**21-22(2)程序设计2大作业要求**

2022-05-05

**1.简述**

对股票数据进行外排序、索引、指标计算、预测分析、聚类分析(可选)、可视化等处理。

**2.数据集准备**

由老师从tushare网站下载A股部分股票的2021年12月30日之前的Daily数据。

当编译器为32位时，数据集的大小应该小于2GB，保证32位程序能够正确读取数据文件以及正确处理文件偏移量，以便进行后续处理；如果采用64位编译器，则没有这个限制。

数据文件的名称为input.txt，文件的内容如下：

|  |
| --- |
| ts\_code,trade\_date,open,high,low,close,pre\_close,change,pct\_chg,vol,amount  600519.SH,20211230,2041.0,2088.98,2028.0,2075.0,2041.0,34.0,1.6659,35261.36,7272866.657  600519.SH,20211229,2150.0,2154.25,2041.0,2041.0,2138.18,-97.18,-4.545,54049.02,11195625.154  …… |

第一行为表头，内容为“ts\_code, trade\_date, open, high, low, close, pre\_close, change, pct\_chg, vol, amount”，分别表示股票代码、交易日期、开盘价、最高价、最低价、收盘价、临近收盘价、涨跌幅、涨跌百分比、交易量、交易金额等。

其余各行为实际数据，每一行的各个数据项(字段)用“,”隔开。日期的格式为“YYYYMMDD”，比如20181228表示2018年12月28日。节假日或者周末等非交易日没有价格数据。

