参赛队员姓名：方书田

中学：上海包玉刚实验学校

省份：上海市

国家/地区：中国/华东

指导教师姓名：张笑钦教授
Daniel Mac Leon

论文题目：基于 GPS 大数据时空特征识别的出租车高效益寻客多目标优化调度策略

基于 GPS 大数据时空特征识别的
出租车高效益寻客多目标优化调度策略

摘要：

城市居民搭乘出租车出行已成为日常生活的一部分，以出租车为代表的浮动车具有客源和车辆分布不均、时空随机性大等特点，常导致车辆空载率过高、居民打车等待时间过长等现象。随着智能运输系统（ITS）技术的发展，出租车高效益寻客模式的研究，已成为智慧交通领域最热门的课题之一。但现有的研究成果大多关注提升出租车载客数量，忽视对载客效益的分析和评价，对于以提升整体效益、达成供需负载均衡为目标的高效益客源推荐策略等方面的研究，还存在空白。

本文以纽约市出租车 GPS 大数据为数据源，辅以时序图和散点图等时空特征识别技术，研究影响出租车高效益寻客的各类指标，确定以高效益寻客和供需均衡为多重目标，构建了出租车高效益实时寻客策略，从全局考虑并兼顾司机个人效益，对空载出租车进行实时调度，推荐到最佳效益载客区域，提升整体效益，缩短乘客打车等待时间，达到经济效益和社会效益的双提升。本文最后介绍了原创的多层网格划分优化的设计方法，规避因地理环境等特殊情况对调度策略的影响，采用历史数据拟合给出两点之间行驶距离的精确值，替代了传统的路径计算方法，为距离求解问题提供了新的解决思路。本文的研究方法和成果具有普适性，可以推广复制到任何有浮动车 GPS 数据基础的城市，具有广泛的应用前景。

关键词： GPS 大数据，时空特征，供需均衡，多目标优化，高效益寻客，预测调度

The Multi-objective Optimization of Predicative Scheduling for High Profitable Passengers Seeking Based on Spatial-temporal feature of Taxi GPS Big Data

Abstract:

Serving as the unreplaceable role in daily transportation, Taxi cabs facilitate people's life and tarry them in the meantime, basically due to the high empty loading ratio and long waiting time for the factors of unpredictability and uneven vehicle distribution. As the Techniques of ITS (Intelligent Transport System) evolve and refine, more researches spotlight on the method of guest-finding, resulting in prevalence. However, so far, none of the research ever analyzes or evaluates the efficiency of taxi loading profits from the aspect of taxi drivers or even the strategy of balancing demand-supply in the case of taxi guest loading from the aspect of government and companies. This paper aims to investigate the indices that impact the efficiency of guest-finding qualified by the multiple objectives (balance of supply and demand, maximum profits, and lowest cab vacancy time) with the help of feature recognition techniques (Sequence Diagram, Scatter Diagram, Contour Diagram) and the annual data of New York City. Ultimately, a credible guest-finding strategy to attain the possible highest profits has been settled. From the point of whole, the strategy makes up to the real-time taxi dispatching, real-time recommendation of best pick-up zone. Besides benefiting the citizens from saving time to hail the taxi, this strategy improves the profits of taxi cabs as a whole, reaching the highland of economic and social effectiveness. At the last part of the paper, an innovative multilayer grid to divide the city into zones is designed. This, with the accurate distance between two points regressed from the history data, evades the negative impacts of possible physical obstacles and substitutes the traditional route calculation, providing a new idea of optimizing the driving routes. The analyzing methods and results provided by this paper adapt various cases that at least have real-world vehicle transporting data. Currently, there are over xx% of city embarking the vehicle data collecting programs, which facilitate and disseminate our methods to a larger scale.

Keywords: GPS Big data, Spatial-temporal feature, Multi-objective Optimization, High profitable Passengers seeking, Predicative Scheduling

目 录

一、 项目背景

选择交通大数据作为研究课题其实非常偶然，那是一天傍晚，我和妈妈坐出租车从市区回学校，路上我随口问司机，空载时怎么寻客？他说也没有什么好办法，老司机纯粹凭经验，新司机只能靠运气，言语中充满着工作的艰辛，这深深触动了我。我曾接触过交通 GPS 大数据，也学过数学建模，于是设想能否借助数学工具来提升司机经济效益，减少城市资源的浪费，这个想法得到指导老师的赞同和鼓励。我阅读了大量的文献资料，对项目的背景和意义有了更深入的理解和思考，觉得这课题非常有价值，也具备研发可行性。

1、城市交通除公交、地铁之外，出租车在人们的出行中扮演着非常重要的角色。但现实中常出现这样的尴尬情景：一边是出租车满大街转悠，另一边是乘客打不到车；一边是出租车扎堆，另一边是一辆车也没有。这种情况随着“滴滴”、“首汽”等网约车平台的出现有所改观，但营运车辆空载后到接单的这个时段依然存在着盲目行驶，热点区域车辆扎推等现象，对于乘客而言，依然存在着打车等待时间过长等问题，城市交通效率有待提升。

2、城市居民出行具有随机性，不同时间、不同区域的乘客分布也不均匀，并且会随着城市的发展、道路的拓展以及周边环境的变化而快速变化，依靠经验实现出行供需的精准对接是不可能的，必须借助大数据的收集和分析研究才能实现。出租车 GPS 大数据的收集是实时且透明的，它记录了乘客的上车时间、下车时间、上车地点、下车地点、行驶里程、费用等信息，这为课题挖掘乘客随机行为背后的出行规律，实现以出租车运营效益、运力和交通资源均衡分布为多重优化目标，制定实时高效寻客策略提供了基础数据。

3、目前，国内外借助出租车 GPS 数据开展寻客策略研究，大多关注载客数量或者载客路径的推荐，往往会出现载客数量提升，但效益没明显提升的情况，忽视了司机的核心利益。另外，对于出租车公司和政府公共部门关心的提升整体经济效益，提高城市运输能力等方面的研究，还存在空白。本文研究重点聚焦于单位时间效益最大、避免车辆扎堆、空载过长等问题，兼顾司机个人利益、出租车公司集体效益、公共部门运力提升等要求，填补了之前研究的空白和不足。

二、 国内外研究现状

出租车 GPS 数据包含位置、时间、费用等多方面信息，已成为交通领域课题研究的基础[1]，主要有两个研究方向：一是城市道路交通状况分析研究；二是交通行为模式挖掘研究，寻客策略属于后者的范畴。目前，司机寻客策略研究已取得多项成果，大致归为三类：通过挖掘司机行为特征[2] [3]，进行寻客推荐；通过分析客源时空分布特征[4] [5]，进行寻客推荐；研究路径推荐算法[6] [7]，进行寻客推荐。但以上研究重点集中在载客热点区域推荐和推荐路径算法等方面，很少有提升司机运营效益寻客策略的研究，更未查到从城市整体交通资源负载均衡的角度进行高效益寻客推荐的研究，而这两方面恰恰是寻客策略所要解决的根本性问题。本文将在已有的研究成果的基础上，量化客源效益指标，对 GPS 大数据进行时空分析和数据挖掘，建立客源流量和客源效益的时空特征库，以提升单位时间效益为优先目标，以交通资源均衡分布和提升运力为辅助目标，进行多目标优化，充分考虑整个城市出租车整体的供需情况、出租车到达推荐区域的时间成本等，建立高效益寻客预测模型，为司机提供实时高效益寻客策略，为车辆运营机构提供在线车辆的调度策略，从而达到提升司机收益和乘客满意度。

三、 数据源分析和预处理

大数据在数据采集和传输过程中，经常会由于干扰和人为原因造成数据丢失或失真，这些错误的数据记录被称为异常值。我们研究的大数据来自纽约市出租车GPS数据库[8]，包括 2016 年 1 月到 6 月约 7000 万条的黄色出租车和 900 万条绿色出租车载客记录，根据课题研究需要，提取了上车时间、上车经度、上车纬度、下车时间、下车经度、下车纬度、基本时程费、总消费金额、行驶路程在内的 9 个字段信息。为了保证数据的客观性和真实性，提高数据挖掘的质量，获得更准确的分析结果，我们对提取数据进行预处理，包括异常数据的修正和删除。其中删除有两种情况：一是直接删除，如上下车经纬度、上下车时间信息缺失或错误，红字冲正（退款）造成金额数据为负数等；二是修正无效后的记录删除。文本重点讨论如何进行数据修正。

3.1 基于莱以达（Pauta）准则的异常数据检测

异常数据包括缺失数据和失真数据，前者容易判断，后者比较隐蔽，需要特定方法进行检测，这里采用莱以达（Pauta）准则[9]进行异常值检验。以99%的置

信概率为标准，以三倍标准偏差为极限，若超此界限，则认定为非随机误差的样本，是需要修正的异常数据，需要做异常标识，待下一步修正处理。

3.2 多元线性回归实现数据修正

采用多元线性回归方法[10]分析 GPS 大数据，发现存在多条线性相关表达式，如：基本时程费与行驶路程、行驶时间的相关性；行驶路程与两点直线距离的相关性；行驶路程与行驶时间的相关性等。前两者的 R^2（可决系数）都在 95%左右，拟合优度很高。下面以基本时程费与行驶路程、行驶时间的相关性分析为例，实现异常数据的修正。

对剔除异常数据后的 GPS 数据进行金额、路程、时间等关键字段特征分析，设 X_1 为行驶时间，X_2 为行驶距离，Y 为收费金额，采用 MATLAB cftool[11]工具，以经验公式（3-1）去拟合，可确定回归方程系数 a_0，a_1，a_2，$R^2=0.9553$。样本数据散点和拟合曲面[11]如图 3-1。

$$Y=a_0+a_1X_1+a_2X_2 \qquad (3\text{-}1)$$

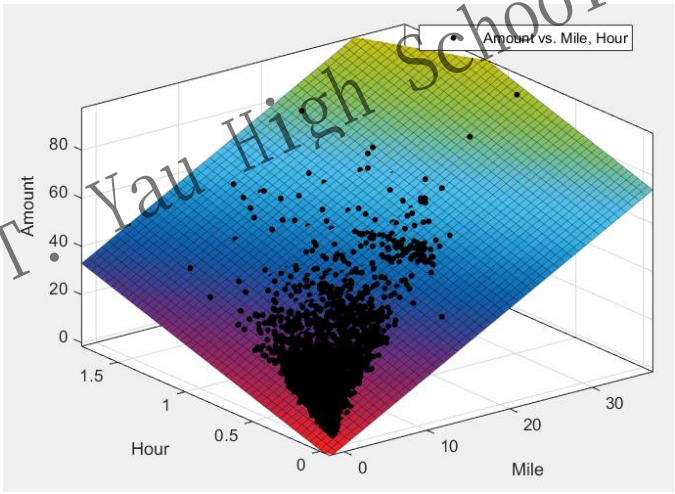

图 3-1 样本数据关于时程费与行驶时间、距离的散点拟合曲面图

可见，基本时程费与时间、路程具有较高的相关性，这三项数据仅有一项异常，可根据公式 3-2 实现有效回填。数据预处理完成质量保障了后续研究工作的精确度。

$$\begin{cases} X_1 = \dfrac{Y - a_2 X_2 - a_0}{a_1} \\ X_2 = \dfrac{Y - a_1 X_1 - a_0}{a_2} \end{cases} \qquad (3\text{-}2)$$

四、 高效益寻客指标的定义和量化

提高出租车的经济效益，关键在于客源带来的单位时间效益高，车辆在单位时间内载客的次数要多[12]。其中，单位时间收入定义为效益指标，取决于行驶路程和行驶时间，该指标在一定程度上还能反映当前道路的通行情况；换乘时间间隔即单位时间内载客的数量定义为运力指标，该指标取决于客流量。

传统的出租车寻客策略大多数只关注单位运力指标，很少考虑单位效益指标。这样往往会出现载客数量很多，但每单金额少收入反而不高，或者单次载客收入很高但到下次载客所花的时间很长，导致单位效益下降。本文研究的高效益寻客策略，要求综合考虑单位效益和单位运力。我们以客流量和客源效益为研究对象，借助 GPS 大数据，量化效益指标和运力指标，通过时序分布图[13]和区域散点图[11]，研究高效益客源关于时段和区域的分布情况，验证量化指标的准确性，为课题研究指明方向。

4.1 出租车 GPS 大数据的降维计算

面对 GB 级别的出租车 GPS 大数据，我们希望找出一些基本规律，采用类似于同类项合并方式，降低数据维度，相似度分析提供了很好的处理手段。对每小时流量分布做相似度分析，发现半年内所有周一的出行规律几乎一致，同理所有的周二、周三、周四、周五、周六和周日也具有高相似性的出行规律，相关系数高达 0.96 以上，详细见附录。由此得出结论：出租车出行规律与星期几相关，与日期无关。对出行量做均值处理，标定 $\overline{Q_i}$ 为星期 i 的出行流量

$$\overline{Q_i} = \frac{\sum_{j=1}^{j_{\max}} Q_{ij}}{j_{\max}} \qquad (4\text{-}1)$$

将 $\overline{Q_i}$ 按小时间隔做时序分析，见图 4-1。可以发现周一到周五的出行曲线非常相似，但周五晚上的出行特征和周六晚上接近，周日晚上的出行特征和周一晚上接近，这个特征和我们日常生活情况非常吻合，为了分析结果更准确，我们将时间

