交通情况与经济、疫情传播的交互分析

目录

一、 研究背景

本研究的分析对象是共计一亿余条的交通出行记录数据，包含了中国各省的交通数据，时间跨度由 2019 年到 2021 年，其中有班次代码、发车日期、发车时间、乘车站名称、到达站名称、座位类型六个变量。

二、 数据预处理

2.1. 数据简介

首先，我们使用 PySpark 读入原始数据 csv 文件，并按变量说明进行简单预处理（舍去第四列、第五列，并更改其余列的列名），得到初始数据框如下：

```
+--------+--------+--------+------------+----------+--------+
|班次代码|发车日期|发车时间|  乘车站名称|到达站名称|座位类型|
+--------+--------+--------+------------+----------+--------+
|    null|    null|    null|        null|      null|    null|
|    null|    null|    null|        null|      null|       9|
|    null|    null|    null|        null|      null|       9|
|    8009|20200222|  110000|  大同汽车站|      浑源|       1|
|    8009|20200222|  110000|  大同汽车站|      浑源|       1|
|    8006|20200224|   92000|  大同汽车站|      浑源|       1|
|    8006|20200224|   92000|  大同汽车站|      浑源|       1|
|    8010|20200225|  104000|  大同汽车站|      浑源|       1|
|    8010|20200225|  104000|  大同汽车站|      浑源|       1|
|    8010|20200225|  104000|  大同汽车站|      浑源|       1|
|    8010|20200225|  104000|  大同汽车站|      浑源|       1|
|    8006|20200226|   92000|  大同汽车站|      浑源|       1|
|    8010|20200226|  104000|  大同汽车站|      浑源|       1|
|    8010|20200226|  104000|  大同汽车站|      浑源|       1|
|    8006|20200226|   92000|  大同汽车站|      浑源|       1|
|    8010|20200226|  104000|  大同汽车站|      浑源|       1|
|    8010|20200226|  104000|  大同汽车站|      浑源|       1|
|    1069|20200227|  112000|浑源县汽车站|      大同|       1|
|    1069|20200227|  112000|浑源县汽车站|      大同|       1|
|    8008|20200227|  100000|  大同汽车站|      浑源|       1|
+--------+--------+--------+------------+----------+--------+
only showing top 20 rows
```

数据中共包含六个变量，简介如下表：

名称	数据类型	格式说明	示例
班次代码	string	长度 4-8 位	KS1057

发车日期	string	yyyyMMdd	20200507
发车时间	string	Hmmss	90000
乘车站名称	string	无	苏州北广场汽车站
到达站名称	string	无	常熟南站
座位类型	string	1：普通座 2：商务座 3：上铺 4：下铺 9：其他	1

2.2. 数据清洗

读入数据后，我们检测出了如下几种数据有错漏的情况，并将它们从原数据中去除，以便展开后续分析。

首先是变量缺失，其中包括整行缺失和部分缺失。由于变量缺失情况整体不算严重，我们选择了去除所有出现缺失值的数据行。

其次是变量格式错误，如发车时间并非 5-6 位数字，而是 3-4 位数字；或是座位类型不属于 1、2、3、4、9；或是在车站名称处却出现了乘车人的年龄、性别。

为了处理该类数据错漏，我们首先尝试了对每个变量设置一个 **UDF 函数**，分别用于**判断各个变量是否符合格式要求**，并依此对数据框进行多次筛选。由于后几次筛选几乎没有筛掉数据行，我们判断各个变量上的格式错误常常出现在同一个数据行内，因此我们最终根据出行时间和座位类型二者进行了筛选，筛选后其他变量的格式错误也基本消失。

为了筛选出格式正确的出行时间，我们首先将发车日期和时间合并到同一列，

再对此列使用 PySpark 内置的 to_timestamp 函数。该函数在格式正确匹配时会返回 timestamp 格式的数据，十分便于后续的时间相关操作；在格式错误时会返回 null。通过去除缺失值，我们便完成了筛选过程。

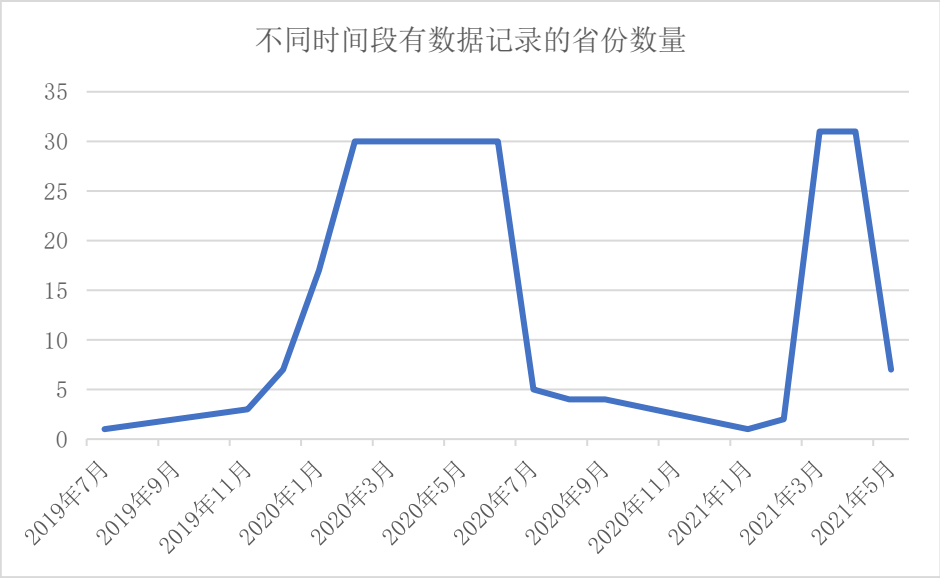
将各类错漏数据去除后，共留下了约七千万条格式标准、便于分析的数据行，我们将其储存为新的 parquet 文件，以便于后续的快速读写。

2.3. 地理特征提取

我们收集了每条出行数据的乘车站名称和到达站名称并去重，得到了五万余个车站名称。接下来，我们使用地图 API 获取了每个车站所在省份及经纬度信息（这里我们是将获取地理信息的 API 操作借助 udf 函数函数化定义，进而对每一列同时操作来生成一列新的变量）。通过该信息，我们为原数据添加了乘车站名称、乘车站经度、乘车站纬度、到达站名称、到达站经度、到达站纬度六个新变量，并通过起止站点的经纬度信息计算出了车次的运行距离。我们发现，全部 34 各省份都有。

除此之外，我们使用了中国地理标准区划，将 34 个省份分为 7 个行政地理分区，如下：

行政地理分区	对应省份
华北	北京市、天津市、河北省、山西省、内蒙古自治区
东北	黑龙江省、吉林省、辽宁省
华东	上海市、江苏省、浙江省、安徽省、江西省、山东省、福建省、台湾省
华中	河南省、湖北省、湖南省
华南	广东省、广西壮族自治区、海南省、香港特别行政区、澳门特别行政区
西南	重庆市、四川省、贵州省、云南省、西藏自治区



