《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 4 月 15 日	实验地点	A 理 119		
学年学期	2021-2022(2)	实验性质	□验证性 ■设计性 □综合性		

一、实验目的

1) 掌握目录和文件的管理；
2) 掌握 Tushare 的使用；
3) 掌握 Matplotlib 的使用；

二、实验项目内容

绘制股票折线图：

1)从 Tushare 下载四个工业领域(自己选择)股票在 2022 年 3 月的日线数据；

2)在同一个 figure 中绘制四个子图，每个子图中绘制一个领域中月成交量最大的三支股票的日 K 线(折线图)；

三、实验过程和结果

本次实验，我首先从 Tushare 随机抽取下载四个工业领域股票在 2022 年 3 月的日线数据，在同一个 figure 中绘制四个子图，每个子图中绘制一个领域中月成交量最大的三支股票的日 K 线(折线图)。

具体实验过程和实验结果如下。

1. 实验过程

随机抽取下载四个工业领域股票在 2022 年 3 月的日线数据：

```python
import os
import pandas as pd
import numpy as np
import tushare as ts
import time
import matplotlib.dates as mdates
import matplotlib.pyplot as plt

plt.rcParams['font.sans-serif'] = ['SimHei'] # 显示中文
plt.rcParams['figure.autolayout'] = True     # 调节间距

# 数据下载
def downloadData(data_path, token):
    if os.path.exists(data_path):
        # 如果该文件夹存在则意味着数据已下好，则不需要重新执行下载的操作
        print("使用已下载的数据")
        return
    else:
        os.mkdir(data_path)
    print("执行数据下载")
    # 初始化端口
    pro = ts.pro_api(token)
    # 获取全部股票列表
    stock_basic = pro.stock_basic(exchange='', list_status='L', fields='ts_code,symbol,name,area,industry,list_date')
    # 获取不重复的工业领域列表
    industries = list(set(stock_basic['industry'].values))
    # 随机抽取4个工业进行下载
    wanted_industries = list(np.random.choice(industries, 4, replace=False))
    # 打印选择的四个工业
    print("此次选择的四个工业是: ", wanted_industries)
    # 删去None类别
    industries = [k for k in wanted_industries if k!=None]

    for i in range(0, len(industries), 3):
        for industry in industries[i:i+3]:
            industry_codes = industry_codes = stock_basic[stock_basic.industry == industry].ts_code.values
            print(industry, ":", industry_codes)
            downloadStockTrans(pro, industry, industry_codes, data_path)
        # 每当一个工业类别的数据下好了以后等待30s，避免访问限制500次/min的问题
        time.sleep(30)
```

下载 3 月的成交数据，获得各个工业类别种交易额前三的股票：

```
# 下载指定工业的股票的交易数据
def downloadStockTrans(pro, industry, industry_codes, basedir):
    # 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)
        # 下载从2022年3月1号到3月31日的数据
        stock_trans = pro.daily(ts_code=code, start_date='20220301', end_date='20220331')
        # 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)

def processCategory(category_path, category_files):
    name_list = []
    vol_list = []
    # 这个列表是为了可以让股票代码和股票文件对应上
    category_path_dict = {}
    for category_file in category_files:
        # 粘合每支股票的路径
        category_file_path = os.path.join(category_path, category_file)
        # 读入每只股票的数据，调整index为trade_date
        content = pd.read_csv(category_file_path, index_col="trade_date")
        # 读入成交量列并且累和获得总交易量
        volume = np.sum(content["vol"])
        # 存入category_dict中
        category_path_dict[category_file.replace(".txt", "")] = category_file_path
        name_list.append(category_file.replace(".txt", ""))
        vol_list.append(volume)

    # 制作成pandas数据集
    category_dict = {"股票编号": name_list, "总交易量": vol_list}
    # 对数据进行从高到低的排序
    category_frame = pd.DataFrame(category_dict).sort_values(by="总交易量", ascending=False)
    # 获取交易量前3的股票
    max_names = list(category_frame.iloc[0:3, 0])
    return max_names, category_path_dict
```