显示某只股票的K线图时，一般只用到股票代码、交易日期、开盘价、最高价、最低价、收盘价、以及交易量。

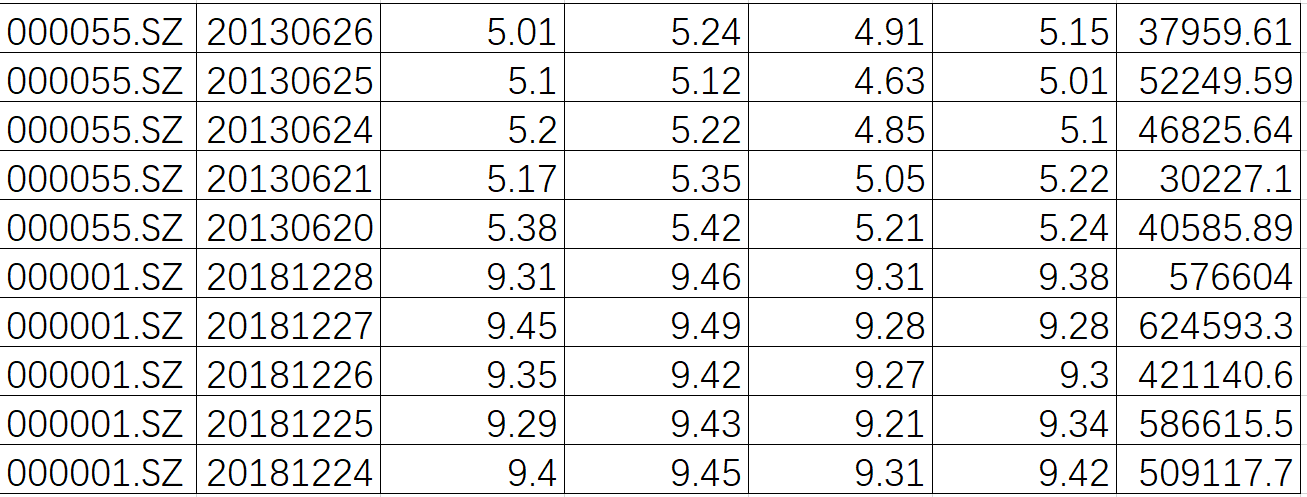
**3.任务描述**

**3.1对股票数据进行排序(必须使用外排序，可以用于排序的内存大小为32MB)：**

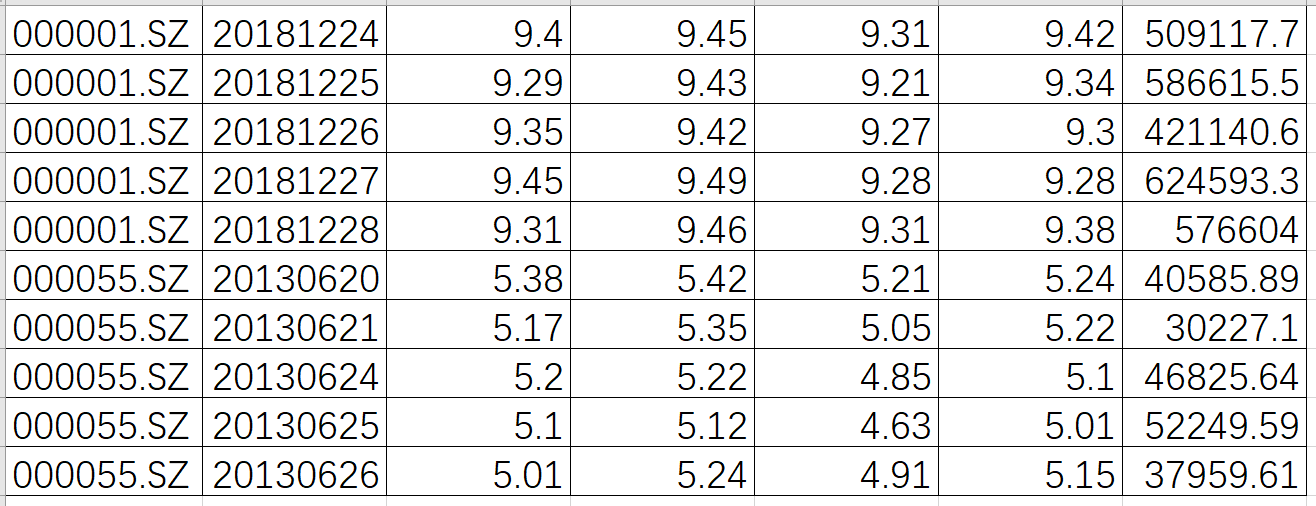
对原文件input.txt进行外排序，排序标准是首先按照股票代码的字典序进行排序，如果股票代码相同则按照日期从小到大进行排序，输出文件取名output.txt。

注意，为了排除干扰，下图的每行数据只给出了股票代码、交易日期、开盘价、最高价、最低价、收盘价、交易量等；实际数据以“2.数据集准备”一节的描述为准。

比如源文件input.txt如下(注意在文件里字段间用逗号隔开，这里是示意图)。



那么经过排序后，结果output.txt如下(注意在文件里字段间用逗号隔开)。



即全部价格数据已经按照股票代码的字典序、以及交易日期从小到大排序了。

**3.2 加快数据访问（一批小文件、或者索引文件）：**

想办法加快数据访问，具体有两个方案可供选择。

方案1——针对每只股票、每个月，创建一个文件。以便通过股票名称和月份就可以访问到相应数据，比如“output\_000001.SZ201812.txt”，表示代码为“000001.SZ”的股票的“201812”月份的数据(这时候，任务1生成一批小文件output\_....txt)。

方案2——建立索引：针对output.txt文件，针对每只股票、每个月份的价格数据在output.txt中的最开始一条数据在文件里的偏移量，建立索引项，生成文件index.txt(这时候，任务1只生成一个大文件output.txt)。以二进制文件方式打开output.txt，即可正确提取和设置文件的偏移量。

比如，对于上述output.txt(实际数据比这里展示的更多，但是不影响原理解释)，由于其数据里包含000001.SZ和000055.SZ两只股票各1个月的数据，于是index.txt文件的内容如下。

|  |
| --- |
| 000001.SZ,201812,{在output.txt文件里该股票该月第一条数据开始的文件偏移量}  000055.SZ,201306,{在output.txt文件里该股票该月第一条数据开始的文件偏移量} |

由于output.txt是经过排序的，index.txt也是有序的。

**3.3对股票价格进行可视化：**

开发QT程序，由用户指定股票代码和月份，采用3.2所述技术方案的帮助，快速定位某只股票、某个月的数据，在QT界面显示该月的价格数据的K线图。

比如借助index.txt的帮助，定位某只股票、某个月第一条数据在output.txt文件中的文件偏移量，从output.txt文件依次提取该股票该月份的数据，一行、一行地读，读完一个月的数据即可，然后可视化出来。参考QTCustomPlot程序，编写可视化代码。

