《时空数据处理与组织课程实习》

实习报告

学　　院:　　遥感信息工程学院

班　　级:　　2001

学　　号:　　2020302131201

姓　　名:　　常耀文

实习地点:　　101 机房

指导教师:　　李晓雷

2022 年　　5 月　　30 日

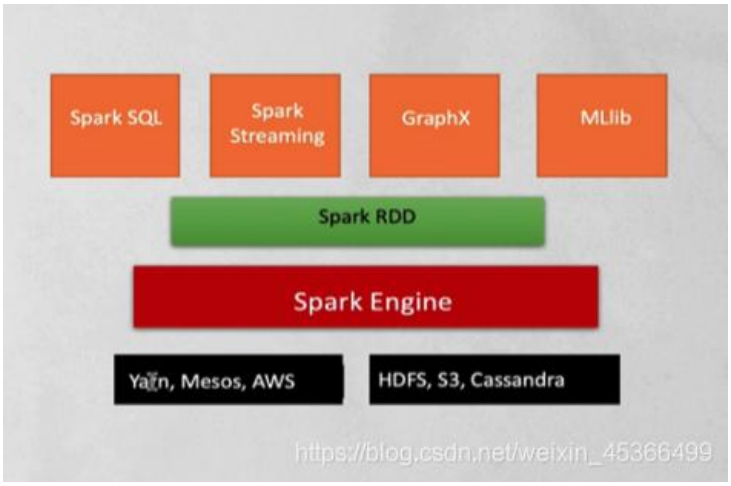
目录

㈠ 实验设计要求与数据

1．实验目的

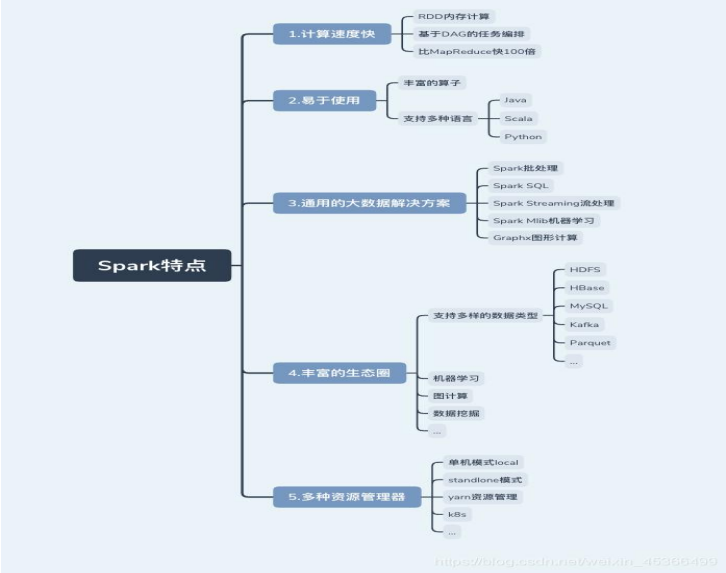
本次课程设计的任务就是利用大数据工具 Spark，综合采用课程中介绍的方法进行车辆轨迹的初步分析，包括车辆停留点分析、加减速监测等，为以后进一步的海量轨迹数据的深入挖掘和应用打下基础，学习 spark，必须要了解 spark 的基本概念与 spark 的特点：

①spark，是一种"One Stack to rule them all"的大数据计算框架，期望使用一个技术堆栈就完美地解决大数据领域的各种计算任务。Apache 官方，对 Spark 的定义就是：通用的大数据快速处理引擎。Spark 使用 Spark RDD、Spark SQL、 Spark Streaming，MLlib，GraphX 成功解决了大数据领城中，离线批处理、交互式查询、实时流计算、机器学习与图计算等最重要的任务和问题。Spark 除了一站式的特点之外，另外一个最重要的特点，就是基于内存进行计算，从而让它的速度可以达到 MapReduce、Hive 的数倍甚至数十倍！现在已经有很多大公司正在生产环境下深度地使用 Spark 作为大数据的计算框架，包括 eBay.Yahool、 BAT、网易、京东、华为、大众点评、优酷土豆、搜狗等等。



（spark 的整体架构）

②Spark 同时也获得了多个世界顶级 IT 厂商的支持，包括 IBM、 Intel 等。Spark，是一种通用的大数据计算框架，I 正如传统大数据技术 Hadoop 的 MapReduce、Hive 引擎，以及 Storm 流式实时计算引擎等，Spark 包含了大数据领城常见的各种计算框架：比如 Spark Core 用于离线计算，Spark SQL 用于交互式查询，Spark Streaming 用于实时流式计算，Spark MILlib 用于机器学习，Spark GraphX 用于图计算。Spark 主要用于大数据的计算，而 Hadoop 以后主要用于大数据的存储（比如 HDFS、Hive，HBase 等），以及资源调度（Yarn）。Spark+Hadoop 的组合，是未来大数据领域最热门的组合，也是最有前景的组合！因此学习 spark 编程对于未来处理海量数据具有极大的好处，为我们处理海量空间数据具有极其重要的意义。



（spark 的特点）

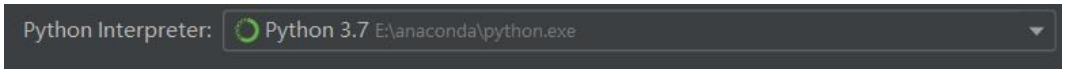
③本次实验有助于我们整理综合之前所学的 spark 知识，将每次 spark 编程的内容应用到我们的实际练习之中，帮助我们掌握与学习知识。

2．实验环境

⑴本次编程实验环境是在 windows 操作系统下，基于 python3.7 实现的。

（windows 版本）

⑵编程语言是基于 anaconda 库下的 Python 语言，其中 python 版本为 3.7



（python 语言版本）

⑶实验所采用编程语言，IDE 以及所用的包

本次使用的编程语言是 python，所使用的编程工具是 PyCharm，下图是在本次实验中引入的所有包，包含 findspark, math, pandas, pyspark, pyspark.sql, pandas， 代码文件只有一份 car_speed_analyse.py

```
import findspark
import math
```

```
from pyspark import SparkConf
from pyspark.sql import SparkSession, Window
import pandas as pd
```

① 其中 findspark 是用来初始化 spark 环境

② PySpark 是 Spark 为 Python 开发者提供的 API。我使用了其中的 pyspark.SparkConf，两者的具体用途是：

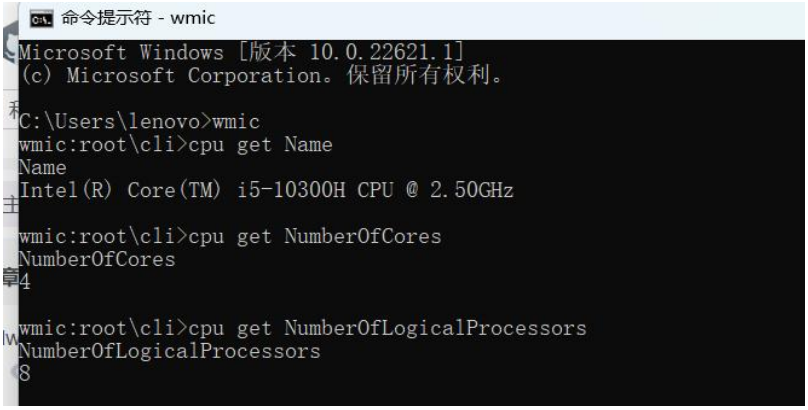
pyspark.SparkConf 类提供了对一个 Spark 应用程序配置的操作方法。用于将各种 Spark 参数设置为键值对。

③ pyspark.sql,数据科学家处理的大多数数据在本质上要么是结构化的，要么是半结构化的。为了处理结构化和半结构化数据集，PySpark SQL 模块是该 PySpark 核心之上的更高级别抽象,。PySpark SQL 支持从许多文件格式系统读取，包括文本文件、CSV、ORC、Parquet、JSON 等。