西北	陕西省、甘肃省、青海省、宁夏回族自治区、新疆维吾尔自治区

由此我们提取了乘车站和到达站所属的行政地理分区，以供更广泛的研究所需。最终数据集如下所示：

```
+--------+------------------+---------------+---------+--------+----------+------------------+-----------+----------+----------+------------------+-----------+----------+----------+
----------+--------+
|班次代码|             发车时间|       乘车站名称|到达站名称|座位类型|乘车站省份|乘车站行政地理分区|乘车站经度|乘车站纬度|到达站省份|到达站行政地理分区|到达站经度|到达站纬度|      距离|
+--------+------------------+---------------+---------+--------+----------+------------------+-----------+----------+----------+------------------+-----------+----------+----------+
----------+--------+
|  KS1057|2020-05-07 09:00:00|苏州北广场汽车站|   常熟南站|       1|     江苏省|                华东|120.608475| 31.330946|     江苏省|                华东| 120.74239| 31.628
862|  35.478|
|  KS3197|2020-05-07 10:50:00|        苏州南站|   常熟南站|       1|     江苏省|                华东|120.638145|  31.27728|     江苏省|                华东| 120.74239| 31.62
8862|  40.325|
|  GT1001|2020-05-07 17:40:00|          沙溪站|     太仓站|       1|     湖南省|                华中|109.902885| 26.756468|     江苏省|                华东| 121.19665| 31.5
10124| 1216.72|
|  GT1001|2020-05-07 17:40:00|          沙溪站|     太仓站|       1|     湖南省|                华中|109.902885| 26.756468|     江苏省|                华东| 121.19665| 31.5
10124| 1216.72|
|  KT3117|2020-05-07 09:30:00|          朝阳站|   嘉定刷卡|       1|     辽宁省|                东北| 120.44165| 41.578815|     上海市|                华东| 121.19923| 31.36
9852| 1137.19|
|  KS3414|2020-05-07 09:30:00|        苏州南站|     连云港|       1|     江苏省|                华东|120.638145|  31.27728|     江苏省|                华东| 119.22161| 34.59
6653| 392.042|
|  KC1241|2020-05-07 09:30:00|        常熟南站|苏州火车站|       1|     江苏省|                华东| 120.74239| 31.628862|     江苏省|                华东|120.611176| 31.33
0664|  35.416|
|  KC1241|2020-05-07 09:30:00|        常熟南站|苏州火车站|       1|     江苏省|                华东| 120.74239| 31.628862|     江苏省|                华东|120.611176| 31.33
0664|  35.416|
|  KC1241|2020-05-07 09:30:00|        常熟南站|苏州火车站|       1|     江苏省|                华东| 120.74239| 31.628862|     江苏省|                华东|120.611176| 31.33
0664|  35.416|
|  ZS7530|2020-05-07 10:55:00|苏州北广场汽车站|   昆山周庄|       1|     江苏省|                华东|120.608475| 31.330946|     江苏省|                华东| 120.84788| 31.113
865|   33.18|
```

2.4. 数据不全检测

除了明显的缺失值以外，由于收集、清洗过程中的种种原因，原始数据还存在隐性的不全之处。例如，天津市仅在 2021 年 3 月到 4 月存在数据记录，其他时间段没有从天津出发的车次被记录在数据中。为此，我们对每个省都检测了有数据记录的月份，并统计哪些月份的省份数据最全，结果如下所示。

由此，我们选择了 2020 年 2-6 月以及 2021 年 3-4 月两个时间段作为我们的

主要分析时段。其中，2020 年 2-6 月数据共包含 30 个省的出行记录（没有港澳台地区和天津），2021 年 3-4 月的数据共包含 31 个省的出行记录（没有港澳台地区）。

三、 可视化分析——探究疫情与交通状况的联系与防疫规律

自 2020 年 1 月起，全球经历了一场新冠肺炎疫情的抗疫时期，而新冠肺炎这一传染性极强的病毒在过去一段时间内对于我国的交通出行也造成了极大的影响。因而我们将通过 PySpark 对交通大数据汇总统计，并对汇总统计数据可视化，以此来反映疫情对我国交通状况的影响。

3.1. 数据准备

通过**限定时间范围的逻辑索引、按列索引以及 groupby 汇总统计**等方法提取出了各个省份间的交通流动人次数据。而时间范围根据数据预处理中所述，2020 年 1 月到 6 月份，以及 2021 年 3、4 月份各个省的数据都较为全面，因而我们也将主要针对这两个时间段进行分析。

数据格式 1	各省情况					
省份	流入人次	流出人次	内部流动人次	行政地理划分	时间	历史感染情况
西藏自治区	287	3680	18195	西南	2020	……
……						

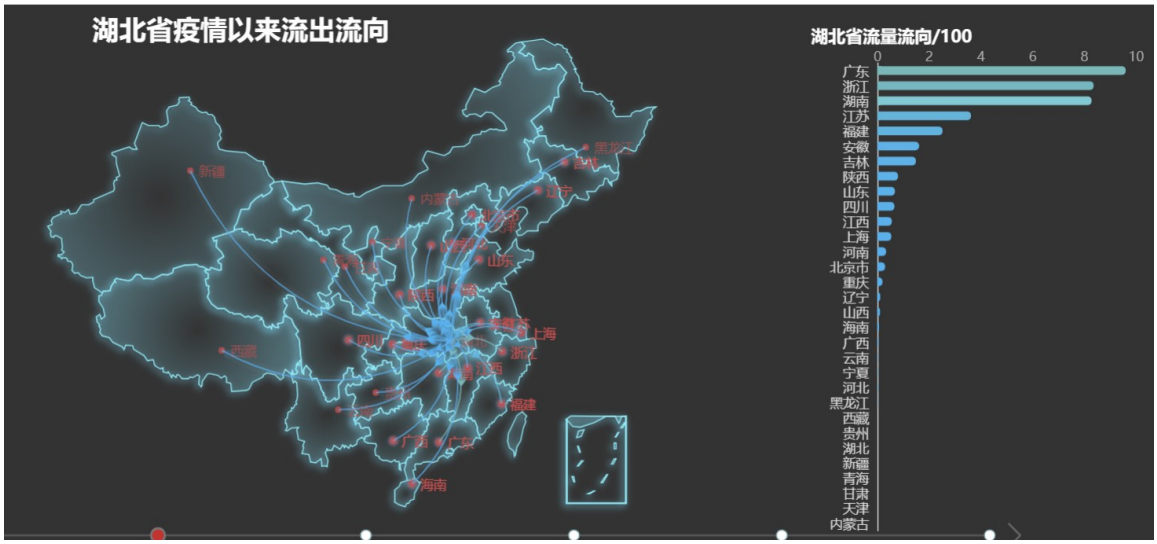
数据格式 2	省际流动		
省份	省份	流动人次	时间
西藏自治	北京	3680	2020