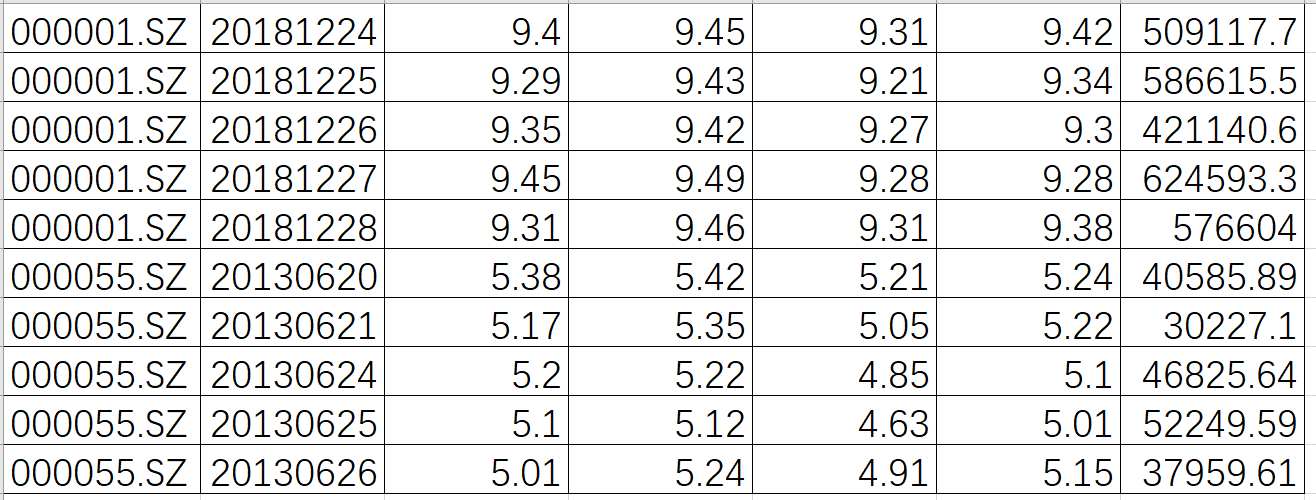
备注：index.txt的内容可以完全载入内存，在内存建立股票代码、月份、文件偏移量的对照表，以便快速查找。

**3.4 指标计算：**

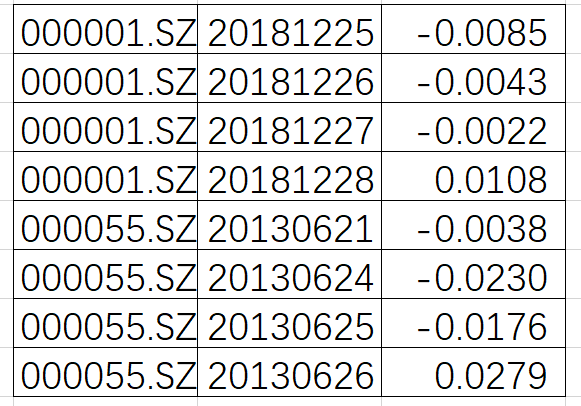
针对output.txt文件中的每只股票的相邻天的股票价格(close)，把某天价格减去前一天的价格得到差分值，除以前一天价格得到一个百分比(即(close price today – close price yesterday)/ close price yesterday)，生成diff.txt文件。这个差分值，表达的是隔天收益率。

