

## 《Big Data Analytics》实验报告

年级、专业、班级	2018 级机械 1 班	姓名	易弘睿	学号	20186103
实验题目	简单数据处理				
实验时间	2022 年 3 月 19 日	实验地点	A 理 119		
学年学期	2021-2022(2)	实验性质	<input type="checkbox"/> 验证性 <input checked="" type="checkbox"/> 设计性 <input type="checkbox"/> 综合性		
<b>一、实验目的</b> 1) 掌握 Numpy 的使用; 2) 掌握 Pandas 的使用; 3) 掌握 Tushare 的使用;					
<b>二、实验项目内容</b> 统计和输出每个工业(industry)中成交量最大的股票代码(并按照成交量降序输出), 主要过程如下: 1)下载“股票列表”, 从中获取工业类别和每个工业类别的股票代码; 2)分别下载每个工业类别中股票在 2021 年的日 K 线数据; 3)计算每只股票在 2021 年的总成交量; 4)找出每个工业类别中成交量最大的股票; 5)得到所有工业类别的成交量最大股票的数据集, 按照成交量降序排列后输出。					
<b>三、实验过程和结果</b>  本次实验, 我通过使用 Tushare 获取全部股票列表, 并获取不重复的工业列表, 然后根据指定工业的类别获取股票代码列表, 并下载股票交易数据。下载完成后, 在每个工业类别下, 计算每只股票在 2021 年的总成交量, 得到每个工业类别成交量最大的股票, 最后降序排列并输出。 具体实验过程和实验结果如下。 1. 实验过程 1.1 第一部分: 数据集下载 完成股票数据输出的函数。这一部分主要在老师提供的样例中进行修改, 主要修改的内容包括: (1) 获取工业列表后去除 NONE 的类别, 否则无法按类别完成股票交易数据的下载; (2) 使用 time 函数, 在每类工业股票数据下载完成后休息 30s, 以解决访问限制 500 次/min 的问					

题。

```
import tushare as ts
import pandas as pd
import os      # 引用操作目录用来下载和分类每个industry的目录
import shutil
import time    # 用作每分钟500次的权限

#清空数据集的文件夹
def clearData():
    basedir = r".\data"
    if os.path.exists(basedir):
        shutil.rmtree(basedir)
    os.mkdir(basedir)

#获取全部股票列表
def getStockBasic():
    stock_basic = pro.stock_basic(exchange='', list_status='L', fields='ts_code,symbol,name,area,industry,list_date')
    return stock_basic

#获取不重复的工业领域列表
def getCodesAndIndustrty(stock_basic):
    industries=stock_basic['industry'].values
    industries=list(set(industries))
    return industries

#获取指定工业的股票代码列表
def getIndustryCodes(stock_basic,industry):
    industry_codes=stock_basic[stock_basic.industry==industry].ts_code.values
    return industry_codes

#下载指定工业的股票的交易数据
def downloadStockTrans(industry,industry_codes):
    basedir = r".\data"
    #创建存放指定工业交易数据的文件夹
    subdir = os.path.join(basedir, industry)
    os.mkdir(subdir)

    for code in industry_codes:
        trans_filename=os.path.join(subdir,code+".txt")
        print(code,trans_filename)
        stock_trans=pro.daily(ts_code=code, start_date='20210101', end_date='20211231')
        #按照交易日期进行升序排列
        stock_trans=stock_trans.sort_values(by='trade_date',ascending=True)
        #讲pandas的dataframe写入文件, 指定写入时去掉行索引
        stock_trans.to_csv(trans_filename,index=False)

#处理所有工业领域的股票数据
def processIndustries(industries):
    for i in range(0,len(industries),3):
        for industry in industries[i:i+3]:
            industry_codes = getIndustryCodes(stock_basic, industry)
            print(industry,":",industry_codes)
            downloadStockTrans(industry, industry_codes)
            time.sleep(30)    #等待30s. 解决访问限制500次/min的问题

#连接数据库获取工业列表
pro = ts.pro_api('0ff3b82a52cd617c7590b7204cf959ce951cec5e8f6a9385bb6bd655')
stock_basic=getStockBasic()
industries=getCodesAndIndustrty(stock_basic)
print(industries)
clearData()

#去除工业类别为NONE的类并完成下载
industries1 = [k for k in industries if k != None]
print(industries1)
processIndustries(industries1)
```

## 1.2 第二部分：数据处理

完成股票数据的处理。主要分为三步：（1）得到每只股票在 2021 年的总成交量；（2）得到每个工业类别中成交量最大的股票；（3）得到所有工业类别的成交量最大股票的数据集并按照成交量降序排列后输出。