坐标平移 6 个小时，将周日 18：00 以后与周五 18 点以前拟合为工作日曲线，周五 18：00 到周日 18：00 拟合为周末曲线，如图 4-2。

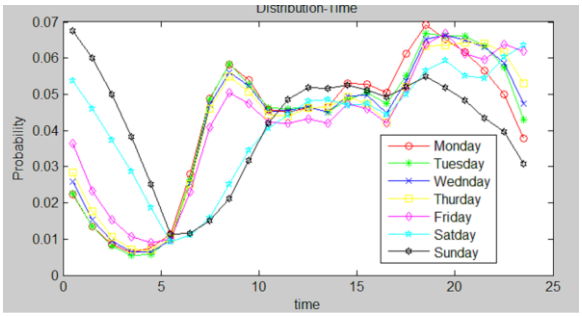

图 4-1 周一至周日出行特征时序分布图

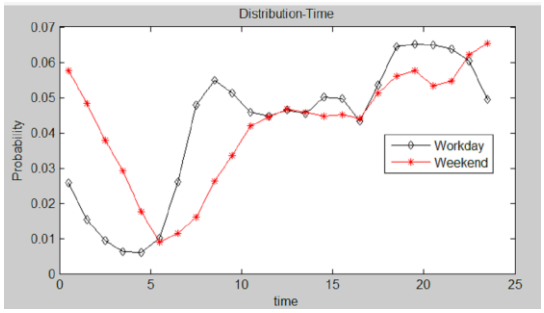

图 4-2 工作日与周末出行特征时序分布图

通过降维分析，我们只需聚焦周末和非周末这两类特征数据，极大地简化了模型的复杂度，大幅提升算法的计算速度。

4.2 客流量和客源效益的时序分析

我们以 1 小时为单位，将 1 天划分成 24 个时段。其中，t 时段内的上车人数定义为 t 时段客流量，记作 $N(t)$。图 4-3 可以非常清晰的看出，非周末的出行早高峰在 8 点前后，10 点出行量相对平稳，晚高峰在 18 点左右，20 点接近平稳；周末的早高峰在 11 点左右，晚高峰在 19 点左右，并且在 22 点左右还会出现第三个出行高峰。另外，非周末的早晚高峰比周末的早晚高峰出行流量要大，这个结果吻合居民日常出行规律。

客源效益的量化计算方法为：统计该时段内所有车辆的总收益和行驶总时间，将比值作为单位客源效益，记作：

$$B(t)=\frac{\sum_{i=1}^{N(t)} a_i(t)}{\sum_{i=1}^{N(t)} T_i(t)} \qquad (4\text{-}2)$$

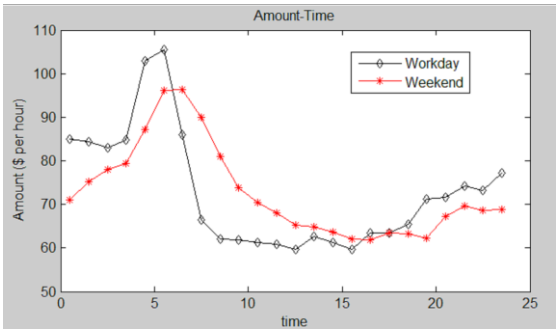

图 4-3 工作日与周末的客源效益特征时序分布图

从图 4-3 可以看出效益时序和客流量时序存在负相关，凌晨 5 点之前，客流量稀少，交通路况良好，加上计费补贴，客源效益处于一天的最高点，6 点之后，客流量逐步上升，客源效益直线下降，在出行高峰期客源效益到达低谷，原因在于出行流量上升导致交通拥挤，车辆行驶速度变慢，单笔效益变低，9 点之后到下午 3 点基本平稳，下午 4 点之后客源效益逐步上升，在晚高峰时也保持上升趋势，这可能和行驶距离有关，乘客选择出租车出行到路况良好、距离又较远的地点聚会或回家。周末晚高峰时段出行流量上升，但客源效益小幅下降，可能的情况是居民出行短途选择出租车，长途选择自备车，受短途交通状况影响，客源效益拉低。

根据上述时序分布图的分析结果，我们可以明确关于客源量和客源效益指标的定义和量化能反映居民的出行规律，同时效益指标与交通状况相关。

4.3 客流量和客源效益的空间分析

与时间分析时按单位小时做切片的方法类似，我们将纽约城市按 1km*1km 网格进行空间切片，客流量指标的量化方式不变，统计一天中该区域的上车人数记作 N_{ij}，考虑到区域车辆供需比对效益造成的影响，我们对客源效益指标进行修正，引入区域客源效益指标，记作：

$$B'_{ij} = \frac{A_{ij}}{T_{ij}} \times \frac{Nu_{ij}}{Nd_{ij}} \quad (4\text{-}3)$$

其中 A_{ij} 为 (i, j) 区域内出发的所有车辆总收益，T_{ij} 为该区域出发车辆载客所花的总时间，Nu_{ij} 为该区域上车的客源数，Nd_{ij} 为该区域下车的单数，即空车数。图 4-4 和 4-5 采用客源分布热力图直观显示客流量和客源效益在空间分布特征。

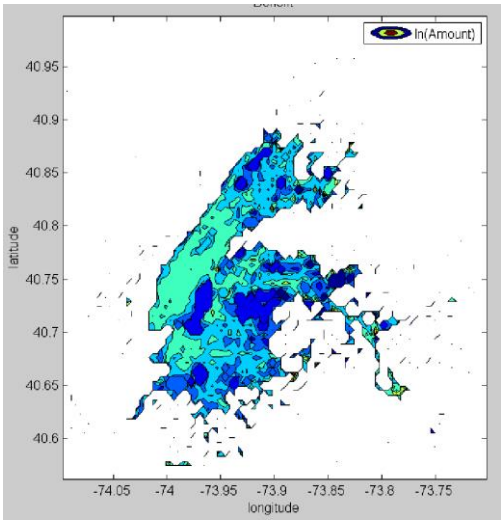

图 4-4 客源效益空间分布特征图

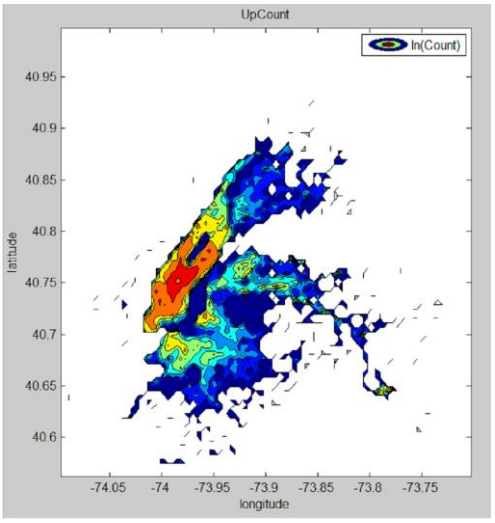

图 4-5 客源量空间分布特征图

从客流量来看，经纬度 [-73.995,40.765]到[-73.97,40.74]的空间区域是整个城市出行最为密集的地方，0.5%的空间面积占据了 23.45%的出行量，但客源效益热点分布相对分散，这反映出客流量热点地区虽然交通需求量大，但同时也存在出租车扎推、车速慢的情况。图 4-4 中可以看出在客流量稀疏的地区存在着多个效益热点，反映了出租车数量少客源稀疏区域交通状况好，行驶距离长，反而有较高的客源效益。因此，本文研究如何通过调度做到全局范围内车辆和客源的供需比相匹配，实现高效益寻客策略，是非常有价值的。

图 4-4 中也存在客流量分布和效益分布双高的区域，为[-73.79,40.64]到[-73.775,40.65]附近，对比纽约市行政地图，辨识出该区域为肯尼迪机场，可以清晰看出一条从该区域到市中心的出行轨迹，意味着机场来往的交通行为比较有规律，并且流量轨迹和效益轨迹高度相似。下面我们以机场区域为例，采用等高线图通过分析特定区域的流动性特征，进一步验证客流量指标和效益指标的合理有效。

4.4 区域空间的流动性分析

图 4-6 为周末出行特征，图 4-7 为非周末出行特征。可以看出，位于右下角的机场区域出行流量特征在凌晨 1 点到 5 点基本无人，6 点跳跃式增长，9 点之后缓慢回落，下午 1 点开始持续增长保持繁忙状态一直到下午 6 点，晚上 7 点流量开始回落，保持状态到夜里 12 点。工作日的流量与周末最大的区别在于工作日的 7：00 到 11 点出行流量是非常稀疏的，而夜里 11 点和 12 点以及凌晨则非常繁忙。上述分析结果吻合城市人群的出行特征，周末会在早上出行，工作日则偏向于晚上出行，流量特征和机场的航班情况也高度吻合，流动性分析结果客观合理。

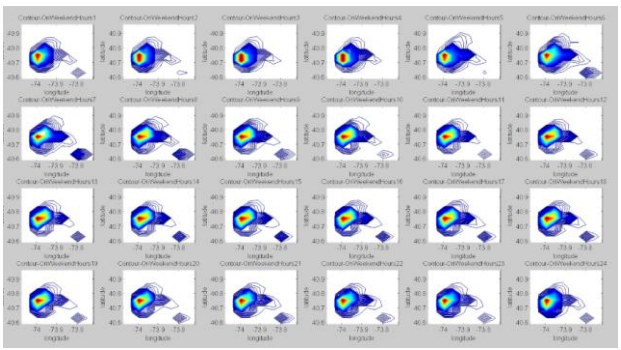

图 4-6 周末客流量时空分布特征图

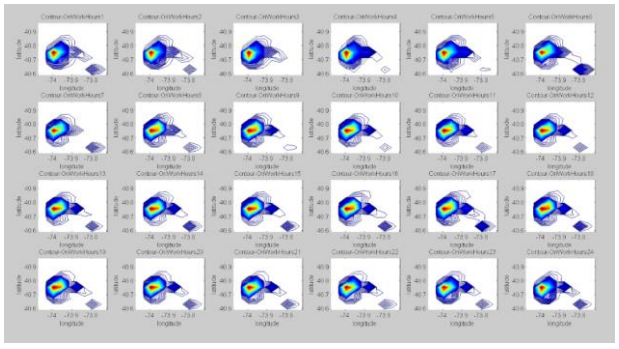

图 4-7 工作日客流量时空分布特征图

五、多目标规划的高效益寻客推荐策略

综上所述，在出行流量高的区域和时间段，出租车搭乘到乘客的可能性大，但区域交通拥挤，短途出行概率高。为获取最高效益的载客点，需要对到达区域的出行流量、道路状况、载客收益等因素进行综合评价。我们对时间进行切片划分，对路网区域进行网格化划分，计算任意网格任意时段的客流量分布、效益分布和载客成功概率，结合出租车前往目标区域的时间成本，以高效益、供需均衡为多重目标建立优化模型[14] [15]，求得最佳寻客推荐点，形成一套完整的出租车高效益寻客推荐策略，详细见流程图 5-1。

