

2021 第六届“数维杯”大学生

数学建模竞赛论文

题 目 基于地理空间数据的外卖配送模型

摘 要

外卖服务已经成为各大城市人群每日生活中重要的一部分，其内卷现象的日益严重让如何优化外卖平台、骑手、商家与消费者之间的关系变得至关重要。本文随机选取北京市为例，从餐厅地理位置入手，根据大数据分析，为外卖送餐做出了量化外卖整单质量的数学模型，并给出了对于外卖行业的最佳决策方案。

针对问题一，本文通过地理空间数据，建立关于安全配送的时间量化模型，首先利用 python 爬虫，爬取地理空间数据，借助 Arcgis 软件计算了餐饮地点与住宅区的距离，通过 Matlab 进行数据分析。对于服务质量，采取一级模糊综合评判模型对骑手的服务质量进行综合评价。对于完成质量的奖惩措施，本文将根据上述评分策略来建立对应的奖惩制度。

针对问题二，本文在问题一模型的基础上进行调整。通过计算骑手收入，根据实际情况，得到相对最优结果。对于静态提成，根据大数据结果，本文以每单 5 元进行提成，配送时间 0-2km 需 20min，2-4km 需 30min 为标准，将其与商品质量进行同问题的赋分和奖惩措施。对于动态提成，在固定的 5 元提成上加 2% 的订单总价和 2.5km 外每增加 1km 加 0.5 元作为每单提成，奖惩措施同问题一。

针对问题三，本文在问题一和问题二的基础上进行调整，将天气的影响量反应在模糊综合评判模型的权重改变上，并且配送费、提成金额和配送时间随着天气风险等级的增加而有所提升。

针对问题四，本文通过建立纳什均衡模型，给出平台、商家、骑手、买家策略集合以及对应的概率集合。通过博弈结果，以期望收益为筛选条件，最终给出对商家抽成 18.1%，骑手的基础提成金额 7.06 元的最佳模糊决策向量。

针对问题五，本文在前几问的基础上针对配送距离进行模型优化，增加了长距离(4-10km)的配送模式，并且对于跨区域不同商品的配送范围进行严格约束。

关键词 python 爬虫 地理空间数据分析 模糊综合评判模型 纳什均衡

一、问题分析

1.1 问题一分析

通过大量收集美团、饿了么等外卖平台的实际情况，我们将外卖配送范围设置为以住宅区为中心的 4km 内。将影响骑行安全的所有因素最终量化为配送时间。为满足骑行安全与高质量服务等因素的要求，从配送时间和服务质量两个方面进行评估。服务质量采取一级模糊综合评判模型对骑手的服务质量进行综合评价；配送时间由等单时间、路程时间、等红绿灯时间组成，最终根据大数据来规定大致范围，根据比例赋分。最后将各个部分的分数按一定比例汇总得到总分。对于骑手的奖惩机制，我组采用一级模糊综合评判方法对骑手的骑行安全与服务质量进行量化考核，针对不同的考核项目，我组根据其重要程度以及美团外卖平台的侧重，设计了不同的权重，最终确定模糊综合判断矩阵。

1.2 问题二分析

对于问题二相当于问题一的一个延伸，因此我们在问题一的基础上进行调整。对于静态提成，根据每单的送餐时间与服务质量的量化结果进行评定，在每单的基础固定提成上加设不同等级的惩罚与奖励。对于动态提成，在每单的基础固定提成上，根据每单的距离以及订单价格按一定比例进行增加，奖惩措施同问题一。

1.3 问题三分析

因为不同极端气候等突发状况对于外卖配送影响不同，本文首先将极端气候按照风险等级分类。此问中我们仅考虑大雨、大雪、雾霾、大风天气对于外卖配送的影响，在问题二的基础上，根据影响程度调整配送时间评分机制，进而改变外卖骑行安全和服务质量的量化结果。同时本文在提成机制上面也根据天气影响程度进行相应的改动，其余部分依然沿袭问题一所用模型，从而得到结果。

1.4 问题四分析

根据博弈纳什均衡理论：有限策略式博弈一定存在混合策略纳什均衡，其中，有限是指，每一方都有有限种纯策略。将外卖平台、商家、骑手、消费者四方抽象为有限的策略式博弈对局。设置他们可能采取的策略为一个策略集，针对每种策略采取的概率建立每种策略的概率集合。最终通过该策略式博弈对

局计算每一方的期望收益，从而得到最终的决策向量。

1.5 问题五分析

对于包含多种因素的长距离配送设定策略。本文先将区域由大数据以及相应地理进行划分，规定长距离配送范围在 4km 至 10km。同时配送模式由对消费者收取的配送费、对商家收取的配送费、对骑手的提成、配送时间这五个因素进行量化。对于跨区域配送，本文考虑到不同类别商品对于其质量的不同要求，对其配送范围进行约束，超过范围的区域一律禁止配送。

二、模型假设与符号说明

2.1 模型假设

- (1) 假设骑手接单后立即赶往商家，等餐时间指骑手到达餐饮地点后，还没有出餐，所等待的时间。
- (2) 假设骑手一对一接单，或在临近餐饮地点接送往相同或附近地点的多单，即不考虑有一定距离的多次配送问题。
- (3) 对于红绿灯，假设遇到红绿灯的概率服从区间长度为红绿灯时长的均匀分布，即： $T_{\text{灯}} \sim U(0, \beta)$ 。
- (4) 假设在骑手接单后不考虑退单问题。
- (5) 假设顾客对于骑手的评价客观实在，不存在估计抹黑差评的情况。
- (6) 在问题一、二中，不考虑配送费用相关。
- (7) 在所有问题中不考虑平台优惠。