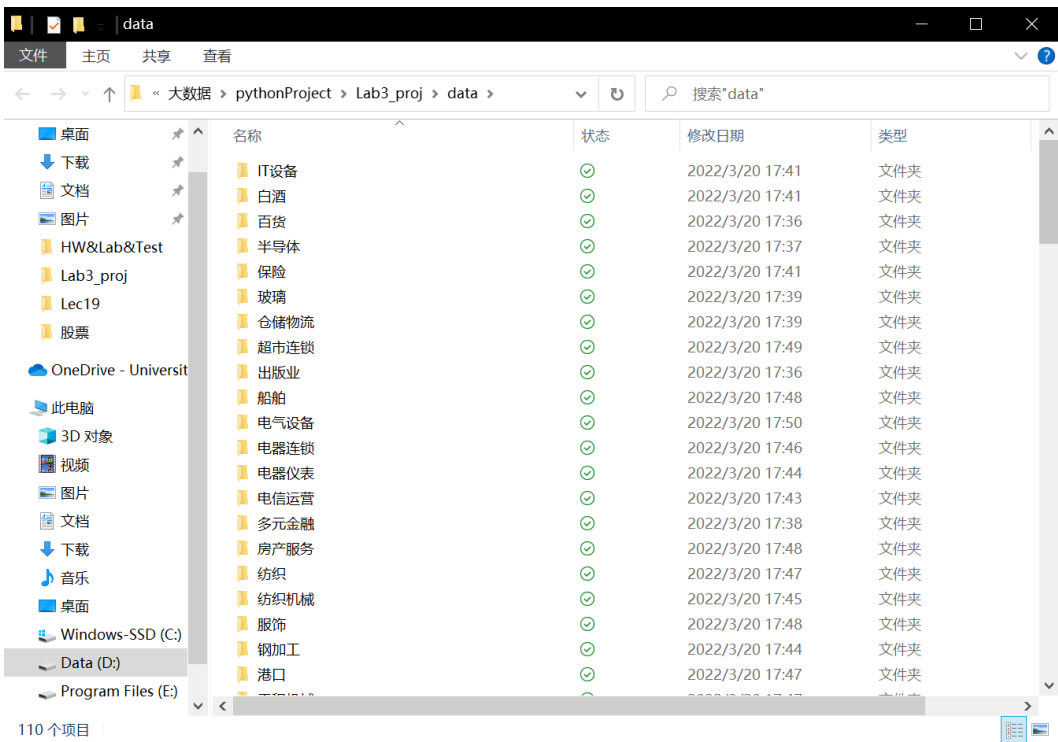
```
# 得到每只股票在2021年的总成交量
filename=os.listdir('./data/')
all_2021=dict()
for i in filename:
    one_type=dict()
    for j in os.listdir('./data/'+i+'/'):
        data=pd.read_csv('./data/'+i+'/'+j)
        one_type[j[:-4]]=round(sum(data['vol']),2)
    all_2021[i]=one_type
print('每只股票在2021年的总成交量',all_2021)

# 得到每个工业类别中成交量最大的股票
all_max=dict()
for i,j in all_2021.items():
    li=sorted(j.items(), key=lambda d: d[1], reverse=True)
    all_max[i]=li[0]
print('每个工业类别中成交量最大的股票',all_max)

# 得到所有工业类别的成交量最大股票的数据集并按照成交量降序排列后输出
lji=sorted(all_max.items(), key=lambda d: d[1][1], reverse=True)
print('按照成交量降序排列后输出',lji)
name=['类别','data']
lji=pd.DataFrame(columns=name,data=lji)
lji.to_csv('成交量最大股票降序排列.csv',encoding='gbk')
```