区			
……			

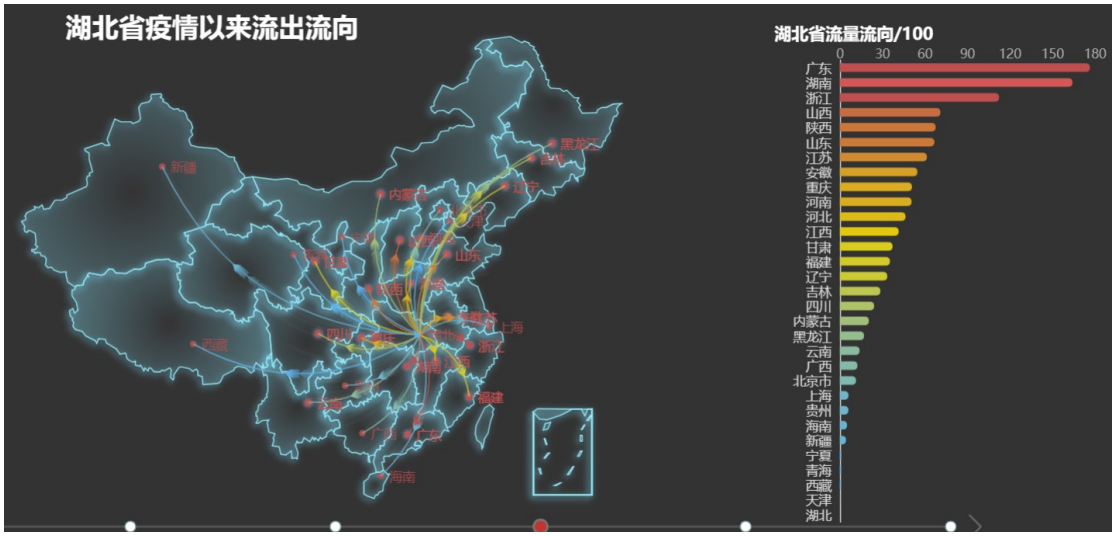
注：其中历史感染数据来自于 https://github.com/eAzure/COVID-19-Data/ 该项目从官方网站爬虫，爬取到的数据为各个省每日的累计确诊、新增确诊、治愈以及死亡人数，经过汇总预处理后得到合并到上述表格中的数据。

3.2. 疫情爆发半年内的交通状况改变

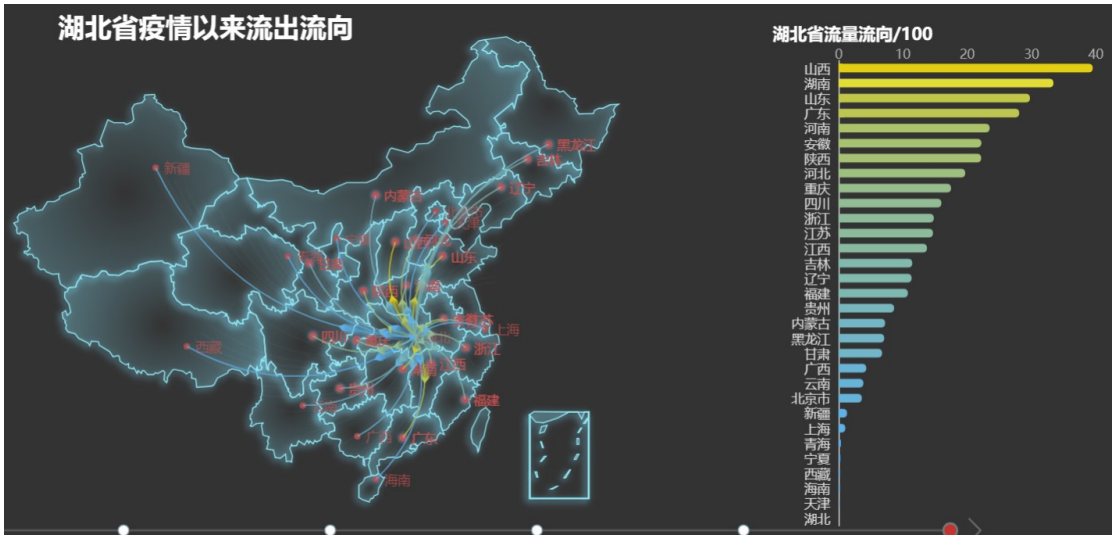
3.2.1. 以湖北省为中心的客流情况变化



2020 年 2 月 湖北省出发客流情况



2020 年 4 月 湖北省出发客流情况

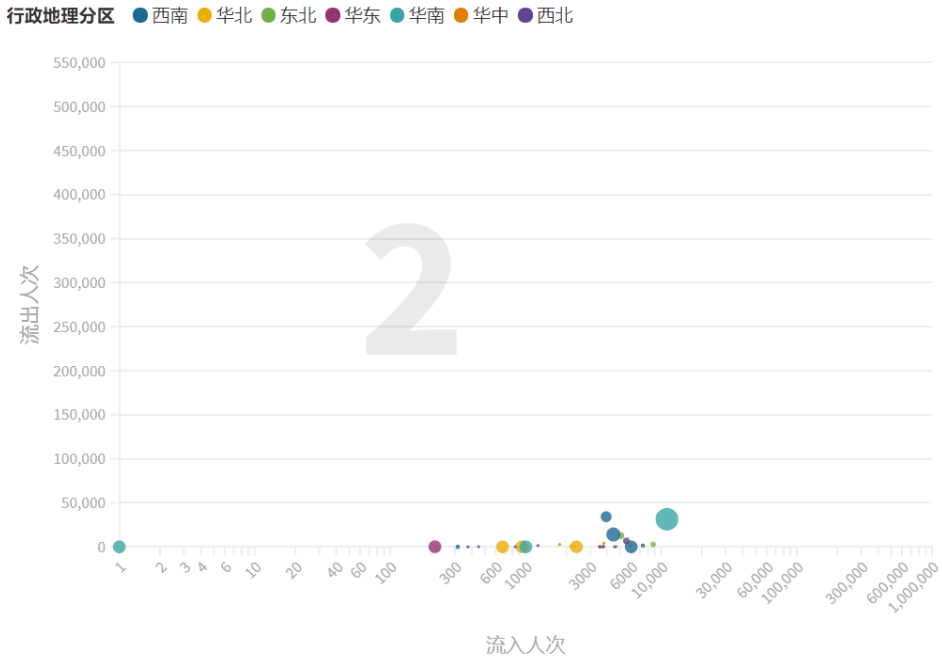


2020 年 6 月 湖北省出发客流情况

在从上面三个图可以看出，自从疫情爆发以来，以湖北省为中心，交通量急剧减小，说明疫情对于交通的打击之大。而可以看到，约 4 月份开始以湖北省为中心的交通流逐渐恢复正常，但在 6 月份左右，可以看到客流量又有一个下跌，通过查阅疫情历史新闻了解到，这是由于 6 月份以北京为首开始出现的疫情反弹所造成的。

