

## 3.2 文本数据读写

Markdown by 陈威霖

### 1) 写入txt文本

Python 内置的“open()”函数是用来读写数据的**最基本函数**，它不仅可用于读写文本数据，还可以读写二进制数据（随后介绍）。

#### “open()”函数有3个常用参数

- 第一个参数是文件名，是其唯一的位置参数
- 第二个参数“mode”是第一个关键字参数，表示读写状态和是否为二进制
- 第三个参数是“encoding”参数，表明数据是以何种编码形式存储的

下面定义一个子函数，生成一个数据。

```
In [3]: import numpy as np
def creatdat(year,b=0,size=30):## 定义名为creatdat的子函数
    time = ['-'.join([str(year+i),'01','01'])
            for i in range(size)]
    time = np.array(time, dtype='datetime64')
    I = np.arange(size)/size
    dat = b*I*size
    for i in range(1,11):
        A = np.random.random()
        B = np.random.random()
        dat += A*np.sin(I*i*np.pi*2+B*np.pi)
    return time,dat
```

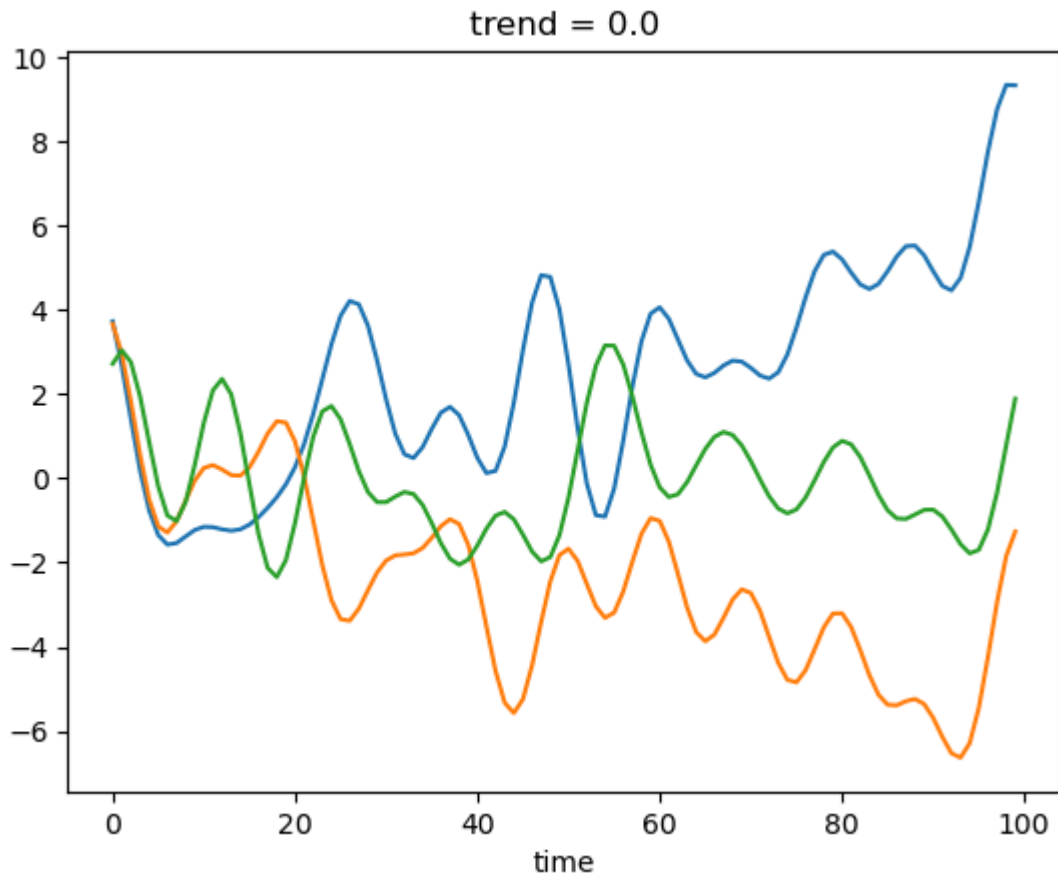
上面的函数用到了随机数，生成一个带趋势（b）的时间序列（年）

1. 主函数利用for循环生成了2100-2199年100年的3个不同趋势的序列;
2. 利用open存储为data.txt

```
In [6]: dat = []
for b in [0.05,-0.05,0]:
    time, tmp = creatdat(2100,b=b,size=100) ##调用creatdat函数
    dat.append(tmp)
```

利用前面学习过的xarray可以看看数据长什么样

```
In [11]: import xarray as xr
da = xr.DataArray(dat,dims=('trend','time'),
                  coords={'time':np.arange(100),'trend':[0.05,-0.05,0]})
for x in da:
    x.plot()
```



- 利用open将数据存储起来
- 使用with语句可以节省close
- "\*"可以将数据分解（元组可分解成各个元素，数据从行开始分解）

注意：写入数据时，需要以字符串的格式写入，所以需将数据转换成字符串。

```
In [12]: fnc = lambda x: '%10.3f' %(x) ##匿名函数Lambda，没有名字的函数，简洁
with open('data.txt','w') as f:
    for t,x,y,z in zip(time,*dat): ## 利用“zip()”函数将时间“time”与3 个时间序列“x,y,z”打包
        tmp = ''.join(list(map(fnc,[x,y,z])))
        tmp = str(t)+tmp+'\n'
        f.write(tmp)