2.2 符号说明

符号	含义
d	两个经纬度左边点 $(x_i, y_i), (x_j, y_j)$ 间的距离
R	地球半径
x_i	第 i 个点的经度

y_i	第 i 个点的纬度
D_i	在搜索半径内的第 i 个餐饮地点距离该住宅区的距离
\bar{D}	住宅区附近餐厅距离该住宅区的平均距离
σ_D	住宅区附近餐厅距离该住宅区的平均距离的标准差
β	红绿灯的等待时长
\bar{v}	骑手的车速
b_i	商家类别
$T_{\text{等}}$	等单时间
$T_{\text{行}}$	骑行时间
$g(x)$	等红绿灯的概率密度
$G(x)$	等红绿灯的分布函数
$T_{\text{灯}}$	等红绿灯的时间
$T_{\text{平}}$	平峰配送时间
$T_{\text{高}}$	高峰配送时间
$T_{\text{极}}$	极端天气下的配送时间
$H_{\text{订}}$	订单价格
E_i	第 i 方收益
M_1	静态提成
M_2	动态提成
p	配送费用
M_i	与 i 相关金额
q_i	评判的第 i 个等级
α	特殊天气计价
s_i	第 i 种影响因素

三、模型建立与求解

3.1 问题一的模型建立与求解

3.1.1 配送距离模型的建立及求解

3.1.1.1 模型建立

针对本问的要求和所需方案,我们认为最核心的部分是平台需要合理的规划配送时间,这其中需要包含考虑:骑手接单后等单时间、骑行的时间和路况情况所消耗时间。基于地理空间数据,采用 0-1 规划模型,概率分布模型进行数学建模,将送餐过程中的一些问题量化为时间问题,并对其建立为可用函数估计和计算的模型。通过地理经纬度坐标,计算住宅区和餐饮地点之间的距离。并用 ArcMap 进行地理空间数据分析,利用 Matlab 进行数据统计。

在 Arcgis 软件中进行住宅区 (x_i, y_i) 和餐饮地点 (x_j, y_j) 的经纬度坐标点,到距离 (d) 的计算:

$$d = R \times \arccos[\cos(y_i) \times \cos(y_j) \times \cos(x_i - x_j) + \sin(y_i) \times \sin(y_j)]$$

由于外卖平台设计了配送范围,通过对美团、饿了么等外卖平台进行大量的数据统计,最终我们确定以直线距离 4km 为配送范围,即以住宅区为中心,4km 的圆为半径,从而得出每个餐饮地点距离该住宅区的距离 (D_i) 并计算平均距离 (\bar{D}) 和标准差 (σ_D) :

$$\bar{D} = \frac{1}{n} \sum_{i=1}^n D_i$$

$$\sigma_D = \sqrt{\frac{\sum_{i=1}^n (D_i - \bar{D})^2}{n}}$$

3.1.1.2 模型求解

为更好的估计餐饮地点距离居民区的距离,利用百度提供的 API 地图开发工具,通过 python 爬虫的方式,利用 requests 函数库对目标网站发起请求,在百度 API 服务器接受响应后,获取到本组需要的餐饮地点和住宅区的相应内容,并以 json 字符串形式显示。通过 python 编程,将 json 字符串写入 excel 表格中,

从而获取到北京市五环内的餐厅，小吃店，奶茶店，面包店等有潜在外卖配送可能的地点的经纬度信息(python 源代码请见附件 1)。

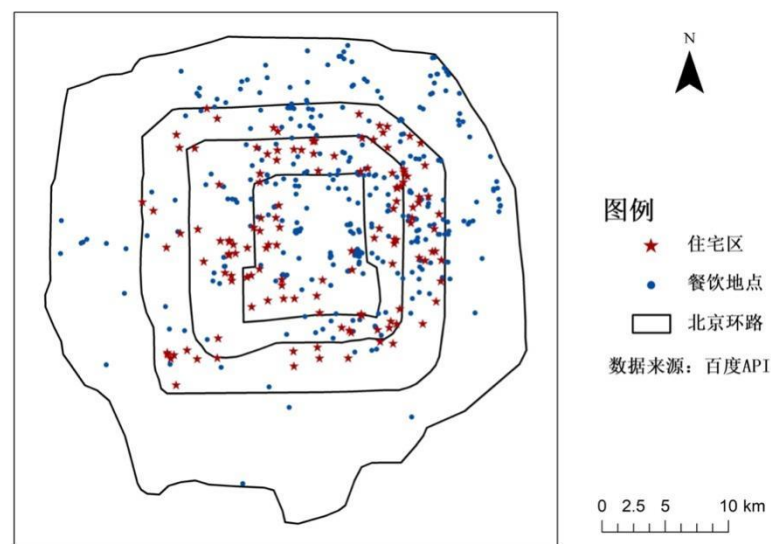


图 3-1 住宅区和餐饮地点的空间分布图(以北京为例)

在 Arcgis 软件中进行住宅区(x_i, y_i)和餐饮地点(x_j, y_j)的经纬度坐标点，从而得到每个餐饮地点距离该住宅区的距离(D_i)。部分结果如下（全部结果请见附件 2）：

表 3-1 餐饮地点到住宅区距离

序号	住宅区编号	餐饮地点编号	距离(单位: Km)
1	0	10	3.322
2	0	6	1.201
3	0	4	3.343
4	0	19	3.088
5	0	17	3.019
6	1	3	2.103
7	1	13	1.804
8	1	11	3.171
9	1	2	2.283

10	1	24	2.810
11	1	18	2.869
12	1	15	2.381
13	1	14	2.381
14	2	10	1.267
15	2	6	1.943
16	2	4	1.765
17	2	19	2.555
18	2	17	1.776
19	3	10	2.794
20	3	6	0.318

