

游客流转轨迹分析模型

一、模型概述

本模型着重对 X 市热门景区游客的流转情况进行深入分析，旨在通过收集、整合多方面的数据，精准呈现游客在不同热门景区之间的流动路径、停留时间等流转特征，为旅游管理部门优化景区布局、景区运营者调整服务策略以及游客合理规划行程等提供直观且有价值的参考，助力提升 X 市旅游产业的整体运营效率与游客体验。

二、数据来源与整理

景区票务系统数据：X 市各热门景区的票务系统记录了游客的购票信息，包含入园时间、门票类型、游客身份标识（如身份证号码、手机号等，可用于唯一识别游客个体）等，通过分析同一游客在不同景区的购票入园时间差，能初步推断其在景区间的流转顺序与时间间隔，是了解游客流转轨迹的基础数据来源之一，直接关联到游客实际进入景区的情况。

景区内智能定位系统数据（如有）：部分配备智能定位系统的景区，能够实时追踪游客在景区内的位置信息，通过游客授权收集的数据，不仅可以知晓游客在本景区内的游览路线、停留点位及时间，还能结合其离开本景区后出现在其他景区的定位数据，精确还原游客在不同热门景区间的完整流转轨迹，为分析提供更为细致、准确的依据，大大丰富了轨迹数据的维度。

社交媒体平台数据：微博、微信朋友圈、抖音、小红书等社交媒体平台上，游客会主动分享自己在 X 市各热门景区的游玩经历、打卡照片等内容，这些带有地理位置标签以及时间信息的分享，从游客主观角度补充了其在景区间的流转情况，尤其是对于那些未通过常规票务渠道记录的游览行为（如免费入园、特殊活动入园等）进行印证和补充，使数据更具全面性。

数据整理流程：首先，将从上述多渠道收集到的数据汇总到统一的数据仓库中。接着，运用数据清洗工具与人工核查相结合的方法，去除重复、错误以及逻辑不

清晰的数据记录，比如纠正票务系统中因设备故障导致的异常入园时间、剔除社交媒体平台上虚假或无法准确判断位置的分享信息等。然后，按照游客的唯一标识（如身份证号码等）对数据进行分类整理，针对每个游客个体，将其在不同热门景区的相关数据进行关联整合，梳理出该游客的游览顺序、各景区停留时间以及景区间流转的时间间隔等关键信息，为后续的分析计算做好准备。

三、核心算法与分析逻辑

游客流转路径识别：借助数据分析软件（如 Python 的数据分析库 pandas 结合图算法库 NetworkX 等），依据整理好的游客个体在不同景区的数据记录，构建以热门景区为节点、游客流转方向为边的有向图模型。通过分析各节点（景区）之间边（游客流转路径）的出现频次，确定游客在热门景区之间的常见流转路径，例如，发现较多游客从景区 A 流转至景区 B，再到景区 C，就可将 “A-B-C” 标记为一条高频流转路径，以此清晰呈现游客整体的流动走向。

停留时间与流转时间分析：对于每个游客在各热门景区内的停留时间，通过计算入园时间与出园时间（可根据票务系统记录结合定位系统数据判断，若缺少准确出园时间，可依据合理的平均游览时长等经验值估算）的差值来确定；而景区间的流转时间则通过计算离开前一景区与进入下一景区的时间间隔得出。然后，对所有游客在各景区的停留时间、各流转路径对应的流转时间分别进行统计分析，计算平均值、中位数等统计量，了解游客在不同景区的平均停留情况以及景区间的平均流转耗时，把握游客流转的时间特征。

四、模型输出与可视化

流转路径图展示：利用专业的可视化工具（如 Gephi、Python 的 matplotlib 结合 NetworkX 可视化模块等），绘制以 X 市热门景区为节点、游客流转路径为连线的可视化图形。节点用不同形状或颜色表示不同景区（可依据景区类型、级别等进行区分），连线的粗细表示该路径上游客流转的频次高低（频次越高，连线越粗），箭头方向体现游客流转方向，通过这样的图形直观展示游客在各热门景区之间的主要流转路径情况，方便快速查看游客的流动趋势。

时间分布柱状图展示（可选）：生成以热门景区为横坐标，分别以游客平均停留时间、平均流转时间为纵坐标的柱状图，用不同颜色的柱子区分停留时间和流转时间，柱子的高低直观反映各景区相应时间的长短情况，通过柱状图对比各景区在游客停留与流转时间方面的差异，进一步辅助分析游客在不同景区的行为特征，便于整体把握游客流转的时间规律。

