时变环境下建邺区充电站多目标配置实验报告

需要打开的数据和服务包括：

理论空间地图MXD

启动MongoDB服务；

运行代码工程；

实验报告文档复现一遍；

# 时变建邺区实验

现有做研究工程，都会面对以下几个困难和挑战：

思路是逐步明晰的；过程中会经常调整思路；需要符合工程的规范；便于复盘和回忆；

形成的成果有：数据、工程、文档、绘图、地图、实验日志

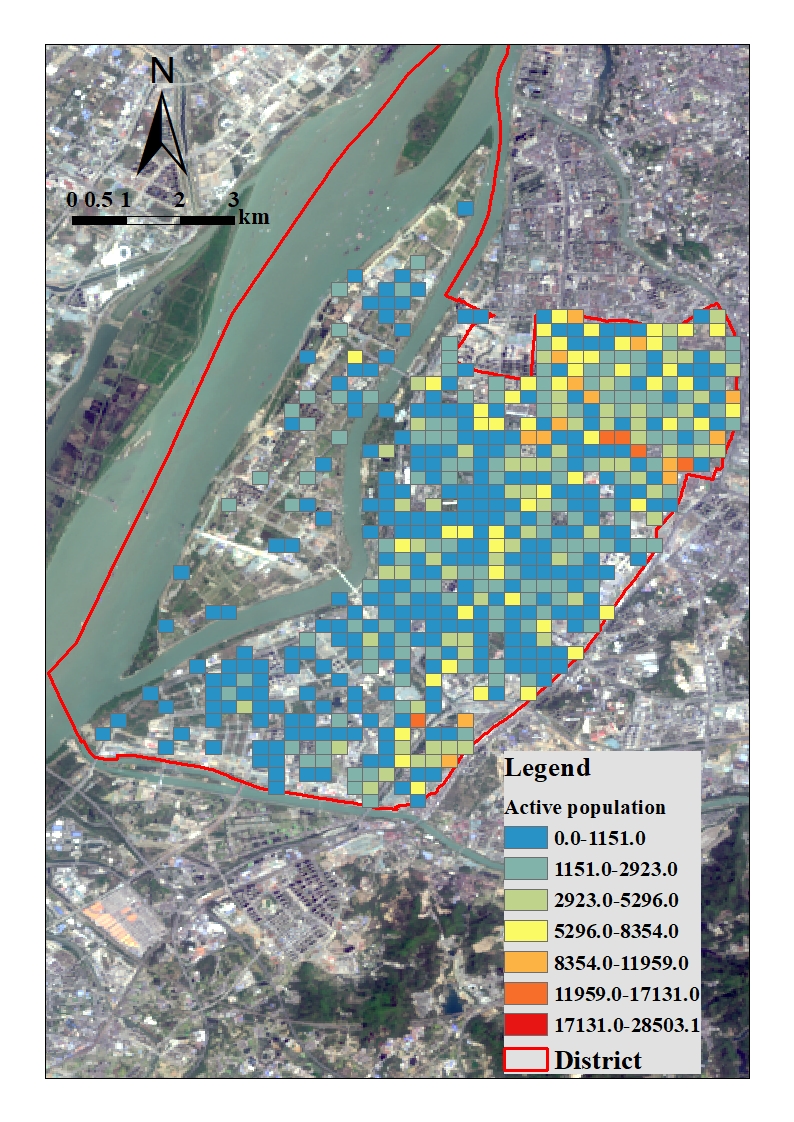
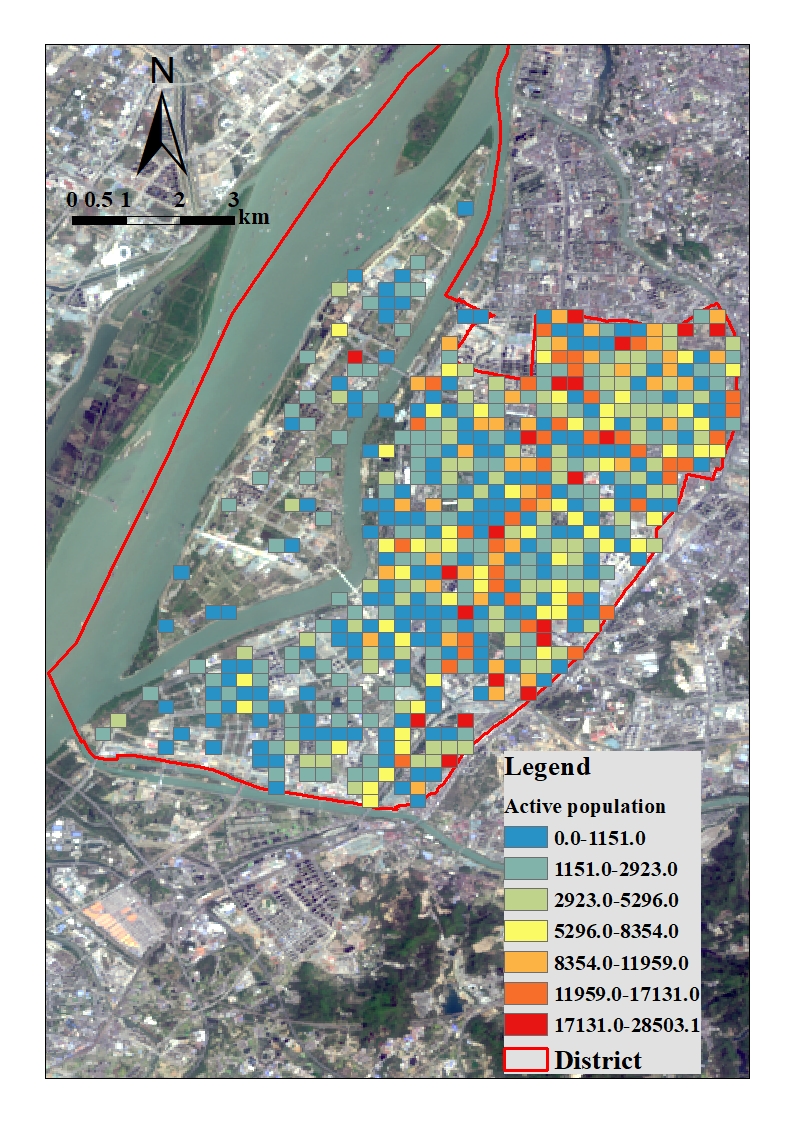
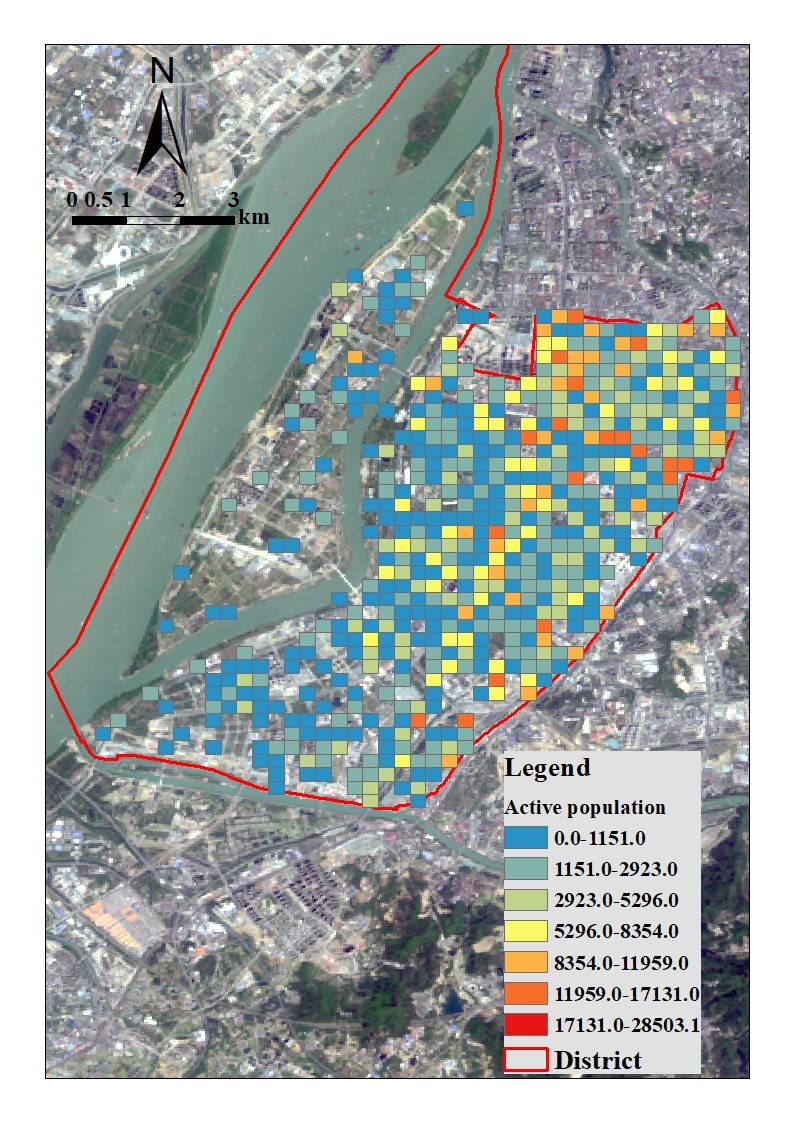
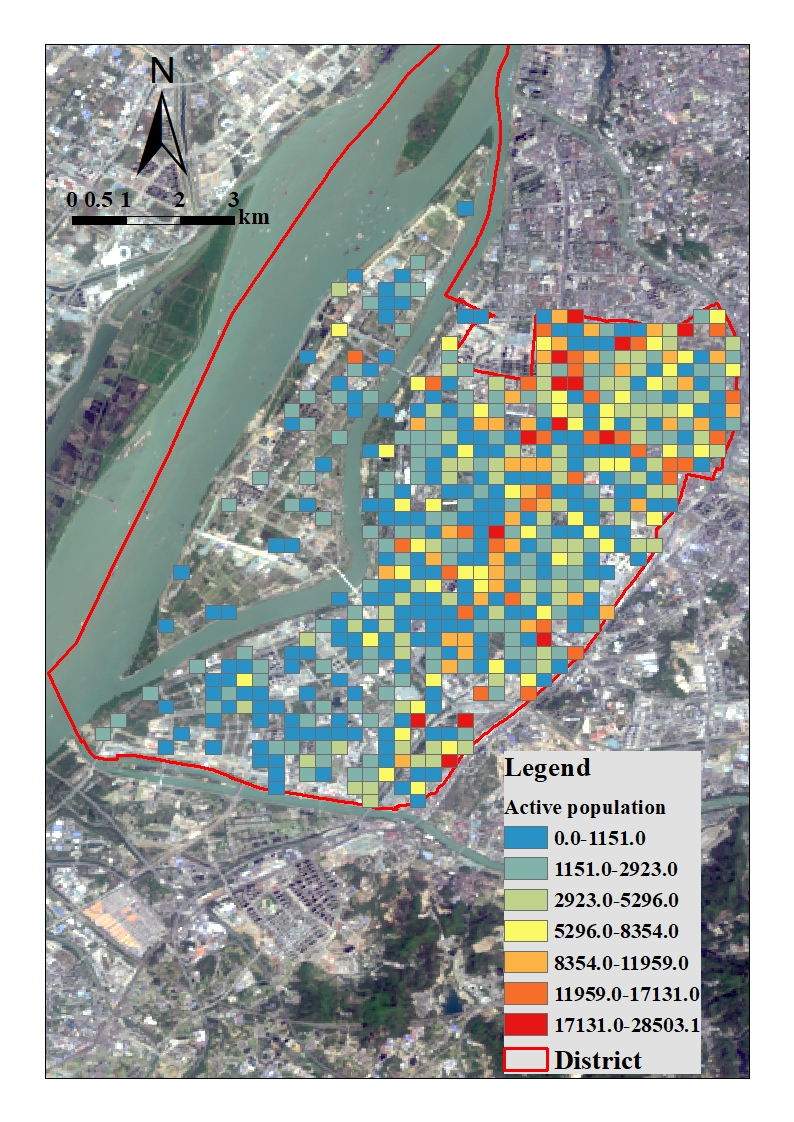
使用的工作空间仍是powerstation；