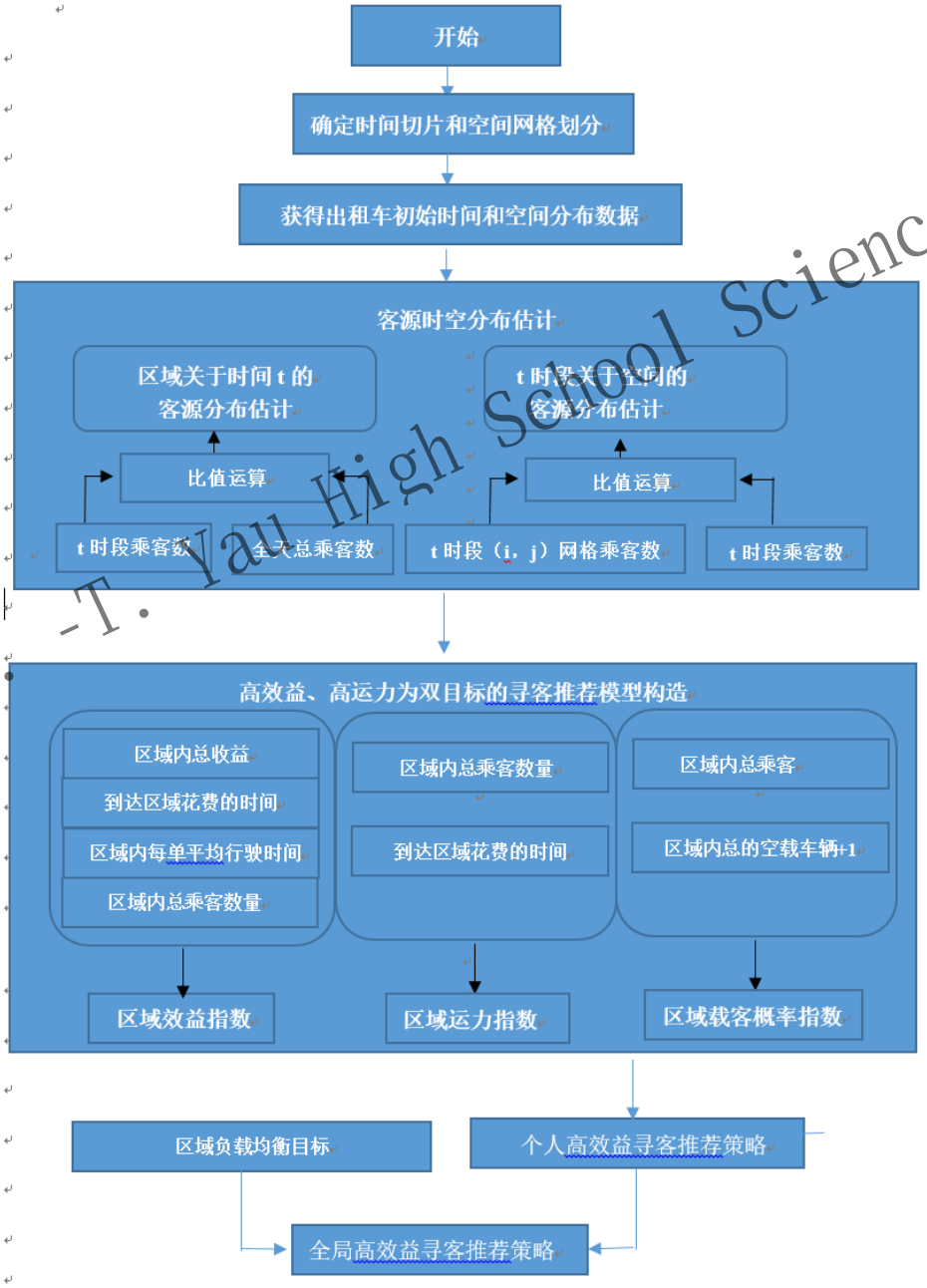

图 5-1 多目标规划的高效益寻客推荐策略算法流程图

5.1 网格划分

网格划分可以有两种方法，一是按单位距离进行划分，如上文的网格效益空间分布采用的是这种方法，单位距离设定为 1km*1km；另一种按 n*m 进行划分，两者都可行，但后者更加灵活，记 $x_{ij}, y_{ij}, i=1,2,\cdots m, j=1,2,\cdots n$ 为第（i，j）个网格区域中心点的经纬度，$x_{min}, y_{min}, x_{max}, y_{max}$ 表示纽约行政区域最小经度纬度和最大经纬度，可得每个空间网格的尺寸为 $\Delta x \times \Delta y$。

$$\text{其中} \begin{cases} \Delta x = \dfrac{(x_{\max} - x_{\min})}{n} \\ \Delta y = \dfrac{(y_{\max} - y_{\min})}{m} \end{cases} \qquad (5\text{-}1)$$

第（i，j）个区域中心点经纬度为：

$$\begin{cases} x_{ij} = x_{\min} + (j-1)\Delta x + \dfrac{\Delta x}{2} \\ y_{ij} = y_{\min} + (i-1)\Delta y + \dfrac{\Delta y}{2} \end{cases} \qquad (5\text{-}2)$$

5.2 网格距离的定义

网格距离决定了出发点到推荐点所花的时间成本，是提升高效益寻客策略精准度的关键指标。我们将网格距离定义为两个区域中心的距离，因为一般道路都是南北走向或者东西走向，所以该距离的计算可近似为延经度方向距离差加上延纬度方向距离差。于是可得第（i，j）个网格到第（s，k）个网格距离定义为：

$$d_{ijsk} = \left|x_{ij} - x_{sk}\right| U_x + \left|y_{ij} - y_{sk}\right| U_y \qquad (5\text{-}3)$$

其中，U_x 表示在该网格一度纬线对应的实际长度，U_y 表示在该网格一度经线对应的实际长度。任意纬度的一度经线长度一样，而一度纬线的长度与行政区域的经度值 θ 相关，且有 $U_x = U_y \cos(\theta)$。

5.3 区域关于时间和空间的乘客分布估计[16]

记 $N_{ij}(t)$ 为在 t 时段内，第（i，j）个网格发生的总乘车记录数。N 表示所有的乘车记录总数。得到整个行政区域关于时间的乘客分布估计：

$$P(t) = \frac{\sum_j \sum_i N_{ij}(t)}{N} \qquad (5\text{-}4)$$

以及在 t 时段关于空间上的乘客分布估计：

$$P_{ij}(t)=\frac{N_{ij}(t)}{\sum_{j}\sum_{i}N_{ij}(t)} \qquad (5\text{-}5)$$

5.4 个人寻客策略推荐模型

对于出租车司机个体，空载出租车的寻客目标为提高单位时间收益与接单数量，其中效益指标为主要目标，运力指标为次要目标，将空载出租车寻客策略转化为双目标问题，双目标函数[14]表征如下：

$$\begin{cases}\max \dfrac{A_{sk}(t_{ijsk})}{N_{sk}(t_{ijsk})\times(T_{ijsk}+\overline{T_{sk}(t_{ijsk})})}\\ \max \dfrac{N_{sk}(t_{ijsk})}{T_{ijsk}}\end{cases} \qquad (5\text{-}6)$$

其中，(i,j) 为出租车当前所在的网格区域，(s,k) 为出租车前往的网格区域，$A_{ij}(t)$ 表示 t 时段在第 (i,j) 个网格区域内发生乘车记录的总金额；$N_{sk}(t)$ 表示 t 时段第 (s,k) 个网格区域内发生的乘车总数量，T_{ijsk} 表示从第 (i,j) 个网格区域前往第 (s,k) 个网格区域的行驶时间，$\overline{T_{sk}(t)}$ 表示 t 时段在第 (s,k) 个网格区域内发生乘车记录的平均行驶时间，t_{ijsk} 表示 t 时段从第 (i,j) 个网格区域出发，到达第 (s,k) 个网格区域的时段。

如上述，可能会出现出租车被推荐运力严重过剩的区域，造成出租车扎堆现象，因此我们引入载客成功概率指数 R，记 $R_{sk}(t)$ 为 t 时段在 (s,k) 网格区域内成功载客的概率，由该区域的供需比 r 决定。定义为：

$$R_{sk}(t_{ijsk})=\begin{cases}r & (r<1)\\ 1 & (r\geq 1)\end{cases} \qquad (5\text{-}7)$$

其中，r 为 t 时段 (s,k) 区域的供需比，为乘客数量与空载车数量之比，记作：

$$r=\frac{Nu_{sk}(t_{ijsk})}{Nd_{sk}(t_{ijsk})+1} \qquad (5\text{-}8)$$

对公式（5-6）的双目标函数进行修正

$$\begin{cases}\max \dfrac{A_{sk}(t_{ijsk})\times R_{sk}(t_{ijsk})}{N_{sk}(t_{ijsk})\times(T_{ijsk}+\overline{T_{sk}(t_{ijsk})})}\\ \max \dfrac{N_{sk}(t_{ijsk})}{T_{ijsk}}\end{cases} \qquad (5\text{-}9)$$

5.4.1 约束条件

（1）关于行驶距离的约束。

$$d_{ijsk}=\begin{cases}\left|x_{ij}-x_{sk}\right|U_x+\left|y_{ij}-y_{sk}\right|U_y & (i,j)\neq(s,k)\\ \frac{1}{2}\sqrt{U_x^2+U_y^2} & (i,j)=(s,k)\end{cases} \tag{5-10}$$

（2）关于行驶时间的约束。

$$T_{ijsk}=\frac{d_{ijsk}}{v_{ij}} \tag{5-11}$$

$v_{ij}(t)$表示t时段在第(i,j)个网格区域内发生的乘车记录的平均速度。

（3）关于到达时段的约束。

由于出租车在t时段从第(i,j)个网格区域出发，花费T_{ijsk}小时到达第(s,k)个网格区域，到达时的时段为t_{ijsk}，故有

$$t_{ijsk}=\mathrm{mod}([t+T_{ijsk}],24) \tag{5-12}$$

其中，$[x]$表示对x向下取整，$\mathrm{mod}(M,N)$指M对N的余数

（4）关于起末点的约束。

$$1\le i,s\le m,\quad 1\le j,k\le n$$

5.4.2 模型的建立

综合效益目标和运力目标，在上述约束条件下，我们建立了个人寻客策略的双目标优化模型[14][15]。

$$\begin{cases}\max\dfrac{A_{sk}(t_{ijsk})\times R_{sk}(t_{ijsk})}{N_{sk}(t_{ijsk})\times(T_{ijsk}+\overline{T_{sk}(t_{ijsk})})}\\ \max\dfrac{N_{\mathrm{sk}}(t_{ijsk})}{T_{ijsk}}\end{cases} \tag{5-13}$$

$$s.t.\quad\begin{cases}d_{ijsk}=\begin{cases}\left|x_{ij}-x_{sk}\right|U_x+\left|y_{ij}-y_{sk}\right|U_y & (i,j)\neq(s,k)\\ \frac{1}{2}\sqrt{U_x^2+U_y^2} & (i,j)=(s,k)\end{cases}\\ T_{ijsk}=\dfrac{d_{ijsk}}{v_{ij}}\\ t_{ijsk}=\mathrm{mod}([t+T_{ijsk}],24)\\ 1\le i,s\le m,\quad 1\le j,k\le n\end{cases} \tag{5-14}$$

5.4.3 模型求解和分析

分析出租车 GPS 大数据，得到乘车记录的经纬度范围为[-74.03,40.57]到[-73.76,40.89]，即面积为 16.75 英里×22.09 英里的矩形，再将城市空间划分为20×20的网格，即 0.84 英里×1.10 英里的区域，按从南到北，从西到东依次进行编号。选取有代表性的 3 个区域，分别为：人流密集、交通拥堵的华尔街，在（9,2）网格；客流量中等、单客效益高的皇后区购物中心，在（11,12）网格；客流量稀少的布鲁克林林肯梯田公园，在（7,8）网格。模型求解[17]得到以下高效益寻客推荐结果，见下表。

表 5-1 华尔街区域空载出租车寻客推荐结果表

初始区域（9,2）华尔街								
时段	1	2	3	4	5	6	7	8
推荐区域	（9,2）	（9,2）	（10,2）	（10,2）	（9,3）	（9,4）	（9,2）	（9,1）
时段	9	10	11	12	13	14	15	16
推荐区域	（9,3）	（9,1）	（9,3）	（9,2）	（9,2）	（9,2）	（9,2）	（9,2）
时段	17	18	19	20	21	22	23	24
推荐区域	（9,2）	（9,2）	（9,2）	（9,2）	（9,2）	（9,2）	（9,2）	（9,2）

表 5-2 皇后区购物中心区域空载出租车寻客推荐结果表

初始区域（11,12）皇后区购物中心								
时段	1	2	3	4	5	6	7	8
推荐区域	（13,12）	（13,12）	（7,13）	（5,19）	（5,10）	（12,11）	（11,12）	（11,12）
时段	9	10	11	12	13	14	15	16
推荐区域	（9,11）	（11,12）	（11,12）	（11,12）	（13,13）	（11,12）	（11,12）	（11,12）
时段	17	18	19	20	21	22	23	24
推荐区域	（11,12）	（11,12）	（11,12）	（11,12）	（11,12）	（11,12）	（11,12）	（13,13）

表 5-3 布鲁克林林肯梯田公园区域空载出租车寻客推荐结果表

初始区域（7,8）布鲁克林林肯梯田公园								
时段	1	2	3	4	5	6	7	8
推荐区域	（5,18）	（5,18）	（7,13）	（5,19）	（5,10）	（9,4）	（7,8）	（5,17）
时段	9	10	11	12	13	14	15	16
推荐区域	（9,11）	（7,8）	（8,7）	（7,9）	（7,8）	（8,17）	（7,7）	（7,9）
时段	17	18	19	20	21	22	23	24
推荐区域	（7,7）	（7,7）	（3,8）	（7,8）	（6,17）	（5,18）	（7,8）	（6,18）

从上表可看出，客流量密集的华尔街区域，单位时间运力强，足以弥补交通拥挤造成的效益损失，因此寻客调度基本以自身网格为中心进行短途寻客；客流量中等的皇后区购物中心在客流稳定的时间段，在自身及周边网格寻客，在客流下降的时间段，如夜里 24 点到凌晨 5 点，则推荐前往高效益的两个机场；客流量稀少的布鲁克林林肯梯田公园，除了高峰期在周边区域寻客，其他时间基本推荐到肯尼迪机场。为了更直观地展示寻客调度结果，图 5-2 展示寻客热点目的地分布情况。与“空载就到闹市区寻客”的直觉不同，推荐指数最高的区域是两个机场（图中红色和黄色区域），其次是紧挨拉瓜迪亚机场的游乐场和布鲁克林大桥周边区域（图中深绿色区域），人流最为密集的曼哈顿区域推荐热度与周边大体持平（图中蓝绿色区域）。

