

# 《时空数据分析与挖掘实习》

## 实习报告

学    院： 遥感信息工程学院

班    级： 20F10

学    号： 2020302131201

姓    名： 常耀文

实习地点： 教学实验大楼 1-102

指导教师： 杨宗亮

2023 年    5    月    22    日

## 目录

1 实习目的与意义 .....	3
1.1 实习目的 .....	3
1.2 实习意义 .....	3
2 实习操作软件概述 .....	3
2.1 ArcGIS 平台 .....	3
2.2 PyCharm 平台 .....	4
3 实习内容 .....	4
3.1 中国人口分布特点研究 .....	4
3.1.1 实验目的 .....	4
3.1.2 时空数据获取与处理 .....	5
3.1.3 时空数据分析与挖掘 .....	12
3.2 华盛顿犯罪分析 .....	19
3.2.1 实验目的 .....	19
3.2.2 时空数据获取与处理 .....	19
3.2.3 时空数据分析与挖掘 .....	22
5 实习总结与体会 .....	29
5.1 实习总结 .....	29
5.2 实习体会 .....	29

## 1 实习目的与意义

### 1.1 实习目的

本次实习为 2023-2024 年《时空数据分析与挖掘》课程的集中综合实习，实习时间为 2023. 5. 13 至 2023. 5. 20。

实习基于 Python 与 ArcGIS，利用数据进行相关处理并且进行有关规则的寻找和探索，进行时空的分析与挖掘并且可视化显示，实验 1 通过对中国人口分布情况进行分析，找出人口分布的特点，以此为经济、商业活动以及政府决策提供参考。实验 2 通过对华盛顿的犯罪事件进行分析与探究，对于时空数据分析与挖掘的过程有了大致的了解，对所学的知识有了更加深刻的认识。

本次实习主要分为 2 部分，总结如下图所示：

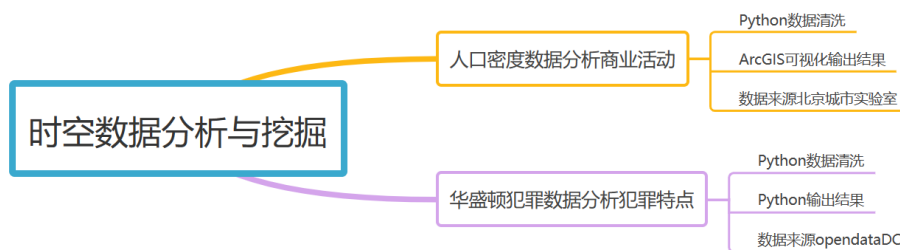


图 1 《时空数据分析与挖掘》综合实习任务概况

### 1.2 实习意义

通过对于实习内容的分析与实际的操作流程，总结了本次实习的意义主要有以下 5 点：

- ① 将《时空数据分析与挖掘》的课堂理论与实践相结合，深入掌握数据分析与数据挖掘的基本概念和原理；
- ② 加强数据处理和数据挖掘的基本技能训练，培养学生分析问题和解决问题的能力；
- ③ 让学生深入实际实践课堂所学知识，同时将课堂所学知识运用到实际生产过程中。
- ④ 培养具有严格的科学思维，具备较强的计算机等现代工具应用能力，能够综合运用数学、自然科学、工程基础和专业知识，分析并解决遥感科学与技术相关领域的科学与工程问题的遥感学子。
- ⑤ 培养学生思考问题的能力，能够利用所学知识分析实际问题，开阔个人视野

## 2 实习操作软件概述

### 2.1 ArcGIS 平台

ArcGIS 是一个全面的用于收集、组织、管理、分析、交流和发布地理信息的系统。作为世界领先的地理信息系统（Geographic Information System, GIS）构建和应用平台，由美国 ESRI 公司开发的 ArcGIS 可供全世界的人们将地理知识 应用到政府、企业、科技、教育和媒体等领域。ArcGIS 可以发布地理信息，以便所有人都可以在任何地点通过 web 浏览器、移动设备（例如智能手机和台式计算机）来使用。



图 2 ArcGIS 概念图

## 2.2 PyCharm 平台

---

PyCharm 是一种 Python IDE (Integrated Development Environment, 集成开发环境), 带有一整套可以帮助用户在使用 Python 语言开发时提高其效率的工具, 比如调试、语法高亮、项目管理、代码跳转、智能提示、自动完成、单元测试、版本控制。此外, 该 IDE 提供了一些高级功能, 以用于支持 Django 框架下的专业 Web 开发。



图 3 PyCharm 概念图

## 3 实习内容

本次实习选择了两个任务进行分析与挖掘, 因此实习内容分别包括两项实验内容, 第一项实验内容是对于中国城市人口分布情况进行分析, 找出中国人口分布的特点, 并且以此为经济, 商业活动和政府决策提供参考, 第二项任务是使用华盛顿犯罪数据进行时序分析与犯罪预测, 找出华盛顿犯罪的规律与问题, 并且可视化输出, 本次实习均将从**时空数据获取及处理, 时空数据分析与挖掘, 时空数据可视化展示**等方面对实习进行介绍, 下面将详细阐释各部分内容。

### 3.1 中国人口分布特点研究

---

#### 3.1.1 实验目的

通过对中国人口分布情况进行分析, 找出人口分布的特点, 以此为经济、商业活动以及政府决策提供参考。

### 3.1.2 时空数据获取与处理

#### (1) 数据来源

本次实习的使用数据来源于北京城市实验室，网址如后链接：[\(20 Expanded parcels during 2012-2017 by MVP-CA - Beijing City Lab\)](#)，进入网址后选择如下图所示的位置进行下载数据与相关论文。



图 4 数据来源图

#### (2) 数据格式

将数据下载后，打开压缩文件夹，发现所给的数据格式为 shapefile 格式，具体展示如图所示：

ChinaDensity300dpi	2014/1/30 23:44	JPG 文件	1,495 KB
Data descriptions	2015/8/27 14:17	文本文档	3 KB
PopCensus2010_township.dbf	2015/8/27 14:16	DBF 文件	6,973 KB
PopCensus2010_township.prj	2015/8/25 13:54	PRJ 文件	1 KB
PopCensus2010_township.sbn	2015/8/25 13:54	SBN 文件	420 KB
PopCensus2010_township.sbx	2015/8/25 13:54	SBX 文件	19 KB
PopCensus2010_township.shp	2015/8/25 13:54	SHP 文件	1,191 KB
PopCensus2010_township.shp	2015/8/25 13:54	Microsoft Edge HT...	11 KB
PopCensus2010_township.shx	2015/8/25 13:54	SHX 文件	341 KB

图 5 数据格式图

通过过往使用数据的经验，决定使用 ArcGIS 打开这样的 shapefile 格式的文件，于是利用 ArcGIS 打开这样的点状 shp 数据，数据共有 **43536 条**。每一个数据对应一个乡镇街道，包含有以下字段：ID，总人口数，男性数，女性数，15 岁以下人口数，15 岁到 65 岁人口数，65 岁以上人口数，地址和一个空间点坐标，具体如图 6 所示：

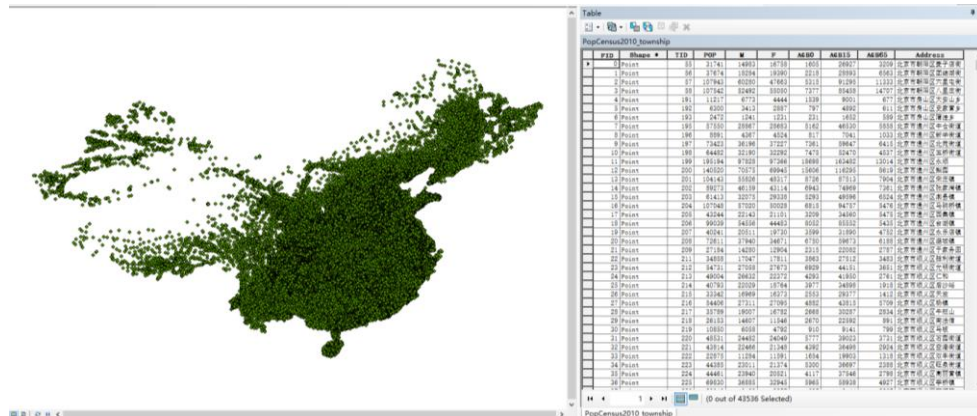


图 6 ArcGIS 可视化显示数据

### (3) 数据补全

#### ➤ 乡镇街道政区边界数据

北京城市实验室仅提供了 2010 年普查对应的乡镇街道单元点数据，要生成密度图以及制作其他有完整行政区边界的图至少需要乡镇街道政区边界数据，因此全国乡镇街道的矢量数据通过网络资料的查找在（[麻辣 GIS 公众号](#)）获取，共有 **43655 条数据**，此数据表现为多边形数据，每个多边形包含省县乡三个字段，格式为 shp 文件，具体情况如下图所示：

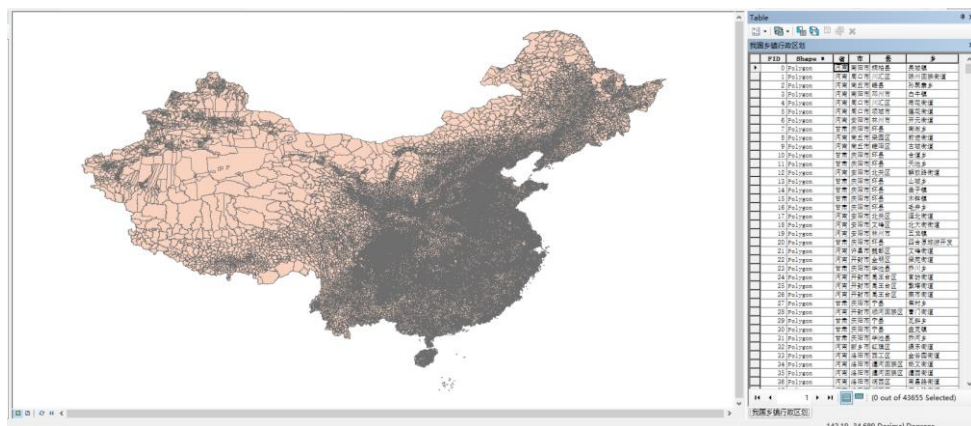


图 7 ArcGIS 可视化乡镇街道政区边界数据

#### ➤ 中国行政区划数据

北京城市实验室仅提供了 2010 年普查对应的乡镇街道单元点数据，要生成密度图以及制作其他有完整行政区边界的图至少需要行政区划数据，中国 1 级、2 级行政区划数据来源于网络搜索。网上这种行政级别较高的数据相对易于寻找，最后使用了一个外国网站上提供的数据，数据年份为 2020 年。此数据表现为多边形数据，每个多边形中有多个字段，其中包括国家、省、市等信息，格式为 shp 文件，具体如图所示。



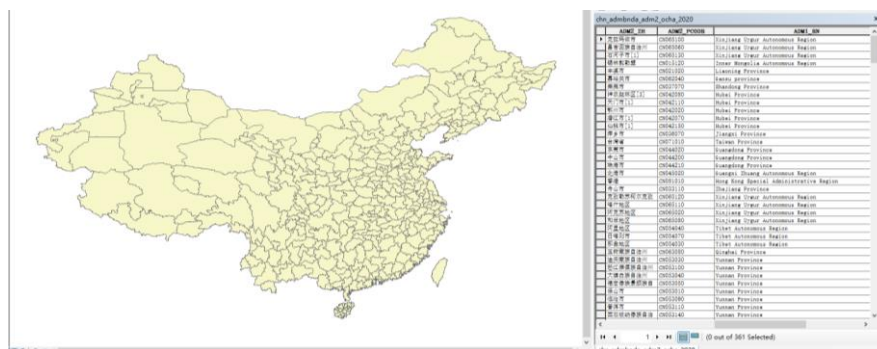


图 7 ArcGIS 可视化中国 1, 2 级行政区划数据

#### (4) 数据处理技术理论

##### (一) 空间连接

空间连接是指根据要素的相对空间位置将连接要素中的行匹配到目标要素中的行。它根据空间关系将一个要素类的属性连接到另一个要素类的属性。目标要素和来自连接要素的被连接属性写入到输出要素类。