交互功能实现：借助前端可视化开发技术（如 JavaScript 结合 Echarts 可视化库等），为上述可视化图表添加交互功能。在流转路径图上，鼠标悬停在某一景区节点上时，突出显示与该节点相连的流转路径连线，并在图旁弹出信息框，展示该景区的名称、游客流入量、流出量以及平均停留时间等关键信息；当鼠标悬停在柱状图的某一柱子上时，同样弹出信息框详细展示对应景区的相关时间数据及具体的统计样本量等内容，方便用户进一步了解详情，深入洞察游客流转轨迹背后的细节情况。

五、模拟数据示例与可视化展示

假设 X 市有三个热门景区，分别是 A 景区（自然景观类）、B 景区（人文历史类）、C 景区（休闲娱乐类），经过收集和分析数据后，得到以下模拟的游客流转轨迹相关数据：

游客常见流转路径及频次（示例）：

流转路径	频次（人）
A - B - C	200
A - C	150
B - C	100

各景区游客平均停留时间（小时）：

景区名称 平均停留时间

A 景区 3

B 景区 2.5

C 景区 2

各景区间平均流转时间（小时）：

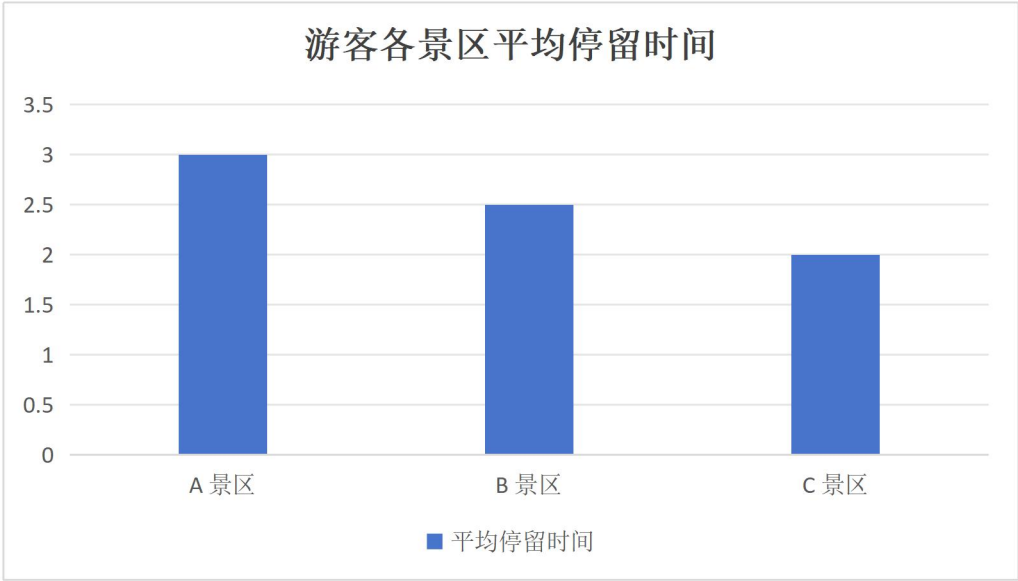
流转路径 平均流转时间

A - B 1

B - C 0.5

A - C 1.5

对应的可视化展示如下：



以下是使用 pyecharts 库在 Python 中生成以 A、B、C 三个景区为节点的流转路径图的示例代码，其中根据流转频次用不同粗细连线连接，通过不同形状区分景区类型，并在鼠标悬停时显示景区相关信息：