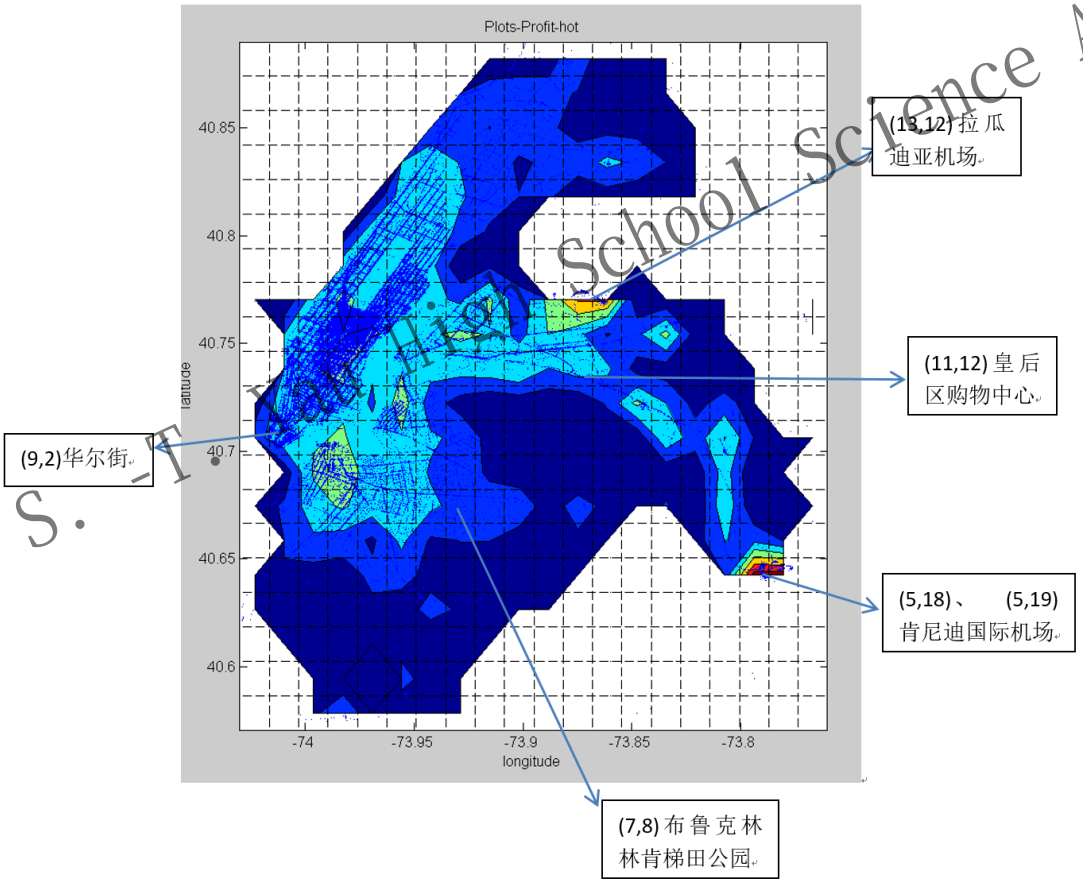

图 5-2 代表不同区域的空载出租车寻客推荐结果空间分布图

进一步按时间切片可以获得任一单位时间的推荐热点特征，见图 5-3，可以反映随着时间段变化推荐热点区域的流动特征，如早上 6 点在曼哈顿岛与其他行政区交接处综合效益是最高的，而早上 11 点钟往肯尼迪机场区域则能获得最高的

预期效益，有了上述图表信息支撑，出租车司机盲目寻客问题得到了很好的解决。

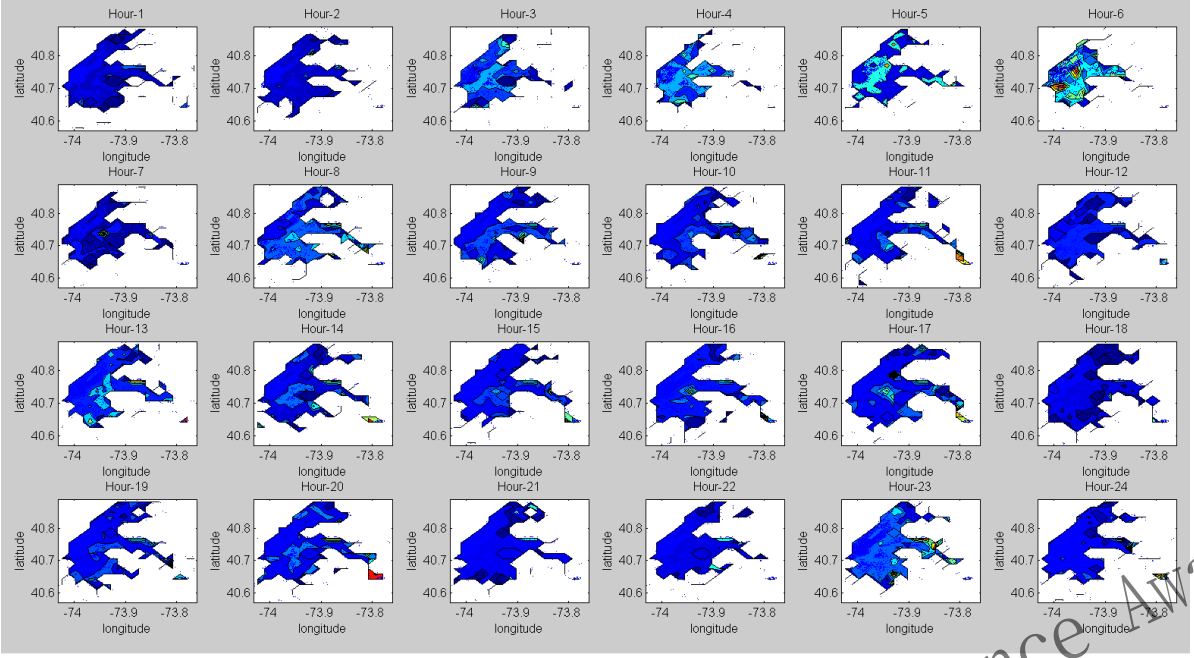

图 5-3 推荐热点区域随时间变化的流动特征图

5.5 集体策略寻客推荐模型

只考虑个人策略，即使加入载客成功概率，还是会出现高效益区域出租车扎堆，运力过剩的现象。为了使出租车资源均衡分布，达到运力与客流量基本一致，我们设计了集体高效益寻客策略，从全局角度实现空车调度，在效益优先的前提下最大程度利用出租车资源，并避免热点区域供远大于需的局面。

5.5.1 模型的建立

出租车资源的均衡分布是指任意地区的客户能够具有相同的打车满意度，也就是任意时刻，任意区域的乘客具有相同的概率打到出租车，即出租车运力分布与乘客空间分布相一致。全局调度应尽可能实现出租车空间分布接近于乘客空间分布 $P_{ij}(t)$。

$$P_{ij}(t)=\frac{N_{ij}(t)}{\sum_{j}\sum_{i}N_{ij}(t)} \qquad (5\text{-}15)$$

这样，对于每个司机来说，接到一单生意的概率是一样的，但因为每个区域每单生意平均收益并不一致，为了保持公平性，适当增加高效益区域运力，以每单平均收益 $C_{ij}(t)$ 来修正原出租车配置分布并归一化[18]得：

$$P_{ij}(t)=\frac{P_{ij}(t)C_{ij}(t)}{\sum_{j}\sum_{i}P_{ij}(t)C_{ij}(t)} \qquad (5\text{-}16)$$

称 $P_{ij}(t)$ 为出租车配置的最优目标分布，其中 $C_{ij}(t)$ 为 t 时刻第（i, j）网格区域的乘客单笔生意平均收益。

（1）决策变量

记 $x_{ijsk}(t)$ 为从 (i, j) 区域调度到 (s,k) 区域的空载出租车比例。其中

$$i,s=1,2,\cdots m \quad j,k=1,2,\cdots n \quad 0\le x_{ijsk}(t)\le 1$$

（2）目标函数

时空分析已计算得到每辆出租车从第 (i, j) 个区域到第 (s,k) 个区域的距离为 d_{ijsk}，若调度总数为 $N\bullet x_{ijsk}(t)$，其中 N 为出租车公司拥有的出租车总数，单位距离的调度成本记为 K。则公司 t 时刻整个出租车调度成本为

$$Z_1(t)=\sum_{k}\sum_{s}\sum_{j}\sum_{i}KNd_{ijst}x_{ijst}(t) \qquad (5\text{-}17)$$

显然全局完成调度目标的最优策略应该是调度成本 Z_1 最低。又由于 N,K 为常数，所以 Z_1 最低的策略等价于 Z 最低，即

$$\min Z(t)=\sum_{k}\sum_{s}\sum_{j}\sum_{i}d_{ijst}x_{ijst}(t) \qquad (5\text{-}18)$$

（3）约束条件

对于任意区域 (i, j)，调度出去的出租车总数不能超过原有在该区域的出租车总数，即：

$$\sum_{k}\sum_{s}x_{ijsk}(t)\le T_{ij}(t) \qquad (5\text{-}19)$$

其中 $T_{ij}(t)$ 表示 t 时刻出租车的初始分布，该数据可以实时查询车载 GPS 定位数据取得（本课题以历史数据中该区域 t 时段下车车辆数来模拟分布）。调度结束后，出租车分布应该与最优的目标分布相一致，即任意区域 (i, j) 有

$$T_{ij}(t)-\sum_{k}\sum_{s}x_{ijsk}(t)+\sum_{k}\sum_{s}x_{skij}(t)=P_{ij}(t) \qquad (5\text{-}20)$$

综上所述，加入符号限制和概率性质可以得到集体策略规划模型：

$$\min Z(t)=\sum_{k}\sum_{s}\sum_{j}\sum_{i}d_{ijst}x_{ijst}(t)$$

$$s.t.\begin{cases}\sum_{k}\sum_{s}x_{ijsk}(t)\le T_{ij}(t) \quad i=1,2\cdots m;\ j=1,2\cdots n\\ T_{ij}(t)-\sum_{k}\sum_{s}x_{ijsk}(t)+\sum_{k}\sum_{s}x_{skij}(t)=P_{ij}(t) \quad i=1,2\cdots m;\ j=1,2\cdots n\\ x_{ijsk}(t)\le 1 \quad i,s=1,2\cdots m;\ j,k=1,2\cdots n\\ x_{ijsk}(t)\ge 0 \quad i,s=1,2\cdots m;\ j,k=1,2\cdots n\end{cases} \tag{5-21}$$

5.5.2 模型的求解和验证

同样将城市空间分割为 20*20 个网格，分别选取 4 月 1 号和 4 月 3 号早上 8 点早高峰、下午 3 点和晚上 8 点的时空切片，出租车初始分布以延时 3 个半小时后的下车车辆数的分布来模拟。按出租车保有量为 1.5 万辆计算，以 4 月 1 号（工作日）早上 8 点调度结果为例，见图 5-4，左图为各网格车辆调出情况，右图为各网格车辆调入情况。

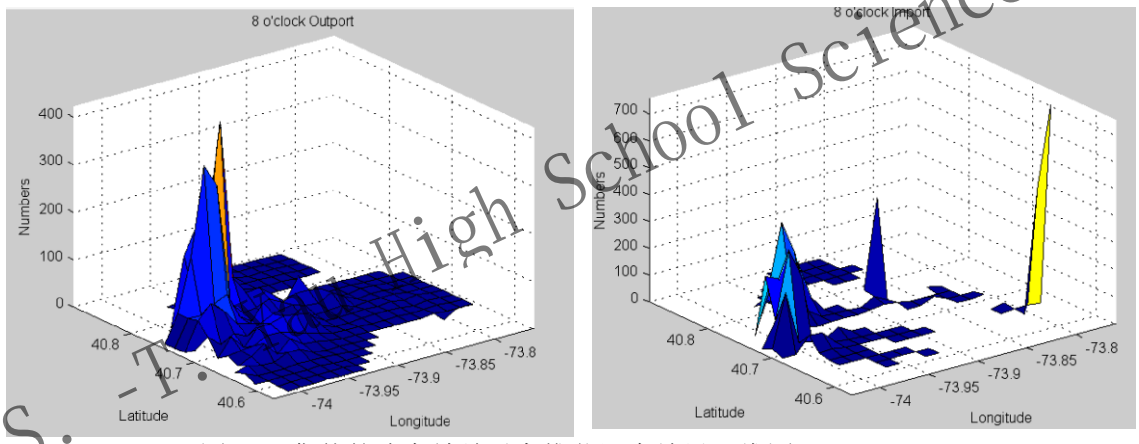

图 5-4 集体策略高效益寻客推荐调度结果三维图

其他 5 个时段的调度情况不再累述，表 5-4 记录了以上 6 个时段调度结果的统计数据，并对调度结果有效性进行了验证，具体验证方法见 5.5.3

表 5-4 集体策略寻客推荐调度结果统计及效益提升验证表

时段	工作日			周末		
时段	早上 8 点	下午 3 点	晚上 8 点	早上 8 点	下午 3 点	晚上 8 点
车辆调出区域数量	92	106	116	97	84	86
车辆调入区域数量	58	45	47	49	39	41
参与调度车辆数量比例	30.55%	17.06%	22.61%	24.21%	18.13%	15.60%
效益提升率	19.49%	5.47%	16.07%	16.41%	5.28%	10.71%