##### (二) 密度聚类

密度聚类亦称"基于密度的聚类" (density-based clustering)，此类算法假设聚类结构能通过样本分布的紧密程度确定。通常情形下，密度聚类算法从样本密度的角度来考察样本之间的可连接性，并基于可连接样本不断扩展聚类簇以获得最终的聚类结果。

##### (三) DBSCAN

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) 是一种著名的密度聚类算法，它基于一组"邻域" (neighborhood) 参数 ( $\epsilon$ ,  $MinPts$ ) 来刻画样本分布的紧密程度。 $\epsilon$  (eps) 邻域，对  $x_j \in D$ ，其  $\epsilon$ -邻域包含样本集  $D$  与  $x_j$  的距离不大于  $\epsilon$  的样本，即

$$N_{\epsilon}(x_j) = \{x_i \in D | dist(x_i, x_j) \leq \epsilon\}$$

1. DBSCAN 相关术语:
2. 核心对象 (core object): 若  $x_j$  的  $\epsilon$ -邻域至少包含  $MinPts$  个样本，则  $x_j$  是一个核心对象。
3. 噪声 (noise): 样本集  $D$  中不属于任何簇的样本被认为是噪声 (noise) 或异常 (anomaly) 样本。
4. DBSCAN 将"簇"定义为: 由密度可达关系导出的最大的密度相连样本集合。
5. DBSCAN 算法先选数据集中的一个核心对象为"种子" (seed)，再由此出发确定相应的聚类簇。算法先根据给定的邻域参数 ( $\epsilon$ ,  $MinPts$ ) 找出所有核心对象；然后以任一核心对象为出发点，找出由其密度可达的样本生成聚类簇，直到所有核心对象均被访问过为止。

#### (5) 数据处理方法

由于数据格式为 shp 文件，因此本次数据的处理需要考虑到对于 shp 文件的处理，对于 shp 文件进行处理我们需要考虑到对于 shp 数据的处理和分析，因此需要引入下列的包或者库进行实践分析：

- ① **Numpy**: NumPy (Numerical Python) 是 Python 的一种开源的数值计算扩展。这种工具可用来存储和处理大型矩阵，比 Python 自身的嵌套列表 (nested list structure) 结构要高效的多 (该结构也可以用来表示矩阵 (matrix))，支持大量的维度数组与矩阵运算，此外也针对数组运算提供大量的数学函数库
- ② **Pandas**: pandas 是基于 NumPy 的一种工具，该工具是为解决数据分析任务而创建的。Pandas 纳入了大量库和一些标准的数据模型，提供了高效地操作大型数据集所需的工具。pandas 提供了大量能使我们快速便捷地处理数据的函数和方法。你很快就会发现，

它是使 Python 成为强大而高效的数据分析环境的重要因素之一。

- ③ **Matplotlib:** 是一个 Python 的 2D 绘图库，它以各种硬拷贝格式和跨平台的交互式环境生成出版质量级别的图形。
- ④ **Gdal:** GDAL(Geospatial Data Abstraction Library)是一个在 X/MIT 许可协议下的开源栅格空间数据转换库。它利用抽象数据模型来表达所支持的各种文件格式。它还有一系列命令行工具来进行数据转换和处理
- ⑤ **Geopandas:** GeoPandas 是一个开源项目，可以更轻松地使用 python 处理地理空间数据。GeoPandas 扩展了 Pandas 中使用的数据类型 DataFrame，允许对几何类型进行空间操作。GeoPandas 的目标是使在 python 中使用地理空间数据更容易。它结合了 Pandas 和 Shapely 的能力，提供了 Pandas 的地理空间操作和多种 Shapely 的高级接口。

GeoPandas 可以让您轻松地在 python 中进行操作，否则将需要空间数据库，如 PostGIS。利用上述 Python 包可以实现对于数据的处理分析，并且支持数据的可视化输出。下图为配置环境的版本信息。

```
SYSTEM INFO
-----
python      : 3.10.2 (tags/v3.10.2:a58ebcc, Jan 17 2022, 14:12:15) [MSC v.1929 64 bit (AMD64)]
executable  : D:\myCode\python\data_mining\venv\scripts\python.exe
machine     : Windows-10-0.22000-SP0

GEOS, GDAL, PROJ INFO
-----
GEOS        : None
GEOS lib    : None
GDAL        : 3.4.2
GDAL data dir: None
PROJ        : 8.2.1
PROJ data dir: D:\myCode\python\data_mining\venv\lib\site-packages\pyproj\proj_dir\share\proj

PYTHON DEPENDENCIES
-----
geopandas   : 0.10.2
pandas      : 1.4.2
fiona       : 1.8.21
numpy       : 1.22.3
shapely     : 1.8.1.post1
rtree       : None
pyproj      : 3.3.1
matplotlib : 3.5.2
lapclassify: None
geopy       : None
psycopg2    : None
gealchemy2  : None
pyarrow     : None
pygeos      : None
None
```

图 8 GeoPandas 各种依赖的安装结果

- ① **将行政区划数据和人口数据连接:** 将街道单元点数据和行政边界数据进行空间连接以获取结果。连接后统计街道 单元点数据在行政边界数据上的重复情况，保证在后续的统计中可以实现统计行政区划的人口密度数据，具体实现代码如下所示：

```
1. import geopandas
2. import pandas
3. import matplotlib.pyplot as plt
4.
5. pandas.set_option('display.max_columns', None)
6. pandas.set_option('display.width', 1000)
7. # pandas.set_option('display.max_colwidth', 500)
8.
9. boundary_data = geopandas.read_file('./我国乡镇行政区划/我国乡镇行政区划.shp')
10. # print(boundary_data.head())
11. # print(boundary_data.crs)
12. # print(len(boundary_data))
13. point_data = geopandas.read_file('./DT19new/PopCensus2010_township.shp')
14. # print(point_data.head())
```



```
15. # print(point_data.crs)
16. # print(len(point_data))
17.
18. # result = geopandas.sjoin(point_data, boundary_data)
19. # print(result.head())
20. # print(len(result))
21. # result = result.duplicated(subset='index_right').sum()
22. # print(result)
23. #
24. # result = geopandas.sjoin(boundary_data, point_data)
25. # print(len(result))
26. #
27. # result = result.duplicated(subset='index_right').sum()
28. # print(result)
29.
30. result = geopandas.sjoin(boundary_data, point_data, how='left')
31. print(result.head(20))
32. print(len(result))
```

利用上述代码，可以获得街道单元点数据与行政边界数据连接的结果，结果展示如下图所示：

	省	市	县	乡	geometry	index_right	TID	POP	M	F	AGE0	AGE15	AGE65	Address	
0	河南省	南阳市	桐柏县	吴城镇	POLYGON ((113.58872 32.43346, 113.58846 32.426...		20964.0	22626.0	24958.0	13011.0	11947.0	5962.0	16986.0	2090.0	河南省南阳市桐柏县吴城镇
1	河南省	周口市	川汇区	陈州镇	POLYGON ((114.64966 33.62388, 114.65080 33.619...		21619.0	23105.0	16429.0	8255.0	8174.0	3861.0	11424.0	1124.0	河南省周口市川汇区陈州镇
2	河南省	商丘市	睢县	陈果集乡	POLYGON ((115.20239 34.28866, 115.21044 34.297...		21075.0	22743.0	38626.0	19543.0	19083.0	7653.0	27322.0	3521.0	河南省商丘市睢县陈果集乡
3	河南省	商丘市	夏邑县	白道口乡	POLYGON ((112.22179 32.48472, 112.21518 32.484...		20995.0	22638.0	39318.0	26434.0	18982.0	8218.0	27478.0	3642.0	河南省商丘市白道口乡
4	河南省	周口市	川汇区	陈州镇	POLYGON ((114.64870 33.61342, 114.61615 33.605...		21416.0	23182.0	70214.0	35376.0	35038.0	11027.0	52637.0	4750.0	河南省周口市川汇区陈州镇
5	河南省	周口市	川汇区	陈州镇	POLYGON ((114.64870 33.61342, 114.61615 33.605...		21417.0	23183.0	28922.0	14292.0	14630.0	4971.0	21226.0	2725.0	河南省周口市川汇区人和街道
6	河南省	周口市	川汇区	陈州镇	POLYGON ((114.64870 33.61342, 114.61615 33.605...		21414.0	23180.0	65542.0	22853.0	22689.0	8081.0	34241.0	3308.0	河南省周口市川汇区陈州镇
7	河南省	周口市	商水县	范店乡	POLYGON ((114.95150 33.38884, 114.95451 33.386...		NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
8	河南省	安阳市	林州市	开元街道	POLYGON ((113.81201 36.06553, 113.80594 36.065...		20129.0	21721.0	78286.0	29359.0	40927.0	14636.0	49818.0	5840.0	河南省林州市城郊乡
9	河南省	安阳市	林州市	开元街道	POLYGON ((113.81201 36.06553, 113.80594 36.065...		20112.0	21704.0	42847.0	21799.0	21848.0	7948.0	35046.0	1831.0	河南省林州市开元街道
10	河南省	安阳市	林州市	开元街道	POLYGON ((113.81201 36.06553, 113.80594 36.065...		40848.0	43963.0	4272.0	2231.0	2041.0	798.0	3175.0	299.0	河南省林州市城郊乡
11	河南省	商丘市	睢县	陈果集乡	POLYGON ((115.64947 34.43188, 115.61922 34.427...		21087.0	22672.0	31697.0	15617.0	16080.0	6864.0	23718.0	3323.0	河南省商丘市睢县陈果集乡
12	河南省	商丘市	睢县	陈果集乡	POLYGON ((115.62093 34.37129, 115.61945 34.371...		21027.0	22693.0	42635.0	21133.0	21582.0	6564.0	31728.0	4349.0	河南省商丘市睢县陈果集乡
13	河南省	安阳市	林州市	开元街道	POLYGON ((114.33948 36.10163, 114.33947 36.101...		40861.0	43936.0	18059.0	9285.0	8774.0	3229.0	12936.0	1894.0	河南省安阳市林州市城郊乡
14	河南省	安阳市	林州市	开元街道	POLYGON ((114.33948 36.10163, 114.33947 36.101...		40859.0	43934.0	17187.0	8967.0	8220.0	3247.0	12491.0	1449.0	河南省安阳市林州市城郊乡
15	河南省	安阳市	林州市	开元街道	POLYGON ((114.33948 36.10163, 114.33947 36.101...		20821.0	21686.0	12929.0	6512.0	6417.0	2429.0	9726.0	774.0	河南省安阳市林州市城郊乡
16	河南省	安阳市	林州市	开元街道	POLYGON ((114.33948 36.10163, 114.33947 36.101...		40867.0	43962.0	7843.0	4118.0	3725.0	1449.0	5804.0	590.0	河南省安阳市林州市城郊乡
17	河南省	安阳市	林州市	开元街道	POLYGON ((114.33948 36.10163, 114.33947 36.101...		40866.0	43931.0	28322.0	11879.0	11947.0	4228.0	17195.0	1999.0	河南省安阳市林州市城郊乡
18	河南省	安阳市	林州市	开元街道	POLYGON ((114.33948 36.10163, 114.33947 36.101...		40858.0	43933.0	18074.0	9333.0	8921.0	3188.0	13165.0	1621.0	河南省安阳市林州市城郊乡
19	河南省	安阳市	林州市	开元街道	POLYGON ((114.33948 36.10163, 114.33947 36.101...		40873.0	43948.0	11878.0	6175.0	5712.0	1949.0	8917.0	1002.0	河南省安阳市林州市城郊乡

图 9 空间连接结果

从数据统计结果上可以看出，街道点数据有 43536-43531=5 个点未找到对应的行政区划，比例很小，问题不大；有 3928 个点重复落在了已有数据的行政区划上，比例不到 10%，也可以接受；对于行政区划来说，总共只有 43536-3928=39608 个区划找到了对应点，有 43655-39608=4047 个区划没有对应点，导致在绘图时会出现不少数值为 0 的间断区划。