注意，为了排除干扰，下图的每行数据只给出了股票代码、交易日期、开盘价、最高价、最低价、收盘价、交易量等；实际数据以“2.数据集准备”一节的描述为准。

比如对于下述output.txt文件(注意在文件里字段间用逗号隔开)。



经过计算生成的diff.txt的文件内容为(注意在文件里字段间用逗号隔开)。

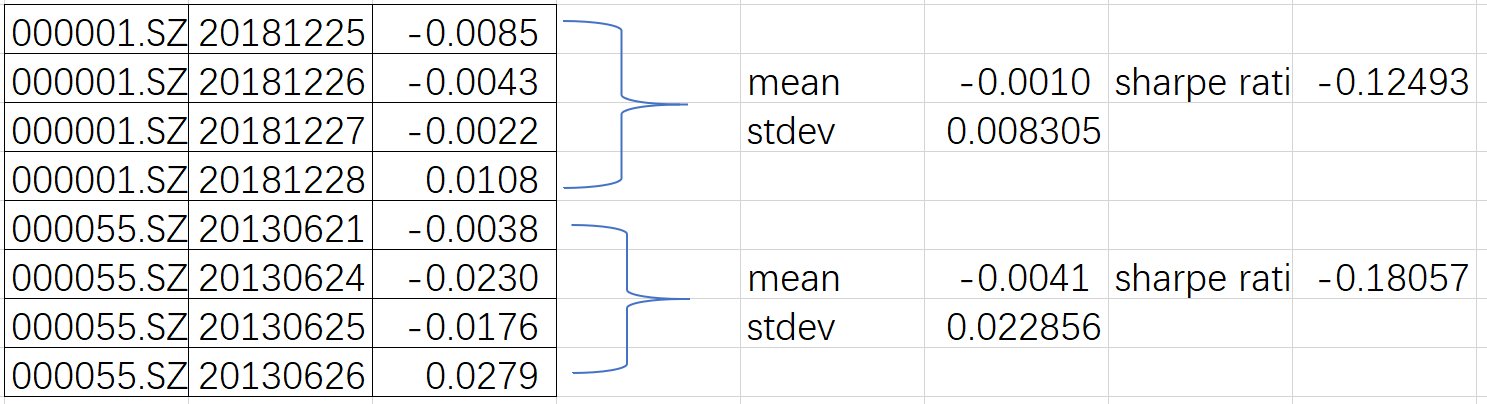


注意，每只股票的某个月的第一条数据没有前一天，那么该条数据无法计算差分。比如，某只股票某个月的价格数据有20条，那么差分数据有19条；此外，不同股票之间没有差分一说。比如，上述output.txt文件里，000001.SZ股票有5条数据，则diff.txt文件里000001.SZ股票有4条差分数据。

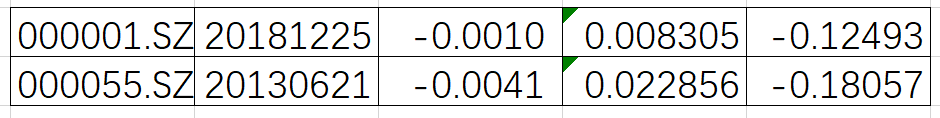
基于diff.txt文件，针对每只股票的每个月的数据，计算该月份收益率的均值、以及收益率的标准差。每只股票的每个月的收益率均值除以其标准差，得到夏普指数。夏普指数衡量了某只股票某个月里的单位风险的收益。均值、标准差按照如下公式进行计算：

|  |
| --- |
| 样本数据  均值  标准差 |

然后每只股票的每个月的收益率均值、标准差、夏普指数写入另一个文件sharpe.txt。收益率均值、标准差、以及夏普指数的计算过程如下图所示：



sharpe.txt文件的内容如下(注意在文件里字段间用逗号隔开)，各个字段分别为股票代码、某个月份、收益率均值、收益率标准差、夏普指数（20181225表示2018年12月份，25日可以不用保存）：



**3.5 查询某个月Sharpe指数排名前30的股票：**

开发QT程序，用户指定某个月份，提取该月份Sharpe指数排名前30的股票，按照夏普指数从大到小排序，以柱状图显示。

备注：Sharpe.txt文件较小，直接载入内存进行后续处理即可。

提示：可以采用横向的柱状图，横坐标是夏普指数，纵坐标是各个股票，按照夏普指数从大到小的顺序从上到下排序。

**3.6 预测价格变动：**

指定某只股票某个月份，请自行选用某种预测方法，预测该股票下一个月的价格。把预测价格和实际价格作对比，在QT界面上显示实际的价格和预测的价格，并且计算RMSE均方误差指标。

比如，指定股票600519.SH，指定2021年01月，请预测该股票2021年02月份的价格为，与600519.SH的实际的2021年02月份的价格y作对比，计算均方误差指标，其中RMSE=