5.5.3 策略的有效性验证

1、估计 t 时刻出租车空载率

记 $K(t)$ 为 t 时段空载率，由客户需求时数等于出租车提供时数得：

$$\sum_j \sum_i N_{ij}(t) Tave(t) = M \bullet T_w (1-(K(t)\ ) \tag{5-22}$$

$$K(t)=1-\frac{\sum\sum N_{ij}(t)\ Tave\ (t)}{MT_w}$$

其中 $Tave(t)$ 表示平均每单行驶时长，T_w 和 M 为常量，分别表示单位时长和城市出租车保有量。

2、估计 (i, j) 区域 t 时段单位时间收益

$$\overline{C_{ij}}(t) = \frac{C_{ij}(t)}{Tave(t)} \times (1-K(t)) \tag{5-23}$$

$C_{ij}(t)$为 t 时刻第 (i, j) 网格区域的乘客单笔生意平均收益。

3、估计 t 时刻全局收益 $A(t)$

$$A(t) = \sum_j \sum_i \overline{C_{ij}}(t) \bullet \min\left\{T_{ij}(t) \bullet M \bullet (1-K(t)), P_{ij}(t) \bullet N(t) \bullet Tave^{ij}(t)\right\} \tag{5-24}$$

其中 $Tave^{ij}(t)$ 表示 (i, j) 区域 t 时刻平均每单行驶时长，$T_{ij}(t)$ 表示 t 时刻出租车的分布，$P_{ij}(t)$ 表示 t 时刻客户的分布。

4、结果验证

调度前后的效益对比见图 5-5，其中红色上三角曲线表示为调度前的总效益 $\sum A_2(t)$，黑色十字曲线表示为调度后的总效益 $\sum A_1(t)$。

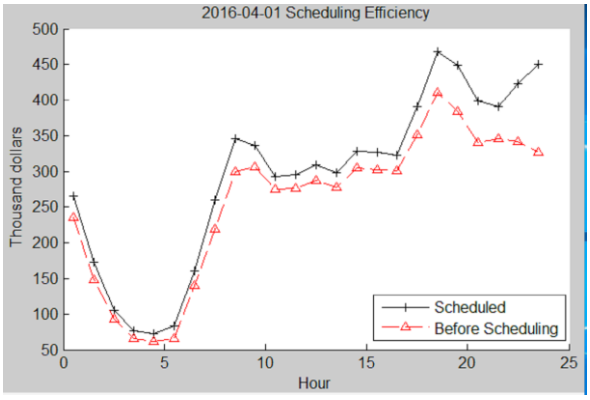

图 5-5 集体策略日效益提升时序分布图

我们将调度有效性定义为：

$$U_{\text{day}}=\frac{\sum A_2(t)-\sum A_1(t)}{\sum A_1(\text{t})}\times 100\% \qquad (5\text{-}25)$$

则当天的效益提升率为 13.98%。同理，对 2016 年 4 月份的效益提升进行分析，月收益提升率定义为：

$$\mu=\frac{\sum_{30\text{天}} U_{day}}{30\text{天}} \qquad (5\text{-}26)$$

如图 5-6，按公式 5-26 计算 4 月的效益提升率为 11.4%，策略有效性非常明显。

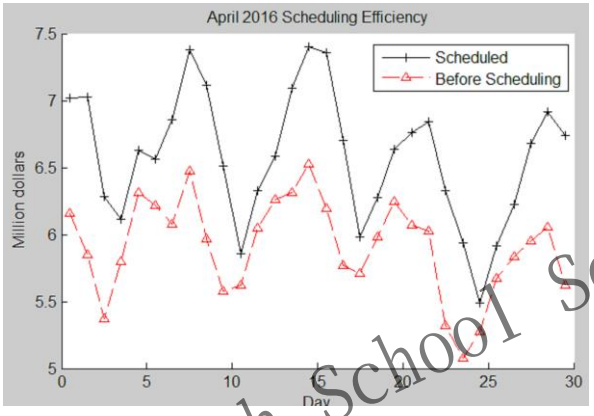

图 5-6 集体策略月效益提升时序分布图

六、多层次网格划分算法实现

在高效益寻客策略中，网格划分方法会影响到策略的精准度，两个区域中心距离 d_{ijsk} 的取值决定了寻客行驶时间，直接影响效益指标、运力指标和调度成本。无论是采用单位距离划分法还是数量等分法划分网格空间，对于像北京这类道路规则有序、地理环境单一的平原地貌城市，d_{ijsk} 参数值和区域两点之间的实际行驶距离基本接近，但对于诸如纽约、香港这类地理环境多样化的岛屿或者丘陵地貌城市，区域内经纬度相近的两个点可能隔着一条河，隔着一座山等天然屏障，必须绕道行驶，导致实际行驶距离和 d_{ijsk} 参数值相差非常大，我们引入网格之间各点行驶距离，采用均方差表示稳定性指标，记作：σ_{ijsk}

$$\sigma_{\text{ijsk}}=\frac{\sqrt{\sum_{i=1}^{n}\frac{1}{n-1}(d_i-\bar{d})^2}}{\bar{d}} \qquad (6\text{-}1)$$

设 $(i,j)\rightarrow(s,k)$ 中有 n 条记录，行驶距离为 $d_1,d_2,...,d_n$，$\bar{d}=\sum_{i=1}^{n}d_i$，$\sigma_{\text{ijsk}}$ 值越小则

区域行驶距离稳定性越强，若$\sigma_{ijsk} \le 0.1$[16]，则认为网格之间各点行驶距离稳定，若σ_{ijsk}过大，表明该区域可能存在复杂的地理环境或特定的交通管制。为达到区域内各点的行驶特征基本一致，我们设计了多层网格划分优化算法，降低σ_{ijsk}指标，使得区域内行驶特征趋于一致。

算法的基本思路为：对优化之前的网格做σ_{ijsk}计算，标识不稳定的网格位置，结合地理环境，以影响稳定性因素为划分准则，将整体空间进行第一层次划分，使得独立区域之间任意两点的行驶距离相对稳定；再对每个独立区域采用数量等分法二度划分网格空间，标识为(b,i,j)，其中(i,j)为网格编号，b 为第一层次区域编号。统计区域之间的行驶记录数d_{ijsk}，并计算σ_{ijsk}得

$$d^{r}_{ijsk}=\begin{cases} d_{ijsk} & n_{ijsk}=0 \\ \bar{d} & n_{ijsk}\neq 0 \text{且} \sigma_{ijsk}\le 0 \\ \text{二度化分网格后计算} & n_{ijsk}\neq 0 \text{且} \sigma_{ijsk}>0 \end{cases} \qquad (6\text{-}2)$$

重复划分，最终使得(b,i,j)网格区域内各点到其他区域内各点的距离基本保持稳定，$d^{r}_{ijsk}=\bar{d}$。该算法在提升策略精准度的同时，极大简化了两点行驶距离求解问题，下面具体讨论如何进行多层网格优化划分。

6.1 基于稳定性特征和客流量时空特征划分空间区域

从行政地图（图 6-1）可以了解，纽约市有五个行政区：曼哈顿（Manhattan）、皇后区（Queens）、布鲁克林区（Brooklyn）、布朗克斯区（The Bronx）、斯塔滕岛（Staten Island），其中布鲁克林区和皇后区同在长岛，与其他 3 个区之间彼此分别被江河隔开。从客流量空间分布散点图（图 6-2），可知斯塔滕岛没有出行流量，因此图 6-2 的左下角没有出行信息，除曼哈顿区域外，肯尼迪机场、拉瓜迪亚机场为出行相对集中的热点区域。

图 6-1 纽约市行政地图

图 6-2 客流量空间分布散点图

由图 6-3 可以看出行驶距离不稳定的区域集中在曼哈顿岛与其他行政区的交界处，在图中显示为红色和蓝绿色，σ_{ijsk} 大于 0.15，最高达 0.2511；曼哈顿岛以及其他行政区的腹地，σ_{ijsk} 一般稳定在 0.06-0.07 之间。这些边界区域交通环境比较复杂，有桥、隧道直达对岸，也有些则必须绕道而行，而对于行政区腹地，地形基本单一。综合以上特征，我们将纽约市分为 5 大空间区域，分别为肯尼迪机场、拉瓜迪亚机场、曼哈顿区、布朗克斯区，以及同在长岛的布鲁林区和皇后区，用折线框在地图上勾勒，并记录各顶点的经纬度坐标，见图 6-4。

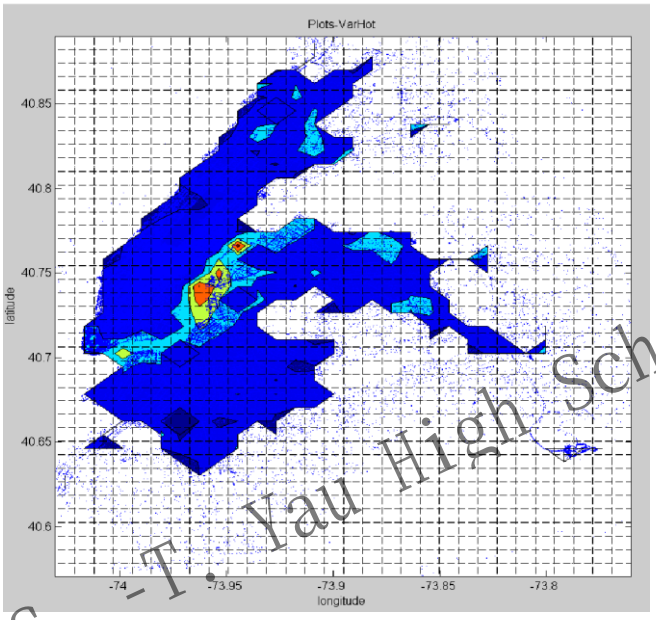

图 6-3 行驶距离稳定性空间分布特征图

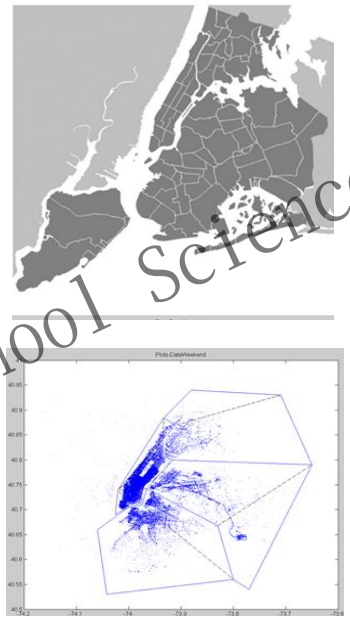

图 6-4 空间区域折线划分图

6.2 通行记录与区域的匹配算法

分别将以折线框勾勒出的独立区域拆分成一至多个凸多边形，见图 6-4，各个凸多边形所包含的散点，构成该独立区域的出行记录集。几何构造算法如下：

1、取凸多边形各顶点的经纬度平均值，作为伪中心(x_0, y_0)，其中，三角形是重心，其他凸多边形围成的平面区域，必定覆盖该点；

2、用斜率截距法，计算凸多边形一条边的直线方程 $y = kx + b$。将伪中心 x_0 代入方程，计算 $y' = kx_0 + b$，若 $y' \geq y_0$，表明直线在上方，取不等式 $y \leq kx + b$，否则取 $y \geq kx + b$。对于平行于 y 轴的边，其直线方程为 $x = b$，则只须判断 x_0 与 b 的大小，就知道(x_0, y_0)在直线右边还是左边，取对应的不等式 $x \geq b$ 或者 $x \leq b$；

3、重复步骤 2，将得到的各个不等式进行“与”操作，从总出行记录中进行筛选，获得该凸多边形所包含的点。

通行记录匹配到对应的独立区域之后，接下来的工作和之前的网格划分类似：取每一个独立区域的最小经纬度和最大经纬度围成矩形框，按$n \times m$进行划分，方法同 5.1，唯一的区别是标识该网格属于哪个独立区域。对经过网格优化之后的通行记录进行行驶距离稳定性计算，最差的稳定性指标值为 0.0833，比之前的 0.2511 提升了近 3 倍，两点之间行驶距离基本趋于稳定，证明了优化算法的有效性。

七、结论

本文在客源时空分布不均匀，出租车运行效率有待提升的背景下，结合国内外出租车寻客推荐策略研究成果及不足，通过对纽约出租车 GPS 大数据进行数据挖掘和时空分析，设计和实现了多目标优化的高效益寻客推荐策略，并进行模拟和验证。该策略可方便地复制应用到其他有出租车 GPS 数据的城市，也可用于网约车等浮动车辆的预测调度。主要的创新工作包括以下几点：

1、大数据预处理和降维分析方法。采用多元线性回归方法将缺失和错误的 GPS 数据进行回填，并通过相关性分析将大数据进行降维处理，降低了模型的复杂度和数据的计算量，提高了数据分析的准确性。

2、对影响客源效益的指标因素进行分析，建立了效益指标、客流量指标和载客概率指标，结合 GPS 大数据的时空分析识别高效益客源特征，合理解释了指标定义，为高效益寻客推荐模型提供了现实支撑。

3、建立多目标优化的高效益寻客策略，并进行验证。考虑出租车到达高效益载客热点区域的时间成本及到达后的出租车供需关系等情况，以高效益载客、供需均衡分布为多重目标，研究并实现了高效益实时寻客策略，填补之前研究的空白和不足，为出租车司机、车辆运营公司以及交通管理部门做调度决策提供一种有效、科学的方法，实现经济效益和社会效益的综合提升。

4、自行设计多层网格划分优化算法，巧妙地规避了由于城市地形屏障或交通管制等情况带来的网格划分不合理，解决了网格之间各点行驶距离不稳定的问题，同时根据历史数据方便地得到d_{ijsk}参数值，极大简化了求解两点之间行驶距离问题，提升了高效益寻客策略的精准度。

参考文献：

[1] 张红 等.出租车 GPS 轨迹大数据在智能交通中的应用[J].《兰州理工大学学报》 , 2016 , 42 (1) :109-114

[2] L Li ,S Wang, FY Wang. An Analysis of Taxi Driver's Route Choice Behavior Using the Trace Records[J].IEEE Transactions on Computational Social Systems, 2018 , PP (99) :1-7

[3]Liu, L., Andris, C., Bidderman, A., Ratti, C.: Revealing taxi drivers mobility intelligence through his trace. MovementAware Applications for Sustainable Mobility: Technologies and Approaches, (2010), 105-120.