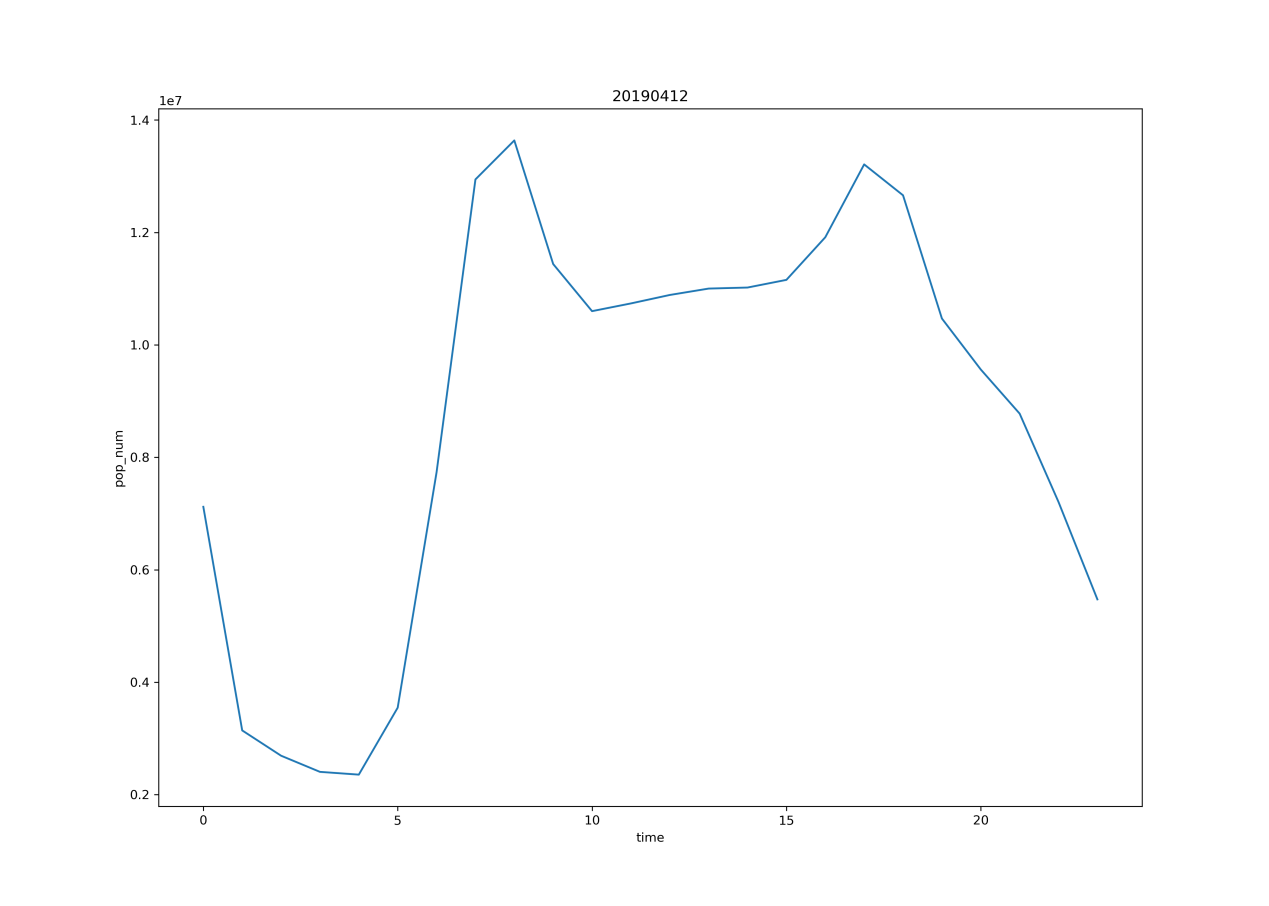
### 准备代码工程和数据SDE工程

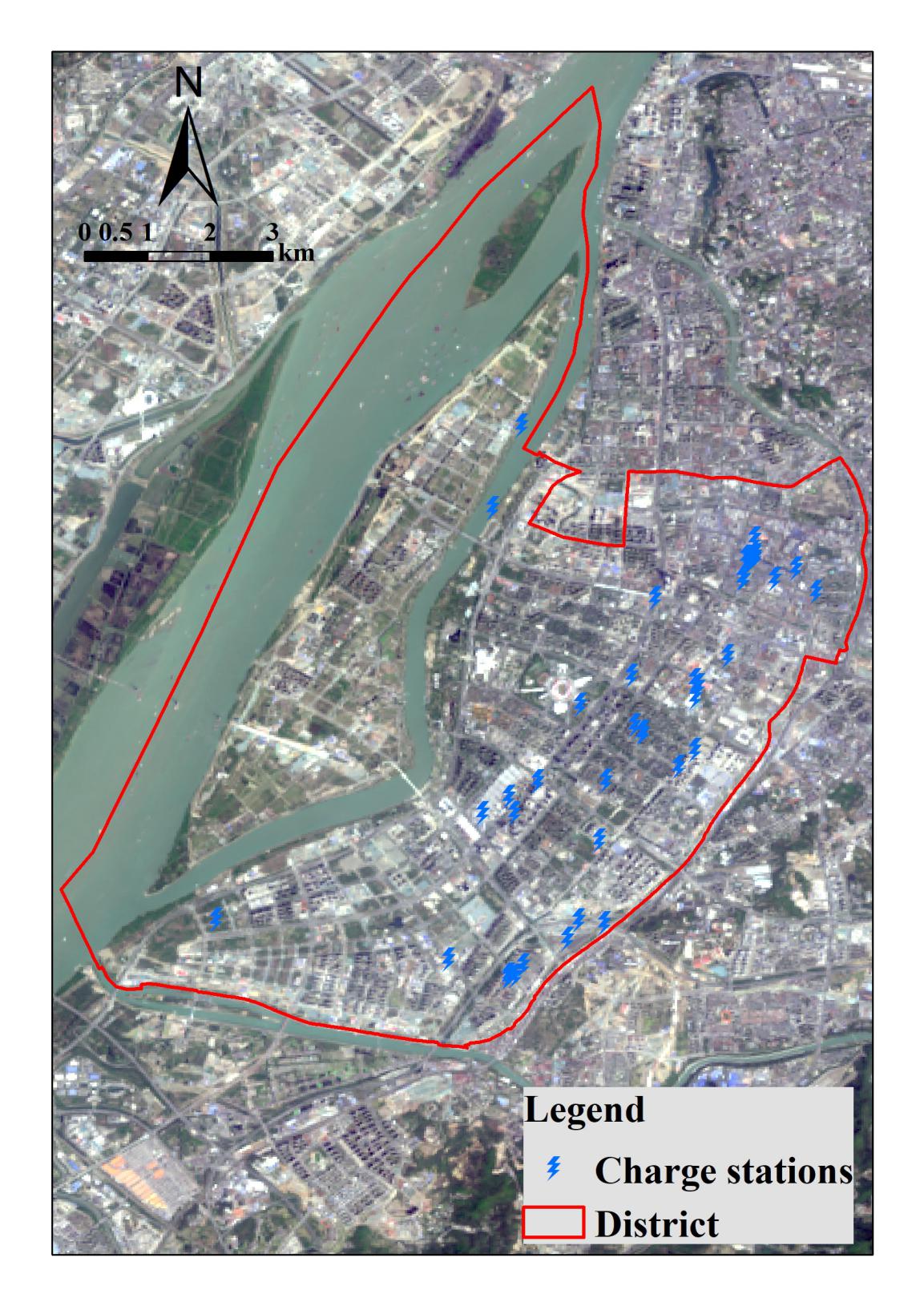