注：表 1 表示从此编号下的餐饮地点，到该编号下住宅区的距离

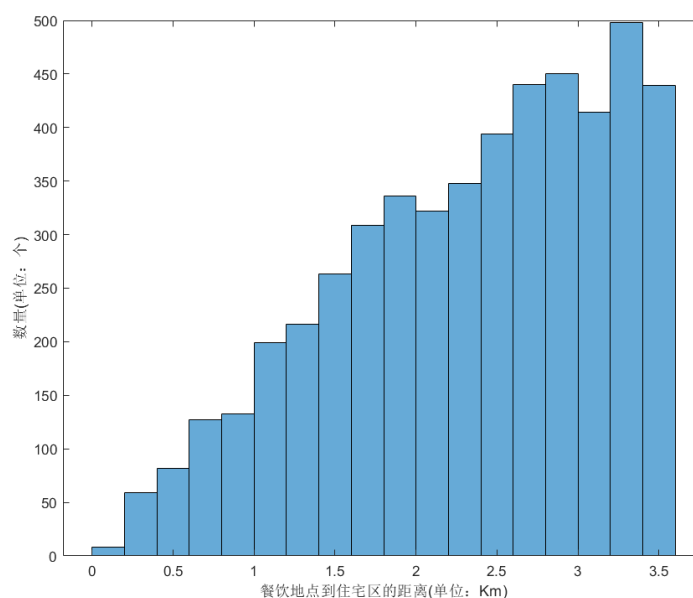


图 3-2 餐饮地点到住宅区距离

利用 matlab 画出频率分布直方图：其距离的平均值 (\bar{D}) 为：2.333m，最小值：0.064km，最大值：3.597km，标准差 (σ_D) 为：0.8492

3.1.2 配送时间模型的建立及求解

3.1.2.1 模型建立

为更好的度量骑手的骑行安全及快速送达,我们将影响骑行安全的所有因素最终量化为配送时间,即在一定时间范围内,显然骑手是在保证了骑行安全的情况下将外卖送达。为简化模型,在此问中我们的配送时间仅考虑等单时间、骑行时间、等红绿灯时间,以及对高峰和平峰不同情况的区分,而忽略路况、天气和一些意外情况。

首先考虑等单时间,其中包括骑手接单、赶往商家、商家制作等时间。为简化模型,我们假设外卖骑手在商家附近,消费者一下单,外卖骑手立刻接单并赶往商家,商家立刻开始制作。因此等单时间我们仅需要考虑商家的制作时间,和极少量的骑手赶往商家的时间。通过分析上述给出的商家,我们对商家进行大致分类,计商家 b 有:

$$b = \begin{cases} b_1 & \text{商家为快餐店和小吃店} \\ b_2 & \text{商家为正规餐馆} \end{cases}$$

根据大量数据统计及大数据分析,对于不同商家我们给出等单时间 $T_{\text{等}}$ 的分段函数:

$$T_{\text{等}} = \begin{cases} 0 \sim 10 \text{min} & b_1 \\ 5 \mp 20 \text{min} & b_2 \end{cases}$$

其次考虑骑行时间。在 2018 年中国工信部颁布的《电动自行车安全技术规范》中要求电动车的最高车速不得高于 25km/h,出于骑行安全、城市路况和外卖所需快速到达的要求考虑,规定骑手的车速(\bar{v})在 20km/h—25km/h,同时假设骑手在骑行过程中保持匀速运动,则有骑行时间 $T_{\text{行}}$:

$$T_{\text{行}} = \frac{D_i}{\bar{v}}$$

对于路程中遇到的信号灯的情况,则根据遇到的信号灯个数(n)来延长配送时长。根据实际情况假设每 600m 设置一红绿灯,且每个红绿灯的时长均为 1.5min。

用 0-1 规划函数 $f(x)$ 来表示遇到红绿灯的情况:

$$f(x) = \begin{cases} 0 & \text{未遇到红绿灯} \\ 1 & \text{遇到红绿灯} \end{cases}$$

以及红绿灯的等待时长 β 。遇到红绿灯后,需要等待的概率服从区间长度为 β 的均匀分布($U(0, \beta)$)。其概率密度表达式:

$$g(x) = \begin{cases} \frac{1}{\beta} & , 0 \leq x \leq \beta \\ 0 & , \text{其他} \end{cases}$$

分布函数为在该区间上对于密度函数的积分：

$$G(x) = \int_0^x \frac{1}{\beta} dx, a \leq x \leq b$$

分布函数表达式为：

$$G(x) = \begin{cases} 0 & , x < 0 \\ \frac{x}{\beta} & , a \leq x \leq b \\ 1 & , x \geq b \end{cases}$$

故，因红绿灯而延长的时间

$$T_{灯} = n \times \beta \times f(x) \times G(x)$$

综合上述三种时间，我们得到平峰时总配送时间 $T_{平}$ 有

$$T_{平} = T_{等} + T_{行} + T_{灯}$$

最后考虑到外卖订餐有高峰及平峰的区分，随着订单量的增加，为保障商品质量，根据实际情况，我们在高峰时适当增加平峰时间的 30%，以进行周转，则高峰时配送时间 $T_{高}$ 有：

$$T_{高} = (1 + 30\%)T_{平}$$

3.1.2.2 模型求解

在一次外卖配送中，若骑手接到小吃和快餐店(b_1)的订单，且该时段处于平峰时期，接到订单后立即赶往商家，则最少时间：

$$\begin{aligned} T_{平} &= T_{等} + T_{行} + T_{灯} \\ T_{等} &= \begin{pmatrix} 0 \sim 10min & b_1 \\ 5 \mp 20min & b_2 \end{pmatrix}_{min} \\ (T_{行})_{min} &= \frac{D_i}{v} \approx 7 \\ T_{灯} &= \left(n \times \beta \times f(x) \times \int_0^x \frac{1}{\beta} dx \right)_{min} \approx 4.5 \end{aligned}$$