U:\myCode\python\data_mining\venv\scripts\python.exe; U:\myCode\python\data_mining\data_mining.py														
	TID	POP	M	F	AGE0	AGE15	AGE65	Address	geometry	index_right	省	市	县	乡
0	55	31741	14983	16758	1685	26927	3209	北京市朝阳区麦子店街道	POINT (116.46299 39.93407)	31891	北京市	北京市	朝阳区	麦子店街道
1	56	37674	18284	19390	2218	28893	6563	北京市朝阳区团结湖街道	POINT (116.46063 39.93061)	31518	北京市	北京市	朝阳区	团结湖街道
2	57	107943	60280	47663	5315	91295	11333	北京市朝阳区六里屯街道	POINT (116.47341 39.92800)	22229	北京市	北京市	朝阳区	六里屯街道
71	100	83454	41203	42251	5993	66371	11090	北京市丰台区方庄	POINT (116.49558 39.92139)	22229	北京市	北京市	朝阳区	六里屯街道
3	58	107542	52492	58050	7377	85458	14707	北京市朝阳区八里庄街道	POINT (116.48993 39.91547)	31413	北京市	北京市	朝阳区	八里庄街道
43531														
3928														

图 10 数据统计结果

- ② **重新连接数据**:之后重新进行连接，保留所有行政区划数据对街道单元点数据进行连接，提取出人口相关信息并基于行政区划数据的索引进行分组后求和，然后基于行政区划数据的索引与行政区划数据进行连接，得到每个地区的人口相关数据，代码如下所示：

```
1. result = result[['POP', 'M', 'F', 'AGE0', 'AGE15', 'AGE65']]
```

```
2. print(result.head(10))
3. result = result.groupby(level=0).sum()
4. print(result.head(10))
5. print(len(result))
```

运行上述代码，发现重新连接后的数据量由 47583 变为了 43655，运行情况如下所示：

	POP	M	F	AGE0	AGE15	AGE65
0	24958.0	13011.0	11947.0	5962.0	16906.0	2090.0
1	16429.0	8255.0	8174.0	3881.0	11424.0	1124.0
2	38626.0	19543.0	19083.0	7683.0	27322.0	3621.0
3	39338.0	20436.0	18902.0	8218.0	27478.0	3642.0
4	144678.0	72321.0	72357.0	25999.0	107904.0	10775.0
5	0.0	0.0	0.0	0.0	0.0	0.0
6	113133.0	51158.0	61975.0	22584.0	82856.0	7693.0
7	4272.0	2231.0	2041.0	798.0	3175.0	299.0
8	31697.0	15617.0	16080.0	6064.0	23710.0	1923.0
9	42635.0	21133.0	21502.0	6566.0	31720.0	4349.0
43655						

图 11 数据重新连接结果

- ③ **计算每个行政区的面积并且输出结果：**使用 geopandas 提供的函数 area 可以快速实现面积计算，需要注意的是计算前需要修改一下投影坐标系，我使用的是西安 80 坐标系高斯克吕格投影 108E (epsg: 2381)，不然面积会使用经纬度数据进行计算，结果没有意义。同时面积的结果单位为平方米，需要除以 1000000 将单位转为平方千米。代码如下所示：

```
1. result = boundary_data.merge(result, left_index=True, right_index=True)
2. print(result.head())
3. print(len(result))
4.
5. result['area'] = result.to_crs(2381).area / (10 ** 6)
6. result['POP/area'] = result['POP'] / result['area']
7. result['M/area'] = result['M'] / result['area']
8. result['F/area'] = result['F'] / result['area']
9. result['AGE0/area'] = result['AGE0'] / result['area']
10. result['AGE15/area'] = result['AGE15'] / result['area']
11. result['AGE65/area'] = result['AGE65'] / result['area']
```

输出结果为 shp 文件并且使用 utf-8 编码。

```
1. result.to_file('./output/result.shp', encoding='utf8')
```

- ④ **对于数据进行空间聚类分析。**考虑到数据本身为街道单元点，因此首先采用 DBSCAN 对数据进行处理，首先要对数据进行读取并且将坐标系转换，数据本身基于 WGS84 坐标系，坐标值均为经纬度，不能很好的反映两点间距离，因此将坐标系转换为西安 80 坐标系高斯克吕格投影 108E (epsg: 2381)。具体代码如下所示：

1. 数据读取和坐标系转换:

```
2. data = geopandas.read_file('./DT19new/PopCensus2010_township.shp').to_crs(2381)
```

- ⑤ 筛选出有价值的点。由于很多街道单元点人口很少,对人口分布分析价值很低,因此首先筛选出有价值的街道单元点数据。计算数据的各项指标,发现均值为 30650 人,中位数为 21975 人,四分位数的 Q3 为 39531 人。衡量各项指标后决定使用接近 Q3 值得 40000 人作为筛选标准,选出人口大于 40000 的街道单元点作为待处理数据,具体代码如下:

```
1. data = data[data['POP'] >= 40000]
2. geom_data = data['geometry']
3. dataframe 转换为 list 后转为 array-like:
4. def point_to_list(point_list):
5.     output_list = []
6.     for point in point_list:
7.         output_list.append([point.x, point.y])
8. return output_list
9.
10. result = geom_data.to_list()
11. result = point_to_list(result)
12. result = numpy.array(result)
```

- ⑥ 利用 DBSCAN 进行初始化并且预测转换结果为 dataframe 并且重建索引,将数据转换为 Geodataframe 并且修改坐标系,代码如下所示:

```
1. # DBSCAN 初始化并进行预测:
2. y_pred = DBSCAN(eps=40000, min_samples=10)
3. y_pred = y_pred.fit_predict(result)
4. # 结果转为 dataframe:
5. pred_label = pandas.DataFrame(y_pred, columns=['label'])
6. # 对筛选后结果重建索引:
7. geom_data = geom_data.reset_index(drop=True)
8. # 将 series 转为 dataframe:
9. geom_data = pandas.DataFrame(geom_data)
10. # 数据合并:
11. geom_data['label'] = pred_label['label']
12. # 结果保存 (需要先转为 geodataframe, 之后顺带改一下坐标系):
13. geom_data = geopandas.GeoDataFrame(geom_data)
14. geom_data = geom_data.to_crs(4326)
15. geom_data.to_file('./output/dbscan.shp', encoding='utf8')
```

运行代码得到以下的实验结果:

	省	市	县	乡	geometry	POP	M	F	AGE0	AGE15	AGE65	area	POP/area	M/area	F/area	AGE0/area
0	河南省	南阳市	桐柏县	吴城镇	POLYGON ((113.58072 32.43346, 113.58846 32.426...	24958.0	13011.0	11947.0	5962.0	16906.0	2090.0	146.900819	169.896943	88.569962	81.326980	40.58
1	河南省	周口市	川汇区	陈州回族街道	POLYGON ((114.64966 33.62388, 114.65080 33.619...	16429.0	8255.0	8174.0	3881.0	11424.0	1124.0	2.826599	5812.284783	2920.478523	2891.814188	1373.6
2	河南省	商丘市	睢县	孙家滩乡	POLYGON ((115.02839 34.28866, 115.01844 34.287...	38626.0	19543.0	19083.0	7683.0	27322.0	3621.0	53.380662	723.595374	366.106364	357.489011	143.92
3	河南省	南阳市	邓州市	白牛镇	POLYGON ((112.22179 32.68472, 112.21510 32.684...	39338.0	20436.0	18902.0	8218.0	27478.0	3642.0	69.627624	564.976914	293.504200	271.472714	118.02
4	河南省	周口市	川汇区	荷花街道	POLYGON ((114.64070 33.61342, 114.63615 33.605...	144678.0	72321.0	72357.0	25999.0	107904.0	10775.0	7.995354	18095.258769	9045.378077	9049.880692	3251.76

Traceback (most recent call last):

图 12 DBSCAN 聚类运行结果

至此，实验一的初步数据准备完成，接下来进入实验一的挖掘与分析过程

### 3.1.3 时空数据分析与挖掘

#### 3.1.3.1 中国人口密度图分析

##### (1) 中国总人口密度图生成与分析

将数据导入到 arcGIS, 修改显示方式和分段值与示例中得相同, 获得如下图所示结果:

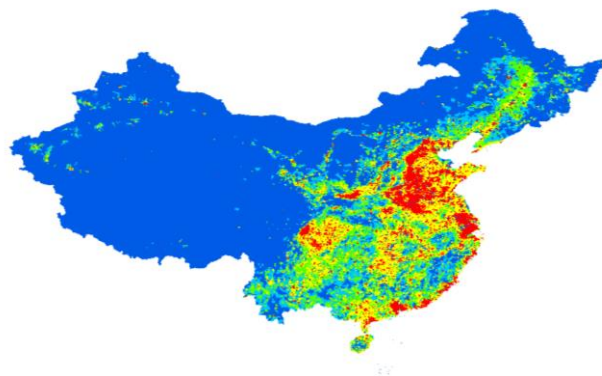


图 13 ArcGIS 可视化制图输出中国总人口密度图

经过添加制图图例后, 可以获得如下图所示的制图结果:

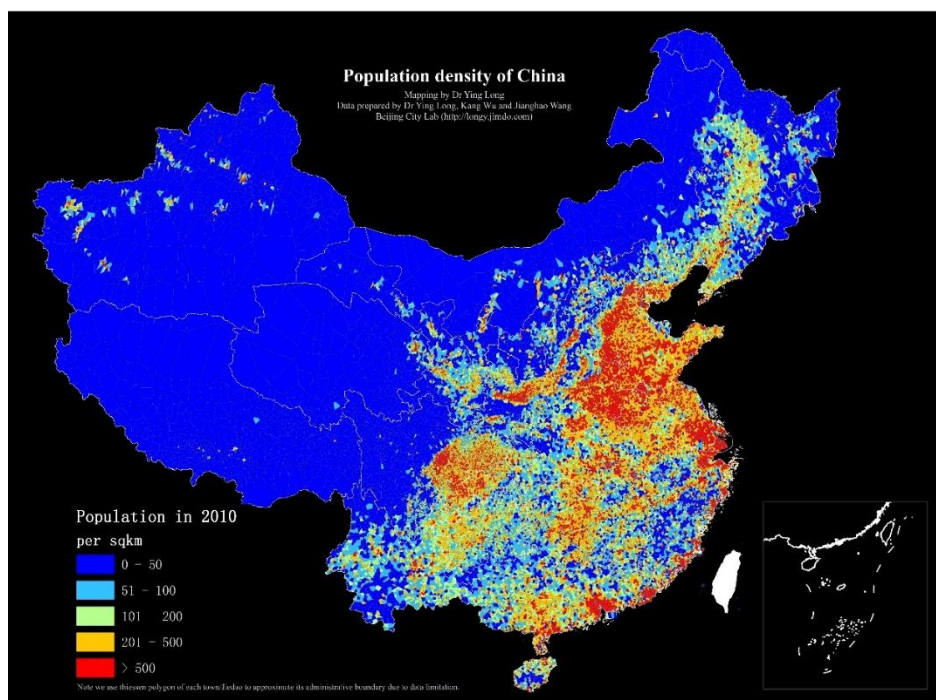


图 14 添加图例后的制图结果

##### ● 分析结论

从上述的中国总人口密度图分析中国人群的聚集程度, 我们不难发现: 中国人口主要分布在黑河腾冲线以南, 其中以北京以南到河南以及长江三角洲一带为人口核心区; 同时存在若干个次级核心, 如珠江三角洲附近, 东南沿海一带, 湖南湖北, 以及四川重庆等地区。人