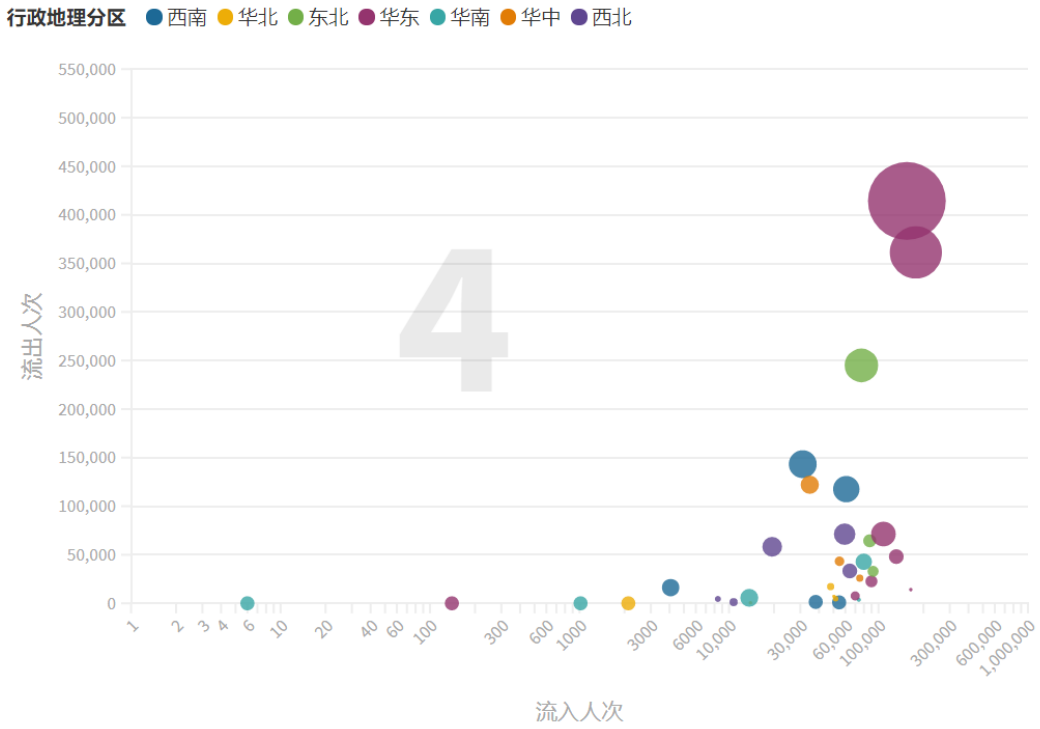
3.2.2. 各个地理行政区域交通受疫情影响情况

注：图中点的大小代表该省内部流动情况。



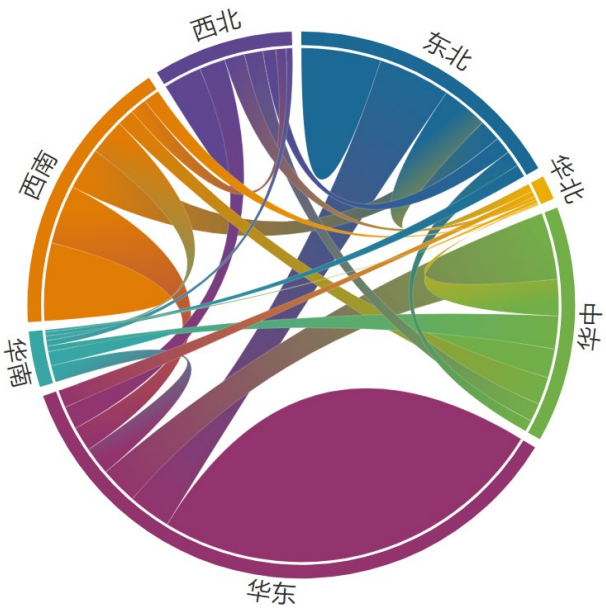
分地理区域各省交通受影响情况 2 月

从图中可以看到，在疫情刚刚爆发初期，各省的交通都受到影响很大，呈现出流出几乎没有的现象，（还保留了一部分流入可能是受疫情影响各地回家的人群）。



分地理区域各省交通受影响情况 4 月

从图中可以看到，在疫情爆发一段时间以后，交通恢复较先较快的地区为华东地区，这与华东地区经济较为繁荣（珠三角、江浙沪等）有较轻的联系，也与华东地区是劳动密集型产业区域，依靠较多的外来劳动力有关。还可以看到，受疫情影响其次恢复较快的地区是西南地区与东北地区，这些地区距离疫情重灾区湖北（华中地区一带）较远，从而疫情得到控制较快，交通恢复较快。总体来看，疫情对交通影响较大，范围广，但部分地区也较快的得到了恢复。

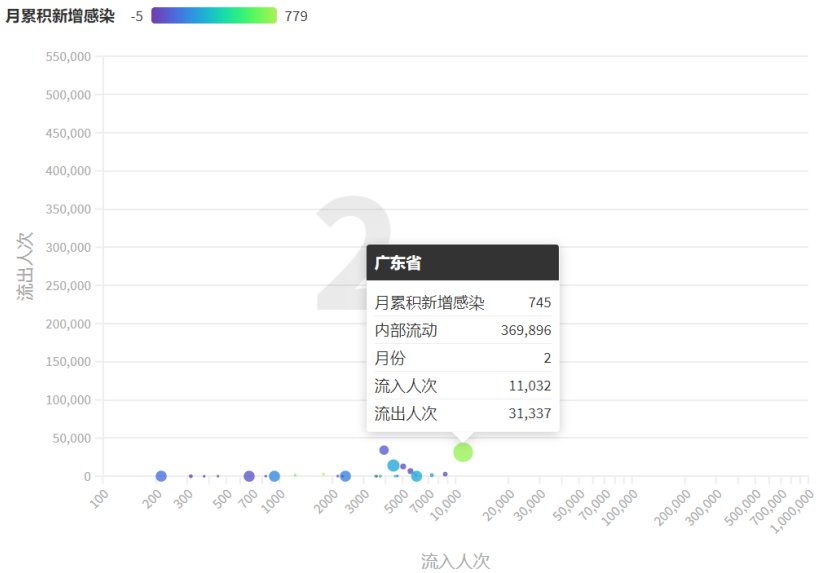


各地理区域相互之间的交通流桑基图

从桑基图也可以看出，受疫情影响最大的两个地区为华南、华北地区，并且从桑基图中我们也可以看到更多的信息，比如，华东地区虽然人口流量大，但是其多数均为内部流动，而非跨区域流动。但是与之相对的，东北地区则有着很多的跨区域流动，这一差别对其疫情状况也会有相应的影响。

