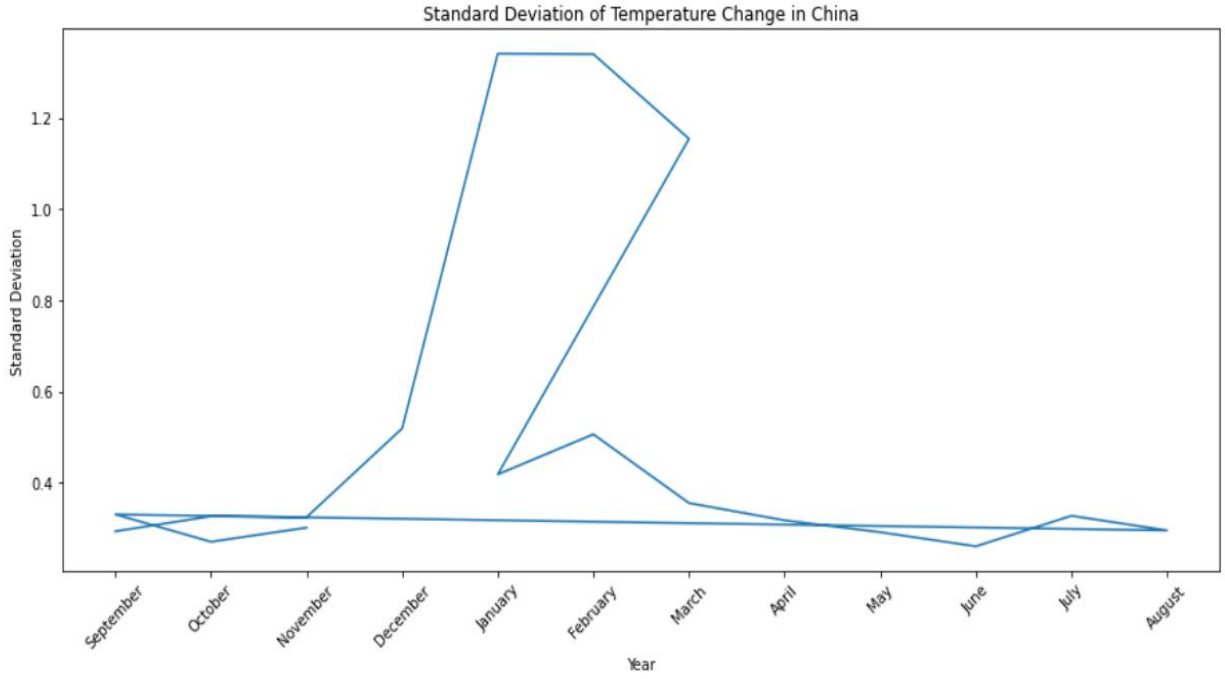

图 8　中国温度变化标准差趋势折线图

图 7、8 可以看出我国冬季标准差大，温度变化的离散程度大，温度不稳定。

五、遇到的问题与解决方法

1.在读入 csv 文件中编码出现问题

```
pandas\_libs\parsers.pyx in pandas._libs.parsers._string_box_utf8()

UnicodeDecodeError: 'utf-8' codec can't decode byte 0xf4 in position 1: invalid continuation byte
```

在运行 data = pd.read_csv(Etemp_path)时出现报错

在查阅相关资料后才发现，Unit 单元地摄氏度是 Latin 字符,

故改为 data = pd.read_csv(Etemp_path, encoding='latin-1')

2.在可视化过程中类型出现问题

```
TypeError: 'value' must be an instance of str or bytes, not a float
```

发现是 months,years 存在重复值, 重复分类在 groupby()中会报错, 进而引起 value 的值出现问题，因此在 year,months 后添加 unique()函数即可解决问题

```
for i in chi.Year.unique():
    plt.plot(chi.Months.loc[chi.Year==str(i)].loc[chi.Element=='Temperature change'],chi.TempC.loc[
# plt.plot(chi.Months.unique(),chi.loc[chi.Element=='Temperature change'].groupby(['Months']).mean(
plt.xlabel('Months',)
```

六、学习总结与反思

本次项目的实践存在几点不足：

1、数据集具体特征的选取，在没有对数据进行清洗以及后续对部分数据的丢弃之前，我们运行出来的结果往往会有问题抑或是运行速度很慢，在对数据进行处理之后，效率才得到了提升；

2、在进行数据分析时，问题最多的就是类型的报错以及索引的使用不当，所有应当更加熟悉 pandas 库，要对 dataframe,series 有一个充分的了解；

因此我认为在项目实践中对应模块的掌握程度很大程度上决定了你对这个数据分析的处理效率，而对于数据进行科学的分析则能够更好地为结果进行服务。

评分参考	得分
(1) 数据描述：数据来源明确，字段描述清楚 (2) 数据预处理：是否进行了有效处理，并阐述了原因 (3) 数据分析：多角度数据分析，描述清楚准确 (4) 可视化展示：多种图表展示，图表类型选择恰当，图表美观完整，对于图表结果能得出相关结论 (5) 问题及解决方法：遇到的问题是否描述清楚，解决方法是否清晰 (6) 文档结构完整、格式排版美观、描述清楚准确	
教师评语	