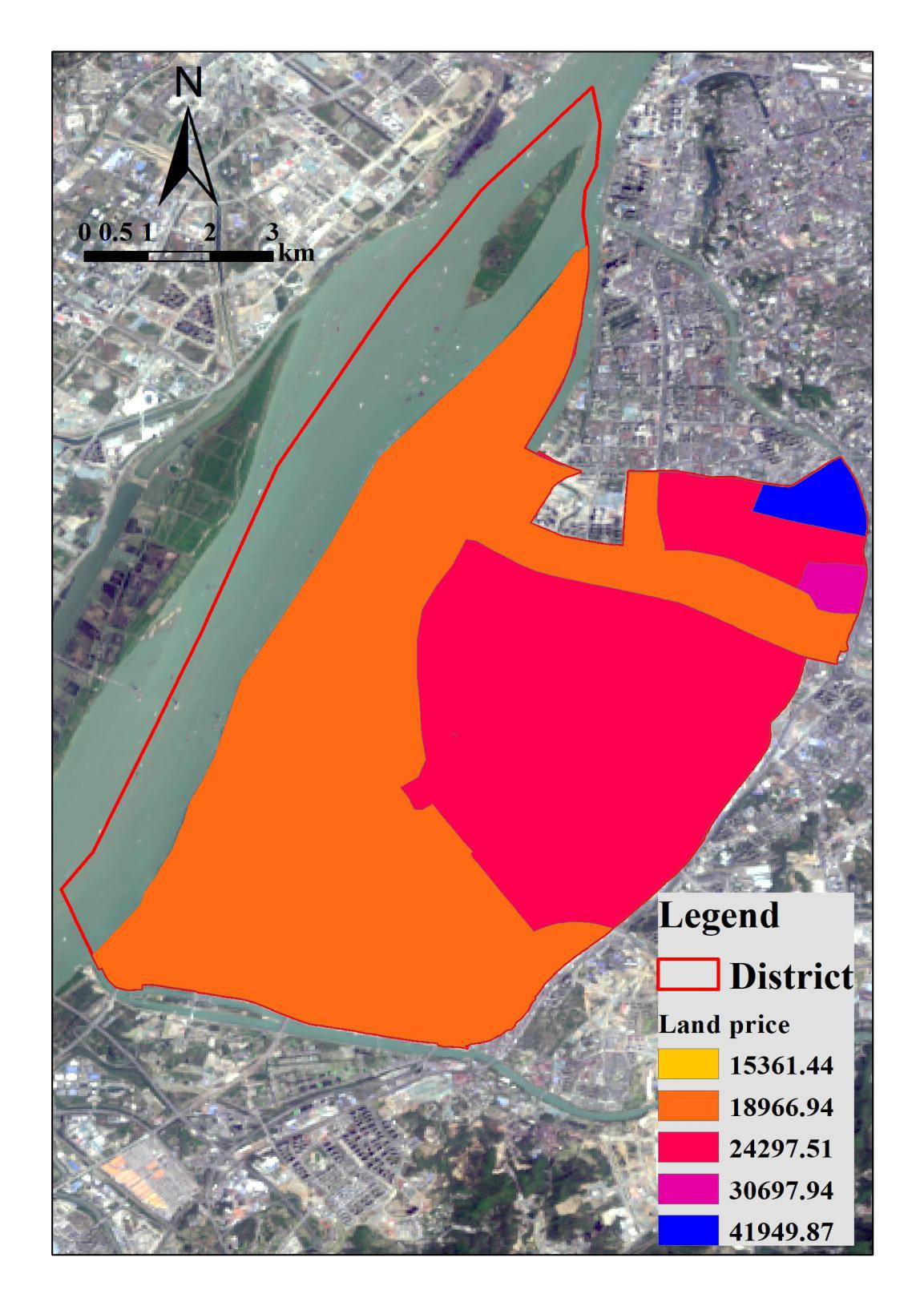
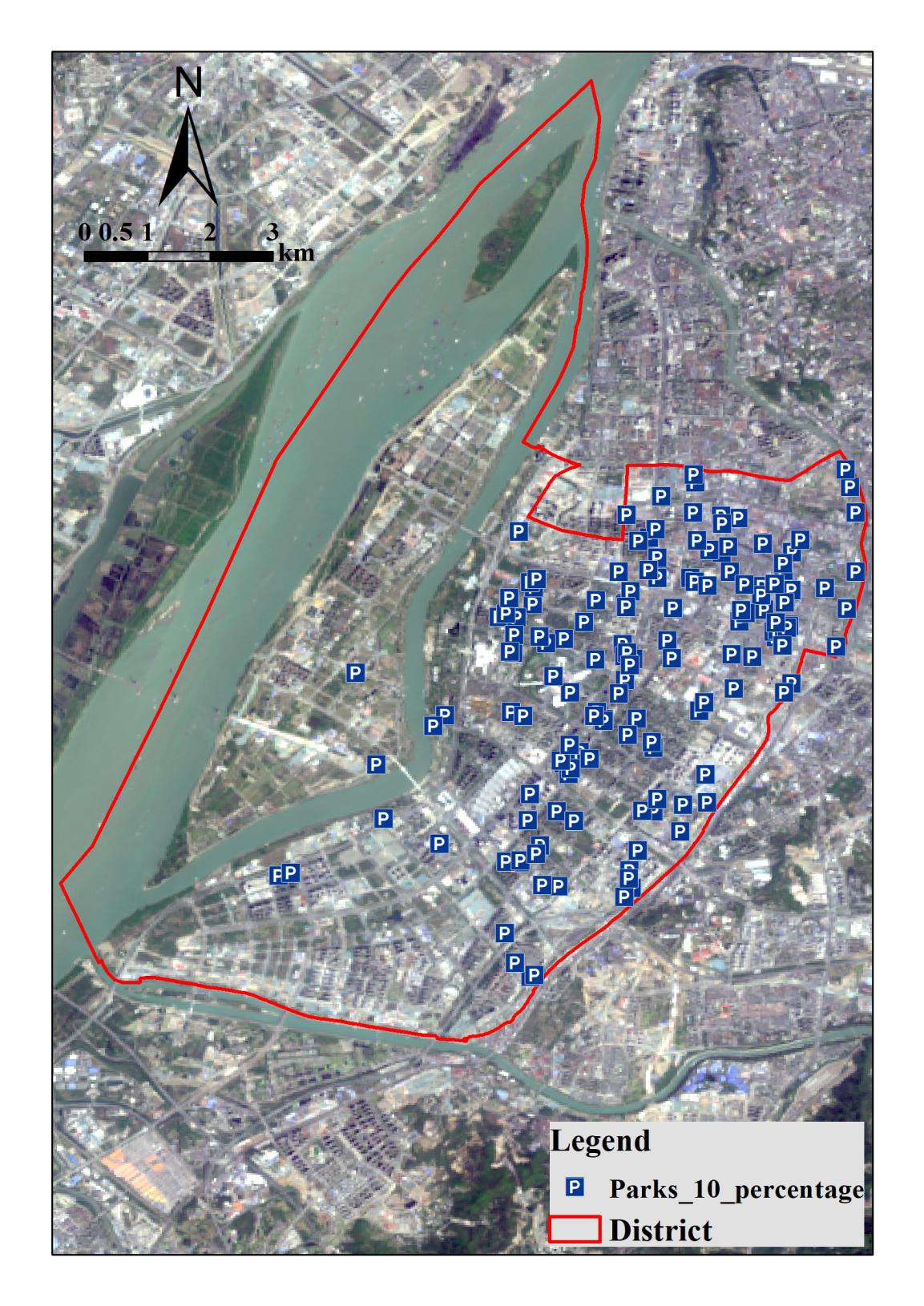
### 首先，准备provider 和 demand图层