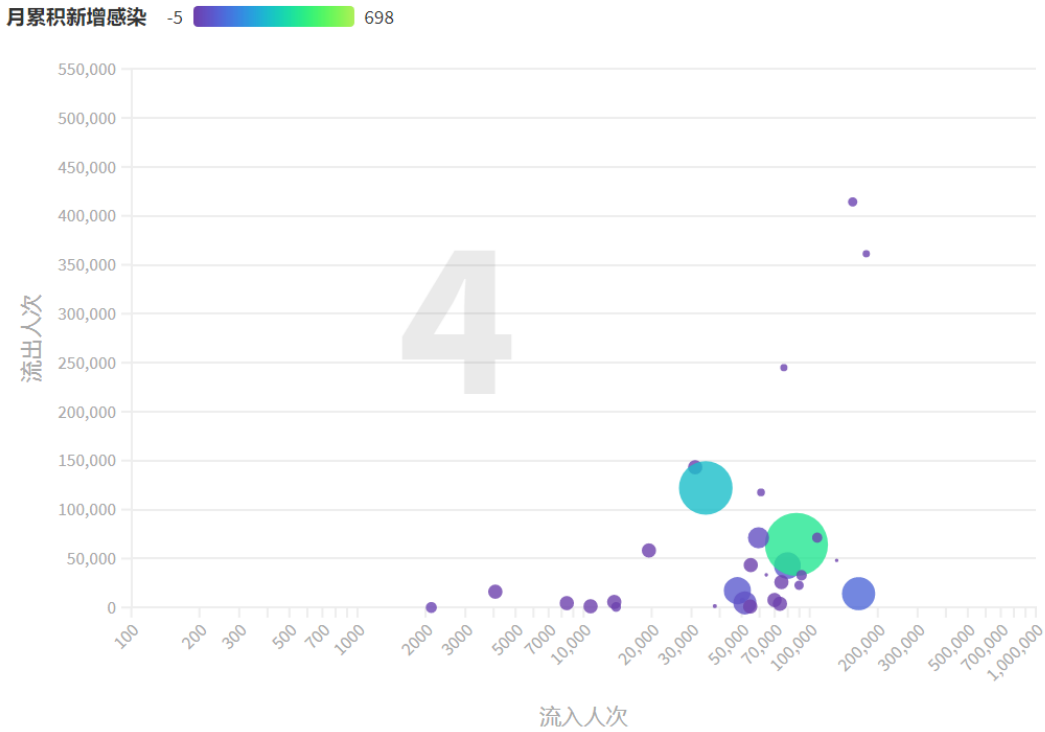
3.2.3. 各个地理行政区域交通与疫情控制关系

为了探究各个地理区域以及各个地理区域内的各省的交通情况与其疫情的发展控制情况的关系，我们进一步绘制了新增感染人数与交通流量情况的气泡图。



2 月份各省交通流量与感染情况气泡图（去除湖北省）

从图中可以看到，在 2 月疫情刚刚爆发之时，各省新增感染人数均较多，且新增感染人数呈现出与该省流入人次较强的关联性。这也反映出，在疫情初期，限制交通对阻止疫情进一步发展的重要性。



而时间来到 4 月份，可以看到，疫情已经基本得到控制，其中左上角为华东地区的浙江省和江苏省等地，其交通流动性较高，但疫情控制情况却较为良好，而右下角的绿色点为黑龙江省，其交通流通性也较高，但是疫情发展却并不乐观。究其原因，在上面的桑基图中也可以看出，华东地区的流动主要是内部区域性流动，因而整个地区相对安全。而黑龙江省则由大量跨区域流动，从而具有较高的输入性病例传播的风险。这也反映出，在后期控制疫情时，避免大规模跨区域性流动，谨防输入性感染与传播的风险是极为重要的。

3.2.4. 总结

从上述疫情发展与交通流情况的交互性分析，我们可以看出疫情会对交通造成较大的影响，但是这种影响是必要的，即在疫情初期全面的控制限制交通对于控制疫情发展具有很大帮助。另一方面，在防疫的后期，在交通上，要从

区域内部性流动开始恢复，而对于跨区域性流动仍要谨慎，跨区域性流动往往会带来输入性病例造成的本地感染与传播风险。

四、 建模分析——交通情况与经济状况等的关系

4.1. 探索车站网络中占据重要地位的车站

随着我国交通规模和客流量的迅速发展，越来越多的交通车站和线路超负荷运行,这使得我国对于城市轨道交通网络的安全运行和运营管理越来越关注。因此,研究轨道交通网中的车站重要度,不仅能够帮助管理者制定和改善管理措施,同时对路网规划和城市布局研究也有重要的意义。

本组基于 PageRank 和社区探测两种算法,以实际的交通网络作为模型基础，计算出车站的重要度排名，为管理者提供决策支持，对重点车站的把控有利于提高城市轨道交通运营管理和服务水平。

4.1.1. 数据准备

通过 SQL 将所有出行数据按照车站的分类汇总，统计每个车站作为始发站或者终点站的总行程数，以及该车站所在省份以及该车站经纬度等信息。数据展示如下：

```
+---+------------+--------+----------+---------+-------+
| id|     Station|Province| Longitude| Latitude|    num|
+---+------------+--------+----------+---------+-------+
|  1|    苍南西站|  浙江省| 120.38268| 27.49879|1230838|
|  2|        温州|  浙江省| 120.69936|27.994267|1196258|
|  3|  义乌火车站|  浙江省|120.042786|29.377123| 973030|
|  4|    东阳东站|  浙江省| 120.25049|29.259472| 929119|
|  5|  无锡客运站|  江苏省| 120.30845|31.592854| 616847|
|  6|    萧山机场|  浙江省| 120.43706|30.234344| 613546|
|  7|    东阳西站|  浙江省| 120.18615|29.298126| 558542|
|  8|  东阳横店站|  浙江省| 120.20354| 29.28787| 505864|
|  9|    南通东站|  江苏省| 120.89203|32.004845| 460216|
| 10|虹桥长途西站|  江苏省|120.845825| 32.00245| 430529|
+---+------------+--------+----------+---------+-------+
only showing top 10 rows
```

通过SQL提取数据中的所有的形成对信息,汇总车站与车站之间的行程总数。数据展示如下：

```
+------------------+------------------+-------+
|      StartStation|       ReachStation|    num|
+------------------+------------------+-------+
|            苍南西站|               温州|1101392|
|            东阳东站|          义乌火车站| 744609|
|            东阳西站|             杭州东| 252622|
|          东阳横店站|          义乌火车站| 216571|
|          东阳横店站|            萧山机场| 191131|
|          昌吉客运站|      乌鲁木齐高铁站| 136342|
|          岱山长途站|             宁波南| 136089|
|          中川机场站|兰州火车站（1号线）| 132122|
|            东阳西站|            萧山机场| 127423|
|乌鲁木齐高铁汽车站|          昌吉新客站| 123830|
+------------------+------------------+-------+
only showing top 10 rows
```