则： $(T_{平})_{min} \approx 10.1$ 分钟；则： $T_{高} = (1 + 30\%)T_{平} \approx 13.13$ 分钟

最多时间：

$$\begin{aligned}
 T_{\text{平}} &= T_{\text{等}} + T_{\text{行}} + T_{\text{灯}} \\
 T_{\text{等}} &= \left(\begin{array}{cc} 0 \sim 10 \text{min} & b_1 \\ 5 \mp 20 \text{min} & b_2 \end{array} \right)_{\text{max}} \\
 (T_{\text{行}})_{\text{max}} &= \frac{D_i}{v} \approx 7 \\
 T_{\text{灯}} &= \left(n \times \beta \times f(x) \times \int_0^x \frac{1}{\beta} dx \right)_{\text{max}} \approx 9
 \end{aligned}$$

则： $(T_{\text{平}})_{\text{min}} \approx 26$ 分钟；则： $T_{\text{高}} = (1 + 30\%)T_{\text{平}} \approx 33.8$ 分钟

若在一次外卖配送中，若骑手接到正规餐厅(b_2)的订单，且该时段处于平峰时期，接到订单后立即赶往商家，则最少时间：

$$\begin{aligned}
 T_{\text{平}} &= T_{\text{等}} + T_{\text{行}} + T_{\text{灯}} \\
 T_{\text{等}} &= \left(\begin{array}{cc} 0 \sim 10 \text{min} & b_1 \\ 5 \mp 20 \text{min} & b_2 \end{array} \right)_{\text{min}} \\
 (T_{\text{行}})_{\text{min}} &= \frac{D_i}{v} \approx 7 \\
 T_{\text{灯}} &= \left(n \times \beta \times f(x) \times \int_0^x \frac{1}{\beta} dx \right)_{\text{min}} \approx 4.5
 \end{aligned}$$

则： $(T_{\text{平}})_{\text{min}} \approx 16.5$ 分钟；则： $T_{\text{高}} = (1 + 30\%)T_{\text{平}} \approx 21.5$ 分钟

最多时间：

$$\begin{aligned}
 T_{\text{平}} &= T_{\text{等}} + T_{\text{行}} + T_{\text{灯}} \\
 T_{\text{等}} &= \left(\begin{array}{cc} 0 \sim 10 \text{min} & b_1 \\ 5 \mp 20 \text{min} & b_2 \end{array} \right)_{\text{max}} \\
 (T_{\text{行}})_{\text{max}} &= \frac{D_i}{v} \approx 7 \\
 T_{\text{灯}} &= \left(n \times \beta \times f(x) \times \int_0^x \frac{1}{\beta} dx \right)_{\text{max}} \approx 9
 \end{aligned}$$

则： $(T_{\text{平}})_{\text{min}} \approx 31$ 分钟；则： $T_{\text{高}} = (1 + 30\%)T_{\text{平}} \approx 40.3$ 分钟

3.1.3 奖惩措施模型的建立及求解

3.1.3.1 模型建立

建立奖惩机制的一级模糊综合评判模型，首先确定因素集合：骑手的表现主

要集中在骑行安全与服务质量。

这些因素构成了评价指标体系集合，即因素集合，记为：

$$S = \{s_1, s_2\}$$

分别对应于：

$$S = \{\text{骑行安全性 } s_1, \text{服务质量 } s_2\}$$

确定评价集合，由于每个指标的评价值的不同，导致了骑手的评判也会出现不同的等级。其中，我组将评价设为四个等级：好，较好，一般，差。将评语集记为：

$$Q = \{q_1, q_2, q_3, q_4\}$$

分别对应于：

$$Q = \{\text{好 } q_1, \text{较好 } q_2, \text{一般 } q_3, \text{差 } q_4\}$$

根据不同评价指标重要程度以及美团外卖平台的侧重，设置各个因素之间的权重分配 (a_i) ，它是 S 的一个模糊向量，记为：

$$P = (0.6, 0.4)$$

分别对应于：

$$P = (\text{骑行安全性 } 0.6, \text{服务质量 } 0.4)$$

其中：

$$\sum_{i=1}^2 a_i = 1$$

最后确定模糊综合判断矩阵。对指标 s_i 来说，各个评语集隶属度为 Q 上的模糊子集，对指标 s_i 的评价记为：

$$V_i = [v_{i1}, v_{i2}]$$

各个指标的模糊综合判断矩阵为：

$$V = \begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix}$$

它是一个从 S 到 Q 的模糊关系矩阵。

对于每一个从 S 到 Q 的模糊关系 $V = (v_{ij})_{n \times m}$ ，对应唯一的模糊变换：

$$T_V: F(S) \rightarrow F(Q)$$

最终得到综合评价的结果： $\text{Result} = P \cdot V$

评价结果即为Q上的一个模糊向量，记为：

$$\text{Result} = [r_1, r_2]$$

令 r 为最后评判结果，

$$r = \max(r_1, r_2)$$

关于奖励部分，内容分为两部分。第一部分为针对每一单的送餐质量进行奖励(M)，

$$M = \begin{cases} 0.2 & r \geq 90 \\ 0.1 & 75 \leq r < 90 \\ 0 & 60 \leq r < 75 \end{cases}$$

针对呈 q_1 ($r \geq 90$) 的结果，为了使外卖骑手能够获得一个与自体力付出成正比的回报，在每单所获基础上额外奖励 0.2 元；针对呈 q_2 (总分 ≥ 75) 的结果，在每单所获基础上额外奖励 0.1 元；针对 q_3 (总分 ≥ 60) 的结果，只获取相应的基础提成。第二部分为针对一天总单数超过 34 的给予额外 10 元奖励，针对每一个月总单数超过 1000，给予额外 300 元奖励，以激励骑手进行更高效的配送服务。