④pandas 包，提高了高性能易用数据类型和分析工具，用于读入初始创建的 dataframe。

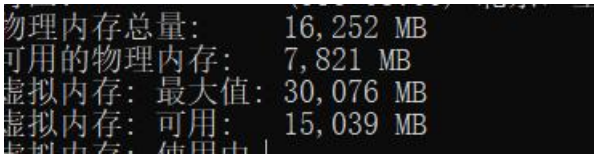
⑷**本机的软硬件环境**

1.CPU 信息：本机的物理 CPU 为 1 个，CPU 线程数为 8，CPU 核心数为 4



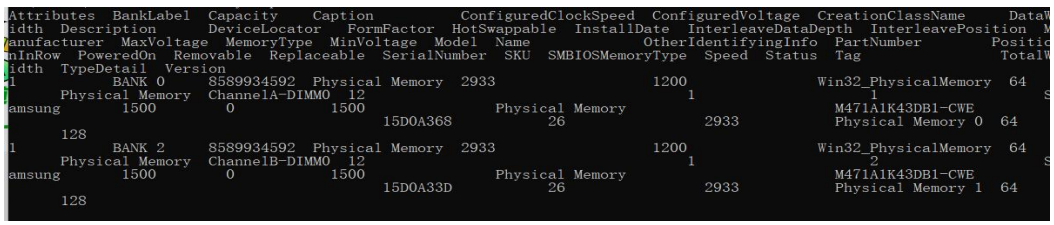
（本机 cpu 信息）

2.内存信息：最大内存支持 16G



（本机内存信息）

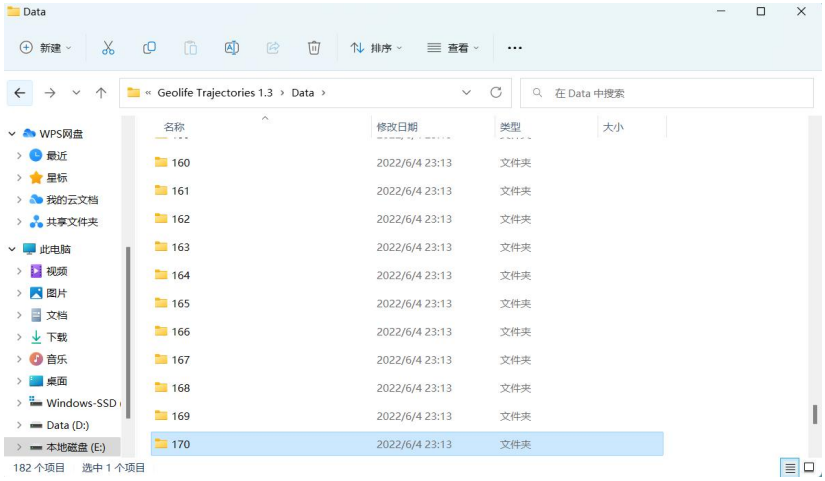
3.内存相关信息



（内存相关信息）

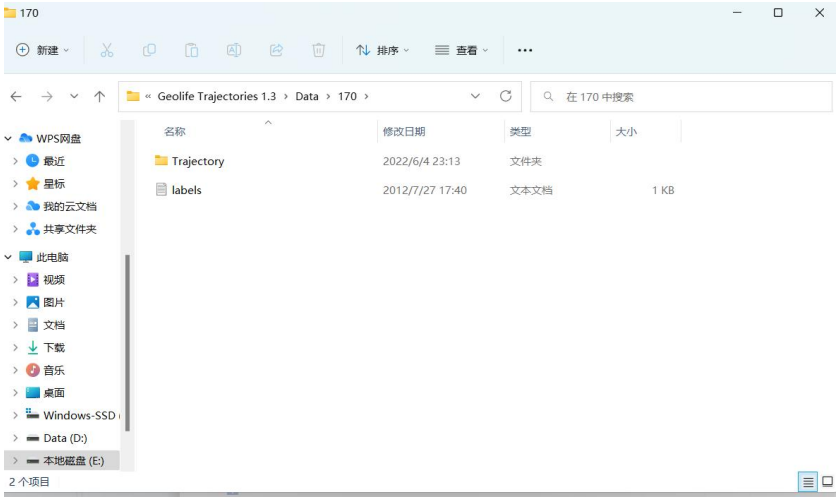
3．课程实验数据

数据采用微软的车辆轨迹数据-Geolife Trajectories 中的编号为 170 号的车辆轨迹数据。所给数据是-Geolife Trajectories-1.3 的数据，需要对于数据进行筛选与选择。



（实验数据）

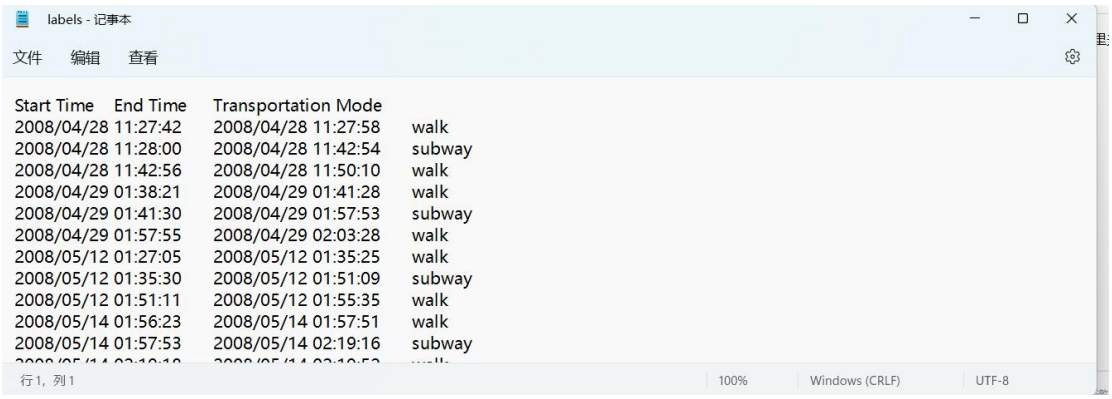
170 号数据的细节：



（实验 170 号数据文件夹）

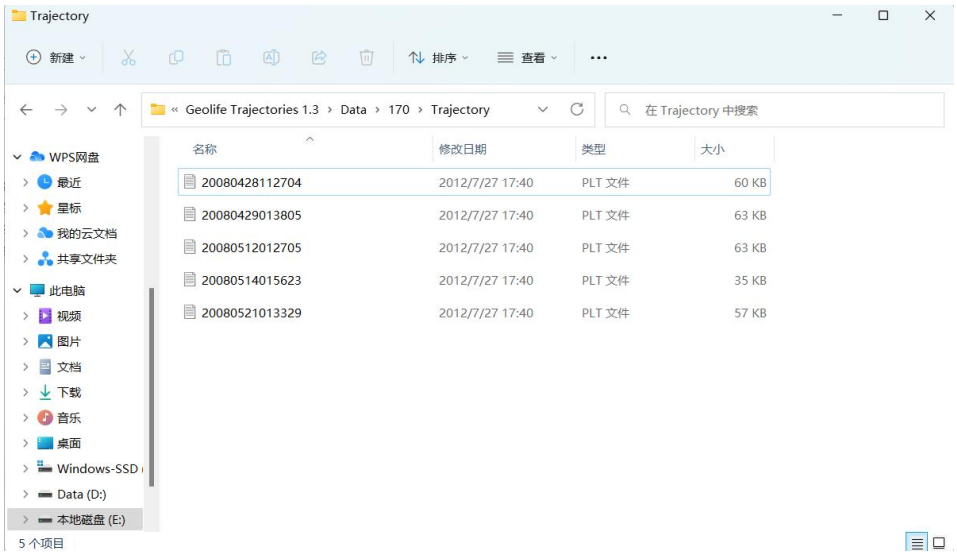
其中所用的数据都在 170 文件夹中：

1. **Lables 文件表示的运输模式与起止时间，与实验设计关系并不密切**



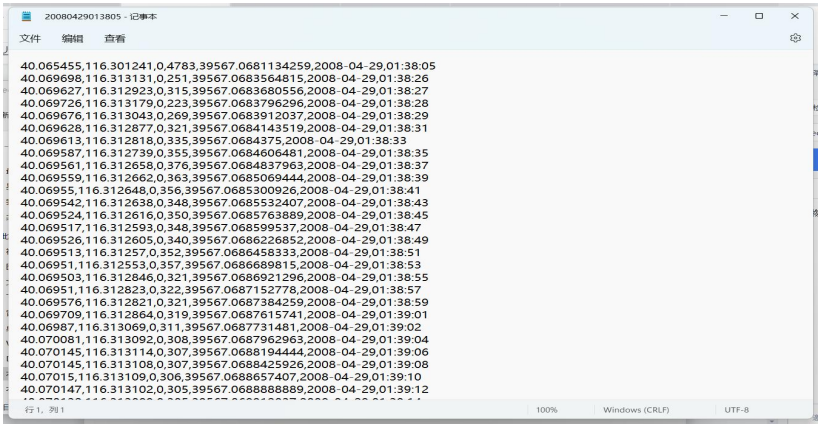
（labels 文件）

2. Trajectory 文件夹中有着 5 个 plt 文件，代表了本次实验设计所用的五个数据。



（Trajectory 文件夹）

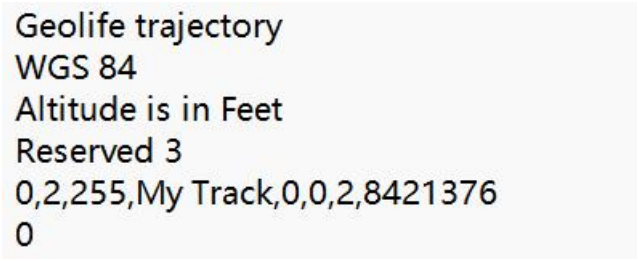
每一个数据都代表了一个 2008 年的一段时间中的车辆运动状态，其中的数据组织格式为：组织分别为纬度，经度，海拔高度，1889 年的时间，2008 年的日期，2008 年的时间，主要使用 pandas 的方法读入相关的内容，按照相应的字段组织方式形成 pandas 的 dataframe：



（车辆运动状态的信息数据）

4．实验数据处理

实验所给的数据带有六行文件首部：



（车辆数据文件前六行）

在实际的应用中需要对这个数据进行处理，可供选择的方法有两种，一种是在处理前提前将文件的前六行数据删除，一种是在使用 pandas 的 read_csv 函数方法，将文件读取到相应的 dataframe 中，其中 skiprows 为 6，跳过前六行，代码如下：

```
df = pd.read_csv(path, header=None, skiprows=6, usecols=[0, 1, 3, 4, 5, 6],
names=['lat', 'lng', 'altitude', 'Day1889', 'Day2008', 'time'])
```