4.1.2. GraphFrames

图结构是一个可以解决很多数据问题的直观的方法。无论是遍历社会网络，餐馆推荐，或者是飞行路径，都可以通过图结构的上下文来快速理解所面临的问题: 顶点(Vertices)、边(edges)和属性(properties)。

再考虑到在本次大作业中我们所使用的车站乘客出行数据集，它天然涵盖了顶点——车站、边——行程、属性——行程数量等元素，因此，我们可以从数据集中提取出车站与行程的图结构数据集，以此研究车站的各种特征。

为此，**我们选择使用 GraphFrames 库进行分析，该类库构建在 Spark DataFrame 之上**，主要具备以下三个优点：一、统一的 API: 为 Python、Java 和 Scala 三种语言提供了统一的接口，这是 Python 和 Java 首次能够使用 GraphX 的全部算法。二、强大的查询功能：GraphFrames 使得用户可以构建与 Spark SQL 以及 DataFrame 类似的查询语句。三、图的存储和读取：GraphFrames 与 DataFrame 的数据源完全兼容，支持以 Parquet、JSON 以及 CSV 等格式完成图的存储或读取。

在 GraphFrames 中，图的顶点(Vertex)和边(edge)都是以 DataFrame 形式存储的，能够完整保存一个图的所有信息，又因 GraphFrames 内置了 GraphX 中的多种算法，我们可以立即利用这个优势，将库中函数例如 pageRank 等函数用于数据复杂查询及分析。

4.1.3. PageRank

4.1.3.1. 介绍

PageRank 算法是一个天才般的算法，原理简单但效果惊人。它是 Google 的两位创始人构建早期的搜索系统原型时提出的链接分析方法，他们借鉴了学术界对论文重要性的评估方法，即“谁被引用的次数多，谁就越重要”，由此提出了算法的工作原理：对到顶点的连接的数量和质量进行计数，从而估计该顶点的重要性，其核心思想有两点：

1) 如果一个节点被很多其他顶点链接到的话，说明这个顶点比较重要，也就是 PageRank 值会相对较高。

2) 如果一个 PageRank 值很高的顶点链接到一个其他的顶点，那么被链接到的顶点的 PageRank 值会相应地因此而提高。

4.1.3.2. 步骤

PageRank 算法可以简单分为两个步骤：

1) 初始化：通过链接关系构建起图结构，为每个顶点设置相同的 PageRank 值。

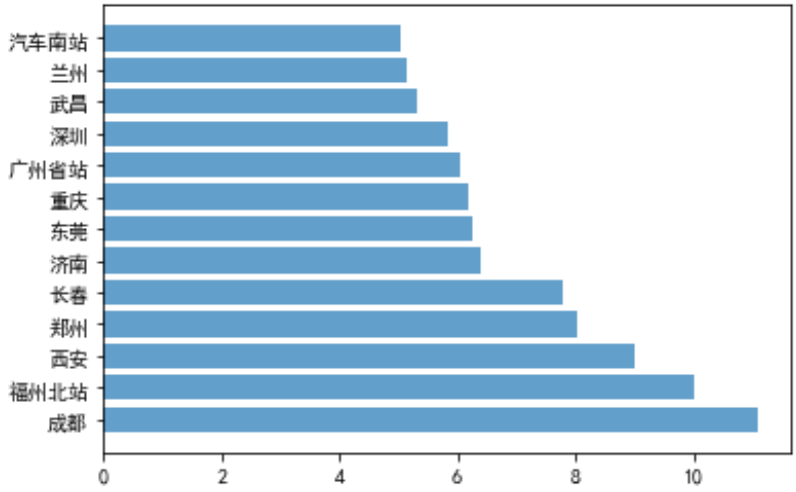
2) 迭代：进行若干轮的迭代计算，顶点当前的 PageRank 值不断更新，直至收敛到平稳分布，最终得到每个结点所获得的最终 PageRank 值。

4.1.3.3. 结果

在 GraphFrames 库中，调用 PageRank 函数，将返回 PageRank 值结果，作为新的 column 追加到 vertices DataFrame 中。根据结果，我们筛选出 PageRank

值大等于 5 的共计十二个站点进行展示。

	车站	PageRank	省份	经度	纬度
0	成都	11.097	四川省	104.066544	30.572269
1	福州北站	10.018	福建省	119.319130	26.111980
2	西安	8.996	陕西省	108.940170	34.341568
3	郑州	8.040	河南省	113.625366	34.746597
4	长春	7.793	吉林省	125.323547	43.817070
5	济南	6.397	山东省	117.120003	36.651215
6	东莞	6.249	广东省	113.751762	23.020536
7	重庆	6.181	重庆市	106.551559	29.563009
8	广州省站	6.046	广东省	113.252312	23.147921
9	深圳	5.851	广东省	114.057869	22.543098
10	武昌	5.316	湖北省	114.299835	30.548456
11	兰州	5.160	甘肃省	103.834305	36.061089
12	汽车南站	5.036	浙江省	119.982819	30.019184



分析此 12 个车站：其中成都、西安、重庆、广州省站等站点本身既是大城市，也是知名的旅游城市；福州北站、深圳、东莞、汽车南站（位于浙江省）等站点的重要性可能与外出务工人员的流动有关；其余郑州、长春、济南、武昌、兰州亦为各地区的重要交通节点。由此我们可以看出，本数据集以及 PageRank 算法具有一定的可信度，分析结果基本符合常识，此方法在实际问题的分析中具

有参考性。

4.1.4. 社区发现：标签传播算法

4.1.4.1. 社区发现

上世纪 60 年代，Herbert Simon 首先提出了复杂系统具有模块结构特性的概念，针对社区的研究由从子图分割问题演化而来，数十年来，各个领域的专家也都逐渐发现社区结构在各种复杂网络中的普遍存在性。

所谓社区，即是图内部连接比较紧密的节点子集合对应的子图，它反映的是网络中的个体行为的局部性特征以及其相互之间的关联关系，研究网络中的社区对理解整个网络的结构和功能起到至关重要的作用，并且可帮助我们分析及预测整个网络各元素间的交互关系。而给定一个网络图，找出其社区结构的过程，就叫做社区发现。

对于本数据集而言，大部分的数据挖掘方法只能将重点放在车站间的直接交通流量上，然而在整个车站网络内，可能存在某些车站，它们彼此间出于某些原因并无显性的直接关系，却以某种形式距离很“近”，这样的结果可能对于人文社科领域或交通路线规划中具有研究意义，也可能在商业推荐上起到一定效果，因此，我们可以尝试使用社区发现方法对数据集进行分析。