口呈现高聚集性特点，分布不均匀。这与我国的经济程度密切相关，发达地区由于经济状况发达，会吸引务工人员源源不断地前往，因此密度会逐渐增加，聚集性强。

## (2) 中国分性别人口密度图生成与分析

转换数据为中国男性的与中国女性，可视化人口密度图并且输出中国男女性分性别展示图如下所示：

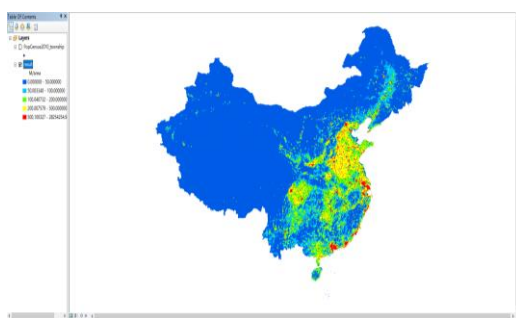


图 15 中国男性人口分布图

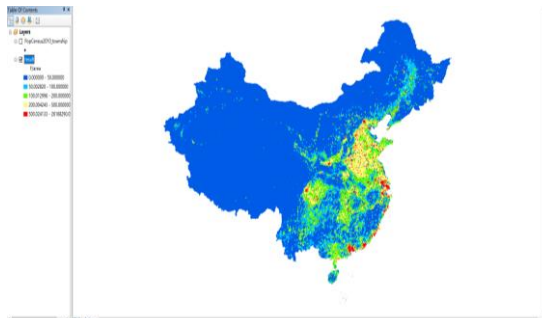


图 16 中国女性人口分布图

然后将男女性人口做差，得出以下的男女性人口分布情况图：

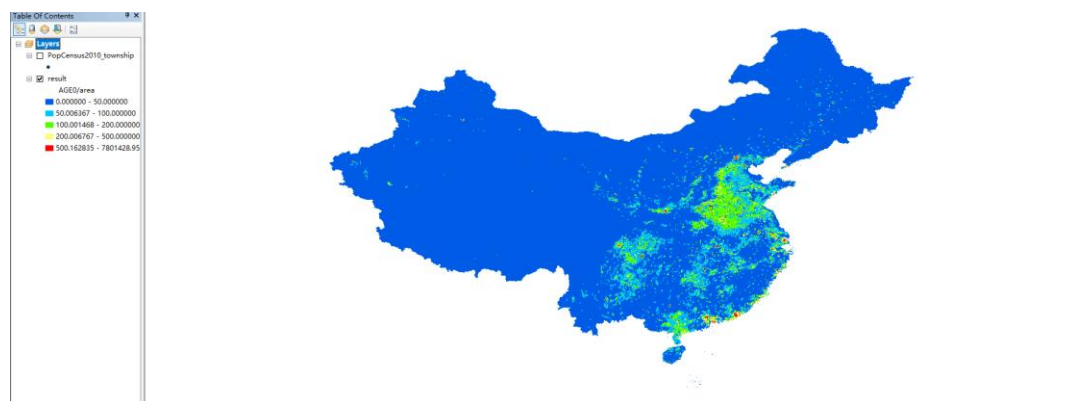


图 17 中国男女性对比分析图

### ● 分析结论

- ① 全国男性人口分布总体上与总人口分布基本相同，但在珠江三角洲和长江三角洲分布的相对密度比总人口要高不少，集聚效应更为明显
- ② 全国女性人口分布总体上和男性人口分布基本相同，没有明显区别，仅人口总数略少，导致局部聚集区相对人数较少。
- ③ 对于全国男女性人口做差后的结果进行分析，我们不难发现，少部分区域呈现聚集效应，而在华北地差量较大，造成这样的原因，通过查阅资料可以发现，排名第一的是北京，天津市紧随其后，分析原因可能是由于以下几点：



《时空数据分析与挖掘实习》  
2020302131201-常耀文

地区	比重		性别比
	男	女	
全 国	51.24	48.76	105.07
北 京	51.14	48.86	104.65
天 津	51.53	48.47	106.31
河 北	50.50	49.50	102.02
山 西	50.99	49.01	104.06
内 蒙 古	51.04	48.96	104.26
辽 宁	49.92	50.08	99.70
吉 林	49.92	50.08	99.69
黑 龙 江	50.09	49.91	100.35
上 海	51.77	48.23	107.33
江 苏	50.78	49.22	103.15
浙 江	52.16	47.84	109.04
安 徽	50.97	49.03	103.94
福 建	51.68	48.32	106.94
江 西	51.60	48.40	106.62
山 东	50.66	49.34	102.67
河 南	50.15	49.85	100.60
湖 北	51.42	48.58	105.83
湖 南	51.16	48.84	104.77
广 东	53.07	46.93	113.08
广 西	51.70	48.30	107.04
海 南	53.02	46.98	112.86
重 庆	50.55	49.45	102.21
四 川	50.54	49.46	102.19
贵 州	51.10	48.90	104.50
云 南	51.73	48.27	107.16
西 藏	52.45	47.55	110.32
陕 西	51.17	48.83	104.79
甘 肃	50.76	49.24	103.10
青 海	51.21	48.79	104.97
宁 夏	50.94	49.06	103.83
新 疆	51.66	48.34	106.85

图 18 中国男女性对比分析图

- ① **一是受经济基础的制约。**在许多农村地区，由于生产力落后，生产和生活主要还是靠体力劳动，很多重体力劳动需要男性承担，人们把发家致富的希望都寄托于男性。同时，农村的社会养老保障体制尚不健全，农民有后顾之忧，这些是产生“男孩偏好”的经济基础。
- ② **二是受生育政策的导向。**我国现行生育政策规定，农村“夫妻只有独生女的”可以生育二胎，而二胎往往成为育龄夫妇生男孩的唯一合法机会，这在客观上造成了出生人口性别比的失衡，强化了男女不平等的思想。加之严格的计划生育政策导致的低生育率也是出生人口性别比升高的条件性原因。出生性别比失调的主要原因在于家庭生育的有计划性与国家生育控制的有计划性之间的矛盾。
- ③ **三是社会性别的不平等。**在现实社会生活中，社会性别不平等现象仍然存在。女性在受教育机会、就业机会、劳动分配、政治生活参与程度等方面与男性仍有一定差距，这也促使人们的生育意愿倾向男孩。
- ④ **四是科技手段的滥用。**在人类社会几千年的历史长河中，以前从来没有引发生人口性别比的明显失衡，直到近 20 多年才出现这一问题，这是因为以往人们尽管有多生男孩的意愿，但是却无力左右生男生女。20 世纪 80 年代以来，B 超、染色体技术用于胎儿性别检测之后，使生育选择性别有了可能性，出生人口性别比升高的问题才显现出来。从以上分析中不难看出，科技手段的滥用也是导致我国出生人口性别比失衡的直接原因。

总而言之，全国男女性人口分布密度与总人口分布密度大致趋同，但呈现个别区域明显聚集现象。但是男女性别比例在各个区域有着差异

### （3）中国分年龄人口密度分布图生成与分析

由于在数据字段中存在着年龄划分区间，15 岁，15-65 岁，65 岁以上，因此，衡量年龄人口在全国各区域的分布也是极为重要的一环，下图为中国各年龄段的人口分布。



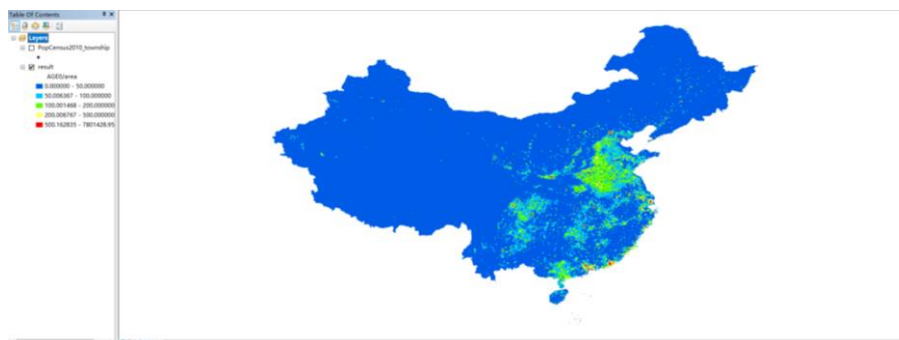


图 19 15 岁以下中国人口分布图

全国 15 岁以下人口分布与总人口分布在总体上趋于一致，但在各聚集区上存在不小的差异。其中最重要的核心区是北京以南，包括河北南部，河南，山东，以及安徽北部区域；同时存在若干个次级中心，如四川重庆地区，湖南，以及广东雷州半岛等区域。与总人口分布相比，最大的不同就是除特大城市附近，长江三角洲和珠江三角洲地区的青少年（15 岁以下）与总人口不成比例，15 岁以下人口偏少。

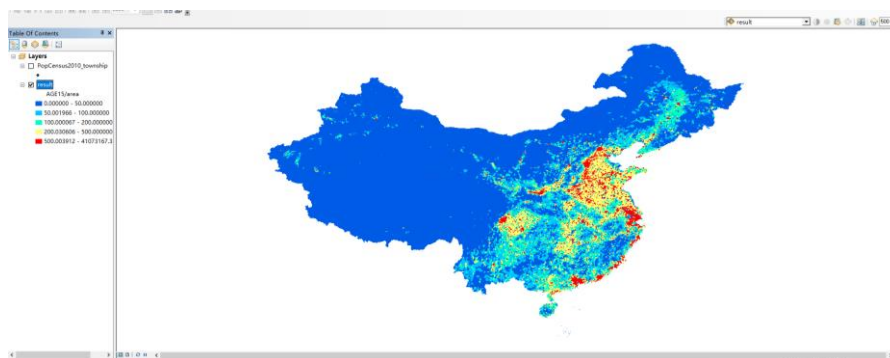


图 20 15-65 岁以下中国人口分布图

全国 15~65 岁人口分布与总人口分布高度近似，仅在少数几个地方存在差异：一是以北京以南到安徽、江浙一带的人口核心区在河南、山东等地的人口相对较少；二是长江三角洲、珠江三角洲附近人口相对较多；三是四川、重庆附近人口相对较少。

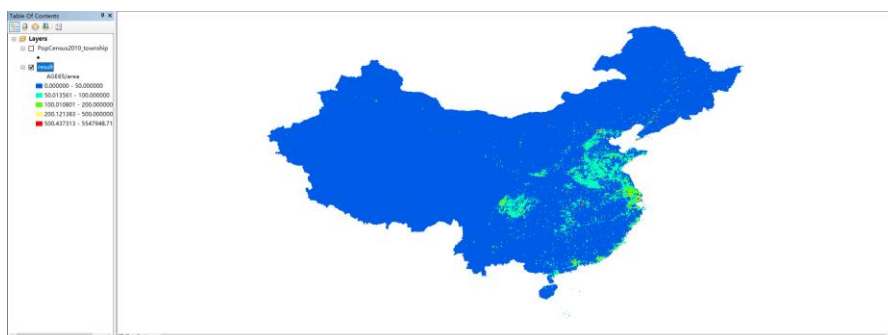


图 21 65 岁以上中国人口分布图

全国 65 岁以上人口与总人口分布比较相近，但在总人口密度较高的东南沿海区域，65 岁以上人口与总人口密度不成比例，总人口密度较高而 65 岁以上人口密度较低。

## ● 分析结论

- ① 不同年龄段的人口密度呈现区域性差异，青老年人群与总人口分布不同
- ② 人口分布与经济发展有着较大联系，经济发达区域劳动力需求较高，因此青壮年人口数量较多，具有较多的社会劳动力，因此 15-65 岁人口分布较为集中这些区域
- ③ 人口分布与教育有着极大关联，15 岁以下人口多集中于教育较为发达的区域，具有不错的受教育情况和背景。
- ④ 与平均死亡年龄与出生率有关，这两项指标会影响当地的老幼年龄段的人数，造成人口分布的不均匀
- ⑤ 与我国人口老龄化的发展现状有着较大关系：

（一）**人口老龄化的省际差异明显**：中度老龄化聚集区已经形成省份看，辽宁是我国人口老龄化程度最严重的省份，老年人口占比为 25.72%，已经进入到中度人口老龄化的后半段。黑龙江、吉林、天津、山东、江苏、上海、重庆、四川的老龄化程度都在 20% 以上，同处于中度人口老龄化阶段，并形成集中连片的中度老龄化聚集区，这些地区人口老龄化程度高与出生率较低或者年轻劳动力的流出有着密切的关系。西藏是人口老龄化程度最低的省份，2020 年仅为 8.52%。西藏人口老龄化程度低的原因有两方面，一是人均预期寿命低，2020 年西藏人均预期寿命已达 71.1 岁，比全国居民人均预期寿命要低 6 岁左右；二是生育水平高，2020 年西藏出生率达 14.6‰，排名全国第一，出生人口多降低了老年人口占比。