关于惩罚部分，针对 q_4 (总分 ≤ 60) 的结果，以首次 10 元，乘二制进行处罚。

3.2 问题二的模型建立与求解

3.2.1 静态配送模型的建立及求解

首先考虑包含多种因素在内的静态提成策略。通过对现今外卖平台提成策略的统计和分析，我们规定每单以固定的 8 元进行提成，以配送时间 T ：

$$T = \begin{cases} 20min & 0 - 2km \\ 30min & 2 - 4km \end{cases}$$

为配送时间的标准。评分机制和奖惩措施同问题一。

同时根据实际情况，考虑到静态订单配送会使外卖骑手为相对轻松的获得更高利润，而拒接距离较远的订单。为避免这种情况的发生，对拒接距离远订单的外卖骑手给予首单为 15 元的乘二制处罚。

3.2.2 动态配送模型的建立及求解

3.2.2.1 模型建立

关于动态部分，我组认为奖励机制的影响因素有：订单价格（ H_{ij} ）、订单路程（ S ）。针对订单路程，除首公里及第二公里外的公里数计算在动态提成的影响范围内。根据不同评价指标重要程度以及美团外卖平台的侧重，设置各个因素所乘以的比例系数（ k_2 ）。对于动态提成（ M_2 ），本组最终决定其在一定金额的奖励基础上进行各个因素相关的调整，记为：

$$M_2 = 7 + H_{ij} * 2\% + (D_i - 2) * 0.5$$

评分机制和奖惩措施同问题一。

3.2.2.2 模型求解

在此问的求解中，根据实际情况，我们以订单价格 30 元为例，得出动态情况下，外卖骑手的提升，部分结果如下：

表 3-2 不同距离动态提成

距离（km）	提成（元）
2.00	7.60
2.40	7.80
2.64	7.92
3.00	8.10
3.36	8.28
3.60	8.40
4.00	8.60

3.3 问题三的模型建立与求解

3.2.1 极端天气下配送费模型的建立与求解

对于极端天气下的外卖订单情况，我们仅就常见天气大雨、大雪、大风、雾霾四种天气进行讨论，从而得出配送费、配送时长和提成的方案。首先是平台对于消费者收取的配送费用，本组参考美团外卖和饿了么配送费标准，并进行了一些改进。对用户的配送费（ $h(x)$ ）收取是关于距离（ x ）的函数，设定 2km 以内不收取

配送费，2-4km 以 2 元为基价，每增加 100m 收取 0.2 元收取配送费，则有：

$$h(x) = \begin{cases} 0 & 0 < x < 2km \\ 2 + 0.2 \cdot (x - 2) \cdot 10 & 2 \leq x \leq 4km \end{cases}$$

通过 Matlab 绘制出配送费随配送距离的函数 ($h(x)$):

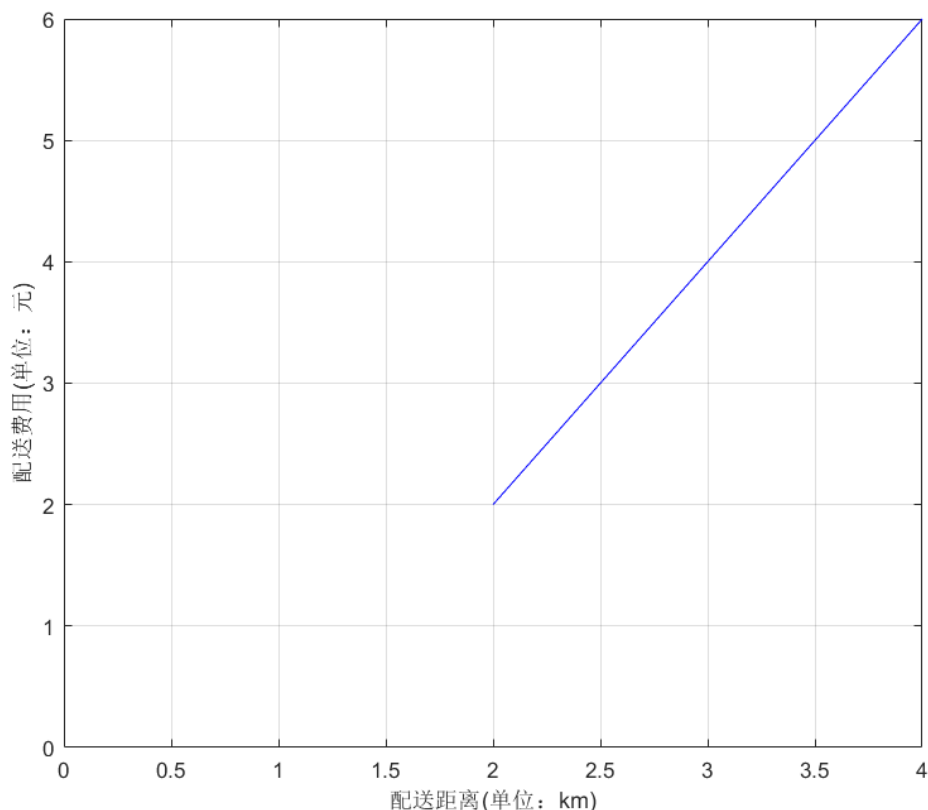


图 3-3 配送费随配送距离的函数

对于特殊天气，规定特殊天气基价(α)。下雨天气，以气象台公布的降水量 (l : 单位为毫米/24h) 为基准，则需要收取的配送费 ($H(x)$):

$$H(x) = \begin{cases} h(x) + 0.3\alpha & 3 \leq l < 10 \\ h(x) + 0.6\alpha & 10 \leq l < 25 \\ h(x) + \alpha & 25 \leq l < 50 \\ \text{骑手不接单} & l \geq 50 \end{cases}$$

下雪天气，以气象台公布的降水量 (l : 单位为毫米/24h) 为基准，则需要收取的配送费 ($H(x)$):

$$H(x) = \begin{cases} h(x) + 0.3\alpha & 0.1 \leq l < 2.4 \\ h(x) + 0.6\alpha & 2.5 \leq l < 4.9 \\ h(x) + \alpha & 5 \leq l < 10 \\ \text{骑手不接单} & l \geq 10 \end{cases}$$