至此，实验数据处理完成，生成一个 pandas 库下的 dataframe，以便于后续的 spark dataframe 数据生成处理。

5．对于实验设计数据与实验设计使用的思考

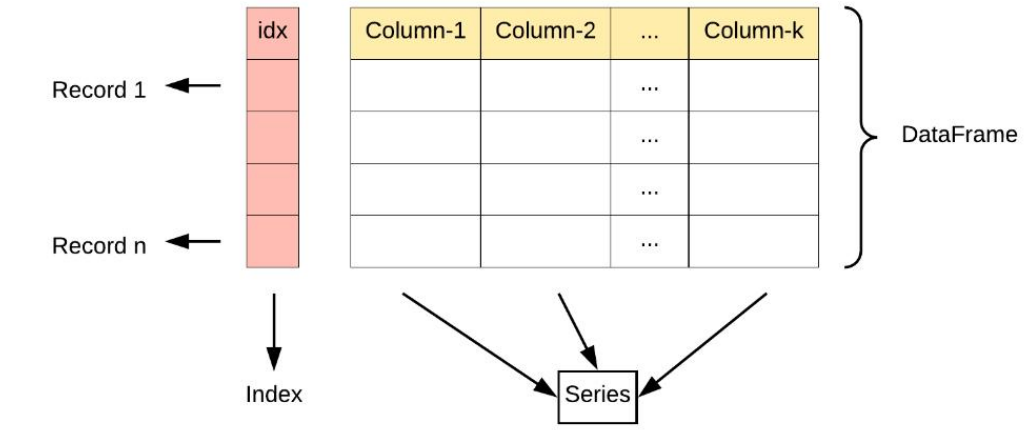
⑴对于实验内容分析数据用处

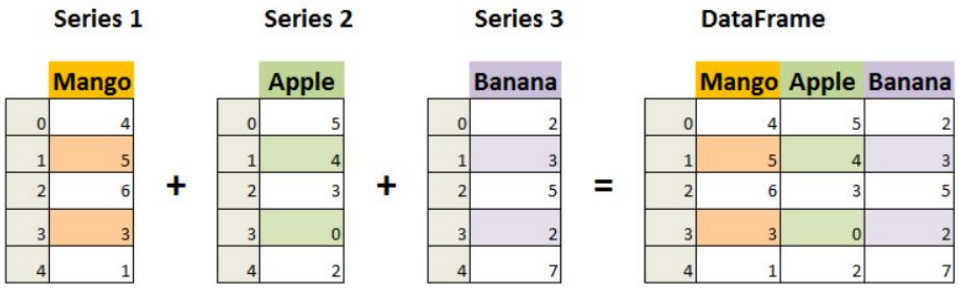
轨迹数据是时空环境下,通过对一个或多个移动对象运动过程的采样所获得的数据信息,包括采样点位置、采样时间、速度等,这些采样点数据信息根据采样先后顺序构成了轨迹数据。近年来,随着各种定位技术的发展和普及,以及无线互联网的高速发展推动下的移动智能终端的更新换代,民用 GPS 设备在移动终端上得到了非常广泛的使用,海量的轨迹数据在日常生活中正在日益积累,如何有效挖掘轨迹数据背后的信息成为一个备受关注的热点。因此我们所具有的数据是来自于数据采用微软的车辆轨迹数据-Geolife Trajectories 中的编号为 170 号的车辆轨迹数据。

学会了如何处理 170 号车的数据，由于所有的车辆具有相同的数据组织形式，就可以通过改变文件名来读入不同的车辆轨迹数据，然后使用相同的方法对于其他的车辆进行处理，并且得到相关的数据结果，可见，代码所运行的操作具有普适性。

⑵对于实验的 dataframe 数据组织格式思考

DataFrame 是一个表格型的数据结构，它含有一组有序的列，每列可以是不同的值类型（数值、字符串、布尔型值）。DataFrame 既有行索引也有列索引，它可以被看做由 Series 组成的字典（共同用一个索引）





（Dataframe 组织结构）

根据个人的理解，dataframe 与数据库中的表类似，又类似于二维数组，因此 dataframe 的数据结构既可以使用类似于 sql 的语句查询，又可以通过类似于数据寻址的方式查询，比如在 spark.sql 中可以使用 sql 语句查询 dataframe,在 pandas 的 dataframe 中可以使用 iloc 与 loc 来寻址 dataframe 中特定字段的数据值。

、

6．实验功能分析与整体思路设计

⑴实验功能分析

车辆轨迹分析应实现如下三种功能：

1. 车辆速率计算

从原始轨迹数据中获取每个轨迹点的经纬度坐标,通过该轨迹点与前一个点的距离和两

个轨迹点的时间差，计算该点的即时速率。根据实际意义，规定起点和终点的即时速率为零。

2. 车辆停留点分析

   停留点识别的算法描述如下：

   1） 遍历轨迹点，检查速率值，若小于阈值，则将上一个点作为停留开始点，记录其序号。

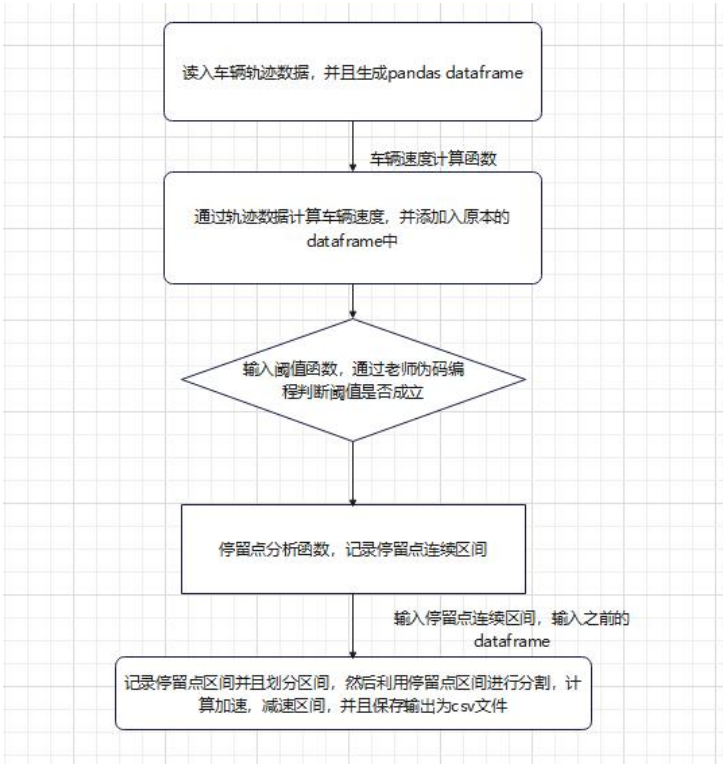
   2） 继续遍历轨迹点，只要轨迹点的速率值仍小于阈值，则将该点作为停留点。

3. 车辆加减速分析

   加速和减速都是持续性的过程，而突发的速率变化则可能是数据采集或预处理阶段的误差引起的正常波动，因此加/减速检测方法可以采用寻找至少连续 N（$N \geq 2$）个点速率单调变化的片段。

（2）实验思路设计

根据实验的功能分析一步一步设计实验所用的函数与相关内容，先构建一张架构设计图：



（实验设计的整体思路）

通过实验架构设计，成功地分析出了整体的需求与所要求得功能，为接下来的代码书写构建框架，节省纠错时间与代码书写时的思考。

7．实验具体实现步骤

⑴ 功能一：车辆速率计算，从原始轨迹数据中获取每个轨迹点的经纬度坐标，通过该轨迹点与前一个点的距离和两个轨迹点的时间差，计算该点的即时速率。根据实际意义，规定起点和终点的即时速率为零。

实现：

① 编写车辆速率计算函数，代码的来源是老师在群中所给的代码实现，想法是通过将经纬度转换为对应的地理距离，利用地理距离与前后两条数据的时间差来计算数据，这里需要特别注意的是使用的单位，具体速度计算函数代码如下：

```
def carspeed(lng1, lat1, lng2, lat2, timeback, timenow):
    #角度转弧度
    lng1 = lng1 * pi / 180
    lat1 = lat1 * pi / 180
    lng2 = lng2 * pi / 180
    lat2 = lat2 * pi / 180
    dlon = lng2 - lng1
    dlat = lat2 - lat1
    a = math.cos(lat1) * math.cos(lat2) * math.sin(dlon / 2) ** 2+math.sin(dlat /
2) ** 2
    distance = 2 * math.asin(math.sqrt(a)) * 6371 * 1000  # 地球平均半径，6371km
    distance = round(distance / 1000, 5)          # 五位小数，单位千米
    speed = distance / ((timenow - timeback) * 24)
    speed = speed/3.6                                          # 距离除以时
间差（秒）得到每秒速度 km\h,转换为 m/s
    return speed
```

注意到的是我们最后得到的数据是以 m/s 来记录的。

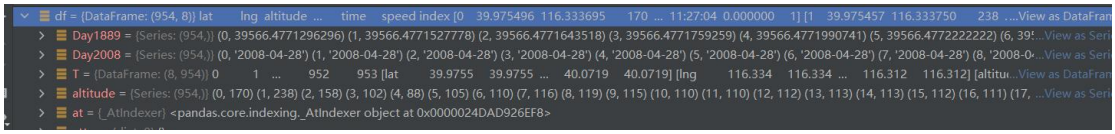
②从文件读取数据，

```
def read_car_information(path):
    df = pd.read_csv(path, header=None, skiprows=6, usecols=[0, 1, 3, 4, 5, 6],
                           names=['lat', 'lng', 'altitude', 'Day1889', 'Day2008',
                                  'time'])
    # 实现要求 1
    # 计算速度，开始和最后都是 0
    car_speed = [0]
    ct = 1
    for i in range(df.shape[0] - 2):
        speed = carspeed(float(df.loc[i]['lng']), float(df.loc[i]['lat']),
float(df.loc[i+1]['lng']), float(df.loc[i+1]['lat']), float(df.loc[i]['Day1889']),
```

```
float(df.loc[i+1]['Day1889']))
        car_speed.append(speed)
        ct += 1
    car_speed.append(0)
    df['speed'] = car_speed
    df['index'] = range(1, df.shape[0] + 1)  # 创建 index 列
    return car_speed, df   # 由于要利用速度来判断停留点，为了便于操作，所以将速度返回
```