运行主函数：

```
def Lab5(root_path):
    # 读入根目录下的所有工业种类
    categories = os.listdir(root_path)
    category_name = []
    max_category_name = []
    max_category_dict = []
    print("每个工业类别中交易量最大的股票统计如下")

    # 遍历所有的种类
    for category_index, category in enumerate(categories):
        # 设置category的路径
        category_path = os.path.join(root_path, category)
        # 读入category路径下的所有文件
        category_files = os.listdir(category_path)
        # 对每种category执行数据处理
        temp_names, temp_dict = processCategory(category_path, category_files)
        category_name.append(category)
        max_category_name.append(temp_names)
        max_category_dict.append(temp_dict)
```

在同一个 figure 中绘制四个子图，每个子图中绘制一个领域中月成交量最大的三支股票：

```
    fig_content_names = ["开盘价格", "收盘价格", "当日最高价格", "当日最低价格"]
    fig_title_names = ["三月成交量最大三支股票开盘价格折线图", "三月成交量最大三支股票收盘价绘图", "三月成交量最大三支股票日最高价折线图", "三月成交量最大三支股票每日最低价折线图"]
    fig_content = ["open", "close", "high", "low"]
    for i in range(4):
        fig = plt.figure()
        fig.suptitle(fig_title_names[i])
        for j in range(4):
            cur_category_name = category_name[j]
            cur_max_names = max_category_name[j]
            cur_max_dict = max_category_dict[j]
            ax = fig.add_subplot(2, 2, j+1)
            # 分别绘制股票的k线
            for max_name in cur_max_names:
                content = pd.read_csv(cur_max_dict[max_name], index_col="trade_date")
                date = pd.to_datetime(content.index, format='%Y%m%d')
                ax.plot(date, content[fig_content[i]])
            ax.legend(cur_max_names)
            ax.set_title('{}类三月月交易量前三'.format(cur_category_name))
            ax.set_xlabel('日期')
            ax.set_ylabel(fig_content_names[i])
            ax.xaxis.set_major_formatter(mdates.DateFormatter('%m-%d'))
            plt.setp(plt.gca().get_xticklabels(), rotation=45, horizontalalignment='right')
        plt.show()

root_path = "Project_5_data"
real_token = "cc11d7cd33fd59f518b2ea86b84dd22a43fa1c276911e7c2c136a2b8"
downloadData(root_path, real_token)
Lab5(root_path=root_path)
```