对于采用的预测办法不做限制，可以是随机预测、根据前几天的涨跌天数和涨跌幅进行判断、进行线性回归预测等。

提示：预测价格变动有两种做法，分别是针对价格进行处理，以及针对隔天收益率进行处理，分别描述如下：

(1)根据前几天的价格数据，预测接下来1天的价格：基于output.txt文件，用户指定某个股票某个月份，提取该月份的数据。

预测下一个月份第一天的价格；然后滚动预测，即把下一个月份第一天的真实价格加入历史数据，预测下一个月份第二天的价格，……。

|  |  |
| --- | --- |
|  | 1.用户指定2021.11月份 |
| 2.提取2021.11月份数据(从output.txt) | 3.预测2021.12月份第1天的价格 |
| 4.把2021.12第1天真实价格(从output.txt)加入历史数据，旧的历史数据可以淘汰 | 5.预测2021.12月份第2天的价格 |
| 6.把2021.12第2天真实价格(从output.txt)加入历史数据，旧的历史数据可以淘汰 | 7.预测2021.12月份第3天的价格 |
| 8.把2021.12第3天真实价格(从output.txt)加入历史数据，旧的历史数据可以淘汰 | … |
| … | n-1.预测2021.12月份最后1天的价格 |
| n.把2021.12最后1天真实价格(从output.txt)加入历史数据，旧的历史数据可以淘汰 |  |
| 2021.12月份真实数据、2021.12月份预测数据，进行可视化和对比 | |

(2)根据前几天的收益率，预测接下来1天的收益率：基于diff.txt文件，用户指定某个股票某个月份，请提取该股票该月份的diff.txt收益率数据。

预测下一个月份第一天的收益率；然后滚动预测，即把下一个月份第一天的真实的收益率加入历史数据，预测下一个月份第二天的收益率，……。

注意为了显示价格，需要把收益率重新转换为价格。参考第一天的价格迭代进行计算，比如第一天的价格为100，第二天的收益率为0.05，第三天的收益率为0.05，那么第二天的价格为105，第三天的价格为110.25)。

**备注：大作业完成任务1到任务6即可。**

**3.7 动画显示历史上各个月份的夏普指数top 30(可选)：**

开发QT程序，提取每个月的top 30夏普指数的股票，timer设置为1秒，每秒显示一个月的股票数据，从最开始的月份到最末尾的月份循环播放。

备注：Sharpe.txt文件较小，直接载入内存进行后续处理即可。

提示：可以采用横向的柱状图，横坐标是夏普指数，纵坐标是各个股票，按照夏普指数从大到小的顺序从上到下排序。

**3.8 K-means聚类(可选)：**

针对某个月份，提取该月份的所有股票的收益率的均值、标准差、以及夏普指数，以均值、标准差作为指标，运行K-means聚类算法(可设定K=9)，计算每只股票的类簇标签，类簇标签为0-8。

开发QT程序，显示散点图，并且根据每个数据点的(每只股票该月的收益率的均值、标准差为一个数据点)所属类簇的不同，渲染不同的颜色，比如red, blue, green, brown, pink, yellow, gray, black, cyan等。

K-means算法的执行过程如下：

(1)在二维平面上(收益率均值、标准差各为一个坐标轴)，在有效数据范围里面生成K=9个初始点P0,P1,P2,…,P8，作为9个类簇的初始中心点；

(2)根据各个数据点(即每只股票该月的均值、标准差)和K个类簇的中心点的欧式距离，选择最近的类簇中心，作为这个数据点的类簇。比如某只股票该月的均值、标准差和P6的距离最近，那么则指定这个数据点的类簇为6；

(3)根据分配给每个类簇的数据点，重新计算每个类簇的中心点；比如把属于类簇6的所有数据点集中起来，对收益率均值、标准差两个字段分别求平均，就是该类簇的新的中心点即P6。

(4)交替执行(2)和(3)，可以设定迭代次数为10、20次；迭代一定次数以后，算法收敛(这里不展开K-means算法的损失函数介绍)。

**4.提交办法**

(1)试验报告(word格式)

每个任务的实现策略描述，包括主要的数据结构、算法流程等；

每个任务的输入界面、运行结果界面以及说明。

(2)源代码(拷贝整个项目文件夹)

把(1)和(2)打包压缩，命名为“学号姓名.zip”提交。