先从文件中读取所有的数据，利用计算所有的速度即 car_speed 为 df 创建新的一列即 speed 列，在根据老师的要求为 df 创建一列 index，最后返回 car_speed 与 pandas 创建的 dataframe。利用 20080428112704.plt 文件进行实例，获得到了 dataframe 与 car_speed

（car_speed）



(dataframe)

至此，问题的第一项解决完毕，获取到了每一个车辆的及时速率，注意第一个点和最后一个点速率为 0，在计算速率时要先给数组首尾赋予为 0 的元素，然后将整体作为 dataframe 的一列。

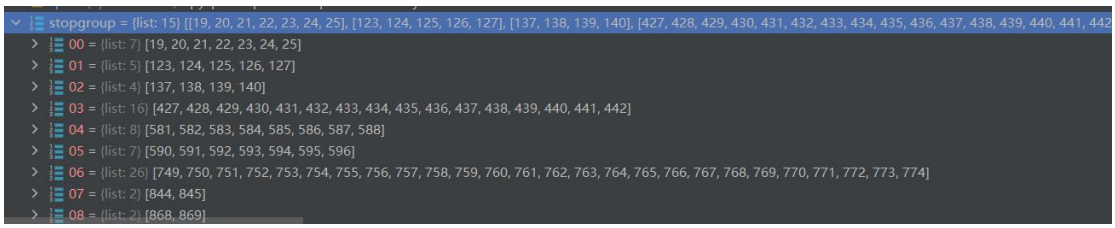
②车辆停留点分析，停留点，经过分析可知停留点是速率小于某一个阈值的点，即根据老师在群中的伪码实现了相关的内容，具体的代码实现如下：

```
def findstop(df, maxspeed, car_speed):
    flag = []  # 标志数组存放停留点下标
    if car_speed[1] < maxspeed:
        flag.append(0)   # 如果第二个点的速率小于阈值，则记录第一个点是停留点
    i = 1
    length = len(car_speed)
    while i < length - 1:  # 根据老师的伪码书写
        # 如果下一个点速度小于阈值，则本点是停留点
        if car_speed[i+1] < maxspeed:
            flag.append(i)
        # 如果上一个点是停留点，且本点速度小于阈值，那么本点也是停留点
        elif i-1 in flag and car_speed[i] < maxspeed:
```

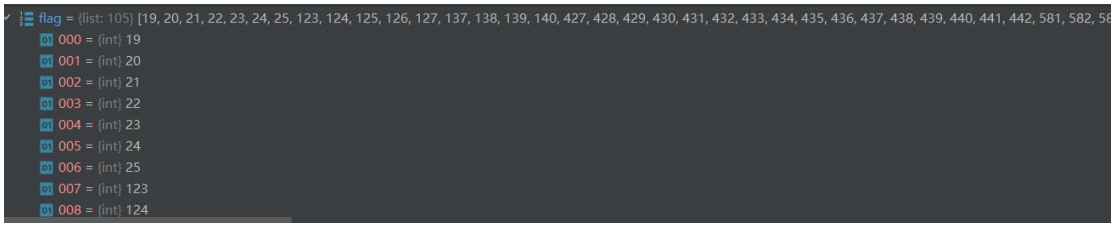
```
            flag.append(i)
        i += 1
    flag.append(length-1)                    # 末尾点速度为0，是停留点，由理解可知
    # 将停留点信息添加到原 dataframe 中
    df['Stop'] = False  # 创建停留点判断的一列，先全部赋值为 false
    for ct in flag:
        df.iloc[ct, 8] = True  # 替换停留点的 bool 的值
    # 由于停止点是有区间的，所以将一组连续的停止点储存在一起
    cp = 0
    stopgroup = []
    temp = []
    while cp < len(flag):
        temp.append(flag[cp])  # 将 flag 中的下标加入到子组中
        if cp != len(flag) - 1:  # 当该点不是末尾点时
            if flag[cp + 1] == flag[cp] + 1:
                cp += 1  # 如果下一个点和本点连续，那么进入下一个循环
            else:  # 如果下一个点和本点不连续，将该子组添加到总分组情况中，将子组清空，进入下一个循环
                stopgroup.append(temp)
                temp = []                    # 清空子组
                cp += 1
        else:                                # 到末尾点时，将该子组添加到总分组情况中，结束循环
            stopgroup.append(temp)
            cp += 1
    return flag, stopgroup, df
```

代码思路如下：主要函数包含三个输入参数 df，maxspeed，car_speed，maxspeed 是我设置的速度阈值，可以根据我自己的要求进行更改，df 与 car_speed 均与之前的操作相关联，先产生一个标志数组，然后判断第一个点是否满足条件，在利用循环，按照停留点识别的算法:：遍历轨迹点，检查速率值，若小于阈值，则将上一个点作为停留开始点，记录其序号。继续遍历轨迹点，只要轨迹点的速率值仍小于阈值，则将该点作为停留点，将结果序号添加到数组中。根据要求计算每一点是否满足停留点的条件，最后将末尾点判断生成 flag（包含全部的停留点排序序号）。然后利用 flag 数组，为 df 创建新的一列，即‘stop’列，先将‘Stop’列的值先赋予 False 值，然后利用 flag 中获取的下标，修改在 Stop 列中的值为 true。最后通过 flag，将所有连续的停留点区间存储到 stopgroup 数组之中。为了后续判断加减速状态做出准备。最后返回 flag，stopgroup， df 三个值。

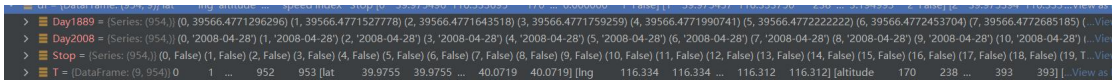
实验获取的中间结果：



（实验获取的停留点区间）



(所存的 flag 数组)



（df 所创造的新列 stop）

③ 车辆加减速分析，加速和减速都是持续性的过程，而突发的速率变化则可能是数据采集或预处理阶段的误差引起的正常波动，因此加/减速检测方法可以采用寻找至少连续 N（N ≥ 2）个点速率单调变化的片段，根据理解，要计算加速度并且通过加速度来判断速度是否单调变化，如果有两个以上加速度均为正值或者负值，那么该两点就划分到加速区间或者减速区间，代码具体如下：

首先在整段数据处理之前，需要将对于每个点计算加速度，加速度计算函数如下：

```
# 计算加速度函数，这加速度算法是看同学们在群中讨论学习到的
def acCalc(v1, v2, t1, t2):
    delta_t = (t2-t1)*24*60*60    # 将单位转换为 s
    deala = (v2-v1)
    ac = deala/delta_t            # 单位：m/s2，除以 3.6 将 km 每小时变为 m/s2
    return ac
```

加速度计算函数需要有 v1，v2 两个速度，以及 t1，t2 两个时间，在计算之前，需要将以天为单位的时间计算为以秒为时间单位的 delta_t,然后利用速度差除以时间差的计算结果。

加减速区间分析函数 findarea

```
def findarea(df, car_speed, stopgroup, flag, spk):
    # 分析车辆运动状态
    # 计算加速度
    ac = []
    acf = []
    length = len(car_speed)
    ac.append({'index': 1, 'acceleration': 0})
    acf.append(0)
    i = 1
    while i < (length-1):
        af = acCalc(car_speed[i], car_speed[i + 1], float(df.loc[i]['Day1889']),
```

```
float(df.loc[i + 1]['Day1889']))
        # 由于分析加减速区间时只需知道加速度的正负，并不要求其具体值，故做如下数据简化
        if af > 0:
            tf = 1  # 加速过程为 1，减速过程为-1，加速度为 0 的情况并不存在，速度为 0，加速度可能不为 0
        else:
            tf = -1
        acf.append(af)
        tempd = {'index': i + 1, 'acceleration': tf}
        ac.append(tempd)
        i += 1
    acf.append(0)
    df['accspeed'] = acf
    ac.append({'index': len(car_speed), 'acceleration': 0})  # 终点由于没有关联点，定义加速度为 0
    newdf = spk.createDataFrame(ac)
    sdf = spk.createDataFrame(df)
    sdf = sdf.join(newdf, on='index')
```