[4] 程静，刘家骏，高勇. 基于时间序列聚类方法分析北京出租车出行量的时空特征[J].《地球信息科学学报》, 2016 , 9 : 1227-1239

[5] Wangsheng Zhang, Shijian Li, Gang Pan. Mining the Semantics of Origin-Destination Flows using Taxi Traces: UbiComp'12, (2012), 5-8.

[6]Yuan, J., Zheng, Y., Zhang, C., Xie, W., Xie, X., Huang, Y.: T-Drive: Enhancing Driving Directions with Taxi Drivers' Intelligence, IEEE Transactions on Knowledge & Data Engineer. 2013 , 25 (1) :220-232.

[7] 程静等.基于出租车 GPS 数据的路段平均速度估计模型[J].西南交通大学学报》 ，2011 ，42 : 638-644

[8]纽约市出租车 GPS 数据库: https://data.cityofnewyork.us/Transportation/2016-Green-Taxi-Trip-Data/hvrh-b6nb.

[9] 异常数据的剔除——拉依达准则和格拉布斯准则 http://www.ilovematlab.cn/thread-88014-1-1.html(出处: MATLAB 中文论坛)

[10] 何晓群.应用回归分析[M].北京:中国人民大学出版社,2015

[11] Matlab 数据的可视化. https://blog.csdn.net/LSGO_MYP/article/details/54972713

[12]徐永明,道路运输行业中如何提高出租车的经济效益[N].《现代商业》,2013 (14) :61-61

[13] 时序图的绘制.http://www.ilovematlab.cn/thread-305650-1-1.html (出处: MATLAB 中文论坛)

[14]范玉妹. 数学规划及其应用[M].北京：机械工业出版社，2018.

[15]姜启源,谢金星,叶俊.数学模型[M].北京：高等教育清出版社，2016.

[16]邓集贤,杨维权,司徒荣等.概率论及数理统计[M].北京：高等教育出版社，2009

[17]刘浩,韩晶. MATLAB R2016a 完全自学一本通 [M].北京:电子工业出版社,2016

[18]《数据归一化和两种常用的归一化方法》[EB/OL].[2016-3-13]. http://blog.csdn.net/thesby/article/details/50878700

致　谢

在研究报告即将完成之时，我的心情非常激动。选择交通大数据的课题，很好玩也很有实际意义，但研究工作量真的非常大，好多知识对于我来说都极具挑战性。凭着初生牛犊不怕虎的勇气，在指导老师的帮助下，我选定数据源，对数据进行整理和清洗，在反复讨论中理解出租车运营数据的含义和作用，定义效益指标和流量指标以及用时空分析手段进行比对和解释，构造多目标优化模型，实现了单人策略和集体策略……整个研究过程的难度和工作量都远超我的预期，特别是数据分析非常繁琐，计算量非常庞大。其中，也遇到很多困难，比如在模型建立过程中，一旦公式调整，或者测试方式改变，又要从头开始。就这样，在一次次的设想、计算、分析、整理的过程中，研究方向和设计思路越来越清晰，技术手段越来越成熟，解决问题的能力也得到很大的提升，最后终于完成了论文的构思和撰写。我想这一段研究经历不仅仅是解决了一个问题，帮助了一些人，它还是我成长道路的一笔巨大财富：它让我深刻体会到数学的强大和美丽，坚定了深入探索数学的决心，更让我感受到科学研究的严谨、细致、耐心和坚持。非常感谢把我带到数学和计算机处理领域的老师们，特别要感谢我的指导老师张笑钦教授和 Daniel MacLeon 老师，在我“山穷水尽疑无路”时，不厌其烦地引导和激发，筛选大量的资料让我参考和学习，引领我进入“柳暗花明又一村”。最后要感谢我的妈妈和那天给我灵感的司机，也感谢我的爸爸帮我校对了全文，包括标点符号，谢谢您们！

队员介绍

方书田，目前就读于上海包玉刚实验学校，高二学生

奖项和荣誉

- 2017 年，HiMCM（美国高中生数学建模大赛）获 Outstanding 奖项
- 2018 年，hack.init() 国际创客马拉松大赛一等奖和设计奖

发表的论文和作品

2018 年，在国际会议“CISN2018 ”发表论文 《The modeling and implementation of Non- rigid motion based on the carcass traits》第一作者

参与课外活动

- 2017 年，组建学校数学建模社团，任社长
- 2018 年，组建三果远程英文支教平台，创始人之一
- 2018 年，MIT LauchX 俱乐部，负责人

本参赛团队声明所提交的论文是在指导老师指导下进行的研究工作和取得的研究成果。尽本团队所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果。若有不实之处，本人愿意承担一切相关责任。

参赛队员：　方书田　　指导老师：　张笑钦教授

Daniel Mac Leon

2018 年 **9** 月 **14** 日

附录：

因篇幅关系，此处仅放关键源代码信息。数据及其他相关信息详见百度云链接：

https://pan.baidu.com/s/15ImOaT5LwWmwdXjLJWReAA

密码：qqn8

第一部分 出租车 GPS 数据字典

第二部分 数据

1、原始数据

2、预处理完毕的数据

3、降维后的数据

4、测试数据

第三部分 源代码

一、GPS 大数据时空特征识别

1、时序分布特征

（1）客流量时序分析

（2）客源效益时序分析

2、空间分布特征

（1）客流量空间分析

（2）客源效益空间分析

3、热点区域（机场）客流量和客源效益时间流动性特征

二、高效益寻客推荐模型求解代码

1、个人策略高效益寻客推荐模型

2、集体策略高效益寻客推荐模型

三、模型优化代码

1、多层次网格划分算法

2、出行记录与区域的匹配算法

第一部分 出租车 GPS 数据字典

Field Name	字段名	描述
lpep_pickup_datetime	上车日期时间	计程器计时开始时间
lpep_dropoff_datetime	下车日期时间	计程器计时结束时间
Trip_distance	行驶里程	计价器报告的英里数.
Pickup_longitude	上车经度	开始计时的经度.
Pickup_latitude	上车纬度	开始计时的纬度.
Dropoff_longitude	下车经度	计程器结束时的经度.
Dropoff_ latitude	下车纬度	计程器结束时的纬度
Fare_amount	基准里程费	按计价器计算的时间和距离的车费.
Total_amount	总金额	乘客的总费用。

第二部分 数据（见百度云盘）

https://pan.baidu.com/s/15ImOaT5LwWmwdXjLJWReAA

密码：qqn8

第三部分 源代码

一、GPS 大数据时空特征识别

```
clear;
clc;
close all;
csvfile='taxi20160420_23.csv';
data=xlsread(csvfile); %读数据
NUMLONG = 50;  %经度划分的列数
NUMLAT = 50;   %纬度划分的列数
Index=find(data(:,3)<-74.16|data(:,3)>-73.7|data(:,4)<40.57|data(:,4)>40.92 ...
    |data(:,5)<0|data(:,5)>200);
data(Index,:)=[];
data(:,1:2)=data(:,1:2)-42461+1;  %时间处理
%时间处理（平移 6 个小时，目的使得礼拜五 18 点后计入周末）
data(:,1:2)=data(:,1:2)-floor(data(1,1))+1+0.25;

%****************时间分布分析*************
data_Mon=data(find((data(:,1)>=4&data(:,1)<5)|(data(:,1)>=11&data(:,1)<12) ...
    |(data(:,1)>=18&data(:,1)<19)|(data(:,1)>=25&data(:,1)<26)),:);
data_Tue=data(find((data(:,1)>=5&data(:,1)<6)|(data(:,1)>=12&data(:,1)<13) ...
```

```
    | (data(:,1)>=19&data(:,1)<20)|(data(:,1)>=26&data(:,1)<27)),:);
data_Wed=data(find((data(:,1)>=6&data(:,1)<7)|(data(:,1)>=13&data(:,1)<14) ...
    | (data(:,1)>=20&data(:,1)<21)|(data(:,1)>=27&data(:,1)<28)),:);
data_Thu=data(find((data(:,1)>=7&data(:,1)<8)|(data(:,1)>=14&data(:,1)<15) ...
    | (data(:,1)>=21&data(:,1)<22)|(data(:,1)>=28&data(:,1)<29)),:);
data_Fri=data(find((data(:,1)>=8&data(:,1)<9)|(data(:,1)>=15&data(:,1)<16) ...
    | (data(:,1)>=22&data(:,1)<23)|(data(:,1)>=29&data(:,1)<30) ...
    | (data(:,1)>=1&data(:,1)<2)),:);
data_Sat=data(find((data(:,1)>=9&data(:,1)<10)|(data(:,1)>=16&data(:,1)<17) ...
    | (data(:,1)>=23&data(:,1)<24)|(data(:,1)>=30&data(:,1)<31) ...
    | (data(:,1)>=2&data(:,1)<3)),:);
data_Sun=data(find((data(:,1)>=10&data(:,1)<11)|(data(:,1)>=17&data(:,1)<18) ...
    | (data(:,1)>=24&data(:,1)<25)|(data(:,1)>=3&data(:,1)<4)),:);

data_Mon(:,1)=data_Mon(:,1)-floor(data_Mon(:,1));
data_Tue(:,1)=data_Tue(:,1)-floor(data_Tue(:,1));
data_Wed(:,1)=data_Wed(:,1)-floor(data_Wed(:,1));
data_Thu(:,1)=data_Thu(:,1)-floor(data_Thu(:,1));
data_Fri(:,1)=data_Fri(:,1)-floor(data_Fri(:,1));
data_Sat(:,1)=data_Sat(:,1)-floor(data_Sat(:,1));
data_Sun(:,1)=data_Sun(:,1)-floor(data_Sun(:,1));

P1=hist(24*data_Mon(:,1),24)/length(data_Mon(:,1));
P2=hist(24*data_Tue(:,1),24)/length(data_Tue(:,1));
P3=hist(24*data_Wed(:,1),24)/length(data_Wed(:,1));
P4=hist(24*data_Thu(:,1),24)/length(data_Thu(:,1));
P5=hist(24*data_Fri(:,1),24)/length(data_Fri(:,1));
P6=hist(24*data_Sat(:,1),24)/length(data_Sat(:,1));
P7=hist(24*data_Sun(:,1),24)/length(data_Sun(:,1));

Time_P=0.5:23.5;figure;
plot(Time_P,P1,'ro-'),hold on
plot(Time_P,P2,'g*-')
plot(Time_P,P3,'bx-')
plot(Time_P,P4,'ys-')
plot(Time_P,P5,'md-')
plot(Time_P,P6,'cp-')
plot(Time_P,P7,'kh-'),
```

```
title('Distribution-Time');
xlabel('time');ylabel('Probability');
hold off

legend('Monday','Tuesday','Wednday','Thurday','Friday','Satday','Sunday');

%************区间分割 NUMLONG*NUMLAT***********

longitude_max=max(data(:,3))+0.00001;latitude_max=max(data(:,4))+0.00001;
longitude_min=min(data(:,3))-0.00001;latitude_min=min(data(:,4))-0.00001;
x_axis=linspace(longitude_min,longitude_max,NUMLONG+1);
y_axis=linspace(latitude_min,latitude_max,NUMLAT+1);
x_step=(longitude_max-longitude_min)/NUMLONG;
y_step=(latitude_max-latitude_min)/NUMLAT;
X_area_center=longitude_min+x_step/2:x_step:longitude_max;
Y_area_center=latitude_min+y_step/2:y_step:latitude_max;

%*************空间分布******************
[xa,ya]=meshgrid(X_area_center,Y_area_center);
data_Work=[data_Mon;data_Tue;data_Wed;data_Thu;data_Fri];
data_Weekend=[data_Sat;data_Sun];

%%%增加画线
P1=hist(24*data_Work(:,1),24)/length(data_Work(:,1));
P2=hist(24*data_Weekend(:,1),24)/length(data_Weekend(:,1));
Time_P=0.5:23.5;figure;
plot(Time_P,P1,'kd-', 'LineWidth',1);
hold on;
plot(Time_P,P2,'r*-', 'LineWidth',1);
title('Distribution-Time');
xlabel('time');ylabel('Probability');
hold off
legend('Workday','Weekend');
%%%增加完毕

%*****************工作日整体空间分布*************
title1='DataWork';
[za_Work cost_Work]=distr(data_Work,title1);
```

```
%*****************双休日整体空间分布*************
title1='DataWeekend';
[za_Weekend cost_Weekend]=distr(data_Weekend,title1);

%*************数据按小时细化（集体策略数据准备）*************
data_Work_hours=cell(24,1);
data_Weekend_hours=cell(24,1);
za_Work_hours=zeros(NUMLONG,NUMLAT,24);
za_Weekend_hours=zeros(NUMLONG,NUMLAT,24);
cost_Work_hours=zeros(NUMLONG,NUMLAT,24);
cost_Weekend_hours=zeros(NUMLONG,NUMLAT,24);
for i=1:24
     data_Work_hours{i}= ...  %按小时数据分类
          data_Work(find(data_Work(:,1)*24<=i&data_Work(:,1)*24>i-1),:);
    title1=['data_Work_hours' num2str(i)];
    [za_Work_hours(:,:,i) cost_Work_hours(:,:,i)]= ...
          distr(data_Work_hours{i},title1);
    data_Weekend_hours{i}= ...
        data_Weekend(find(data_Weekend(:,1)*24<=i&data_Weekend(:,1)*24>i-1),:);
    title1=['data_Weekend_hours' num2str(i)];
    [za_Weekend_hours(:,:,i) cost_Weekend_hours(:,:,i)]= ...
          distr(data_Weekend_hours{i},title1);
    figure(100);subplot(4,6,i);
    contourf(xa,ya,za_Work_hours(:,:,i));                    %按小时分布图
    title(['Contour-WorkHours' num2str(i)]);xlabel('longitude');
    ylabel('latitude');
    figure(101);subplot(4,6,i);
    surf(xa,ya,za_Work_hours(:,:,i));
    axis([-74.16 -73.7 40.57 40.92 0 300]);
    title(['Distribution-WorkHours' num2str(i)]);
    xlabel('longitude');
    ylabel('latitude');
    zlabel(['N_' num2str(i)]);

    figure(102);subplot(4,6,i);
    plot(data_Work_hours{i}(:,3),data_Work_hours{i}(:,4),'.','MarkerSize',1);
    title(['Plots-WorkHours' num2str(i)]);
```

```
    xlabel('longitude');ylabel('latitude');

    axis([-74.16 -73.7 40.57 40.92]);
    figure(103);subplot(4,6,i);
    contour(xa,ya,za_Work_hours(:,:,i),20);
    title(['Contour-OriWorkHours' num2str(i)]);
    xlabel('longitude');ylabel('latitude');

    figure(200);subplot(4,6,i);
    contourf(xa,ya,za_Weekend_hours(:,:,i));                    %按小时分布图
    title(['Contour-WeekendHours' num2str(i)]);
     xlabel('longitude');ylabel('latitude');

    figure(201);subplot(4,6,i);
    %surf(xa,ya,log(za_Weekend_hours(:,:,i)));
    surf(xa,ya,za_Weekend_hours(:,:,i));
    axis([-74.16 -73.7 40.57 40.92 0 300]);
    title(['Distribution-WeekendHours' num2str(i)]);
    xlabel('longitude');
    ylabel('latitude');
    zlabel(['N_' num2str(i)]');

    figure(202);subplot(4,6,i);
plot(data_Weekend_hours{i}(:,3),data_Weekend_hours{i}(:,4),'.','MarkerSize',1);
    title(['Plots-WeekendHours' num2str(i)]);
    xlabel('longitude');ylabel('latitude');

    axis([-74.16 -73.7 40.57 40.92]);
    figure(203);subplot(4,6,i);
    contour(xa,ya,za_Weekend_hours(:,:,i),20);
    title(['Contour-OriWeekendHours' num2str(i)]);
    xlabel('longitude');ylabel('latitude');
end
```