（二）**东北地区人口老龄化程度最高**，东、中、西部差距不大从东、中、西部和东北地区四大板块来看，东北地区是我国人口老龄化最为严重的地区。东北地区老年人口占比达到 24.26%，在近 1 亿的总人口中，老年人口占到 2300 多万，通常来说老年人口占比过大会加重社会负担，影响地区创新力。人口老龄化程度高可能是东北地区经济下滑的原因。排名第二的是中部地区，老年人口占比达到 18.83%。排名第三的是东部地区，老年人口占比达到 18.34%。从省级层面看，人口老龄化程度高的省份东部较多，但从地区层面看，人口老龄化程度则没有那么重，主要是广东等人口流入大省拉低了整个东部地区的人口老龄化程度，广东老年人口占比仅为 12.35%，低于全国 6.75 个百分点。西部地区人口老龄化程度最轻，老年人口占比达到 17.77%。整体来看，除了东北地区以外，东中西部地区人口老龄化程度差距不大，最高值和最低值之差在 0.6 个百分点以内。

（三）**南北方人口老龄化程度略有差异**，北方高于南方分南北方看，北方地区人口老龄化程度略高于南方。北方地区 60 岁及以上人口占比为 19.75%，北方比南方高 1.72 个百分点，但从 65 岁及以上人口占比来看，北方地区为 13.90%，仅比南方高 0.63 个百分点（见图 1）。南方比北方人口老龄化程度低的原因，一是北方的计划生育政策执行比较严格，导致在一段时期内出生人口在低位徘徊；二是北方人口向南方流动。历年常住人口统计数据显示，南方大部分省份均实现人口净流入，而北方省份除新疆、陕西外多表现为人口净流出，伴随着南北经济发展分化，人口也呈现出“南入北出”的格局。

至此，中国人口分布密度分析完成。

### 3.1.3.2 多人口乡镇街道聚类图分析

在使用 DBSCAN 进行聚类分析时，我们可以利用有关的数据进行实际的分析和研究，并且利用数据进行分析实际探究，在这里色痕迹了不同聚类参数下不同的聚类结果，并以此指导实验的最终聚类结果。首先使用 eps 参数为 5000 (m)，min\_samples 为 10，将所有数

据分为 58 个类，可视化如图所示。



图 22 Eps=5000m, min\_sample=10 的聚类结果

可以发现，大多数点均被排除，但能看出有一些距离很近的点分类成功，说明问题出在调参上。接着继续进行调参，以期获得较好结果。在将 eps 参数调整为 40000 (m) 后，共有 9495 个点成功分类，共分为 47 个类，大约有 10%的数据被排除，结果比较理想，因此使用此数据作为结果显示，具体如图所示。



图 23 Eps=40000m, min\_sample=10 的聚类结果

### ● 分析结论

聚类结果来看，中国人口分布呈现明显的聚集特征，有两个主要的类别：北京以南与华南地区，就聚类结果来看，中国人口分布呈现明显的聚集特征，有两个主要的类别。一类主要为北京以南，包括河南河北，江浙以及湖北和安徽等地，另一类主要在南方，包括广东福建以及广西的部分地区。在这两大类之外还有几个较小的聚集区，如四川重庆，云贵以及江西等地。其他地区如东北虽然人口较多，但分布稀疏，未呈现明显的聚集特征。

### 3.1.3.3 市级多人口乡镇街道统计图

依据聚类结果和行政区划空间连接后的数据，使用自然分段法进行划分，可视化后得到如下的两幅图：

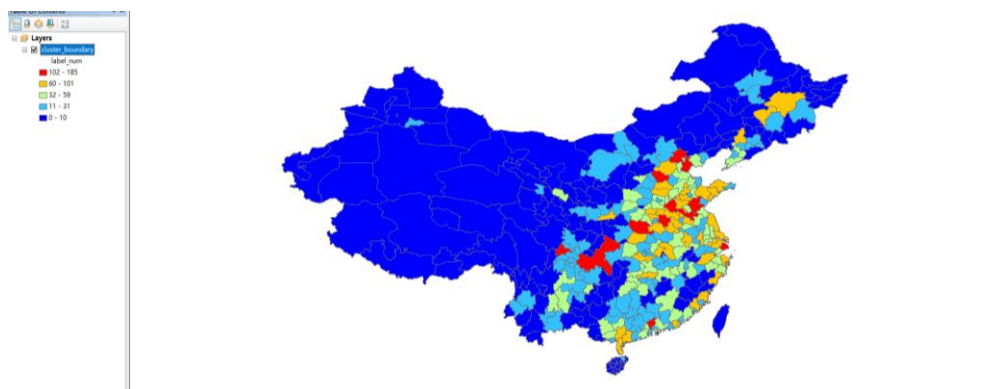


图 24 市级多人口聚集乡镇街道统计图

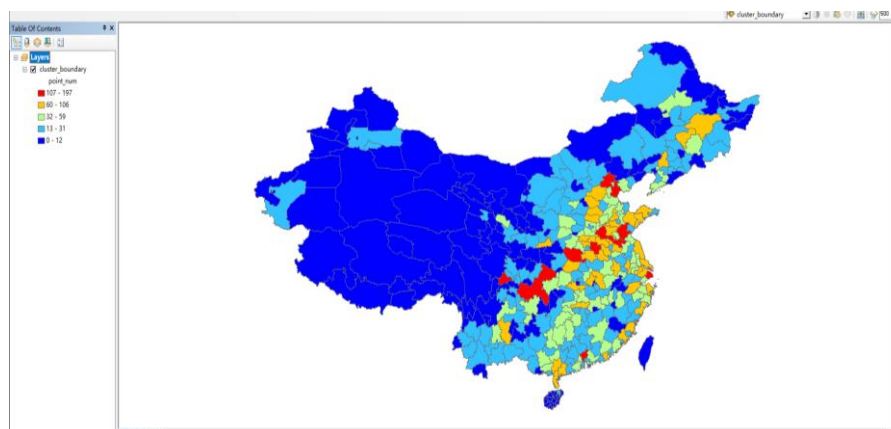


图 25 市级多人口乡镇街道统计图

### ● 分析结论

从图中可以看出，中国高人口聚集的城市主要部分在北京到重庆、成都的连线上，仅有两个例外，一个是上海，一个是广州。同时东北以及新疆乌鲁木齐市附近多人口的城镇较多，但分布相对均匀，聚集性差。至于东南沿海地区从之前的结果中获知人口较多，但分布高度集中于有限的几个乡镇街道，人口分布高度密集，导致多人口乡镇街道数目相对较少，人口聚集性远超其他地区。

### 3.1.3.4 总结

人口空间分布与经济分布高度相关，人口密集区域大多集中于经济发达的地区，经济越发达的区域，人口聚集性越强，而且青壮劳动力多集中于这些经济发达的区域，经济发达的区域，劳动力的需求较高，具有较多的经济社会发展需求，综合影响如下图所示：



图 26 综合影响结果

至此，实验 1 全部结果实现，对于人口的探究初步完成。

## 3.2 华盛顿犯罪分析

### 3.2.1 实验目的

利用华盛顿 2013 年-2022 年的犯罪数据进行分析规律，找出案件发生时间和作案工具和季节性特征的分析，完成对于华盛顿地区犯罪分析等问题的数据挖掘和探究。

### 3.2.2 时空数据获取与处理

#### (1) 数据来源

##### 1. 2013-2022 年美国华盛顿特区犯罪数据

本次实习的使用数据来源于 OpendataDC.gov，网址如后链接：  
([https://opendata.dc.gov/datasets/f516e0dd7b614b088ad781b0c4002331\\_2/explore?location=38.910865%2C-77.020651%2C14.70](https://opendata.dc.gov/datasets/f516e0dd7b614b088ad781b0c4002331_2/explore?location=38.910865%2C-77.020651%2C14.70))，进入网址后选择如下图所示的位置进行下载数据。  
数据格式为 shp 与 csv 格式文件，具体数据下载位置如图所示：

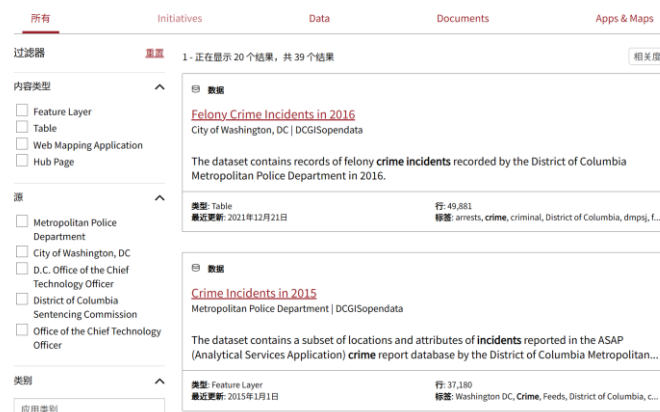


图 27 犯罪数据下载位置

将数据导入到 ArcGIS 后，显示显示如下图所示：

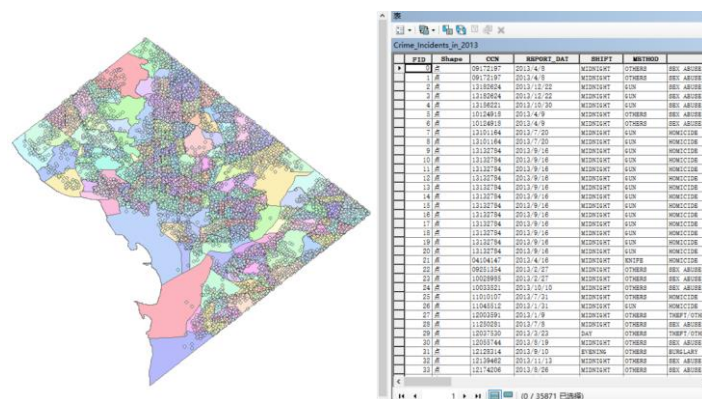


图 28 犯罪数据可视化显示



## 2. 2020 年美国华盛顿特区人口普查数据

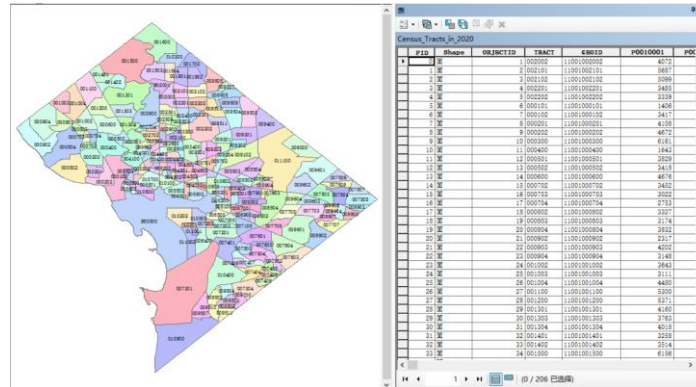


图 29 人口普查数据可视化

### (2) 数据格式

数据类型：csv 格式与 shp 格式

### (3) 数据处理技术理论

(一) **热力图绘制**：热力图 (Heat Map) 是通过密度函数进行可视化用于表示地图中点的密度的热图。它使人们能够独立于缩放因子感知点的密度。热力图能够直观表现数据分布，通过颜色的深浅和亮度的变化，可以让人更好地理解数据密集程度和趋势；热力图可以帮助人们快速识别数据中的规律，找到数据中的异常点和趋势，从而更好地理解数据的含义。使用 pandas 库对 2013-2020 年每年案件数量的 csv 文件进行读取，并去除重复记录，后调用 folium 库绘制热力图。