首先对于每一个点计算加速度，并且对于每一个加速度进行正负判断，当加速度大于 0 时，赋予一个标志值为 1，当加速度小于 0 时，赋予一个标志值为-1,利用 index 与判断标志 acceleration 创造一个键值对，用 index 与 accspeed 创造一个键值对分别存储判断标志与加速度真值,最后利用 spk 的 join 函数通过 index 将两个键值对与原本的 dataframe 连接，生成一个新的 dataframe，注意首尾的加速度均为 0，首尾的加速度标志也为 0。

```
|  1|        39.975496|116.33369499999999|     170|  39566.4771296296|2008-04-28|11:27:04|                0.0|false|                0.0|          0|          non|
|  2|        39.975457|        116.33375|     238|  39566.4771527778|2008-04-28|11:27:06|3.1949932276042303|false|   3.855072777410866|          1|          non|
|  3|        39.975394|116.33374099999999|     158|  39566.4771643518|2008-04-28|11:27:07| 7.050042671549434|false| -0.3700560466456537|         -1|          non|
|  4|39.975334000000004|        116.333745|     102|  39566.4771759259|2008-04-28|11:27:08| 6.679985840499612|false| -1.2199948279724058|         -1|          non|
|  5|39.975359000000005|        116.333651|      88|  39566.4771990741|2008-04-28|11:27:10|4.2399910125326885|false|  0.1975094394775353|          1|          non|
|  6|        39.975317|        116.333745|     105|  39566.4772222222|2008-04-28|11:27:12|4.6350091146880255|false| -0.7525062795719415|         -1|          non|
|  7|39.975367999999996|116.33371399999999|     110|  39566.4772453704|2008-04-28|11:27:14|3.1299933653838004|false| -0.8374968989966267|         -1|          non|
|  8|        39.975394|116.33371799999999|     116|  39566.4772685185|2008-04-28|11:27:16| 1.455002861245108|false| 0.3899953739593266|          1|          non|
|  9|        39.975432|        116.333735|     119|  39566.4772916667|2008-04-28|11:27:18|  2.23499526250249|false| -0.7099982261086848|         -1|          non|
| 10|39.975421999999995|        116.333749|     115|  39566.4773148148|2008-04-28|11:27:20|0.8150016026905589|false|  0.2499972750446127|          1|          non|
| 11|        39.975414|116.33377800000001|     110|   39566.477337963|2008-04-28|11:27:22| 1.314997212613322|false| 0.17000335527453425|          1|          non|
| 12|        39.975396|        116.333809|     110|  39566.4773611111|2008-04-28|11:27:24|1.6550032545434048|false|-0.18500259702156807|         -1|          non|
| 13|        39.975382|116.33383300000001|     112|  39566.4773842593|2008-04-28|11:27:26|1.2849727762038924|false|-0.14999796457403045|         -1|          non|
| 14|         39.97537|        116.33385|     113|  39566.4774074074|2008-04-28|11:27:28|0.9850019369941111|false| 0.06249772260157945|          1|          non|
| 15|        39.975369|        116.333876|     113|  39566.4774305556|2008-04-28|11:27:30|1.1099976471488877|false| 0.06000250381199905|          1|          non|
| 16|         39.97538|        116.333901|     112|  39566.4774537037|2008-04-28|11:27:32|1.2300024187845244|false|-0.21000882592723507|         -1|          non|
| 17|        39.975379|116.33391999999999|     111|39566.477476851804|2008-04-28|11:27:34|0.8100015928581015|false|  0.0024983345033368|          1|          non|
```

（程序运行后的结果）

```
# 用停留点区间对 ac 进行分割
sdf = spk.createDataFrame(df)
sdf = sdf.join(newdf, on='index')
begin = []  # 存储每个停留区间的左端点序号
end = []  # 存储每个停留区间的右端点序号
for i in stopgroup:
    begin.append(i[0])  # 添加左端点的序号
    end.append(i[-1])      # 添加右端点序号
ct = 0
```

```
nonstop = []  # 存储每段非停留点区间
while ct < len(begin) - 1:
    list = ac[end[ct] + 1:begin[ct + 1]]
    nonstop.append(list)
    ct += 1
# 由于每个加速减速区间必须得有 4 个点，因此必须要对结果修正，这个位置是与同学讨论学会的
# 首先删除划分的非停留区间中元素不足 4 的区间,存储在 err 中
error = []
i = 0
while i < len(nonstop):
    if len(nonstop[i]) < 4:
        error.append(nonstop[i])            # 将误差值存在 error 中,然后去除掉不满足条件的元素
    i += 1
temp_nonstop = nonstop
for item in error:  # 在 nonstop 中删除误差值
    if item in temp_nonstop:
        temp_nonstop.remove(item)
nonstop = temp_nonstop
```

对于不连续的数据进行处理，利用之前获得的 stopgroup 点，即连续的停止点区间来获取非停止点的区间，并将非停止的区间加入到 nonstop 数组中，非停止区间即是我们要寻找的加速区间与减速区间，通过判断 nonstop 中的一个元组是否长度小于 4，如果小于 4，将该元组存储在 error 数组中，然后利用 error 数组去除掉 nonstop 中的元素，满足了连续的条件。

```
acc = []  # 加速区间，需要记录
dec = []  # 减速区间，需要记录
for item in nonstop:
    temp = []
    i = 0
    while i < len(item):
        temp.append(item[i])
        if i != len(item) - 1:  # 当该点不是末尾点时
            if item[i+1]['acceleration'] == item[i]['acceleration']:
                i += 1                          # 如果下一点和本点加速状况相同，进入下一循环
            else:                               # 如果下一点和本点加速状况不同，将该组加（减）速区间添加到对应列表中，将 temp 清空，进入下一循环
                if temp[0]['acceleration'] == 1:
                    acc.append(temp)
                else:
                    dec.append(temp)
                temp = []
                i += 1
        else:  # 到末尾点时，将该组加（减）速区间添加到对应列表中，结束循环
```

```
            dec.append(temp)
```

通过判断该点与下一点的状态，利用写好的 acceleration 字段中的 1 与-1 将相应的内容写入到加速区间数组 acc，与 dec 数组，存储完了加速与减速区间。

```
acc_or_desc = []
ct = 0
while ct < len(car_speed):
    temp = {'index': ct, 'speed_asc_desc': 'non'}  # 错误点是
    acc_or_desc.append(temp)
    ct += 1
for i in flag:
    acc_or_desc[i]['speed_asc_desc'] = 'Stop'
for i in acc:  # 加速
    for j in i:
        acc_or_desc[j['index']]['speed_asc_desc'] = 'accelerate'
for i in dec:  # 减速
    for j in i:
        acc_or_desc[j['index']]['speed_asc_desc'] = 'decelerate'

newdf= spk.createDataFrame(acc_or_desc)
sdf = sdf.join(newdf, on='index')
sdf = sdf.sort(sdf['index'].asc())
return sdf
```

创建新的 newdf，其中存储了 speed_asc_desc,包含了加速减速与停止区间的判断，如果不清楚的状态，即有错误的状态就赋予一个 non 值，利用之前存储的停止区间，加速区间，减速区间，将对应的 Stop，acclerate，decelerate 字段写入，然后将临时的 dataframe 与原本的 dataframe 通过 join 的方法合并，得到新的 dataframe。

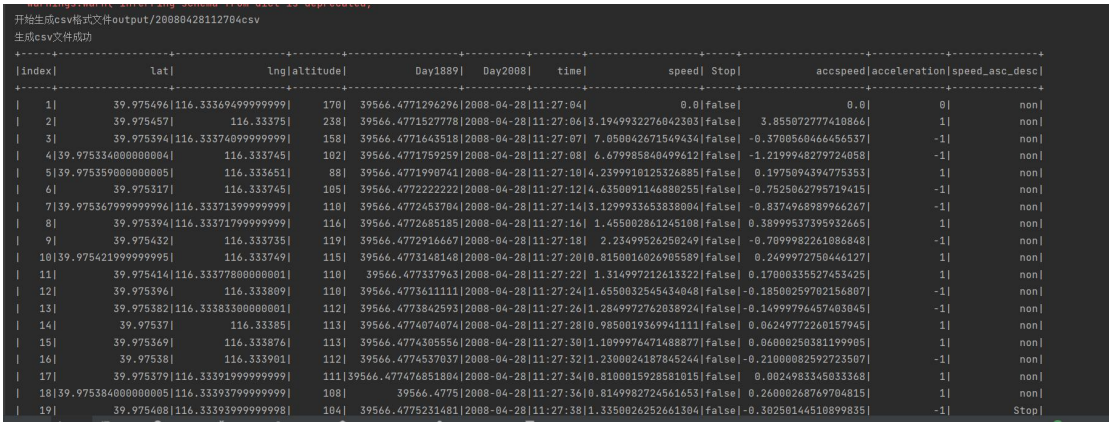
④写入 csv 文件并且保存，代码如下：

```
def output(outpath, df):
    # 输出 csv 文件
    print("开始生成 csv 格式文件{}".format(outpath))
    df.coalesce(1) \
        .write.mode("overwrite") \
        .option("mapreduce.fileoutputcommitter.marksuccessfuljobs", "false") \
        .option("header", "true") \
        .option("delimiter", ",") \
        .csv(outpath)
    print("生成 csv 文件成功")
```