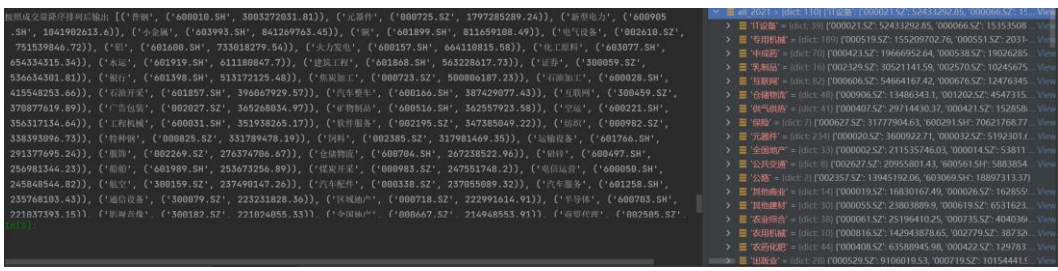
## 2. 实验结果

### 2.1 数据集下载结果



110 个项目

## 2.2 程序运行结果



## 2.3 数据输出结果

成交量最大股票降序排列.csv

文件开始插入页面布局公式数据审阅视图

剪贴板

剪切复制格式刷

等线

11

A^A

B I U

A

文

剪贴板字体

自动保存 关

O15

X ✓ fx

	A	B	C	D	E	F
1	类别	data				
2	普钢	('600010.SH', 3003272031.81)				
3	元器件	('000725.SZ', 1797285289.24)				
4	新型电力	('600905.SH', 1041902613.6)				
5	小金属	('603993.SH', 841269763.45)				
6	铜	('601899.SH', 811659108.49)				
7	电气设备	('002610.SZ', 751539846.72)				
8	铝	('601600.SH', 733018279.54)				
9	火力发电	('600157.SH', 664110815.58)				
10	化工原料	('603077.SH', 654334315.34)				
11	水运	('601919.SH', 611180847.7)				
12	建筑工程	('601868.SH', 563228617.73)				
13	证券	('300059.SZ', 536634301.81)				
14	银行	('601398.SH', 513172125.48)				
15	焦炭加工	('000723.SZ', 500806187.23)				
16	石油加工	('600028.SH', 415548253.66)				
17	石油开采	('601857.SH', 396067929.57)				
18	汽车整车	('600166.SH', 387429077.43)				
19	互联网	('300459.SZ', 370877619.89)				
20	广告包装	('002027.SZ', 365268034.97)				
21	矿物制品	('600516.SH', 362557923.58)				
22	空运	('600221.SH', 356317134.64)				
23	工程机械	('600031.SH', 351938265.17)				
24	软件服务	('002195.SZ', 347385049.22)				
25	纺织	('000982.SZ', 338393096.73)				
26	特种钢	('000825.SZ', 331789478.19)				
27	饲料	('002385.SZ', 317981469.35)				
28	运输设备	('601766.SH', 291377695.24)				
29	服饰	('002269.SZ', 276374706.67)				
30	仓储物流	('600704.SH', 267238522.96)				
31	铅锌	('600497.SH', 256981344.23)				
32	船舶	('601989.SH', 253673256.89)				
33	煤炭开采	('000983.SZ', 247551748.2)				
34	电信运营	('600050.SH', 245848544.82)				
35	航空	('300159.SZ', 237490147.26)				
36	汽车配件	('000338.SZ', 237055089.32)				
37	汽车服务	('601258.SH', 235768103.43)				
38	通信设备	('300079.SZ', 223231828.36)				
39	区域地产	('000718.SZ', 222991614.91)				
40	半导体	('600703.SH', 221037393.15)				
41	影视音像	('300182.SZ', 221024055.33)				

成交量最大股票降序排列

就绪 辅助功能: 不可用

#### 四、实验总结与体会

实验中，主要考察 pandas, numpy, os 包的使用对大量数据文件进行分析。

通过本实验，我更真切认识到不同的算法对于程序运行时间的影响，并通过查询相关资料有效解决了老师给出的样例代码中的如下问题：

（1）获取工业列表后去除 **NONE** 的类别，否则无法按类别完成股票交易数据的下载；

（2）使用 **time** 函数，在每类工业股票数据下载完成后休息 30s，有效解决了 tushare 数据接口每分钟只能访问 500 次的问题。

此外，在数据处理分析方面，除了从指定目录读取文件的操作外，我还练习了通过 **numpy.array** 给多维数组降维。

相比老师的程序，我的程序代码量更小，但 **for** 循环相对更多，之后还要多了解一些并行的代码块以缩短运行时间。

实验报告填写说明：

- 1、第一、二部分由老师提供；
- 2、第三部分填写主要操作步骤（文字）与结果 3（截图）；
- 3、第四部分主要填写遇到的问题与解决问题的方法、总结和体会等；
- 4、报告规范：包含报告页眉、报告的排版、内容是否填写，命名是否规范等内容；
- 5、实验文件与实验报告命名：学号姓名序号.\*（应用对应的扩展名），学号姓名(en).docx，  
例如学号 20161234 的张三同学，他的第一次实验命名为：**20161234 张三(e1)1.docx**