Demand数据采用了多时刻的手机信令人口数据；

Provider数据采用了供给点数据；







### 其次，准备好候选站的数据集；

首先，通过POI数据筛选停车场，得到1720条记录，主要包括：专用停车场、公共停车场、路边停车场四类。顾及快速充电站是以服务公共为导向的，故此选用公共停车场、路边停车场两类作为候选位置点，此外人工核查的方式剔除同一个停车场标记为多个停车场出口的重复情况，保留一条有效记录，得到有效记录数为：1460条。

此时，为了简化计算，分别采用10%，30%，50%对候选集做抽稀；得到三个图层。此时，用10%的结果，来做第一次的运算。

复制park\_jy图层，然后得到park\_ky\_test图层，给park\_jy\_test图层增加10个字段，存放solution

arcpy.env.workspace = r'Database Connections/nanjingclinic.sde'

arcpy.AddField\_management(in\_table="fishnet\_d\_p", field\_name='V\_STD\_08', field\_type="DOUBLE")

### 初始化数据到MongoDB中

修改初始化代码，注意数据结构

注意：provider（PS\_JY）的Key\_ID和petential\_provider（park\_jy\_test）的Key\_ID不可以有重复，所以需要加一个处理：

select \* from PS\_JY t2 where t2.key\_id in (select key\_id from park\_jy\_test )

注意：统计一下三张表的点的数量：

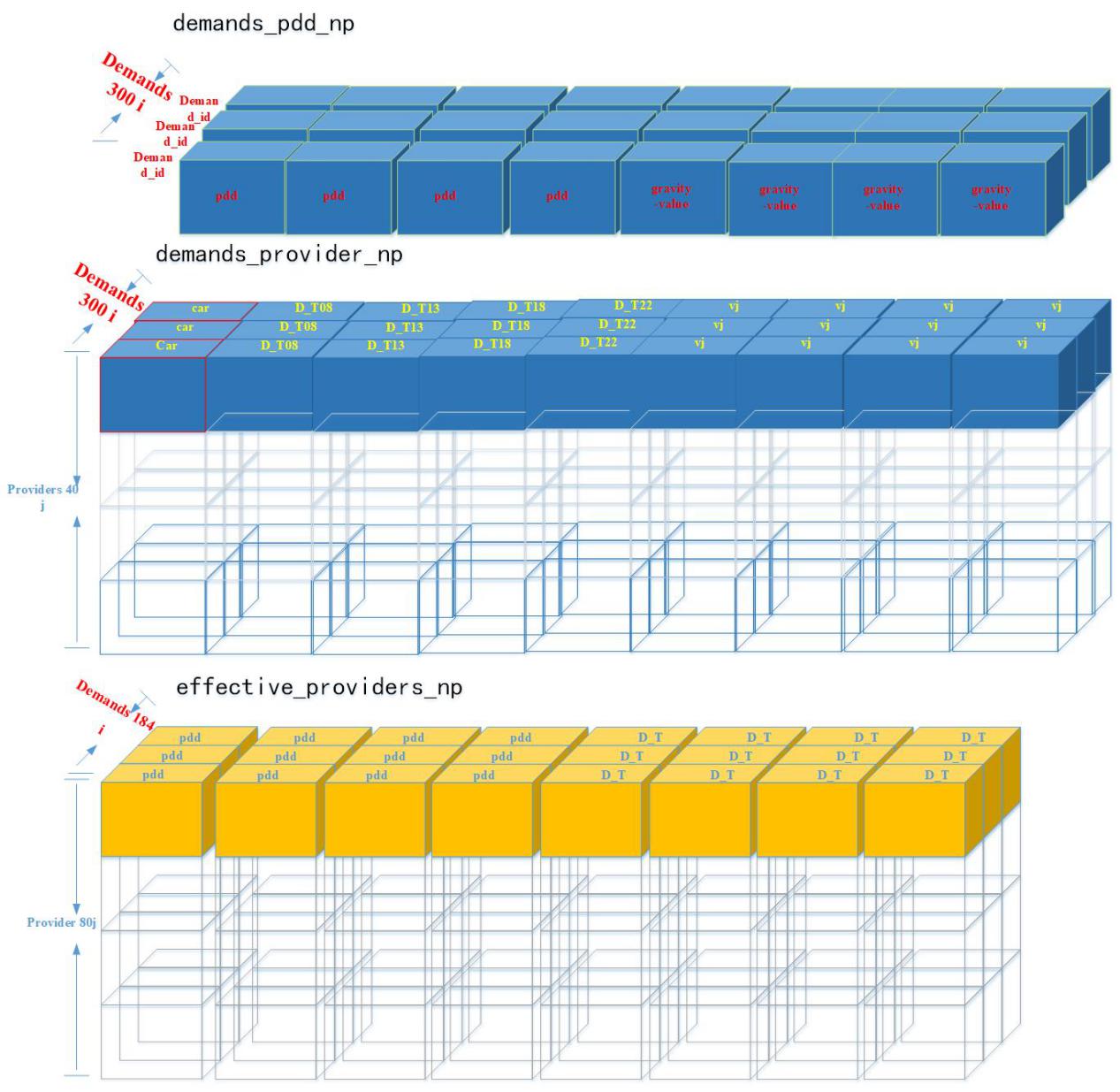
select count(0) from pop\_grid250\_20190412; *---565*