只要包含 outpath 输出路径，按照 spark 的 dataframe 的规则输入 csv 文件即可

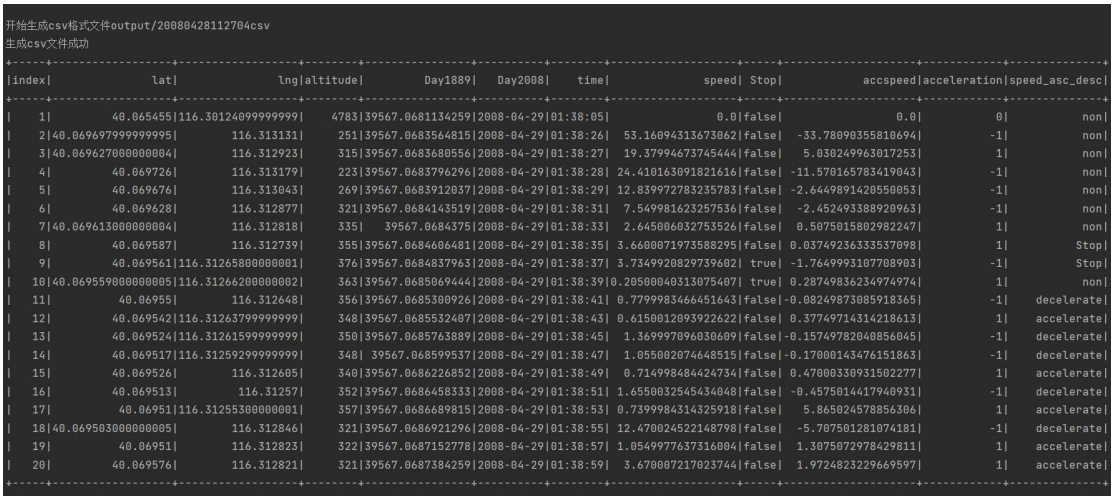
8. 实验成果

实验的主要成果按照图片的方式总结，首先得到五个 csv 文件，分别对应每个数据输入的文件，命名方式按照为 output/outputxxxxxcsv 形式，结果具有类似的形式，生成结果如下：



```
开始生成csv格式文件output/20080428112704csv
生成csv文件成功
+-----+------------------+------------------+--------+------------------+---------+--------+------------------+-----+------------------+------------+--------------+
|index|               lat|               lng|altitude|           Day1889|  Day2008|    time|             speed| Stop|           accspeed|acceleration|speed_asc_desc|
+-----+------------------+------------------+--------+------------------+---------+--------+------------------+-----+------------------+------------+--------------+
|    1|        39.975496|116.33369499999999|     170|  39566.4771296296|2008-04-28|11:27:04|               0.0|false|               0.0|           0|           non|
|    2|        39.975457|         116.33375|     238|  39566.4771527778|2008-04-28|11:27:06|3.1949932276042303|false|   3.855072777410866|           1|           non|
|    3|        39.975394|116.33374099999999|     158|  39566.4771643518|2008-04-28|11:27:07| 7.050042671549434|false| -0.3700560466456537|          -1|           non|
|    4|39.975334000000004|        116.333745|     102|  39566.4771759259|2008-04-28|11:27:08| 6.679985840499612|false| -1.2199948279724058|          -1|           non|
|    5|39.975359000000005|        116.333651|      88|  39566.4771990741|2008-04-28|11:27:10|4.2399910125326885|false|  0.1975094394775353|           1|           non|
|    6|        39.975317|        116.333745|     105|  39566.4772222222|2008-04-28|11:27:12|4.6350091146880255|false| -0.7525062795719415|          -1|           non|
|    7|39.975367999999996|116.33371399999999|     110|  39566.4772453704|2008-04-28|11:27:14|3.1299933653838004|false| -0.8374968989666267|          -1|           non|
|    8|        39.975394|116.33371799999999|     116|  39566.4772685185|2008-04-28|11:27:16| 1.455002861245108|false| 0.389995373959532665|           1|           non|
|    9|        39.975432|         116.33373|     119|  39566.4772916667|2008-04-28|11:27:18|  2.234995262502491|false| -0.7099982261086848|          -1|           non|
|   10|39.975421999999995|         116.33374|     115|  39566.4773148148|2008-04-28|11:27:20|0.8150016026905589|false|  0.2499972750446127|           1|           non|
|   11|        39.975414|116.33377800000001|     110|   39566.477337963|2008-04-28|11:27:22| 1.3149972126133222|false| 0.17000335527453425|           1|           non|
|   12|        39.975396|         116.33380|     110|  39566.4773611111|2008-04-28|11:27:24|1.6550032545434048|false|-0.18500259702156807|          -1|           non|
|   13|        39.975382|116.33383300000001|     112|  39566.4773842593|2008-04-28|11:27:26|1.2849972762038924|false|-0.14999796457403045|          -1|           non|
|   14|         39.97537|         116.33385|     113|  39566.4774074074|2008-04-28|11:27:28|0.9850019369941111|false| 0.062497722601579 45|           1|           non|
|   15|        39.975369|         116.33387|     113|  39566.4774305556|2008-04-28|11:27:30|1.1099976471488877|false| 0.06000250381199905|           1|           non|
|   16|         39.97538|         116.33390|     112|  39566.4774537037|2008-04-28|11:27:32|1.2300024187845244|false|-0.21000082592723507|          -1|           non|
|   17|        39.975379|116.33391999999999|     111|39566.477476851804|2008-04-28|11:27:34|0.8100015928581015|false|  0.0024983345033368|           1|           non|
|   18|39.975384000000005|116.33393799999999|     108|       39566.4775|2008-04-28|11:27:36|0.8149827245616 53|false| 0.26000268769704815|           1|           non|
|   19|        39.975408|116.33393999999998|     104|  39566.4775231481|2008-04-28|11:27:38|1.3350026252661304|false|-0.30250144510899835|          -1|          Stop|
```

（结果 1）



```
开始生成csv格式文件output/20080428112704csv
生成csv文件成功
+-----+------------------+------------------+--------+------------------+---------+--------+------------------+-----+------------------+------------+--------------+
|index|               lat|               lng|altitude|           Day1889|  Day2008|    time|             speed| Stop|           accspeed|acceleration|speed_asc_desc|
+-----+------------------+------------------+--------+------------------+---------+--------+------------------+-----+------------------+------------+--------------+
|    1|        40.065455|116.30124099999999|    4783|39567.0681134259|2008-04-29|01:38:05|               0.0|false|               0.0|           0|           non|
|    2|40.069679999999995|        116.313131|     251|39567.0683564815|2008-04-29|01:38:26|  53.160943136 73062|false|  -33.78090355810694|          -1|           non|
|    3|40.069627000000004|        116.312923|     315|39567.0683680556|2008-04-29|01:38:27|  19.37994673745444|false|   5.030249963017253|           1|           non|
|    4|        40.069726|        116.313179|     223|39567.0683796296|2008-04-29|01:38:28| 24.410163091821616|false| -11.570165783419043|          -1|           non|
|    5|        40.069676|        116.313043|     269|39567.0683912037|2008-04-29|01:38:29| 12.839972783235783|false| -2.6449891420550053|          -1|           non|
|    6|        40.069628|        116.312877|     321|39567.0684143519|2008-04-29|01:38:31|  7.549981623257536|false|  -2.452493388920963|          -1|           non|
|    7|40.069613000000004|        116.312818|     335|   39567.0684375|2008-04-29|01:38:33|  2.645006032753526|false|  0.5075015802982247|           1|           non|
|    8|        40.069587|        116.312739|     355|39567.0684606481|2008-04-29|01:38:35| 3.6600071973588295|false| 0.03749236333537098|           1|          Stop|
|    9|        40.069561|116.31265800000001|     376|39567.0684837963|2008-04-29|01:38:37| 3.7349920829739602| true| -1.7649993107708903|          -1|          Stop|
|   10|40.069559000000005|116.31266200000002|     363|39567.0685069444|2008-04-29|01:38:39|0.20500040313075407| true| 0.28749836234974974|           1|           non|
|   11|         40.06955|        116.312648|     356|39567.0685300926|2008-04-29|01:38:41| 0.7799983466451643|false|-0.08249873085918365|          -1|    decelerate|
|   12|        40.069542|116.31263799999999|     348|39567.0685532407|2008-04-29|01:38:43| 0.6150012093922622|false| 0.37749714314218613|           1|    accelerate|
|   13|        40.069524|116.31261599999999|     350|39567.0685763889|2008-04-29|01:38:45|  1.369997096030609|false|-0.15749782040856045|          -1|    decelerate|
|   14|        40.069517|116.31259299999999|     348| 39567.0685995537|2008-04-29|01:38:47|  1.055002074648515|false|-0.17000143476151863|          -1|    decelerate|
|   15|        40.069526|        116.312605|     340|39567.0686226852|2008-04-29|01:38:49|  0.714998484424734|false| 0.47000330931502277|           1|    accelerate|
|   16|        40.069513|         116.31257|     352|39567.0686458333|2008-04-29|01:38:51| 1.6550032545434048|false| -0.4575014417940931|          -1|    decelerate|
|   17|         40.06951|116.31255300000001|     357|39567.0686689815|2008-04-29|01:38:53| 0.7399984314325918|false|   5.865024578856306|           1|    accelerate|
|   18|40.069503000000005|        116.312846|     321|39567.0686921296|2008-04-29|01:38:55| 12.470024522148798|false|  -5.707501281074181|          -1|    decelerate|
|   19|         40.06951|        116.312823|     322|39567.0687152778|2008-04-29|01:38:57| 1.0549977637316004|false|  1.3075072978429811|           1|    accelerate|
|   20|        40.069576|        116.312821|     321|39567.0687384259|2008-04-29|01:38:59|  3.670007217023744|false|  1.9724823229669597|           1|    accelerate|
+-----+------------------+------------------+--------+------------------+---------+--------+------------------+-----+------------------+------------+--------------+
```