二、高效益寻客推荐模型求解代码

1、个人策略高效益寻客推荐模型

```
%司机个人载客策略
%要求：
%1、选定出发点，计算一天 24 个时段，最佳载客目的地的分布情况
%2、根据一天 24 个时段，载客目的地利润之和的不同分布，画热力图
%设计：
%网格(lat,lon)表示第 lat 行，lon 列的网格，在散点图上表现方式为：
%     以纬度为行、经度为列进行分割的网格
%单元格，dcel{x,y,t}为 N*M*T 三维单元格，存放 x,y 网格在 t 时刻的出行记录
%     为简化设计，将二维网格转换成一维列向量：
%     dcel{up,t}为 NM*T 的二维矩阵，x,y,up 的转换公式为：
%     up=(x-1)*M+y;    x=ceil(up/M);  y=up-(x-1)*M;
%预期利润，Profit[from,to,t]为三维矩阵，存放 t 时刻从 from 网格到 to 网格的利润
%     网格编号为一位列向量，转换公式如上
%计算公式，Profit(from,to,t)=A(to,t)/(N(to,t)*Ta(from,to)+T(to,t))
%     A(to,t)表示 to 目的地 t 时刻的所有乘车收入
%     N(to,t)表示 to 目的地 t 时刻的所有乘车次数
%     T(to,t)表示 to 目的地 t 时刻的累计乘车时间
%     Ta(from,to)表示从出发地 from 到 to 目的地所花的时间
%供需比约束：Down(to,t)<N(to,t)
%     Down(to,t)表示 t 时刻 to 目的地的下车人数，N(to,t)表示上车人数
clear;
clc;
close all;
MINLON=-74.03;
MAXLON=-73.76;
MINLAT=40.57;
MAXLAT=40.89;
MinTrips = 3;   %一个网格最小的出行记录数，小于这个值，则不推荐
loncos = cosd(40.75);   %纽约中心纬度的 cos 值
N=20;        %网格行个数，对纬度进行划分
M=20;        %网格列个数，对经度进行划分
%读数据，原始列：上下车时间、上下车经纬度、总金额、行驶距离
%     将上下车时间进行计算改写，变为时刻 T(1~24)，行驶时间(单位小时)
csvfile='taxichg-200k.csv';
data(:,1:2)=xlsread(csvfile,'B:C');      %上车下车时间
data(:,3:4)=xlsread(csvfile,'F:G');      %上车经纬度
data(:,5)=xlsread(csvfile,'S:S');        %总金额
data(:,6:7)=xlsread(csvfile,'J:K');      %下车经纬度
```

```
data(:,8)=xlsread(csvfile,'E:E');        %行驶距离

Index=find(data(:,3)<MINLON|data(:,3)>MAXLON|data(:,4)<MINLAT|data(:,4)>MAXLAT|
data(:,5)<2|data(:,5)>200);
data(Index,:)=[];
Index=find(data(:,6)<MINLON|data(:,6)>MAXLON|data(:,7)<MINLAT|data(:,7)>MAXLAT)
;
data(Index,:)=[];       %异常数据删除
data(:,9) = sqrt( ((data(:,3)-data(:,6))*cosd((MINLAT+MAXLAT)/2)).^2 ...
    + (data(:,4)-data(:,7)).^2 ) * 40000 / 360 / 1.609344;
Index=find(data(:,8)<0.15 | data(:,9)<0.1 | data(:,8)<data(:,9)*0.9 ...
   | data(:,8)>3*data(:,9));
data(Index,:)=[];
data(:,2)=24*(data(:,2)-data(:,1));      %行驶时间，单位小时
Index=find(data(:,5)>(30*data(:,2)+2*data(:,9)+2.5)*2|data(:,5)<(30*data(:,2)+2
*data(:,9)+2.5)*0.5);
data(Index,:)=[];
data(:,1)=(data(:,1)-floor(data(:,1)))*24;     %上车时间 0~24(0:0:0~23:59:59)
Index=find(data(:,2)>1 & data(:,5)<5*data(:,2));
data(Index,:)=[];

minlon = min(min(data(:,3)),min(data(:,6))) - 0.0001;
maxlon = max(max(data(:,3)),max(data(:,6))) + 0.0001;
minlat = min(min(data(:,4)),min(data(:,7))) - 0.0001;
maxlat = max(max(data(:,4)),max(data(:,7))) + 0.0001;
lonstep = (maxlon - minlon) / M;     %经度对应的网格大小
latstep = (maxlat - minlat) / N;     %纬度对应的网格大小

dcel = cell(N*M,24); %每个一维网格 1~24 时间段的出行记录，作为检验的中间变量
S = zeros(1,24);     %1~24 时段的行驶总里程
T = zeros(1,24);     %1~24 时段的行驶总时间
Down = zeros(N*M,24);   %每个一维网格 1~24 时间段的下车记录数，要小于该网格上车记
录数则推荐
for i=1:size(data,1)
    lonnum = ceil((data(i,3)-minlon)/lonstep);  %上车点经度对应的网格下标
    latnum = ceil((data(i,4)-minlat)/latstep);  %上车点纬度对应的网格下标
    t = ceil( data(i,1) + 0.00001 );     %上车时间段 1~24
    len = size(dcel{latnum+M*(lonnum-1),t},1)+1;      %对应长度+1
```

```
    dcel{latnum+M*(lonnum-1),t}(len,:) = data(i,:);  %记录赋值
    S(t) = S(t) + data(i,8);    %总里程
    T(t) = T(t) + data(i,2);    %总行驶时间
    lonnum2 = ceil((data(i,6)-minlon)/lonstep);  %下车点经度对应的网格下标
    latnum2 = ceil((data(i,7)-minlat)/latstep);  %下车点纬度对应的网格下标
    t2 = ceil( mod(data(i,1)+data(i,2),24) + 0.00001 );    %下车时间段 1~24
    Down(latnum2+M*(lonnum2-1),t2) = Down(latnum2+M*(lonnum2-1),t2) + 1;
end
V = S ./ T;     %1~24 小时的平均速度
Sdd = sum(S) / sum(data(:,9));  %路程与直线距离的平均比值

Profit = zeros(N*M,N*M,24);     %24 小时从出发点到目的地点的效益
lon = minlon+lonstep/2 : lonstep : maxlon;  %经度网线
lat = minlat+latstep/2 : latstep : maxlat;  %纬度网线
[X Y]=meshgrid(lon,lat);        %二维网格中心经纬度构造
XX=X(:);        %二维网格 x 轴转变成 1 维
YY=Y(:);        %二维网格 y 轴转变成 1 维
for To=1:N*M
    for t=1:24
        n_t = size(dcel{To,t},1);   %目的地 To，t 时刻的记录数
        if(n_t>0)           %目的地有出行记录
            x2 = ceil(To/M);   %目的地对应的经度网格，x 轴
            y2 = To - (x2-1)*M;   %目的地对应的纬度网格，y 轴
            lon2 = minlon + (x2-1)*lonstep+lonstep/2; %目的地对应的经度
            lat2 = minlat + (y2-1)*latstep+latstep/2; %目的地对应的纬度
            tmpX = XX;
            tmpY = YY;
            tmpX(x2) = lon2-lonstep/2;
            tmpY(y2) = lat2-latstep/2;
            UpVsDown = n_t/(Down(To,t)+1);  %接到客人概率：客源/(车源+1)
            if(UpVsDown>1)
                UpVsDown=1;             %表明车少，有车过去，必然会接到客人
            end
            Profit(:,To,t)=sum(dcel{To,t}(:,5))*UpVsDown./ ...    %效益分母
                ( sum(dcel{To,t}(:,2)) ...       %单元格的所有时间
                + ((tmpY-lat2).^2 + ((tmpX-lon2)*loncos).^2).^0.5 ... %地心夹角
                * 40000/360/1.609344 * Sdd / V(t) * n_t); %预估路程
        end
```

```
    end
end

MaxP = zeros(N*M,24);   %MaxP(from,t)表示在 t 时段 from 网格，到最大利润网格的下标
for From=1:N*M
    for t=1:24
        MaxP(From,t) = find(Profit(From,:,t)==max(Profit(From,:,t)));
    end
end

SumP = zeros(N*M,24);   %SumP(To,t)表示 t 时段，所有网格点到 To 网格的利润和
for To=1:N*M
    for t=1:24
        SumP(To,t) = sum(Profit(:,To,t));
    end
end

%%%------------画散点图-------------%%%%
figure(20);
plot(data(:,3),data(:,4),'.','MarkerSize',1);
hold on;
title('Plots-Profit-hot');
xlabel('longitude');ylabel('latitude');

for i=1:M-1     %画经度网格虚线
    DottedLine([minlon+i*lonstep,minlat],[minlon+i*lonstep,maxlat],M);
end
for i=1:N-1     %画纬度网格虚线
    DottedLine([minlon,i*latstep+minlat],[maxlon,i*latstep+minlat],N);
end

%%%---------利润图----------%%%
DestP = reshape(sum(SumP,2),N,M);
DestP(find(DestP==0)) = -inf;
contourf(lon, lat, DestP);  %每个目的地网格的所有时间段的利润和
legend('Profit');%利润
axis([minlon,maxlon,minlat,maxlat]);
```

```
%%%---------利润图----------%%%
Figure(21);
contourf(lon, lat, log(reshape(sum(SumP,2),N,M)));  %所有时间段的利润和
legend('ln(Profit)');
axis([minlon,maxlon,minlat,maxlat]);

figure(22);
for t=1:24
    Index = find(ceil(data(:,1)+0.0001) == t);
    subplot(4,6,t);
    plot(data(Index,3),data(Index,4),'.','MarkerSize',1);
    hold on;
    title(['Hour-' num2str(t)]);
    xlabel('longitude');ylabel('latitude');

    %%%---------利润图----------%%%
    DestP = reshape(SumP(:,t),N,M) ;
    DestP(find(DestP==0)) = -inf;
    contourf(lon, lat, DestP);  %每个目的地网格的所有时间段的利润和
    axis([minlon,maxlon,minlat,maxlat]);
end
```