刮风天气，以气象台公布的风力等级为基准，则需要收取的配送费($H(x)$):

$$H(x) = \begin{cases} h(x) + 0.3\alpha & 4,5 \text{ 级风} \\ h(x) + 0.6\alpha & 6 \text{ 级风} \\ h(x) + \alpha & 7 \text{ 级风} \\ \text{骑手不接单} & \geq 8 \text{ 级风} \end{cases}$$

雾霾天气，以气象台公布的能见度(L : 单位为米)为基准，需要收取的配送费($H(x)$):

$$H(x) = \begin{cases} h(x) + 0.3\alpha & L < 1000 \\ h(x) + 0.6\alpha & 200 \leq L < 500 \\ h(x) + \alpha & 50 \leq L < 200 \\ \text{骑手不接单} & L < 50 \end{cases}$$

3.2.2 极端天气下配送时间模型的建立与求解

对于极端天气下配送时长($T_{\text{极}}$)的量化，基于问题一模型的量化函数求得的平峰时的配送时常($T_{\text{平}}$), 以及高峰时的配送时常($T_{\text{高}}$), 根据不同的天气状况和风险情况，给予骑手设置不同的极端天气下配送时长($T_{\text{极}}$)。

下雨天气，以气象台公布的降水量(l : 单位为毫米/24h) 为基准，则极端天气下配送时长($T_{\text{极}}$):

$$T_{\text{极}} = \begin{cases} 1.3T_{\text{平}}; 1.4T_{\text{高}} & 3 \leq l < 10 \\ 1.5T_{\text{平}}; 1.6T_{\text{高}} & 10 < l < 25 \\ 1.7T_{\text{平}}; 1.8T_{\text{高}} & 25 < l < 50 \\ \text{不接单} & l \geq 50 \end{cases}$$

下雪天气，以气象台公布的降水量(l : 单位为毫米/24h) 为基准，则极端天气下配送时长($T_{\text{极}}$):

$$T_{\text{极}} = \begin{cases} 1.3T_{\text{平}}; 1.4T_{\text{高}} & 0.1 \leq l < 2.4 \\ 1.5T_{\text{平}}; 1.6T_{\text{高}} & 2.5 \leq l < 4.9 \\ 1.7T_{\text{平}}; 1.8T_{\text{高}} & 5 \leq l < 10 \\ \text{不接单} & l \geq 10 \end{cases}$$

刮风天气，以气象台公布的风力等级为基准，则极端天气下配送时长($T_{\text{极}}$):

$$T_{\text{极}} = \begin{cases} 1.3T_{\text{平}}; 1.4T_{\text{高}} & 4,5 \text{ 级风} \\ 1.5T_{\text{平}}; 1.6T_{\text{高}} & 6 \text{ 级风} \\ 1.7T_{\text{平}}; 1.8T_{\text{高}} & 7 \text{ 级风} \\ \text{不接单} & \geq 8 \text{ 级风} \end{cases}$$

雾霾天气，以气象台公布的能见度(L : 单位为米)为基准，则极端天气下配送时长($T_{\text{极}}$):

$$T_{\text{极}} = \begin{cases} 1.3T_{\text{平}}; 1.4T_{\text{高}} & L < 1000 \\ 1.5T_{\text{平}}; 1.6T_{\text{高}} & 200 \leq L < 500 \\ 1.7T_{\text{平}}; 1.8T_{\text{高}} & 50 \leq L < 200 \\ \text{不接单} & L < 50 \end{cases}$$

3.2.3 极端天气下提成模型的建立与求解

对于极端天气下骑手的提成($M_{\text{极}}$)，平台根据不同的天气状况和风险情况，给予骑手不同程度的提成。基于第二问的动态提成(M_2)的规则：

$$M_2 = 8 + H_{ij} \times 2\% + (D_i - 1) \times 50\%$$

下雨天气，以气象台公布的降水量(l : 单位为毫米/24h) 为基准，则骑手收到的极端天气提成($M_{\text{极}}$):

$$M_{\text{极}} = \begin{cases} 1.1M_2 & 3 \leq l < 10 \\ 1.3M_2 & 10 < l < 25 \\ 1.5M_2 & 25 < l < 50 \\ \text{不接单} & l \geq 50 \end{cases}$$

下雪天气，以气象台公布的降水量(l : 单位为毫米/24h) 为基准，则骑手收到的极端天气提成($M_{\text{极}}$):

$$M_{\text{极}} = \begin{cases} 1.1M_2 & 0.1 \leq l < 2.4 \\ 1.3M_2 & 2.5 \leq l < 4.9 \\ 1.5M_2 & 5 \leq l < 10 \\ \text{不接单} & l \geq 10 \end{cases}$$

刮风天气，以气象台公布的风力等级为基准，则骑手收到的极端天气提成($M_{\text{极}}$):

$$M_{\text{极}} = \begin{cases} 1.1M_2 & 4,5 \text{ 级风} \\ 1.3M_2 & 6 \text{ 级风} \\ 1.5M_2 & 7 \text{ 级风} \\ \text{不接单} & \geq 8 \text{ 级风} \end{cases}$$

雾霾天气，以气象台公布的能见度(L : 单位为米)为基准，则骑手收到的极端天气提成($M_{\text{极}}$):

$$M = \begin{cases} 1.1M_2 & L < 1000 \\ 1.3M_2 & 200 \leq L < 500 \\ 1.5M_2 & 50 \leq L < 200 \\ \text{不接单} & L < 50 \end{cases}$$