（结果 2）

```
开始生成csv格式文件output/20080428112704csv
生成csv文件成功
+-----+-----------------+-----------------+--------+-----------------+----------+--------+-------------------+-----+--------------------+------------+-------------+
|index|              lat|              lng|altitude|          Day1889|   Day2008|    time|              speed| Stop|           accspeed|acceleration|speed_asc_desc|
+-----+-----------------+-----------------+--------+-----------------+----------+--------+-------------------+-----+--------------------+------------+-------------+
|    1|         40.07045|116.31296299999998|     492|  39589.0649189815|2008-05-21|01:33:29|                0.0|false|                 0.0|           0|          non|
|    2|          40.0704|         116.3132|     492|  39589.0649305556|2008-05-21|01:33:30|  28.249940111887935|false|  -10.01988526375906|          -1|          non|
|    3|        40.070495|116.31308899999999|     491|  39589.0649421296|2008-05-21|01:33:31|  18.230121180106137|false|  -6.977549119782567|          -1|          non|
|    4|        40.070524|         116.3129|     491|  39589.0649652778|2008-05-21|01:33:33|  4.2749895946259955|false| -0.6999931151931701|          -1|          non|
|    5|        40.070495|          116.312|     491|  39589.0649884259|2008-05-21|01:33:35| 2.87500655734078867|false| 0.08499280868256466|           1|          non|
|    6|40.070471999999995|         116.3128|     490|  39589.0650115741|2008-05-21|01:33:37| 3.0449925884528737|false| -0.8099964859493138|          -1|          non|
|    7|        40.070458|116.31284699999999|     490|  39589.0650347222|2008-05-21|01:33:39| 1.4250028022503638|false|-0.22000197873434876|          -1|          non|
|    8|        40.070456|116.31286999999999|     490|  39589.0650578704|2008-05-21|01:33:41| 0.9849979121095985|false|  0.6750046672123264|           1|          non|
|    9|        40.070429|         116.3128|     489|  39589.0650810185|2008-05-21|01:33:43|  2.3350045917576613|false|-0.25000371076001354|          -1|          non|
|   10|        40.070426|         116.3127|     489|  39589.0651041667|2008-05-21|01:33:45|  1.8349961103767665|false|  0.4100053615995279|           1|          non|
|   11|        40.070402|         116.3127|     489|  39589.0651273148|2008-05-21|01:33:47| 2.6550052210348882|false| -0.8300019057141846|          -1|         Stop|
|   12|        40.070399|         116.3127|     488|   39589.065150463|2008-05-21|01:33:49| 0.9949978909127416| true| -0.2849991108118887|          -1|         Stop|
|   13|        40.070392|         116.3127|     488|39589.065173611096|2008-05-21|01:33:51|0.42500096934602954| true|   8.574995723810037|           1|          non|
|   14|        40.069894|         116.3142|     122|  39589.0652199074|2008-05-21|01:33:55|  34.72499188261762|false| -15.187468358897645|          -1|    decelerate|
|   15|        40.069955|         116.3141|     122|  39589.0652430556|2008-05-21|01:33:57|  4.3499807793672621|false| -0.7899942196067764|          -1|    decelerate|
|   16|        40.069979|116.31408799999998|     120|  39589.0652662037|2008-05-21|01:33:59|  2.7700054471814091|false|-0.29250089893988335|          -1|    decelerate|
|   17|        40.069965|116.31403999999999|     113|  39589.0652893518|2008-05-21|01:34:01| 2.1850049835789993|false| 0.01999475171178681|           1|    accelerate|
|   18|        40.069936|         116.3140|     109|    39589.0653125|2008-05-21|01:34:03|  2.2249945843374861|false| -0.1399953933796971|          -1|    decelerate|
|   19|        40.069919|116.31396399999998|      98|  39589.0653356481|2008-05-21|01:34:05|  1.9450044361835941|false|  -0.092504134860614|          -1|    decelerate|
|   20|        40.069911|         116.3139|      94|  39589.0653587963|2008-05-21|01:34:07|  1.7599957161501011|false|  0.1250047192550946|           1|    accelerate|
+-----+-----------------+-----------------+--------+-----------------+----------+--------+-------------------+-----+--------------------+------------+-------------+
only showing top 20 rows
```

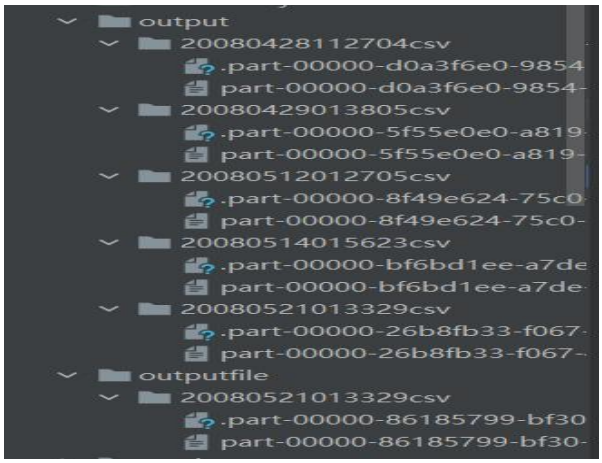
（结果 3）

```
开始生成csv格式文件output/20080428112704csv
生成csv文件成功
+-----+-----------------+-----------------+--------+-----------------+----------+--------+-------------------+-----+--------------------+------------+-------------+
|index|              lat|              lng|altitude|          Day1889|   Day2008|    time|              speed| Stop|           accspeed|acceleration|speed_asc_desc|
+-----+-----------------+-----------------+--------+-----------------+----------+--------+-------------------+-----+--------------------+------------+-------------+
|    1|         40.07045|116.31296299999998|     492|  39589.0649189815|2008-05-21|01:33:29|                0.0|false|                 0.0|           0|          non|
|    2|          40.0704|         116.3132|     492|  39589.0649305556|2008-05-21|01:33:30|  28.249940111887935|false|  -10.01988526375906|          -1|          non|
|    3|        40.070495|116.31308899999999|     491|  39589.0649421296|2008-05-21|01:33:31|  18.230121180106137|false|  -6.977549119782567|          -1|          non|
|    4|        40.070524|         116.3129|     491|  39589.0649652778|2008-05-21|01:33:33|  4.2749895946259955|false| -0.6999931151931701|          -1|          non|
|    5|        40.070495|          116.312|     491|  39589.0649884259|2008-05-21|01:33:35| 2.87500655734078867|false| 0.08499280868256466|           1|          non|
|    6|40.070471999999995|         116.3128|     490|  39589.0650115741|2008-05-21|01:33:37| 3.0449925884528737|false| -0.8099964859493138|          -1|          non|
|    7|        40.070458|116.31284699999999|     490|  39589.0650347222|2008-05-21|01:33:39| 1.4250028022503638|false|-0.22000197873434876|          -1|          non|
|    8|        40.070456|116.31286999999999|     490|  39589.0650578704|2008-05-21|01:33:41| 0.9849979121095985|false|  0.6750046672123264|           1|          non|
|    9|        40.070429|         116.3128|     489|  39589.0650810185|2008-05-21|01:33:43|  2.3350045917576613|false|-0.25000371076001354|          -1|          non|
|   10|        40.070426|         116.3127|     489|  39589.0651041667|2008-05-21|01:33:45|  1.8349961103767665|false|  0.4100053615995279|           1|          non|
|   11|        40.070402|         116.3127|     489|  39589.0651273148|2008-05-21|01:33:47| 2.6550052210348882|false| -0.8300019057141846|          -1|         Stop|
|   12|        40.070399|         116.3127|     488|   39589.065150463|2008-05-21|01:33:49| 0.9949978909127416| true| -0.2849991108118887|          -1|         Stop|
|   13|        40.070392|         116.3127|     488|39589.065173611096|2008-05-21|01:33:51|0.42500096934602954| true|   8.574995723810037|           1|          non|
|   14|        40.069894|         116.3142|     122|  39589.0652199074|2008-05-21|01:33:55|  34.72499188261762|false| -15.187468358897645|          -1|    decelerate|
|   15|        40.069955|         116.3141|     122|  39589.0652430556|2008-05-21|01:33:57|  4.3499807793672621|false| -0.7899942196067764|          -1|    decelerate|
|   16|        40.069979|116.31408799999998|     120|  39589.0652662037|2008-05-21|01:33:59|  2.7700054471814091|false|-0.29250089893988335|          -1|    decelerate|
|   17|        40.069965|116.31403999999999|     113|  39589.0652893518|2008-05-21|01:34:01| 2.1850049835789993|false| 0.01999475171178681|           1|    accelerate|
|   18|        40.069936|         116.3140|     109|    39589.0653125|2008-05-21|01:34:03|  2.2249945843374861|false| -0.1399953933796971|          -1|    decelerate|
|   19|        40.069919|116.31396399999998|      98|  39589.0653356481|2008-05-21|01:34:05|  1.9450044361835941|false|  -0.092504134860614|          -1|    decelerate|
|   20|        40.069911|         116.3139|      94|  39589.0653587963|2008-05-21|01:34:07|  1.7599957161501011|false|  0.1250047192550946|           1|    accelerate|
+-----+-----------------+-----------------+--------+-----------------+----------+--------+-------------------+-----+--------------------+------------+-------------+
```