2、集体策略高效益寻客推荐模型

```
%集体策略最优调度算法
%原则：
%以效益分布与出租车运力分布相匹配的原则进行调度
%最优解为调度的总行驶里程最小
%要求：
%1、求得 24 小时的最优调度策略，以分布方式（分布和为 1）进行进出表达
%2、画出 24 小时调度热力图，分为调出和调入两部分
%设计：
%网格(lat,lon)表示第 lat 行，lon 列的网格，在散点图上表现方式为：
%     以纬度为行、经度为列进行分割的网格
%网格出租车保有量，Down[down,t]为 NM*T 的二维矩阵
%     其中，t 为（下车时间+1）的整点时间，范围 1~24

close all;
clear;
```

```
clc;
MINLON=-74.03;
MAXLON=-73.76;
MINLAT=40.57;
MAXLAT=40.89;
%MinSchedule=2;   %一个网格最小的调度车辆数（万分之？），小于这个值，
则不调度
% loncos = cosd(40.75);   %纽约中心纬度的cos值
N=15;        %网格行个数，对纬度进行划分
M=15;        %网格列个数，对经度进行划分
%读数据，原始列：上下车时间、上下车经纬度、总金额、行驶距离
%    将上下车时间进行计算改写，变为时刻T(1~24)，行驶时间(单位小时)
% csvfile='taxi_20160420.csv';
% tday='Workday';
csvfile='taxi_20160423.csv';
tday='Weekend';

%上车下车时间 %上车经纬度 %下车经纬度 %行驶距离
data(:,[1:4 6:8])=xlsread(csvfile,'A:G');
data(:,5)=xlsread(csvfile,'N:N');          %总金额

Index=find(data(:,3)<MINLON|data(:,3)>MAXLON|data(:,4)<MINLAT|data(:
,4)>MAXLAT|data(:,5)<2|data(:,5)>200);
data(Index,:)=[];
Index=find(data(:,6)<MINLON|data(:,6)>MAXLON|data(:,7)<MINLAT|data(:
,7)>MAXLAT);
data(Index,:)=[];
data(:,9)                                                    =
sqrt( ((data(:,3)-data(:,6))*cosd((MINLAT+MAXLAT)/2)).^2 ...
    + (data(:,4)-data(:,7)).^2 ) * 40000 / 360 / 1.609344;  %直线距离
Index=find(data(:,8)<0.15 | data(:,9)<0.1 | data(:,8)<data(:,9)*0.9 |
data(:,8)>3*data(:,9));
data(Index,:)=[];
data(:,2)=24*(data(:,2)-data(:,1));      %行驶时间，单位小时
Index=find(data(:,5)>(30*data(:,2)+2*data(:,9)+2.5)*2 ...
    |data(:,5)<(30*data(:,2)+2*data(:,9)+2.5)*0.5 | data(:,2)<0);
data(Index,:)=[];
Index=find(data(:,2)>1 & data(:,5)<5*data(:,2));
```

```
data(Index,:)=[];
data(:,1)=(data(:,1)-floor(data(:,1)))*24; %上车时间(0:0:0~23:59:59)

minlon = MINLON - 0.0001;
maxlon = MAXLON + 0.0001;
minlat = MINLAT - 0.0001;
maxlat = MAXLAT + 0.0001;
lonstep = (maxlon - minlon) / M;    %经度对应的网格大小
latstep = (maxlat - minlat) / N;    %纬度对应的网格大小

Distance=zeros(N*M,N*M);          %记录节点间距离
UnitLongiLong=52.3327;
UnitLatiLong=69.043;

for i=1:N*M
    for j=i+1:N*M
        x1 = ceil(j/M);       %出发地对应的经度网格，x 轴
        y1 = j - (x1-1)*M;    %出发地对应的纬度网格，y 轴
        x2 = ceil(i/M);       %目的地对应的经度网格，x 轴
        y2 = i - (x2-1)*M;    %目的地对应的纬度网格，y 轴
     Distance(i,j)=abs(y2-y1)*UnitLatiLong+abs(x2-x1)*UnitLongiLong;
        Distance(j,i)=Distance(i,j);
    end
end

lon = minlon+lonstep/2 : lonstep : maxlon;  %经度网线
lat = minlat+latstep/2 : latstep : maxlat;  %纬度网线
Flagcel = cell(1,24);
Fmincel = cell(1,24);
nums = N*M;
SumOutP = zeros(nums,1);
SumInP = zeros(nums,1);
Up = zeros(N*M,24);   %每个一维网格 1~24 时间段的上车记录数
Cost = zeros(N*M,24);   %每个一维网格 1~24 时间段的营业收入
Down = zeros(N*M,24);   %每个一维网格 1~24 时间段的下车记录数
for i=1:size(data,1)
    lonnum = ceil((data(i,3)-minlon)/lonstep);  %上车点经度网格下标
    latnum = ceil((data(i,4)-minlat)/latstep);  %上车点纬度网格下标
```

```
    t = ceil( data(i,1) + 0.00001 );    %上车时间段 1~24
    Up(latnum+M*(lonnum-1),t) = Up(latnum+M*(lonnum-1),t) + 1;
    Cost(latnum+M*(lonnum-1),t) = ...
        Cost(latnum+M*(lonnum-1),t) + data(i,5);  %网格营收累加
    lonnum2 = ceil((data(i,6)-minlon)/lonstep);  %下车点经度网格下标
    latnum2 = ceil((data(i,7)-minlat)/latstep);  %下车点纬度网格下标
    t2 = ceil( mod(data(i,1)+data(i,2)+0.15,24) + 0.0001 );
    Down(latnum2+M*(lonnum2-1),t2)=Down(latnum2+M*(lonnum2-1),t2)+1;
 end

oop=zeros(nums*nums,24);
for t=1:24
    A=zeros(nums,nums*nums);
    Aeq=zeros(nums,nums*nums);   %表示第 1 个地区运往第 1 到 100 个地区
    for i=1:nums
        a=ones(1,nums);
        a(i)=0;
        A(i,(nums*(i-1)+1):(nums*i))=a;
        Aeq(i,i:nums:nums*nums)=ones(1,nums);
        Aeq(i,(nums*(i-1)+1):(nums*i))=-ones(1,nums);
        Aeq(i,i+(i-1)*nums)=0;
    end
    b=Down(:,t)/sum(Down(:,t));     %下车人数分布
    beq=Cost(:,t)/sum(Cost(:,t))-b;
    LB=zeros(nums*nums,1);
    LU=ones(nums*nums,1);
    [OP1 Fmincel{t} Flagcel{t}]= ...
        linprog(Distance(:),A,b,Aeq,beq,LB,LU);
    oop(:,t) = OP1;

    OP=reshape(OP1,nums,nums);
    OP=OP';
    for i=1:nums
        OP(i,i)=0;
    end

    %画散点图
    figure(10);
```

```
    hold on;
    Index = find(ceil(data(:,1)+0.0001) == t);
    subplot(4,6,t);
    plot(data(Index,3),data(Index,4),'.','MarkerSize',1);
    hold on;
    title([tday ' 调出-' num2str(t)]);
    xlabel('longitude');ylabel('latitude');

    %画热力图
    OutP = sum(OP,2) *10000;
    SumOutP = SumOutP + OutP;
    OutP(find(OutP<0.5)) = -inf;
    contourf(lon, lat, reshape(OutP,M,N));  %所有时间段的利润和
    axis([minlon,maxlon,minlat,maxlat]);

    %画调入热力图
    figure(11);
    InP = sum(OP) *10000;
    SumInP = SumInP + InP(:);
    InP(find(InP<0.5)) = -inf;
    contourf(lon, lat, reshape(InP,M,N));
    axis([minlon,maxlon,minlat,maxlat]);
end
```

三、模型优化代码

```
clc;
% 曼哈顿
manhattan1=[-74.02 40.74;-74.026 40.69;-74 40.705]; %QRC
manhattan2=[-74.02    40.74;-74    40.705;-73.975    40.71;-73.965
40.742]; %QCBA
manhattan3=[-74.02    40.74;-73.965    40.742;-73.916    40.8;-73.934
40.807]; %QAHK
manhattan4=[-74.02 40.74;-73.934 40.807;-73.935 40.84]; %QKL
manhattan5=[-74.02 40.74;-73.935 40.84;-73.933 40.884];
manhattan6=[-73.935 40.84;-73.91 40.87;-73.933 40.884];
s1=StrPolyarea(manhattan1, 'data(:,3)', 'data(:,4)');
s2=StrPolyarea(manhattan2, 'data(:,3)', 'data(:,4)');
s3=StrPolyarea(manhattan3, 'data(:,3)', 'data(:,4)');
```

```
s4=StrPolyarea(manhattan4, 'data(:,3)', 'data(:,4)');
s5=StrPolyarea(manhattan5, 'data(:,3)', 'data(:,4)');
s6=StrPolyarea(manhattan6, 'data(:,3)', 'data(:,4)');
fprintf('Manhattan\n%s ...\n|...\n%s ...\n|...\n%s ...\n|...\n%s ...
\n|...\n%s ...\n|...\n%s\n',s1,s2,s3,s4,s5,s6)

% 布鲁克林
brooklyn1=[-73.965 40.742;-73.975 40.71;-73.8 40.56;-73.83 40.668];
brooklyn2=[-73.975 40.71;-74 40.705;-74.06 40.66;-74.042 40.53; ...
    -73.8 40.56];
s1=StrPolyarea(brooklyn1, 'data(:,3)', 'data(:,4)');
s2=StrPolyarea(brooklyn2, 'data(:,3)', 'data(:,4)');
fprintf('Brooklyn\n%s ...\n|...\n%s\n',s1,s2)

% 皇后（包括两个机场）
queens1=[-73.965 40.742;-73.916 40.8;-73.65 40.79;-73.83 40.668];
queens2=[-73.65 40.79;-73.77 40.54;-73.8 40.56;-73.83 40.668];
s1=StrPolyarea(queens1, 'data(:,3)', 'data(:,4)');
s2=StrPolyarea(queens2, 'data(:,3)', 'data(:,4)');
fprintf('Queens\n%s ...\n|...\n%s\n',s1,s2)

% 肯尼迪机场
kennedy=[-73.793 40.639;-73.775 40.64;-73.776 40.65;-73.794 40.649];
fprintf('Kennedy\n%s\n', ...
        StrPolyarea(kennedy, 'data(:,3)', 'data(:,4)'))

% 拉瓜迪亚机场
laguardia=[-73.86 40.766;-73.86 40.771;-73.874 40.777;-73.876 40.772];
fprintf('Laguardia\n%s\n', ...
        StrPolyarea(laguardia, 'data(:,3)', 'data(:,4)'))

% 布朗克斯
bronx1=[-73.916 40.8;-73.934 40.807;-73.935 40.84; ...
    -73.91 40.87;-73.71 40.93;-73.65 40.79];
bronx2=[-73.91 40.87;-73.933 40.884;-73.88 40.94;-73.71 40.93];
s1=StrPolyarea(bronx1, 'data(:,3)', 'data(:,4)');
s2=StrPolyarea(bronx2, 'data(:,3)', 'data(:,4)');
fprintf('Bronx\n%s ...\n|...\n%s\n',s1,s2)
```

```
function constr = StrPolyarea(xydim, xstr, ystr)
%根据 xydim 平面点坐标构成的封闭凸多边形，程序自动首尾相接
%凸多边形点坐标，二列多行，第一列为 x，第二列为 y
%constr，返回封闭空间的字符型线性表达式
%xstr，x 变量的字符表达
%ystr，y 变量的字符表达
constr = '';
centreX = average(xydim(:,1));    %凸多边形中心 x 轴坐标
centreY = average(xydim(:,2));    %凸多边形中心 y 轴坐标
%fprintf('Centre (%d,%d)\n', centreX, centreY );
for i = 1 : length(xydim)
    if(i==1)    %最后一个点与第一个点尾首相连
        x1 = xydim(end,1);
        y1 = xydim(end,2);
    else
        x1 = xydim(i-1,1);
        y1 = xydim(i-1,2);
        constr = sprintf( '%s ...\n&', constr );
    end;
    x2 = xydim(i,1);
    y2 = xydim(i,2);
    %    fprintf('(%d,%d) (%d,%d)\n',x1, y1, x2, y2);
    %计算(x1,y1)、(x2,y2)的直线方程 y=kx+b 的系数 k、b
    if( x2==x1 ) %平行 x 轴，直线方程为 x = b  (b=x1=x2)
        if( centreX >= x1 )  %中心点在右边，即图形区域的点都在右边 >=x1
            constr = sprintf('%s %s >= %s', constr, xstr, num2str(x1) );
        else
            constr = sprintf('%s %s <= %s', constr, xstr, num2str(x1) );
        end
    else
        k = (y2-y1)/(x2-x1);
        b = y1 - k*x1;
        %    fprintf( 'LINE: y = %d * x + %d\n', k, b );
        cy = k * centreX + b;
        constr = sprintf( '%s %s', constr, ystr );
        if cy <= centreY  %中心点在对应的直线上方
            constr = sprintf( '%s >=', constr );
```

```
            else
                constr = sprintf( '%s <=', constr );
            end;
            constr = sprintf( '%s %s * %s + (%s)', constr, num2str(k), ...
                xstr, num2str(b));
        end;
    end;
```