3.4 问题四的模型建立与求解

3.4.1 外卖服务利益相关者纳什均衡模型的建立

从博弈论角度考虑，

定义外卖平台，商家，骑手，买家四方策略集合分别为 A_1, A_2, A_3, A_4 ，纯策略 $a_i \in A_i$ ，表示为：

$$A_i = \{a_{i1}, a_{i2}, \dots, a_{in}\}$$

共有 n 种决策。针对于每个决策，每一个人的决策集合是一个样本空间，且有一个选择该对策的概率集合，表示为：

$$\Delta(A_i) = \left\{ p_i = \{p_{i1}, p_{i2}, \dots, p_{in}\}, p_{ij} \geq 0, \sum_j p_{ij} = 1 \right\}$$

则，混合决策集合的表示：

$$p_i = (p_{i1}, p_{i2}, \dots, p_{in}) \in \Delta(A_i)$$

那么这四方的博弈结果：

$$p = (p_1, p_2, \dots, p_N), \text{ 其中: } p_i \in \Delta(A_i)$$

那么除去自身的博弈结果：

$$p_{-i} = (p_1, \dots, p_{i-1}, p_{i+1}, \dots, p_N), \text{ 则: } p = (p_i, p_{-i})$$

给定一个策略式博弈(G):

$$G = \{N, \{A_i\}, \{u_i\}\}$$

那么根据四方的博弈结果集合 (p) ，计算四方的期望收益：

$$U_i(p) = \sum_{a \in A} p(a) u_i(a) = \sum_{a=(a_1, \dots, a_N) \in A} p_1(a_1) \times \dots \times p_N(a_N) u_i(a)$$

3.4.2 模型求解

表 3-4-1 商家和平台博弈

		外卖商家	
		廉价食材	高质量食材
外卖平台	高提成点	$E_2 - J, E_1$	$0, 0$
	低提成点	$E_2 - I, E_1 - C$	$E_2 - I, E_1 - F$

C ：为廉价食材而带来的顾客流失影响的收益

F ：因高质量食材的高成本而带来的收益的流失

I ：因为低提成点而造成的平台收益流失

J ：因为高提成点而造成的外卖商家流失的收益

E_1 ：商家的总利润

E_2 ：平台的总利润

表 3-4-2 商家和消费者博弈

		外卖商家	
		廉价食材	高质量食材
消费者	高消费	$0, 0$	E_3, E_1
	低消费	$E_3, E_1 - C$	$E_3, E_1 - F$

C ：为廉价食材而带来的顾客流失影响的收益

F ：因高质量食材的高成本而带来的收益的流失

E_1 ：商家的总利润

E_3 ：消费者的消费金额

表 3-4-3 骑手和平台博弈

		骑手	
		好服务态度	差服务态度
外卖平台	高提成点	0, 0	$E_2 - Z, E_4$
	低提成点	$E_2, E_4 - W$	E_2, E_4

W : 由于骑手送餐态度低迷造成的收入降低

Z : 由于高提成点造成的外卖平台损失

E_2 : 平台的总利润

E_4 : 骑手总收入

根据博弈纳什均衡理论: 有限策略式博弈一定存在混合策略纳什均衡, 其中, 有限是指, 每一方都有有限种纯策略。参考市面各大外卖平台, 以混合策略纳什均衡点为最优决策, 最终, 表示为最佳决策模糊向量:

$$p_f = (p_{1f}, p_{2f})$$

$$p_f = (\text{对商家抽成 } 18.1\%, \text{ 骑手的基础提成金额 } 7.06 \text{ 元})$$

除此之外, 商家在保证食材质量的同时, 可以降低生产成本。平台对于商家的服务类型也要收取年推广费 10000-30000 元, 并且每单收取保底提成 5 元。平台也应该对用户, 根据每单金额, 给予不同程度的优惠, 对于一些商品, 采取特殊的折扣。消费者也可根据自身情况和服务质量, 给骑手少量金额的打赏, 该部分打赏, 平台不从中抽取任何提成。

3.5 问题五的模型建立与求解

3.5.1 长距离配送模型的建立及求解

3.5.1.1 模型建立

首先考虑包含多种因素的长距离配送设定策略。由大数据以及相应地理划分, 我们规定长距离配送范围在 4km 至 10km。对于长距离配送模式, 我组考虑以下几种因素: 对消费者收取的配送费、对商家收取的配送费、对骑手的提成、配送时间。

对消费者收取的配送费，以配送距离作为重要的考虑因素，通过对现有外卖平台上的配送费数据进行分析，我们决定在 4km-10km 设置每单以固定的 5 元进行基础配送费的计算，所得配送费（ p ）为：

$$p = \ln 9 * e^{\frac{x-4}{3}} + 5$$

对于配送时间（ T ），

$$T = \tau(T_{\text{等}} + T_{\text{行}} + T_{\text{灯}})$$

其中 $T_{\text{等}}$ 同问题一，考虑到长距离配送时间长，意外多，我们将平峰和高峰视作相同情况计算，总体乘以根据美团等外卖平台的数据设置一个系数因子： τ 。

$$T_{\text{行}} = \gamma \frac{D_i}{v},$$

γ 为根据实际情况设立的系数因子。

其中 $T_{\text{灯}}$ 则直接根据美团、饿了么等外卖平台数据进行估算：

$$T_{\text{灯}} = 15\text{min}$$

综合上述三种时间，我们得到总配送时间 T 有

$$T = \tau(T_{\text{等}} + T_{\text{行}} + T_{\text{灯}})$$

对于骑手提成，考虑到仅是距离发生较大变化，因此我们在问题二的动态模型基础上，根据现今外卖平台的提成策略稍加改进，进行分段增加提成，则有提成（ M ）：

$$M = \begin{cases} M_{\text{at}} = M_2 + 0.4 * (D_i - 4) * 10 & 4\text{km} \leq t < 6\text{km} \\ M_{\text{bt}} = M_{\text{a6}} + 0.45 * (D_i - 6) * 10 & 6\text{km} \leq t \leq 10\text{km} \end{cases}$$