（结果 4）

```
开始生成csv格式文件output/20080428112704csv
生成csv文件成功
+-----+-----------------+-----------------+--------+-----------------+----------+--------+-------------------+-----+--------------------+------------+-------------+
|index|              lat|              lng|altitude|          Day1889|   Day2008|    time|              speed| Stop|           accspeed|acceleration|speed_asc_desc|
+-----+-----------------+-----------------+--------+-----------------+----------+--------+-------------------+-----+--------------------+------------+-------------+
|    1|        40.070171|116.31309899999999|     488|  39582.0808217593|2008-05-14|01:56:23|                0.0|false|                 0.0|           0|          non|
|    2|        40.070932|116.31258600000001|    1192|  39582.0820023148|2008-05-14|01:58:05| 0.9304902423643492|false|  0.7372501574506645|           1|         Stop|
|    3|40.070809999999996|116.31260400000001|    1169|   39582.082025463|2008-05-14|01:58:07| 2.4049941462164734| true| -1.1174994280417672|          -1|         Stop|
|    4|        40.070889|         116.3126|    1159|  39582.0820486111|2008-05-14|01:58:09|0.17000038773841186| true| 0.15249885687047948|           1|         Stop|
|    5|        40.070884|116.31261599999999|    1154|  39582.0820717593|2008-05-14|01:58:11|0.47499884384732843| true| -0.0974993383469802|          -1|         Stop|
|    6|40.070879999999995|         116.3126|    1154|39582.082094907404|2008-05-14|01:58:13| 0.2800005506176153| true|  1.1824944148850727|           1|          non|
|    7|        40.070861|116.31267700000001|    1133|  39582.0821180556|2008-05-14|01:58:15| 2.6449943943313587|false| -0.0849949303239465|          -1|    decelerate|
|    8|        40.070837|         116.3127|    1113|39582.082141203704|2008-05-14|01:58:17| 2.4750048670664215|false|-0.05250483406676...|          -1|    decelerate|
|    9|40.070809000000004|116.31276799999999|    1094|  39582.0821643519|2008-05-14|01:58:19| 2.3699949763449224|false|-0.22749582316739447|          -1|    decelerate|
|   10|        40.070782|116.31279599999999|    1076|    39582.0821875|2008-05-14|01:58:21| 1.9150043677591686|false| -0.0775041371585689|          -1|    decelerate|
|   11|        40.070761|         116.3128|    1058|  39582.0822106482|2008-05-14|01:58:23|  1.7599957161501011|false|-0.034996010606277644|          -1|    decelerate|
|   12|40.070741999999996|         116.3128|    1041|39582.082233796296|2008-05-14|01:58:25| 1.6900038545759766|false| 0.07000000970587132|           1|    accelerate|
|   13|         40.07072|         116.3128|    1024|  39582.0822569444|2008-05-14|01:58:27| 1.8300035986794148|false| 0.04499607038430483|           1|    accelerate|
|   14|        40.070689|         116.3129|    1009|39582.082280092596|2008-05-14|01:58:29| 1.9199959302034815|false| -0.4749979454309487|          -1|    decelerate|
|   15|        40.070673|         116.3129|     997|  39582.0823032407|2008-05-14|01:58:31|  0.970001907496739| true|-0.29750068600431856|          -1|         Stop|
|   16|        40.070667|         116.3129|     989|  39582.0823263889|2008-05-14|01:58:33| 0.3749990872478909| true|-0.09499954932445881|          -1|         Stop|
|   17|        40.070664|         116.3129|     989|   39582.082349537|2008-05-14|01:58:35|0.18500042195062466| true|0.027499430010082513|           1|         Stop|
|   18|        40.070661|         116.3129|     988|  39582.0823726852|2008-05-14|01:58:37|0.23999941583865014| true|  0.7975042036824453|           1|          non|
|   19|        40.070631|         116.3129|     974|  39582.0823958333|2008-05-14|01:58:39| 1.8350041852940338|false|0.019995877762585993|           1|    accelerate|
|   20|        40.070605|         116.3128|     961|39582.082418981496|2008-05-14|01:58:41| 1.8749960255893372|false|  0.5325059251141194|           1|    accelerate|
+-----+-----------------+-----------------+--------+-----------------+----------+--------+-------------------+-----+--------------------+------------+-------------+
only showing top 20 rows
```

（结果 5）



（csv 文件夹）

由于文件过多，所以使用一个输出的 csv 文件作为例子：

```
index,lat,lng,altitude,Day1889,Day2008,time,speed,Stop,accspeed,acceleration,speed_asc_desc
1,40.070171,116.31309099999999,488,39582.0808217593,2008-05-14,01:56:23,0.0,false,0.0,0,non
2,40.070932,116.31258600000001,1192,39582.0820023148,2008-05-14,01:58:05,0.9304902423643492,false,0.7372501574506645,1,Stop
3,40.070890999999996,116.31260400000001,1169,39582.082025463,2008-05-14,01:58:07,2.4049941462164734,true,-1.1174994280417672,-1,Stop
4,40.070889,116.312607,1159,39582.0820486111,2008-05-14,01:58:09,0.17000038773841186,true,0.1524988568704794 8,1,Stop
5,40.070884,116.31261599999999,1154,39582.0820717593,2008-05-14,01:58:11,0.47499884384732843,true,-0.09749933834609802,-1,Stop
6,40.070879999999995,116.31262,1154,39582.082094907404,2008-05-14,01:58:13,0.2800005506176153,true,1.1824944148850727,1,non
7,40.070861,116.31267700000001,1133,39582.0821180556,2008-05-14,01:58:15,2.6449943934313587,false,-0.0849949303239465,-1,decelerate
8,40.070837,116.312726,1113,39582.082141203704,2008-05-14,01:58:17,2.4750048670664215,false,-0.052504834066768404,-1,decelerate
9,40.07080900000004,116.31276799999999,1094,39582.0821643519,2008-05-14,01:58:19,2.3699949763449224,false,-0.22749582316739447,-1,decelerate
10,40.070782,116.31279599999999,1076,39582.0821875,2008-05-14,01:58:21,1.9150043677591686,false,-0.0775041371585689,-1,decelerate
11,40.070761,116.312827,1058,39582.0822106482,2008-05-14,01:58:23,1.759995716150101,false,-0.03499601060627764,-1,decelerate
12,40.070741999999996,116.312858,1041,39582.082233796296,2008-05-14,01:58:25,1.6900038545759766,false,0.07000000970587132,1,accelerate
13,40.07072,116.31289,1024,39582.0822569444,2008-05-14,01:58:27,1.8300035986794148,false,0.04499607038430483,1,accelerate
14,40.070689,116.31291,1009,39582.082280092596,2008-05-14,01:58:29,1.9199959302034815,false,-0.4749979454309487,-1,decelerate
15,40.070673,116.312919,997,39582.0823032407,2008-05-14,01:58:31,0.970001907496739,true,-0.29750068600431856,-1,Stop
16,40.070667,116.312923,989,39582.0823263889,2008-05-14,01:58:33,0.3749990872478909,true,-0.09499954932445881,-1,Stop
17,40.070664,116.312925,989,39582.082349537,2008-05-14,01:58:35,0.18500042195062466,true,0.027499430010082513,1,Stop
18,40.070661,116.312921,988,39582.0823726852,2008-05-14,01:58:37,0.23999941583865014,true,0.7975042036824453,1,non
19,40.070631,116.312903,974,39582.0823958333,2008-05-14,01:58:39,1.8350041852940338,false,0.019995877762585993,1,accelerate
20,40.070605,116.312875,961,39582.082418981496,2008-05-14,01:58:41,1.8749960255893372,false,0.5325059251141194,1,accelerate
21,40.070567,116.312827,948,39582.0824421296,2008-05-14,01:58:43,2.9400057814849614,false,0.04999378136394825,1,accelerate
22,40.070528,116.312777,935,39582.082465277796,2008-05-14,01:58:45,3.0399935561555127,false,0.43500792185078513,1,accelerate
23,40.070479,116.31271100000001,922,39582.0824884259,2008-05-14,01:58:47,3.9100076889817004,false,0.8724871497144484,1,accelerate
24,40.070412,116.312611,910,39582.0825115741,2008-05-14,01:58:49,5.654986235698194,false,0.8100156281493753,1,accelerate
```

（输出文件的例子）

9．实验遇到的问题与解决

实验遇到的问题，在实验遇到的问题中，结合个人解决问题的过程中的感悟，我在解决第三题时具有较大的问题，在刚开始解决问题时具有较大的问题，不太清楚如何计算加速与减速区间。通过与同学讨论，想到了用加速度来判断是否速度区间是否单调的问题，较好地解决了所具有的问题，首先利用停止区间将原本的总点数区分为非停止区间，然后对于非停止区间利用加减速的 flag 划分区间，分出加速减速区间，然后判断区间的长度如果小于 4，就存储到 error 数组中，然后在原本的加减速区间中去除掉其中的 error 数组，修正了结果。通过与同学的讨论解决了第三问的问题。
其余的代码问题都通过了相应的修正解决了问题，通过查阅了相关资料与内容，均得到了解决，细节处理的问题有 dataframe 中产生空值，在 pandas dataframe 直接对一列赋值时产生了这个问题，通过 spark 的 dataframe 查询语句 join 函数将两张表连接，解决了相应的内容，解决了空值报错的问题。
还有许许多多的小问题，在这里不过多赘述，问题的解决一直是代码书写的一大难题，解决问题的过程收获了许多。

（二）实验设计体会

本次课程设计使我将上课老师所讲授的知识很好地运用，也明白了学习不只是一个人的探究，在第三个任务的第三个问题处理的时候，之前总是会遇到一些问题，在处理加减速区间的问题时，一直没有想好如何寻找连续的点，通过与同学的讨论明白了数据点的获取必须具有四个及以上的数据点，利用如此的数据点判断数据长度，从而实现实验要求，使我受益匪浅，这门课程教会了我如何去解决数据库有关的问题，去使用和学习对于时空数据的处理，在动手操作的过程中收获了许多之前没有体悟到的知识与方法，练习与掌握了 spark 的基础理论与操作，收获了许多的知识，但是由于自身能力的问题，没有能够及时提交作业，在这里想向老师说声抱歉。最后，感谢李老师，李老师辛苦了！