社区划分分为两种，各社区节点集合彼此没有交集的称为非重叠型（disjoint）社区，有交集的称为重叠型（overlapping），本文将采取非重叠型社区划分。

4.1.4.2. 标签传播算法

标签传播算法是不重叠社区发现的经典算法，其核心思想非常简单：相似的数据应该具有相同的 label。算法认为每个结点的标签应该和其大多数邻居的标签相同，将一个节点的邻居节点的标签中数量最多的标签作为该节点自身的标签（bagging 思想）。给每个节点添加标签（label）以代表它所属的社区，并通过

标签的“传播”形成同一个“社区”内部拥有同一个“标签”。

标签传播算法的最大的优点就是算法的逻辑非常简单，相对于优化模块度算法的过程是非常快的，不用 pylouvain 那样的多次迭代优化过程。标签传播算法利用自身的网络的结构指导标签传播，这个过程是无需任何的任何的优化函数，而且算法初始化之前是不需要知道社区的个数的，随着算法迭代最后可以自己知道最终有多少个社区。

4.1.4.3. 结果

经过社区发现，共将所有车站分为 4591 类，这里展示其中数量最多的前二十个标签。

	label	数量
0	5	346
1	101	327
2	1665	239
3	42	227
4	1024	206
5	217	194
6	78	158
7	645	138
8	523	132
9	69	130
10	249	127
11	1469	93
12	1411	93
13	893	75
14	347	65
15	120	54
16	408	54
17	962	50
18	138	49
19	34	44

再择取一个社区进行分析，这里选取社区 34，即上表中数量排行第二十的社区。首先查看其社区内省份分布情况：

	Province	count
0	辽宁省	14
1	吉林省	7
2	黑龙江省	6
3	浙江省	2
4	河北省	2
5	内蒙古自治区	2
6	重庆市	1
7	福建省	1
8	陕西省	1
9	山西省	1
10	甘肃省	1
11	四川省	1
12	山东省	1
13	新疆维吾尔自治区	1
14	河南省	1
15	青海省	1
16	江苏省	1

可以看到，该社区内大部分车站位于东北三省之内，这可以在一定程度上证明了社区发现的可靠性，同时得到了社区内部分来自其他省份的车站，或可进一步研究。

从另一个角度看，该结果可以理解为通过网络关系对车站分类，结果显示，地理关系近的车站更容易被分为同一类。因此，可以将其作为一种在没有相关省份车站数据集情况下对车站的一种无监督标注省份的方法。

4.1.5. 三角形计数

4.1.5.1. 介绍

TriangleCount 算法的工作是“统计每个顶点所在的三角形个数”。所谓“三角形”，即是图中两两相连的三个节点所构成的三角形结构子图，图中的三角形数量可以一定程度上反映图整体的稠密程度与质量，三角形计数为车站聚类提供了重要依据和信息。

4.1.5.2. 结果

	车站	PageRank	数量	三角形计数
0	苏州南站	江苏省	430302	1035
1	抚顺	辽宁省	152071	581
2	淮安北站	江苏省	172464	567
3	南通东站	江苏省	460216	501
4	泰州南站	江苏省	299376	474
5	常熟南站	江苏省	358850	461
6	庄河	辽宁省	96448	418
7	瓦房店	辽宁省	172329	405
8	青岛	山东省	37674	401
9	无锡客运站	江苏省	616847	378
10	丹东	辽宁省	175070	365

4.2. 探究公路交通与省级行政区经济状况的关系

为探究公路交通对省级行政区经济状况的影响，本组将数据集按照省份进行汇总， 使得每一条观测代表一个省份，在公路建设以及客运场景这两个维度下考虑可以选择的聚类变量，最终选择省内车站数量、车站空间分布，省内客运流量（省内客运场景下）、流量出度和流量入度（跨省客运场景下）作为聚类变量。在充分刻画 22 个省、5 个自治区、4 个直辖市（由于港澳台地区数据量过少，将其剔除）共计 31 个省级行政区的公路交通出行特点的基础上，对省级行政区进行聚类，得到基于公路交通出行特点的省级行政区聚类结果。在所得聚类结果的基础上，分析公路交通出行情况相近的省级行政区类，经济状况是否也是相近的。

4.2.1. 确定聚类指标

在确定聚类变量的时候，我们立足于最大程度地捕获蕴含在海量出行记录条中的信息，尽可能从多个视角上去思考可能可以选择的聚类变量，以保证聚类结果的准确性，使得类内城市的公路交通出行的同质性最大，类间的异质性最大。

4.2.1.1. 客运交通设施维度

一个省份的公路建设规模需要根据该省份的公路运输在全国综合运输体系中的作用，再结合地理环境条件来确定。考虑到指标的可获得性，本组用省内的车站数量、车站空间分布情况、客运交通类型来捕获该省的公路建设规模。其中车站数量一定程度上能够体现该省的公路设施数量，省内不同车站的经度与维度的方差能够一定程度上体现该省份车站的分散程度，从而反应省内公路建设的空间分布，省内各个车站作为起始站和终点站的购票座位类型例如上铺、下铺和普通座、商务座可以看作该省份的不同客运交通类型。

4.2.1.2. 场景刻画维度

一般而言，每个省份的公路交通出行都有省内客运和跨省客运两种场景。为更准确地获取每个城市在不同场景下的出行特征信息，我们用省内客运流量来捕获省内客运场景下的出行信息，用流量出度和流量入度来捕获跨省客运场景下的出行信息。其中，流量出度一定程度上能够体现从该省份流出的客流量大小，流量入度能够体现流入该省份的客流量大小。

4.2.2. 数据准备

考虑到指标的可获得性，最终利用 SQL 按照省份进行分类汇总，生成了如下指标：

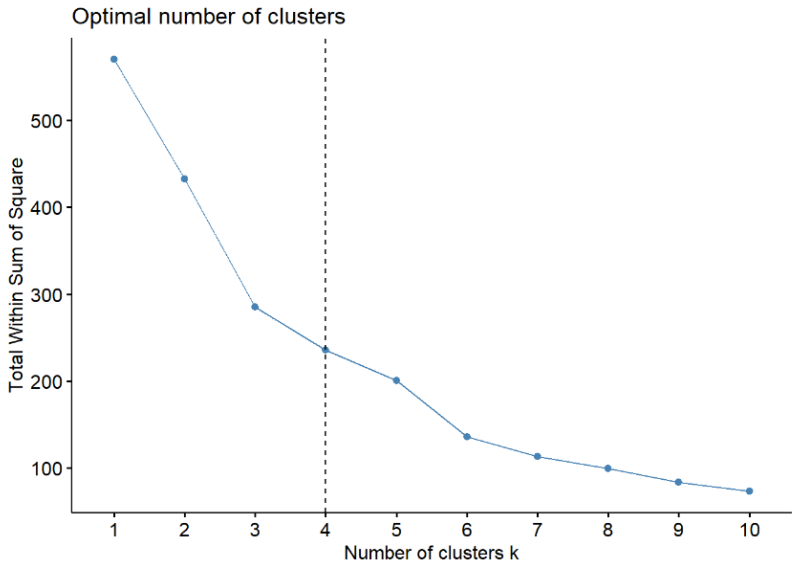
指标名称	指标含义	计算方法
车站数量	车站总数	对该省所有始发站或终点站去重后计数
车站经度分散程度	车站空间横向分散程度	计算该省不同车站经度的方差
车站纬度分散程度	车站空间纵向分散程度	计算该省不同车站经度的方差
流入人次	流入的客流量大小	汇总终点站为该省的出行观测数量
流出人次	流出的客流量大小	汇总始发站为该省出行