select count(0) from PS\_JY; *---40*

select count(0) from parks\_10\_percentage; *---146*

注意，运行initial\_provider.py文件的时候需要打断点条件；

已有的数据结构



### 爬取四个时间段的人口和导航数据；

修改数据集的地址，然后运行，多个时刻的；

运行，注意量超了；

### 对导航数据做简要的拥挤度分析（缺图）

待定（缺图）

### 可达性评价

计算结果：四个时刻（四种人口、四种交通条件）得出不同人口单元中的可达性结果；

制图可视化；

### 不公平性评价

四个时刻的不公平性分别为：141，140,90，166

update pop\_grid250\_20190412 set inequlity08=141.00924594924555,inequlity13=140.84396444692027,inequlity18=91.03367784820455,inequlity22=166.60440345142962 where KEYID=109333

### 准备好规模缺口【预测值】；

新建充电桩总量为100个；

### 修改多目标函数；

行驶时间总和最小；

Waiting time 最小；

公平性最大；

建设成本最低；

覆盖人口最大；

### 修改NSGA2的代码；

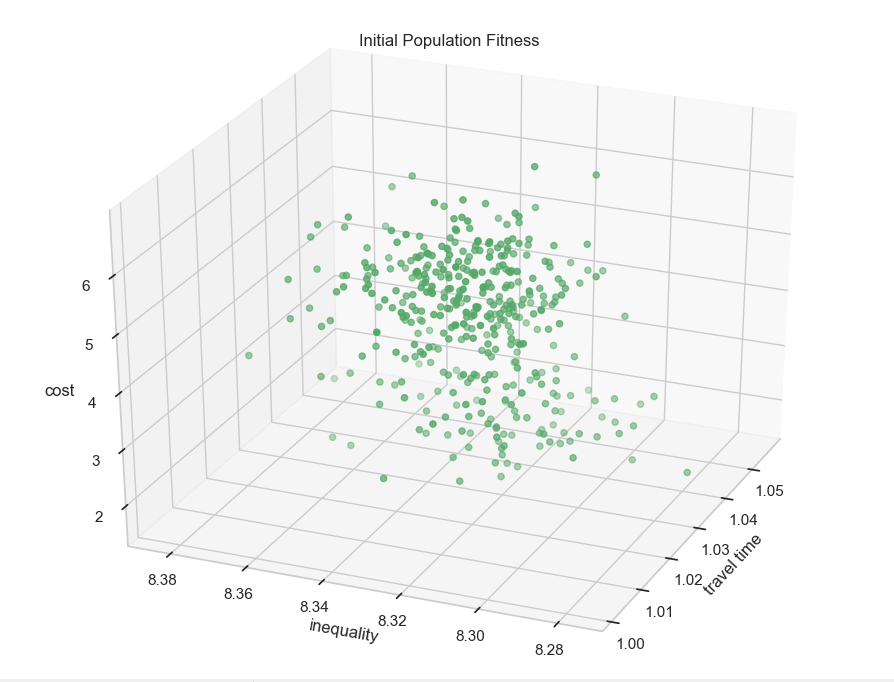
对代码的目标函数做定义，尤其要注意是最小化原则；

数据读入

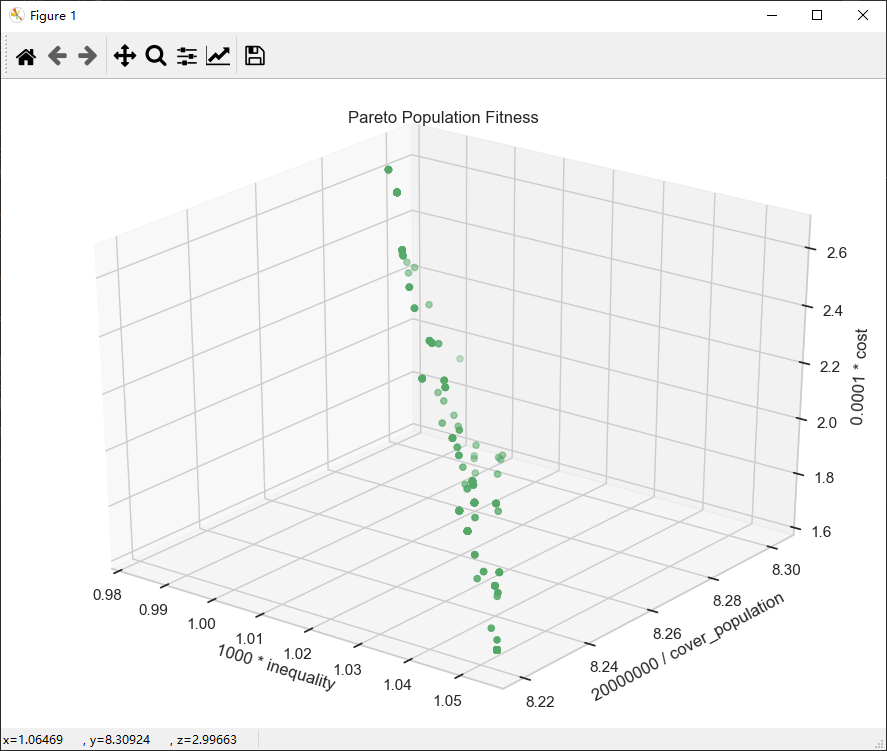
验证通过做了测试；

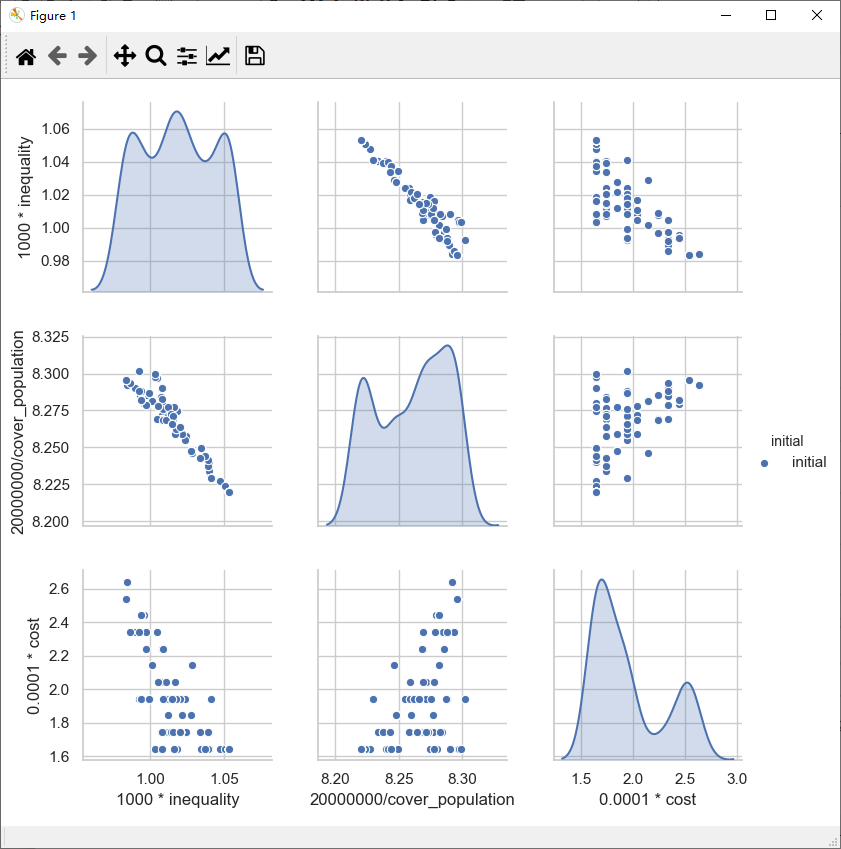
### 运行结果；

为了简化实验，先运行三个目标：行驶时间、不公平性、建设成本

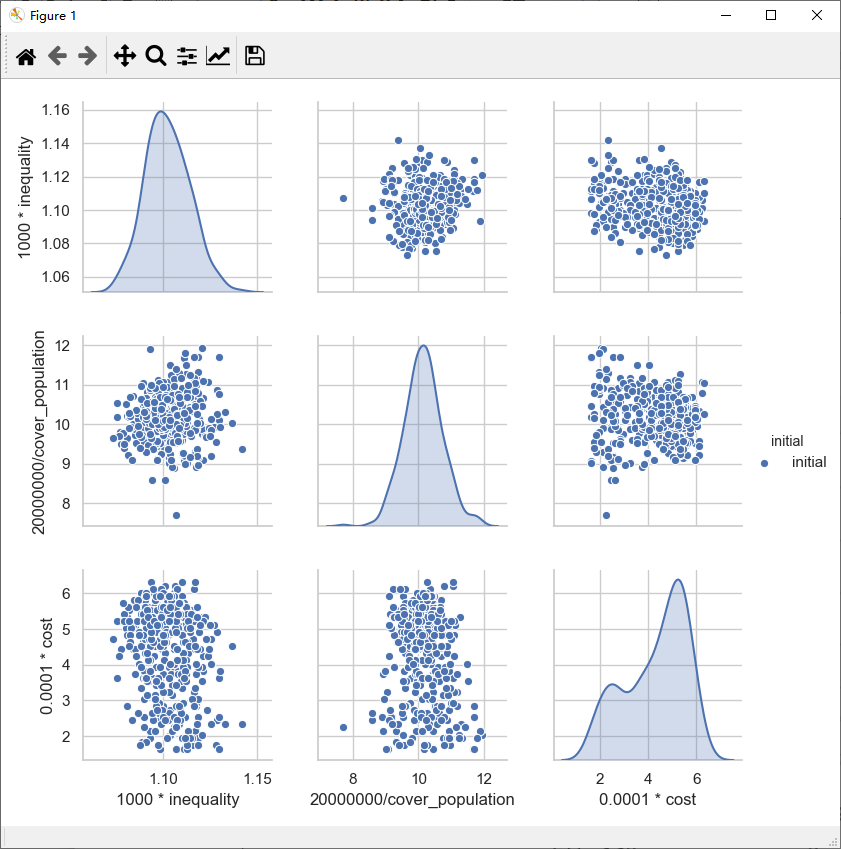
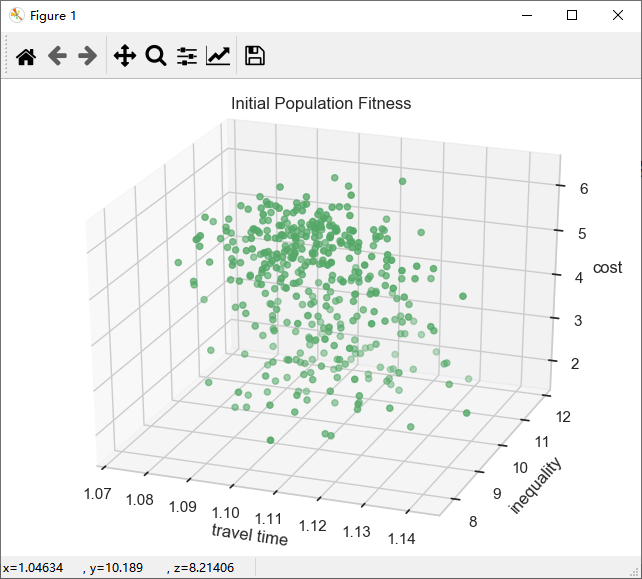