(二) **时间序列分析**：时间序列分解是将时序数据分离成不同成分：Trend (长期趋势)、Seasonal (季节性)、Residuals (残差)。具体分解过程为 (以 Additive Model 为例)：  
Step1: 分解趋势项 (中心化移动均值方法)

$$T_t = \begin{cases} \frac{x_{t-(\frac{f-1}{2})} + x_{t-(\frac{f-1}{2})+1} + \dots + x_{t+(\frac{f-1}{2})}}{f}, t \in (\frac{f+1}{2}, l - \frac{f-1}{2}), \text{当 } f \text{ 为奇数时} \\ \frac{0.5x_{t-(\frac{f}{2})} + x_{t-(\frac{f}{2})+1} + \dots + x_{t+(\frac{f}{2})-1} + 0.5x_{t+(\frac{f}{2})}}{f}, t \in (\frac{f+1}{2}, l - \frac{f-1}{2}), \text{当 } f \text{ 为偶数时} \end{cases}$$

Step2: 分解季节性周期项

采用将原始时间序列减去趋势项:  $S_t = x_t - T_t$  将各个周期内相同频率下的值平均化，得到季节项 figure

$$figure_t = \sum_{i=0}^n \frac{S_{t+i*f}}{f}, t \in (1, f), n = \max(n, n*f \leq l)$$

将 figure 中心化，得到中心化的季节项 Figure，代码可表述为：

$$figure = figure - \text{mean}(figure)$$

最终得到的长度为 f 的季节项

Step3: 计算残值项：

$$e[t] = Y[t] - T[t] - S[t]$$

时间序列分析的应用有：趋势预测、异常检测等。我调用 statsmodels 包来分解时间序列，调用 statsmodels 的 seasonal\_decompose 方法，并且通过设置参数 mode="additive(multiplicative)" 来执行加法(乘法)分解操作。

(三) **空间分析**：莫兰指数分为全局莫兰指数 (Global Moran's I) 和局部莫兰指数 (Local Moran's I)。通常情况，先做一个地区的全局莫兰指数，全局指数告诉我们空间是否



出现了集聚或异常值，但并没有告诉我们在哪里出现，回答有或无。如果全局有自相关出现，接着做局部自相关，局部 Moran' I 会告诉我们哪里出现了异常值或者哪里出现了集聚。


全局莫兰指数	$I = \frac{N}{W} \cdot \frac{\sum_{i=1}^N \sum_{j=1}^N w_{ij} (x_i - \bar{x})(x_j - \bar{x})}{\sum_{i=1}^N (x_i - \bar{x})^2}$
Z 得分	$I = \frac{N}{W} \cdot \frac{\sum_{i=1}^N \sum_{j=1}^N w_{ij} (x_i - \bar{x})(x_j - \bar{x})}{\sum_{i=1}^N (x_i - \bar{x})^2}$
局部莫兰指数	$I = \frac{N}{W} \cdot \frac{\sum_{i=1}^N \sum_{j=1}^N w_{ij} (x_i - \bar{x})(x_j - \bar{x})}{\sum_{i=1}^N (x_i - \bar{x})^2}$
Moran' I 散点图	

表 1 空间分析不同指标

调用 libpysal.weights.distance.Kernel.from\_dataframe() 函数,由 dataframe 构建权重矩阵,后调用 esda.moran.Moran() 和 esda.moran.Moran\_Local() 函数,计算全局莫兰指数和局部莫兰指数,最后用 spplot.esda 库相关函数,如用 moran\_scatterplot() 绘制 Local Moran' s I 散点图,用 plot\_moran() 绘制全局 Moran' s I 分析图像, lisa\_cluster() 绘制 LISA 聚集图。

#### (4) 数据处理

使用 pandas 库、geopandas 库、numpy 库实现多源异构空间数据的输入、处理、输出。使用 pandas.dataframe.groupby() 根据某一字段,对源数据进行归类、计数;使用 pandas.dataframe.apply() 对源数据进行各种灵活的变形处理;使用 pandas.merge() 对两种数据依据某一字段进行合并;使用 geopandas.sjoin() 函数,根据要素的相对空间位置将连接要素中的行匹配到目标要素中的行。

##### ① 绘制热力图:

使用 pandas 对某年犯罪数据的 csv 格式文件进行读入,分别获取"X", "Y" 字段信息作为空间点坐标信息,再用 numpy 库均值函数计算并设置图中心点。Python 的 folium 库提供了绘制热力图的函数,且提供了地图底图,可将热力图与地图底图叠加进行可视化,比较便捷。实现代码如下 (HeatMap.py):

```
1. import numpy as np
2. import pandas as pd
3. import folium
4. from folium.plugins import HeatMap
5. def readfiles(file path ,field name e)
6. df pd.read csv(file path)
```

```

7. #df.dropna(axis=e,how="any',inplace=True) #删除有空白值的数据 df x df["x"]x df
x.values.tolist()df y df["y"]
8. y=df y.values.tolist()
9. data=[]for i in range(len(x)):
10. data.append([y[i],x[i],1])
11. return y,x,data
12.
13. if __name__ == "main":lat,lng,data=readfiles("crime Incidents in 2020.csv")#计算中心点,
也可以自定义
14.Center=[np.mean(np.array(lat ,dtype='float32')),np.mean(np.array(lng,dtype='float32')
)]#初始化地图
15. m-folium.Map(location Center,zoom start-6)#热力图
16. HeatMap(data,radius = 15).add to(m)#愿示底图要翻墙
17. m.save("res.html")

```

## ② 时间序列分析

本部分代码过长,不予展示全部代码,详细代码在提交的 time\_analyse.py 文件中阅读。

## ③ 空间分析:

本部分代码主要使用人口普查数据进行分析, 并且最后输出 shp 文件进行可视化。

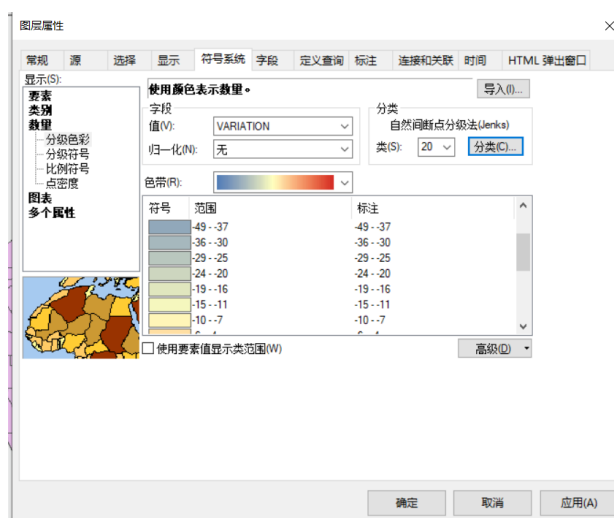


图 30 ArcGIS 可视化

## 3.2.3 时空数据分析与挖掘

### 3.2.3.1 热力图与犯罪热点区域分析

绘制的 2013-2022 年每年案件数量的热力图如下图所示, 可以明显观察到, 犯罪热点区域分布有规律。2013-2021 年, 紫色圈所示区域犯罪事件高发。

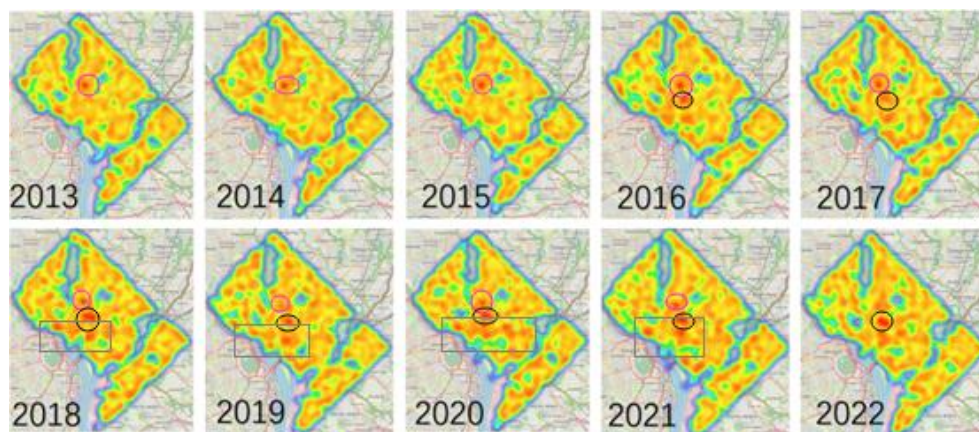


图 31 犯罪热力图

- 分析结论

从 2016 年开始，黑色圈所示区域的犯罪事件日益严峻；从 2018 年开始，灰色框所示的区域开始成为犯罪事件的高发区域；与之形成鲜明对比的是，Washington DC 左角处治安状况较为良好，鲜少被不法分子荼毒。

### 3.2.3.2 时间序列分析

#### (1) 2013-2022 年总案件数时间序列分析

使用 Multiplicative Model，可以获得如下图所示的结果：

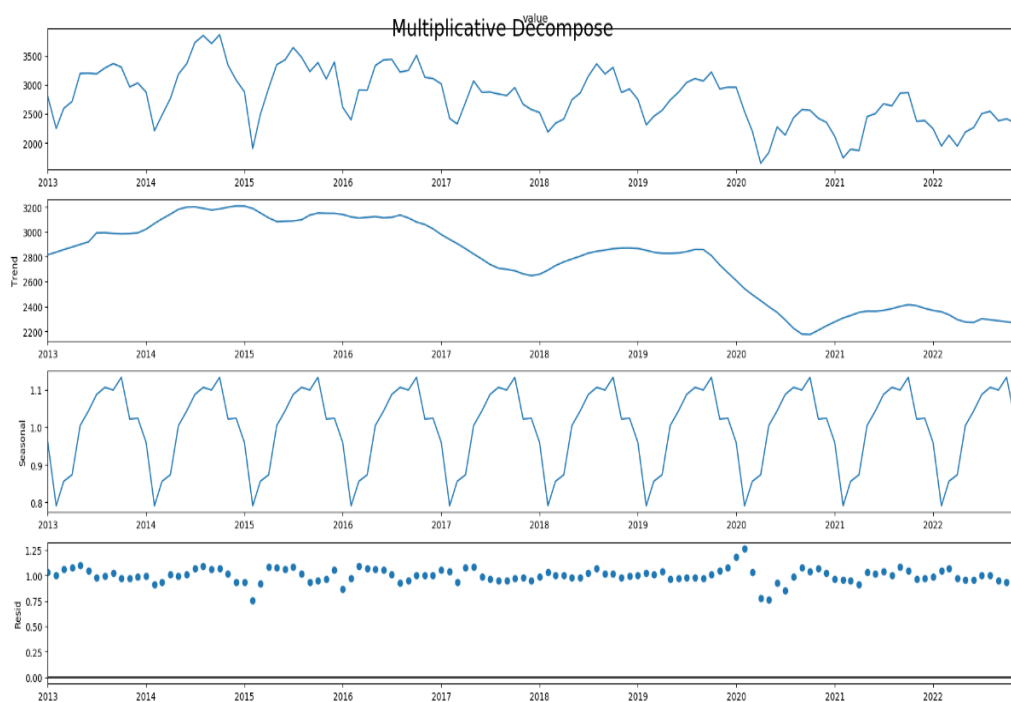


图 32 multiplicative model

使用 Additive Model:

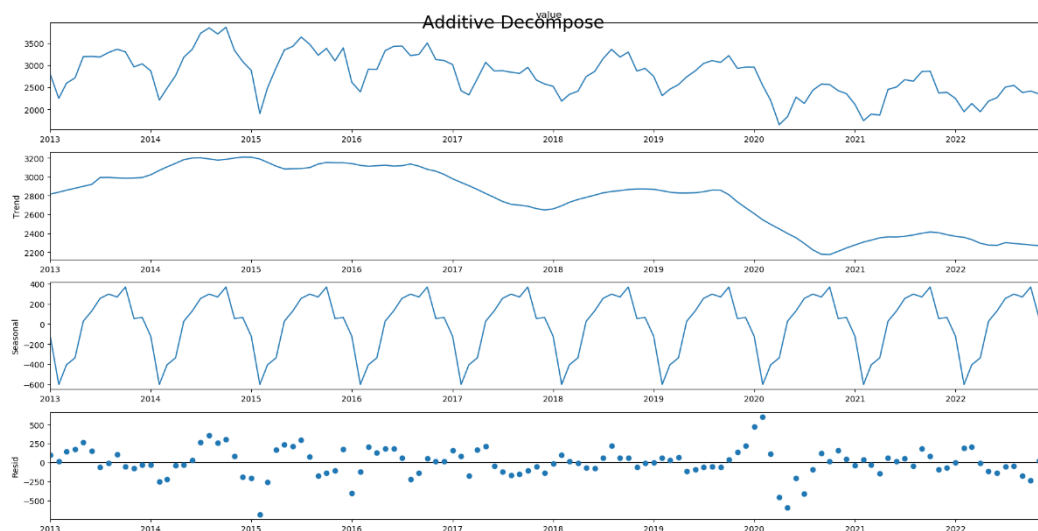


图 33 additive model

### ● 分析结论

使用两种模型进行时间序列分解，残差都比较理想。

- ① 犯罪数量趋势：可以发现，犯罪数量从 2013 年起开始升高，在 2014 年内达到峰值，之后维持高位三年之久。在 2016 年 9 月，犯罪数量开始下降，至 2018 年初又迎来一次小反弹。在 2019 年 10 月，犯罪数量开始大幅跌落，到 2020 年 9 月跌落到低谷，此后犯罪数量缓慢提升，但数量已无法与 2014 年-2017 年相比。

犯罪数量随季节的变化：可以看到犯罪数量有明显的季节特性，在每年 2 月初达到低谷，之后迅速走高，到 9 月份达到峰值，之后迅速回落。

- ② 噪声随时间的变化：Multiplicative 序列分解模型的噪声变化较为平缓，绝大部分噪声落在 0.8-1.2 间，在 2015 年 2 月、2020 年初发生较大波动，可以发现疫情对犯罪数量有十分显著的影响，这两个时间节点是我后续分析时的重点关注部分。Additive 序列分解模型的噪声变化波动较大，在上述两个时间节点的噪声也很显著。

### (2) 2013-2022 各类案件数量时间序列分析

“OFFENSE” 字段反映了犯罪的类型，可分为 “ARSON”， “ASSAULT W/DANGEROUS WEAPON”， “BURGLARY”， “HOMICIDE”， “MOTOR VEHICLE THEFT”， “ROBBERY”， “SEX ABUSE”， “THEFT F/AUTO”， “THEFT/OTHER” 九类。我分别对九类案件数量进行统计案件主要集中在盗窃部分（占 85.17%）。

首先来看 “ARSON” ——纵火案（左下）和 “HOMICIDE” ——故意杀人案（右下）。这两种案件的特点是，数量最少，但对社会的危害性最大。

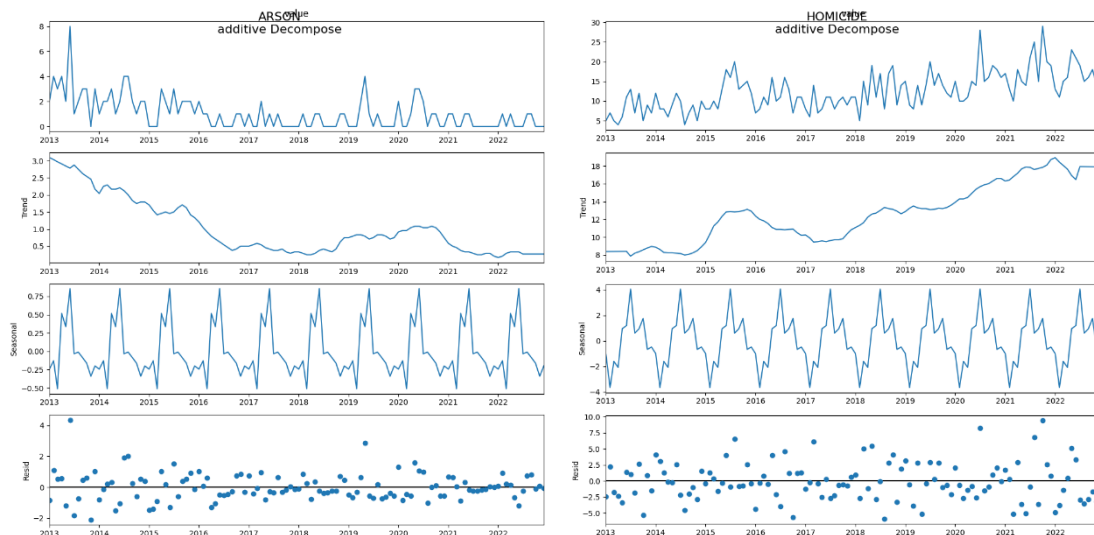


图 34 纵火与故意杀人

再来“ASSULT W/DANGEROUS WEAPON”——使用危险武器斗殴（左下）和“SEX ABUSE”——性骚扰（右下）。这两类案件的样本数量同样是比较少的。

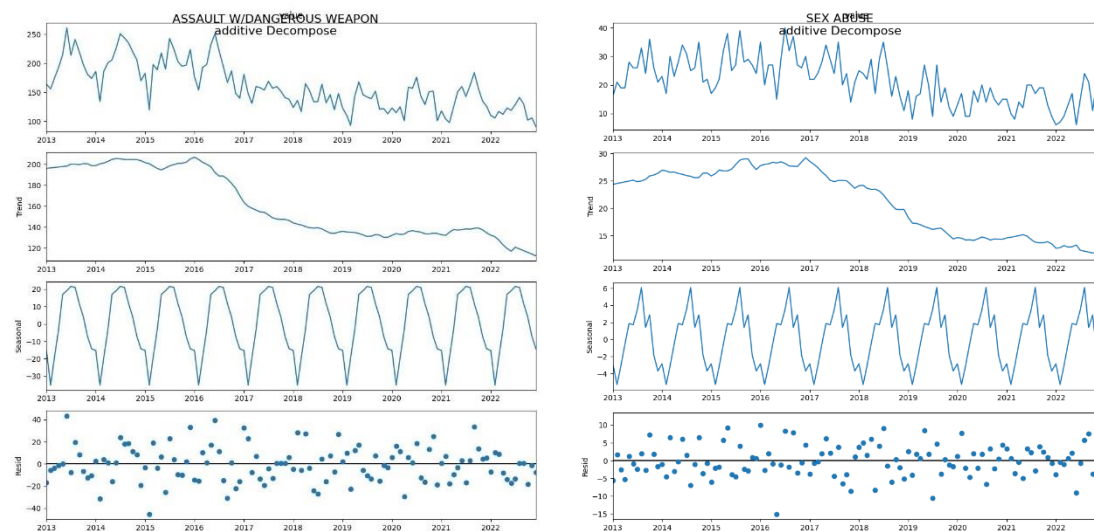


图 35 使用危险武器斗殴和性骚扰

ASSAULT W/DANGEROUS WEAPON（使用危险武器斗殴）的案件数量在 90-270 件/月间波动，2013 年 6 月发生的此类案件最多，为 261 件，2022 年 12 月发生的最少，为 91 件。综合来看，2013-2016 年每月发生的此类案件数量要多于 2017 年及以后。SEX ABUSE（性骚扰）案件在 0-40 件/月间波动，最严重的是 2016 年 7 月，发生了 40 件性骚扰案件，2022 年 1 月和 6 月的此类案件最少，发生了 6 件。整体来看，性骚扰案件在逐年下降。

之后来看”BURGLAY”——入室盗窃案（左下）和“ROBBERY “——抢劫案（右下）。

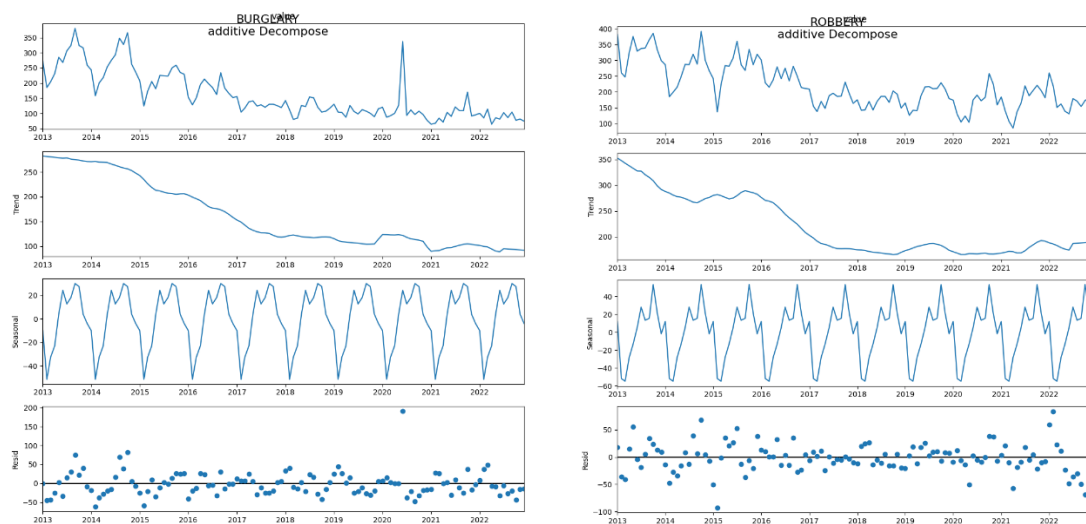


图 36 入室盗窃和抢劫

BURGLARY（入室盗窃）案件数量在 60–380 件/月间波动，2013 年 9 月发生的此类案件最多，为 380 件/月，2021 年 1 月和 2022 年 4 月发生的此类案件数量最少，为 64 件/月。入室盗窃在 10 年间总体呈下降趋势，但是在 2020 年第二季度出现严重的反弹，这可能是由于新冠疫情所导致的。ROBBERY（抢劫）案件在 80–400 件/月间波动，最严重的是 2014 年 10 月，发生了 391 件抢劫案，2021 年 4 月发生的此类案件最少，为 86 件。2017 年之后的抢劫案数量相比之前有所改善。

然后来看” MOTOR VEHICLE THEFT “——重大汽车盗窃案（左下）与” THEFT F/AUTO “——普通汽车盗窃案（右下）。

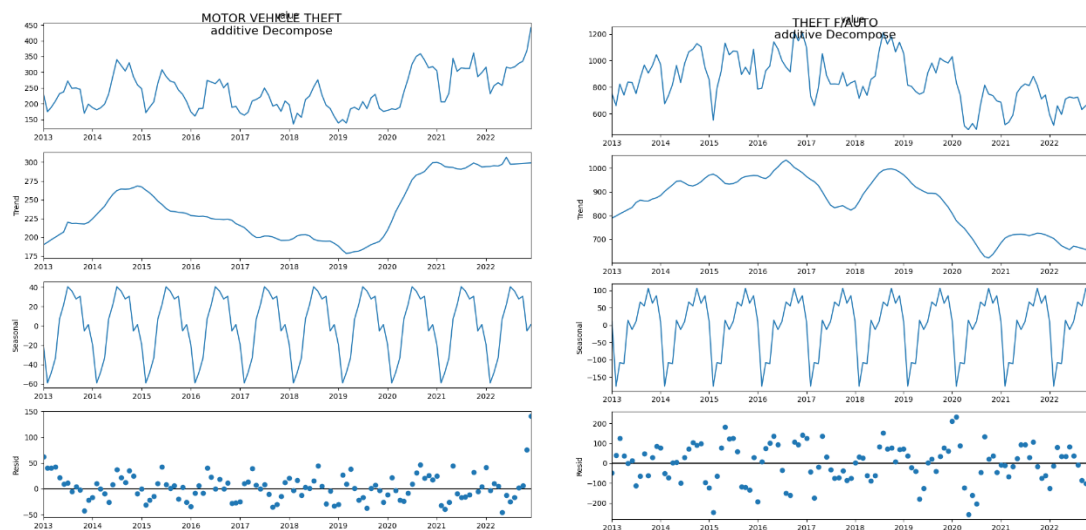


图 37 重大车辆盗窃和普通车辆盗窃

MOTOR VEHICLE THEFT 和 THEFT F/AUTO 的区别是，前者用来表示重大汽车盗窃案件，后者表示普通的汽车盗窃案件。虽然同为汽车盗窃，但两种盗窃案件的数据规律却不同。重大汽车盗窃发生最多的月份是 2022 年 12 月，发生 441 件，最少的月份是 2018 年 2 月，发生 136 件，2020 年后本类汽车盗窃案数量在逐年上升；普通汽车盗窃案发生最多的月份是 2016 年 10 月，发生 1216 件，最少是 2020 年 5 月，发生 481 件，2020 年之后的普通汽车盗