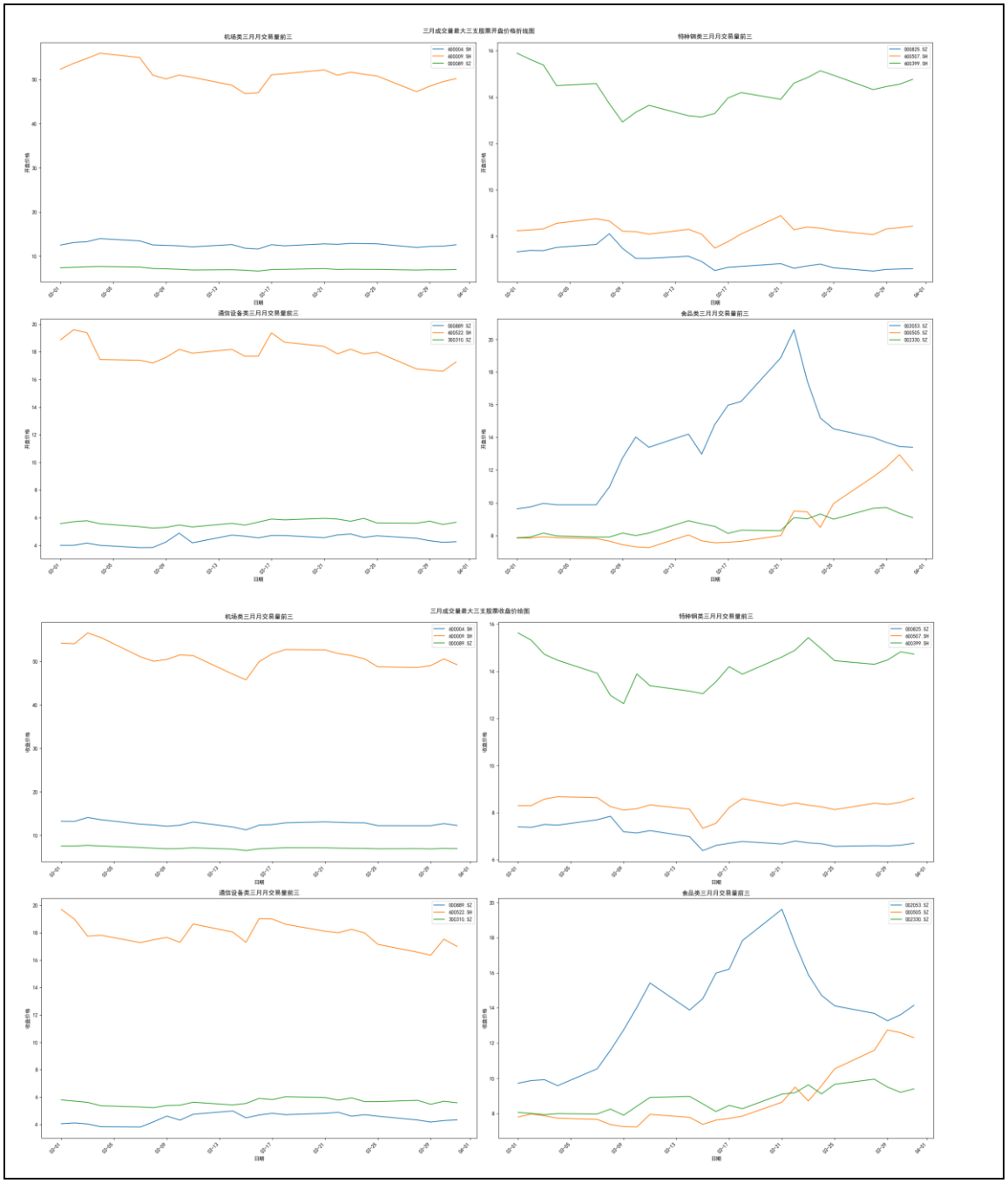
2. 实验结果

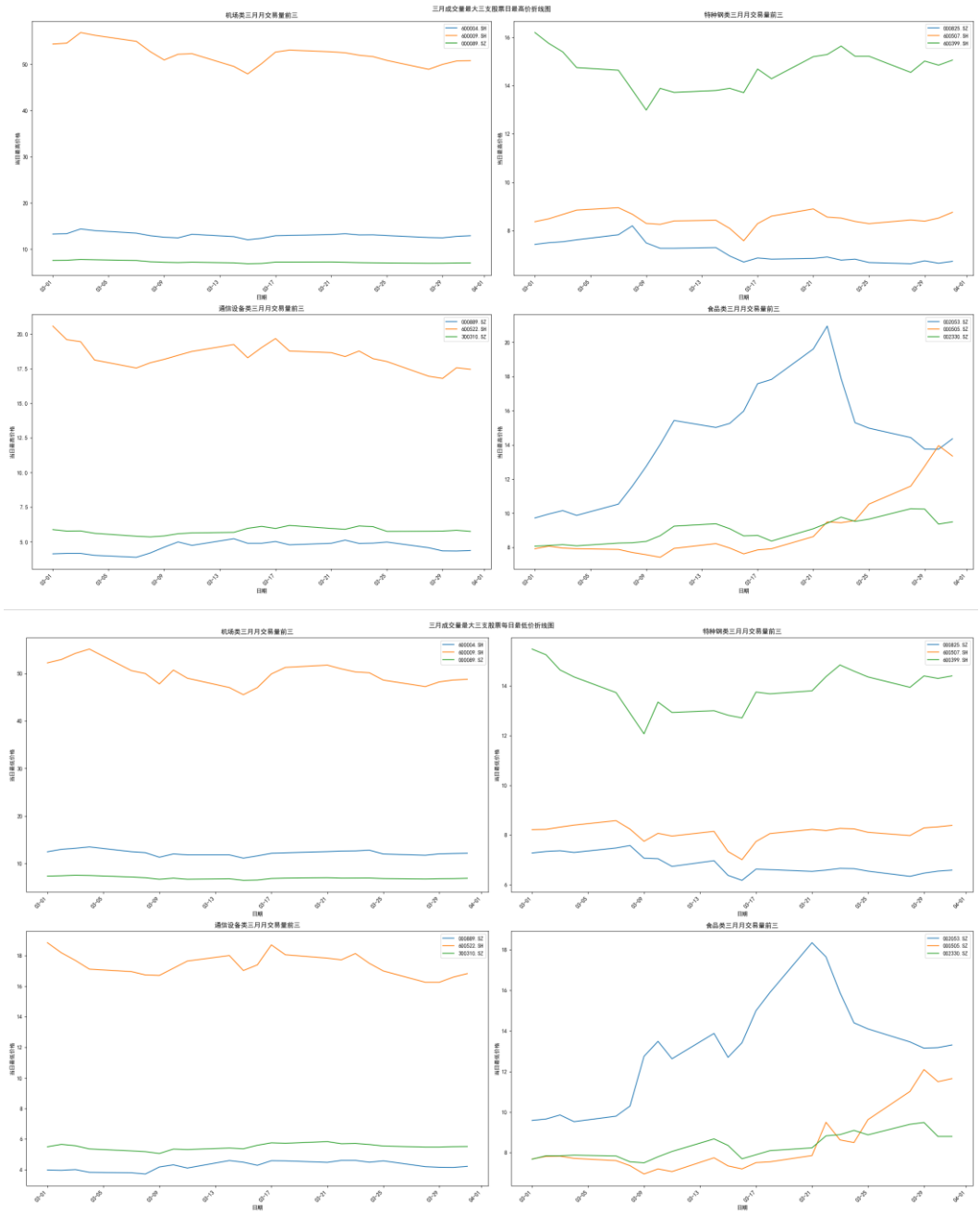
```
"E:\Program Files\Anaconda3\python.exe" "E:\Program Files\PyCharm 2021.2.1\plugins\python\helpers\pydev\pydevconsole.py" --mode

import sys; print('Python %s on %s' % (sys.version, sys.platform))
sys.path.extend(['D:\\OneDrive - University of Cincinnati\\2022 Spring\\大数据\\pythonProject', 'D:/OneDrive - University of Cin

Python 3.8.8 (default, Apr 13 2021, 15:08:03) [MSC v.1916 64 bit (AMD64)]
Backend Qt5Agg is interactive backend. Turning interactive mode on.
使用已下载的数据
每个工业类别中交易量最大的股票统计如下

In[3]:
```


三月成交量最大三支股票开盘价格折线图
机场类三月月交易量前三
特种钢类三月月交易量前三
通信设备类三月月交易量前三
食品类三月月交易量前三
三月成交量最大三支股票收盘价格图
机场类三月月交易量前三
特种钢类三月月交易量前三
通信设备类三月月交易量前三
食品类三月月交易量前三



四、实验总结与体会

本次实验主要是对两个方面的知识进行了强化以及一个方面的知识的学习。

1. 学习了Matplotlib第三方绘图库，熟悉了Matplotlib的绘图方式，可以按照题目需求完成指定数据的绘制。并完成对x，y坐标和标题的修改以及对图例的增加；

2. 提高了Tushare平台使用的熟练度，可以熟练使用Tushare平台下载并操作各类数据；

3. 强化了Pandas第三方库的使用，对数据进行排序，切片以及特殊数据的筛选。

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**