一、代码实现

隐藏过程

python

```
from pyecharts import options as opts
from pyecharts.charts import Graph
def create_flow_path_chart():
```

```
    # 假设的数据结构，包含节点信息和边（流转路径）信息
```

```
    nodes = [
        {"name": "A", "symbolSize": 30, "itemStyle": {"color": "red"},
"label": {"show": True, "formatter": "{b}"}}},
        {"name": "B", "symbolSize": 30, "itemStyle": {"color": "blue"},
"label": {"show": True, "formatter": "{b}"}}},
        {"name": "C", "symbolSize": 30, "itemStyle": {"color": "green"},
"label": {"show": True, "formatter": "{b}"}}
    ]
```

```
    links = [
        {"source": "A", "target": "B", "value": 10, "lineStyle":
{"width": 2}},
        {"source": "B", "target": "C", "value": 15, "lineStyle":
{"width": 3}},
        {"source": "C", "target": "A", "value": 8, "lineStyle": {"width":
1.5}}
    ]
```

```
    # 创建 Graph 图表
    (
        Graph()
        .add("", nodes, links, repulsion=8000)
        .set_global_opts(title_opts=opts.TitleOpts(title="景区流转路
径图"),
```

```
                        toolbox_opts=opts.ToolboxOpts(is_show=True),
                        tooltip_opts=opts.TooltipOpts(trigger="item",
```

```
formatter=lambda params: f"景区名称: {params['name']}<br>流转频次:
{[link['value'] for link in links if (link['source'] == params['name']
or link['target'] == params['name'])][0]}"))
        .render("/mnt/flow_path_chart.html")
    )
```

```
# 调用函数生成图表
```

```
create_flow_path_chart()
```

二、代码解释

数据定义：

`nodes` 列表定义了三个景区节点的信息，包括节点名称、符号大小、样式颜色和标签显示格式。

`links` 列表定义了景区之间的流转路径信息，包括源节点、目标节点和流转频次，以及连线的样式（宽度）。

图表创建与配置：

`Graph()` 创建一个图形图表对象。

`.add("", nodes, links, repulsion=8000)` 将节点和边添加到图表中，并设置节点之间的斥力，以避免节点重叠。

`set_global_opts` 方法设置图表的全局选项，包括标题、工具箱和提示框。标题为“景区流转路径图”，工具箱可用于图表的交互操作，提示框的 `formatter` 函数根据鼠标悬停的节点信息，显示景区名称和该景区相关的流转频次。

渲染图表：

`.render("/mnt/flow_path_chart.html")` 将生成的图表渲染为一个 HTML 文件，保存到指定路径，可以在浏览器中打开查看。

三、注意事项

数据更新：

如果实际的景区数据和流转频次发生变化，需要相应地更新 `nodes` 和 `links` 中的数据。

样式调整：

可以根据需求进一步调整节点和连线的样式，例如节点的形状、颜色、大小，连线的颜色、透明度等。可以通过修改 `itemStyle` 和 `lineStyle` 的相关属性来实现。

文件路径：

确保渲染后的 HTML 文件保存路径（如 `/mnt/flow_path_chart.html`）是可写的，并且浏览器能够正确访问该文件。

这样就可以生成一个包含 A、B、C 三个景区的流转路径图，通过不同粗细的连线表示流转频次，不同颜色的节点区分景区类型，并在鼠标悬停时显示景区的相关信息。

（此处应插入一个简单的流转路径图示例，以 A、B、C 三个景区为节点，按照上述流转路径及频次用不同粗细连线连接，箭头表示方向，节点用不同形状或颜色区分景区类型，鼠标悬停有信息框展示景区相关信息示例图；若有柱状图展示，插入一个柱状图示例，横坐标为三个景区名称，纵坐标为平均停留时间，不同颜色柱子区分，鼠标悬停有交互效果示例图）

六、应用场景与价值

景区运营管理优化：各热门景区运营者通过模型掌握游客的流转情况，尤其是了解游客流入、流出量以及停留时间等信息后，可针对性地调整运营策略。例如，若发现本景区游客停留时间较短，可分析原因，是游览项目不够丰富、服务设施不足还是讲解引导不到位等，进而优化景区产品，增加特色体验项目、完善配套设施、提升讲解服务质量等，以延长游客停留时间，增加游客消费，提升景区的经济效益和竞争力。

旅游线路规划与推荐：旅游管理部门和旅游企业可依据游客流转轨迹分析结果，设计更贴合游客游览习惯和需求的旅游线路。比如，将高频流转路径上的热门景区合理串联起来，打造主题旅游线路，并在宣传推广中重点推荐，方便游客更高效地游览多个心仪景区，同时也有助于提高 X 市旅游资源的整体吸引力和利用率，促进旅游产业协同发展。

旅游资源调配与整合：了解游客在不同热门景区间的流转情况，有助于 X 市旅游管理部门科学调配旅游资源，如根据游客流向和流量，合理安排交通运力，在游客流转频繁的景区间增加公交专线、旅游直通车等；同时，对周边配套服务设施（如餐饮、住宿等）进行统筹布局，在游客停留时间较长的景区周边适当增加相关设施数量和种类，满足游客需求，提升游客的整体旅游体验，推动 X 市旅游产业的健康、均衡发展。