窃案反倒不如之前那么严重了。

- **分析结论**

- ① 纵火罪在 2013 年-2016 年间呈下降趋势，2017-2018 年第三季度都维持低水平，第四季度-2020 年第三季度有所上涨，之后再次下降。2016 年第四季度开始，纵火案件出现频率明显低于之前。故意杀人案一反常态，逐年走高，由最初 2013 年的 8 件/月，到 2015 年初开始为期半年的快速增长，增至 13 件/月，维持半年后，案件数量缓慢减少了一年半，从 2017 年中再次开始增长，截止 2022 年 12 月，故意杀人案已高达 18 件/月。
- ② 使用危险武器斗殴案件在 2013-2015 三年间都维持在 200 件/月左右，处于高位，之后开始下降，到 2019 年初降至 130 件/月，此后维持稳定三年之久，2022 年起继续开始下降，截止 2022 年底，此类案件数降至 110 件/月。性骚扰类案件在 2013-2016 年一直维持在 27 件/月上下，处于高位，之后此类案件开始减少，到 2022 年 12 月，降至每月 10 件以下
- ③ 入室盗窃罪由 2013 年初的 240 件/月开始逐渐下降，截止 2022 年底，降至 100 件/月上下，此类案件在十年间得到明显的改善。抢劫案从 2013 年初的 350 件/月开始逐渐下降，到 2014 年第三季度达到了平稳期，维持在 270 件/月近一年半后，从 2016 年初开始继续下降，从 2017 年中开始保持在低于 200 件/月的较低水平，直至 2022 年底，又有了上升的趋势。
- ④ 重大车辆盗窃案和普通车辆盗窃案的变化趋势比较有趣，二者在 2013-2014 年都保持上升趋势，但从 15 年开始，重大汽车盗窃案开始下降，普通车辆盗窃案在波动中维持高位。在 2020 年之后，普通汽车盗窃案开始下降，而重大车辆盗窃案开始陡增，重大车辆盗窃案在普通车辆盗窃案达到历史低值的时候保持在了历史高值。

### 3.2.3.3 空间分析分析

#### (1) 2019-2020 年各案件数量在不同区域变化

九类案件 2019-2020 年在各人口普查区的变化情况如图所示。案件数量变化由减少->增多对应颜色由蓝色->红色。

- **分析结论**

可以发现，案件增多的区域分布比较发散，而案件减少区域的分布则比较有规律，如在 ARSON 案件减少的区域，在其他案件中也都迎来了案件数量的减少。除了两类车辆盗窃案和其他盗窃案外，其余案件在最下方人口普查区也都减少了（或如 ARSON 保持为 0），我分析可能是由该地区的疫情情况更加严峻所造成的。

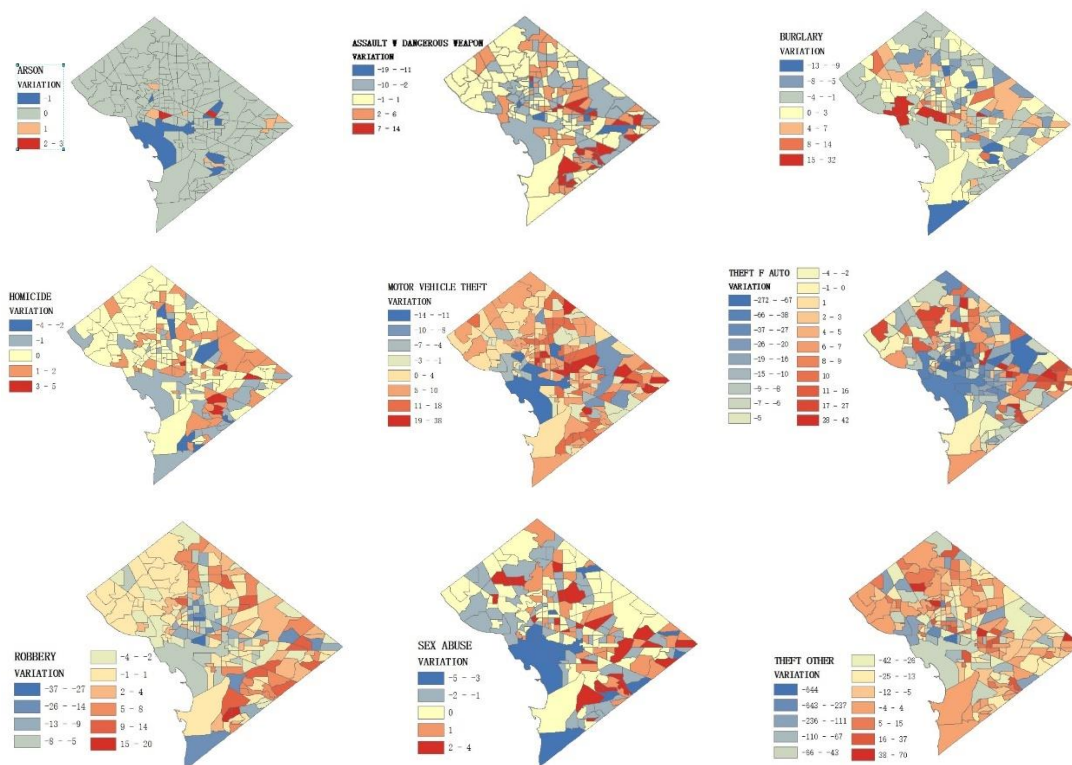


图 38 各类案件变化图

### 3.2.3.4 总结

通过对于实验问题的分析与研究，我们不难发现以下的规律：

- ① 从 2016 年开始，黑色圈所示区域的犯罪事件日益严峻；从 2018 年开始，灰色框所示的区域开始成为犯罪事件的高发区域；与之形成鲜明对比的是，Washington DC 左角处治安状况较为良好，鲜少被不法分子荼毒。
- ② 犯罪数量趋势：可以发现，犯罪数量从 2013 年起开始升高，在 2014 年内达到峰值，之后维持高位三年之久。在 2016 年 9 月，犯罪数量开始下降，至 2018 年初又迎来一次小反弹。在 2019 年 10 月，犯罪数量开始大幅跌落，到 2020 年 9 月跌落到低谷，此后犯罪数量缓慢提升，但数量已无法与 2014 年-2017 年相比。  
犯罪数量随季节的变化：可以看到犯罪数量有明显的季节特性，在每年 2 月初达到低谷，之后迅速走高，到 9 月份达到峰值，之后迅速回落。
- ③ 噪声随时间的变化：Multiplicative 序列分解模型的噪声变化较为平缓，绝大部分噪声落在 0.8-1.2 间，在 2015 年 2 月、2020 年初发生较大波动，可以发现疫情对犯罪数量有十分显著的影响，这两个时间节点是我后续分析时的重点关注部分。Additive 序列分解模型的噪声变化波动较大，在上述两个时间节点的噪声也很显著。
- ④ 纵火罪在 2013 年-2016 年间呈下降趋势，2017-2018 年第三季度都维持低水平，第四季度-2020 年第三季度有所上涨，之后再次下降。2016 年第四季度开始，纵火案件出现频率明显低于之前。故意杀人案一反常态，逐年走高，由最初 2013 年的 8 件/月，到 2015 年初开始为期半年的快速增长，增至 13 件/月，维持半年后，案件数量缓慢减少了一年半，从 2017 年中再次开始增长，截止 2022 年 12 月，故意杀人案已高达 18 件/月。

- ⑤ 使用危险武器斗殴案件在 2013-2015 三年间都维持在 200 件/月左右，处于高位，之后开始下降，到 2019 年初降至 130 件/月，此后维持稳定三年之久，2022 年起继续开始下降，截止 2022 年底，此类案件数降至 110 件/月。性骚扰类案件在 2013-2016 年一直维持在 27 件/月上下，处于高位，之后此类案件开始减少，到 2022 年 12 月，降至每月 10 件以下
- ⑥ 入室盗窃罪由 2013 年初的 240 件/月开始逐渐下降，截止 2022 年底，降至 100 件/月上下，此类案件在十年间得到明显的改善。抢劫案从 2013 年初的 350 件/月开始逐渐下降，到 2014 年第三季度达到了平稳期，维持在 270 件/月近一年半后，从 2016 年初开始继续下降，从 2017 年中开始保持在低于 200 件/月的较低水平，直至 2022 年底，又有了上升的趋势。
- ⑦ 重大车辆盗窃案和普通车辆盗窃案的变化趋势比较有趣，二者在 2013-2014 年都保持上升趋势，但从 15 年开始，重大汽车盗窃案开始下降，普通车辆盗窃案在波动中维持高位。在 2020 年之后，普通汽车盗窃案开始下降，而重大车辆盗窃案开始陡增，重大车辆盗窃案在普通车辆盗窃案达到历史低值的时候保持在了历史高值。

至此实验二的全部实验完成。

## 5 实习总结与体会

### 5.1 实习总结

本次实习老师给予的自由度较大，允许自由选题。我个人在权衡过后，选择使用老师提供的数据进行实习。在进行数据分析挖掘的过程中，遇到了不少的问题，我通过各种方式寻求解决方法，最终将整个数据分析的流程走完，得到了较好的结果，也在过程中有了不少的收获。在此对实习过程中的一些心得体会进行总结：

#### 1. 单一数据的价值有限

初次打开数据后，发现只有点数据，因此很难对这些数据进行有意义的分析。于是上网寻找行政区边界数据，两者结合后分析结果立即变得易于理解，价值提升。

#### 2. 数据标准化并不总是有用

进行空间聚类时，曾尝试对坐标标准化后调参，结果由于参数实际意义消失，很难确定合适的 `eps` 值。因此最终放弃标准化，对应实际意义设置参数，很快就找出了有价值的参数值。

#### 3. 方法的使用要考虑实际意义

聚类时也尝试使用过 K-MEANS 等算法，但聚类结果不可解释，无法进行下一步的分析和总结。个人以为不能为了使用方法而使用方法，要为了分析而使用方法，因此最终使用 DBSCAN 算法以期找出高聚集性人口区域，也得到了较好的结果。

通过本次实习，我对老师理论课上所讲授的内容有了更深入的理解，真正做到对理论知识知行合一。理论课上接触时间序列分析时，感觉距离比较遥远，很多的实际情况和具体的分析方法在脑海里都比较朦胧。但这次实习给了我一个很好的近距离接触机会，让我在实际操作中增进对知识的理解。

### 5.2 实习体会

本次实习内容繁多，我的实习收获也颇为丰富，主要体现在：

本次实习，将《时空数据分析与挖掘》课程中的理论知识运用到实际当中。在理论课程中，我们分别学习了数据分析、数据挖掘的各方面原理，但“纸上谈兵”终觉浅，本次实习是我们第一次实实在在地使用代码进行那些背诵了无数遍的理论操作，是从理论到实践的一次飞跃。

本次实习，全身心的投入使我成长，在实习的过程中，最大的体会是在完成制作时会全身心投入到操作中，有很多时候会觉得实践时间不够充足，以致于不能够完全深入探究一些操作。希望能够延长实习周期，对实习指导书中介绍的各种功能有一定的操作与探究时间，达到更好的实践效果。同时，在实习中也体会到要拥有不断试错、不断改进的耐心。可能在前期实践时对操作不够熟悉，会出现一些不正确或是不够完美的效果，在后期学习中可能会寻找到更合适的方法，可以不断做出修改，这种探究精神与不断改正的耐心是对我们将来的学习，甚至是从事科研工作都十分有益的。

最后，祝贺本次《时空数据分析与挖掘》实习圆满结束，感谢杨老师在实习过程中对于我们的帮助和指导！