优化后



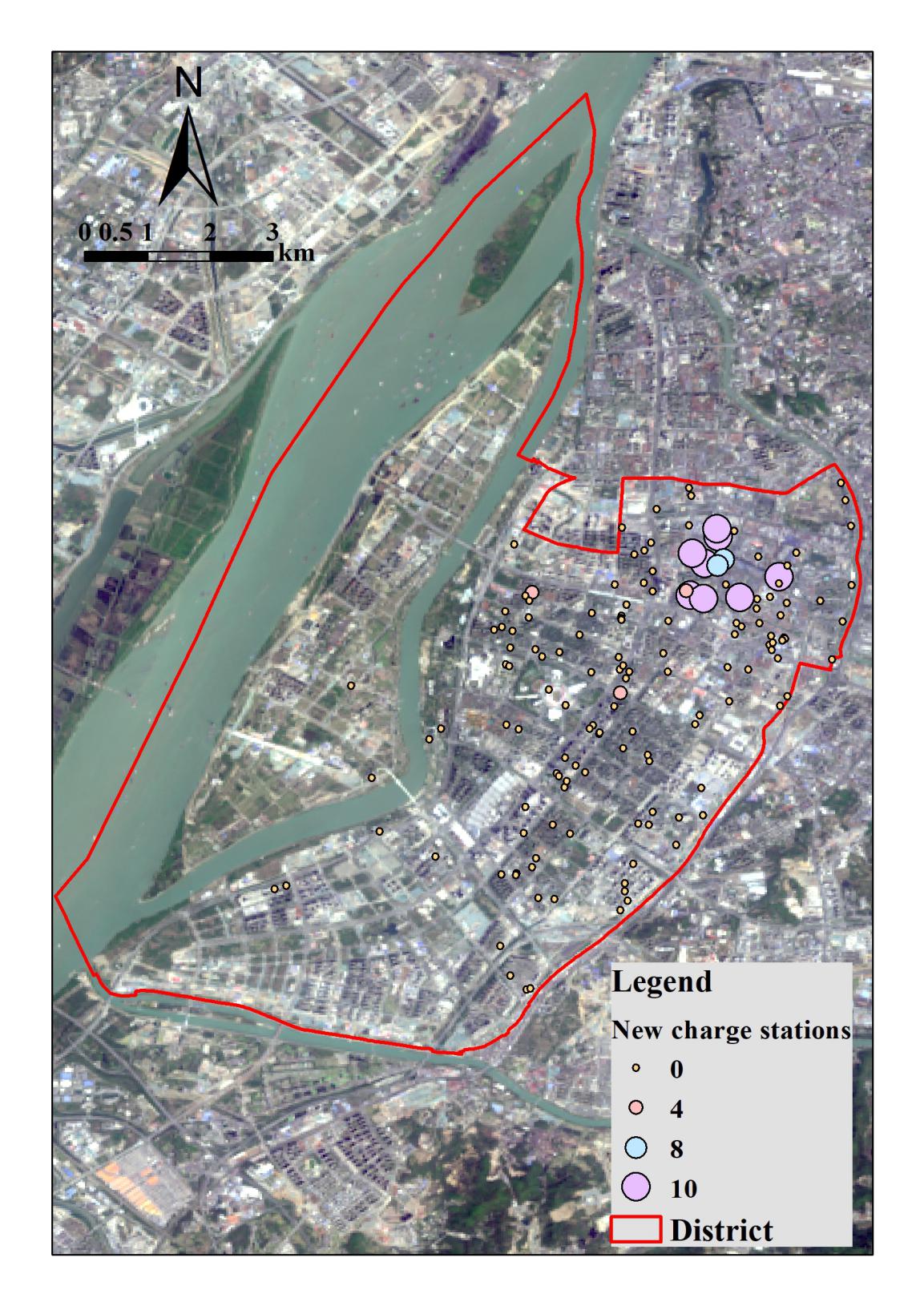


代码调整BuG之后



### 结果空间可视化；

第一版可视化结果，发现结果不太对，后来修改代码过程中发现了BUG，将不公平性的目标函数仍旧写为最大化目标。



### 增加多轮对比试验；

### 增加NSGA3的代码；

5个目标，采用NSGA3来做求解

### 增加量子优化算法；

### 重新运行结果

### 结果空间可视化；

## 1、定量评价指标

Objecteve count 2,3,4

Popsize=100,500,1000

Iteration=1000

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 测试函数 | 求解问题 | 分类 | 种群规模 | 迭代次数 | SS | 耗时 |
|  |  | A1 | 200 | 10 | -0.922 | 15 |
|  |  | A1 | 200 | 10 | 0.227 | 16 |
|  |  | A1 | 500 | 20 | 0.975 | 102 |
|  |  | A1 | 500 | 20 | 0.928 | 104 |
|  |  | A1 | 500 | 40 | 0.865 | 207 |
|  |  | A1 | 500 | 40 | 0.849 | 209 |
|  |  | A1 | 1000 | 20 | 0.890 | 308 |
|  |  | A1 | 1000 | 20 | 0.968 | 303 |
|  |  | A1 | 200 | 20 | 0.140 | 89 |
|  |  | A1 | 200 | 20 | -0.183 | 88 |
|  |  | A1 | 500 | 100 | 0.095 | 900 |
|  |  | A1 | 500 | 100 | 0.0617 | 893 |
|  |  | A1 | 1000 | 100 | 0.0385 | 2389 |
|  |  | A1 | 1000 | 100 | 0.109 | 2374 |
|  |  | A1 | 5000 | 1000 |  |  |
|  |  | A1 | 5000 | 1000 |  |  |