		观测数量
内部流动人次	内部客流量大小	汇总始发站及终点站均为为该省的出行观测数量
始发站普通座车次（商务座上铺、下铺、其他）	始发站的交通类型	汇总始发站为该省且座位类型为普通座（商务座、上铺、下铺和其他）的车次
终点站普通座车次（商务座上铺、下铺、其他）	终点站的交通类型	汇总终点站为该省且座位类型为普通座（商务座、上铺、下铺和其他）的车次

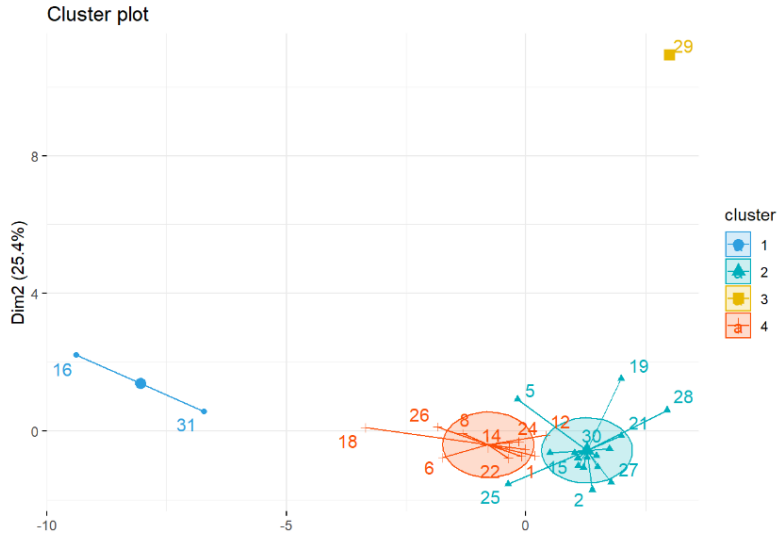
4.2.3. K-means 聚类分析

提取出省份数据后，鉴于其规模较小，我们使用 R 语言进行聚类分析。

首先，利用函数确定最佳聚类数目。从指标上看，可以选择坡度变化不明显的点最为最佳聚类数目，发现聚为四类最合适，在后续尝试中也证实其效果较优，故保留。



确定类数后，采取 K-means 聚类方法进行聚类分析。将具有有效数据的 31 个省份（香港、澳门、台湾三地区存在数据缺失）分为四类：



第一类：″北京市″、″重庆市″、″福建省″、″甘肃省″、″广西壮族自治区″、″海南省″、″河北省″、″吉林省″、″江西省″、″内蒙古自治区″、″宁夏回族自治区″、″青海省″、″山西省″、″上海市″、″天津市″、″西藏自治区″、″云南省″

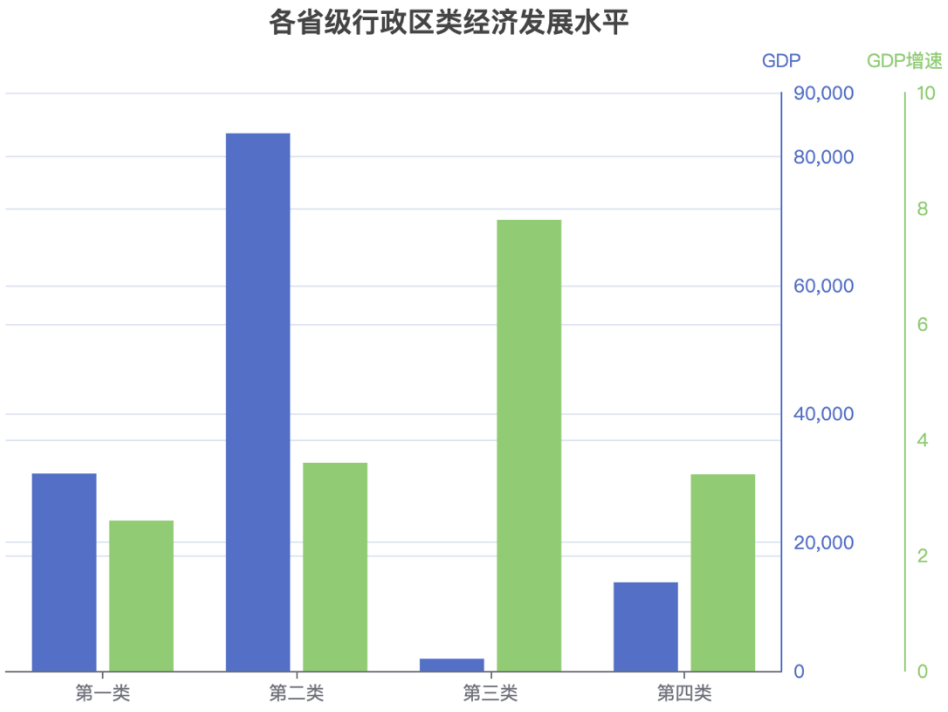
第二类：″江苏省″、″浙江省″

第三类：″新疆维吾尔自治区″

第四类：″辽宁省″、″山东省″、″陕西省″、″四川省″、″安徽省″、″广东省″、″贵州省″、″河南省″、″黑龙江省″、″湖北省″、″湖南省″

将聚类结果显示在地图上：

从图中看出，聚类结果基本符合实际，新疆地区因其特殊性独居一类，江浙地区分为一类，中部地区大致分为一类，其余地区分为最后一类。



绘制各省级行政区类的经济发展水平状况(GDP 和 GDP 增速)。可以看到，根据公路交通出行特点的省级行政区聚类结果得到四个省级行政区类,这四个省级行政区类的 GDP 以及 GDP 增速差异较为明显。可以得出，在公路建设以及客运场景这两个维度下公路交通出行特点相近的省级行政区，经济状况也是相近的。

五、 总结

本研究对中国各省的交通出行记录数据进行了综合分析，使用 PySpark 对共计一亿余条的乘客出行记录进行了数据清洗、特征提取等流程，并通过可视化和数学建模方法挖掘了数据中隐藏的规律，如不同地区乘客的出行特征、各车站的交通重要性、省份交通状况与经济状况的关联等，得出了高可信度的结论。