```

## 2) 读取txt文件

- 利用“open()”函数读取“data.txt”
- 同样，利用with语句可省略close
- 使用readlines函数逐行读入
- 还有readline和read，分别是只读一行和一次性读入
- 读入的数据是字符串
- 需要对字符串进行处理和转换才能得到数字

```
In [15]: with open('data.txt','r') as f:
    for i,x in enumerate(f.readlines()):#依次得到每行
        tmp = x.split() #利用字符串的“split()”函数将一个字符串变成时间、3 个数据的列表
        if i<10: #只看前10行
            print(tmp)
```

```
['2100-01-01', '3.727', '3.665', '2.717']
['2101-01-01', '2.676', '2.961', '3.038']
['2102-01-01', '1.411', '1.827', '2.756']
['2103-01-01', '0.196', '0.569', '1.953']
['2104-01-01', '-0.760', '-0.497', '0.864']
['2105-01-01', '-1.347', '-1.141', '-0.186']
['2106-01-01', '-1.573', '-1.288', '-0.882']
['2107-01-01', '-1.538', '-1.021', '-1.016']
['2108-01-01', '-1.381', '-0.536', '-0.563']
['2109-01-01', '-1.227', '-0.057', '0.315']
```

### 3) 读写csv 数据

- csv数据是以逗号隔开的文本数据
- 读写csv数据，可使用Pandas
- 还可使用Python内置的**csv模块**进行读写
- 利用csv.reader读入
- 读入后是字符串

```
In [16]: import csv
with open('成绩.csv', encoding='utf-8') as f:
    for x in csv.reader(f):#利用“csv.reader()”函数读入数据，利用for循环遍历
        print(x)
```

```
['', 'A 同学', 'B 同学', 'C 同学']
['语文', '65', '50', '75']
['数学', '70', '100', '80']
['英语', '99', '100', '59']
```

**利用csv 模块同样可以写csv 格式文件，此处把“data.txt”文件写成csv 格式文件**

- 写数据用csv.writer逐行写入（writerow）
- 写入的数据也需是字符串

```
In [17]: f0 = open('data.csv', 'w', newline='')
cw = csv.writer(f0)
with open('data.txt', 'r') as f:
    for x in f.readlines():
        tmp = x.split()
        cw.writerow(tmp) ##利用“cw.writerow()”函数写入数据
f0.close()
```

可以打开data.csv看看