对于平台对商家收取的服务费，通过对美团平台的数据统计，我们将其分为三部分：距离收费、价格收费和时段收费，同时平台对商家收取的服务费保底为 5 元。

首先距离收费($M_{\text{距}}$)为：

$$M_{\text{距}} = \begin{cases} 6 & 0 - 4\text{km} \\ 6 + 0.3 * (D_i - 4) * 10 & 6 - 10\text{km} \end{cases}$$

其次根据实际情况，我们规定起送价格为 30 元，则价格收费（ $M_{\text{商}}$ ）为：

$$M_{\text{商}} = \begin{cases} 0 & 0 < x < 30 \text{ 元} \\ (H_{\text{订}} - 30) * 0.13 & x \geq 30 \text{ 元} \end{cases}$$

最后时段收费 ($M_{\text{时}}$) 为:

$$M_{\text{时}} = \begin{cases} 1.5 \text{ 元} & 00:00—03:00 \\ 2 \text{ 元} & 03:00—06:00 \\ 0 \text{ 元} & 06:00—21:00 \\ 1 \text{ 元} & 21:00—24:00 \end{cases}$$

则最终得出平台对商家收取的服务费($M_{\text{服}}$)为:

$$M_{\text{服}} = M_{\text{距}} + M_{\text{商}} + M_{\text{时}}$$

3.5.1.2 模型求解

对于一次 6Km 的外卖配送, 假设其商品价格为 50 元, 配送时段为 6—21 点

$$p = \ln 9 * e^{\frac{x-4}{3}} + 5 \approx 8.6 \text{ (元)}$$

即: 用户所需要缴纳的配送费为 8.6 元

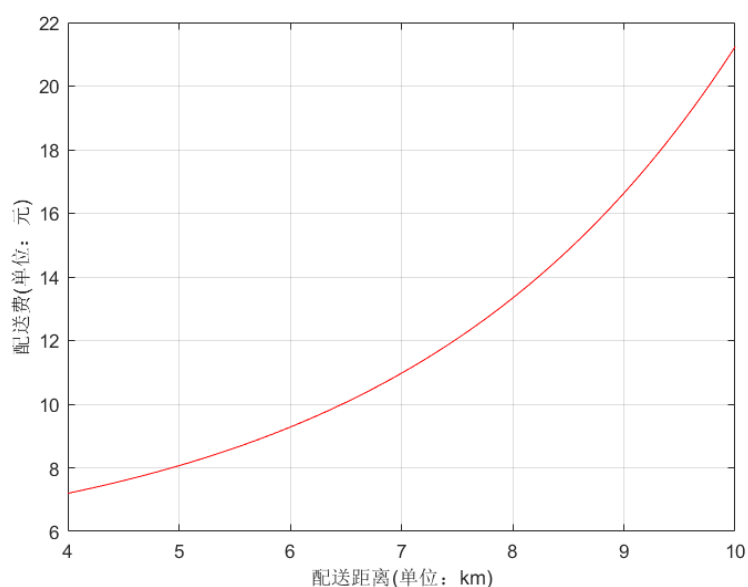


图 3-5-1 配送费随配送距离的函数

对于骑手的配送时长:

$$T = \tau(T_{\text{等}} + T_{\text{行}} + T_{\text{灯}})$$

$$T_{\text{等}} = \begin{pmatrix} 0 \sim 10 \text{ min} & b_1 \\ 5 \mp 20 \text{ min} & b_2 \end{pmatrix}_{\text{ave}}$$

$$T_{\text{行}} = \gamma \frac{D_i}{v} \approx 18 \text{ (分钟)},$$

$$T_{\text{订}} = 15 \text{ (分钟)}$$

τ 取: 1.42, γ 取: 1.02, 则: $T \approx 55$ (分钟)

对于骑手的提成(M):

$$M_2 = 7 + H_{\text{订}} \times 2\% + (D_i - 1) \times 50\% = 8 + 1.5 = 9.5 (D_i \in [0,4])$$

$$M = M_2 + 0.4 * (D_i - 4) * 10 = 17.5 \text{ (元)}$$

对于商家的收费($M_{\text{服}}$):

$$M_{\text{距}} = \begin{pmatrix} 6 & 0-4km \\ 6 + 0.3 * (D_i - 4) * 10 & 6-10km \end{pmatrix}_{6km}$$

$$M_{\text{商}} = \begin{pmatrix} 0 & 0 < x < 30 \text{ 元} \\ (H_{\text{订}} - 30) * 0.13 & 30 \leq x \end{pmatrix}_{50 \text{ 元}}$$

$$M_{\text{时}} = \begin{pmatrix} 1 \text{ 元} & 00:00-03:00 \\ 1.5 \text{ 元} & 03:00-06:00 \\ 0 \text{ 元} & 06:00-21:00 \\ 0.5 \text{ 元} & 21:00-24:00 \end{pmatrix}_{6-21}$$

$$M_{\text{服}} = M_{\text{距}} + M_{\text{商}} + M_{\text{时}} \approx 8.6 \text{ (元)}$$

同理, 对于一次 8km 的外卖配送, 假设其商品价格为 100 元, 配送时段为 06:00—21:00, 则用户需要缴纳的配送费为: 12.6 元, 对于骑手的配送时长, τ 取: 1.42, γ 取: 1.2, 则: $T \approx 69$ (分钟)。对于骑手的提成(M)为 27.5 元 商家的收费($M_{\text{服}}$)为 27.1 元。

3.5.2 跨区域配送模型的建立

通过大数据分析, 我们认为跨区域中所指的区域, 即为商品在保证质量的条件下所允许配送的最大范围。而跨区域配送较难实现, 主要是因为外卖所配送商品其对商品质量的要求很高, 因此我们对各种不同的商品在配送范围上进行严格的划分:

$$\text{配送范围} \begin{cases} < 4km & \text{餐饮食品类} \\ < 15km & \text{生活日用品类} \end{cases}$$

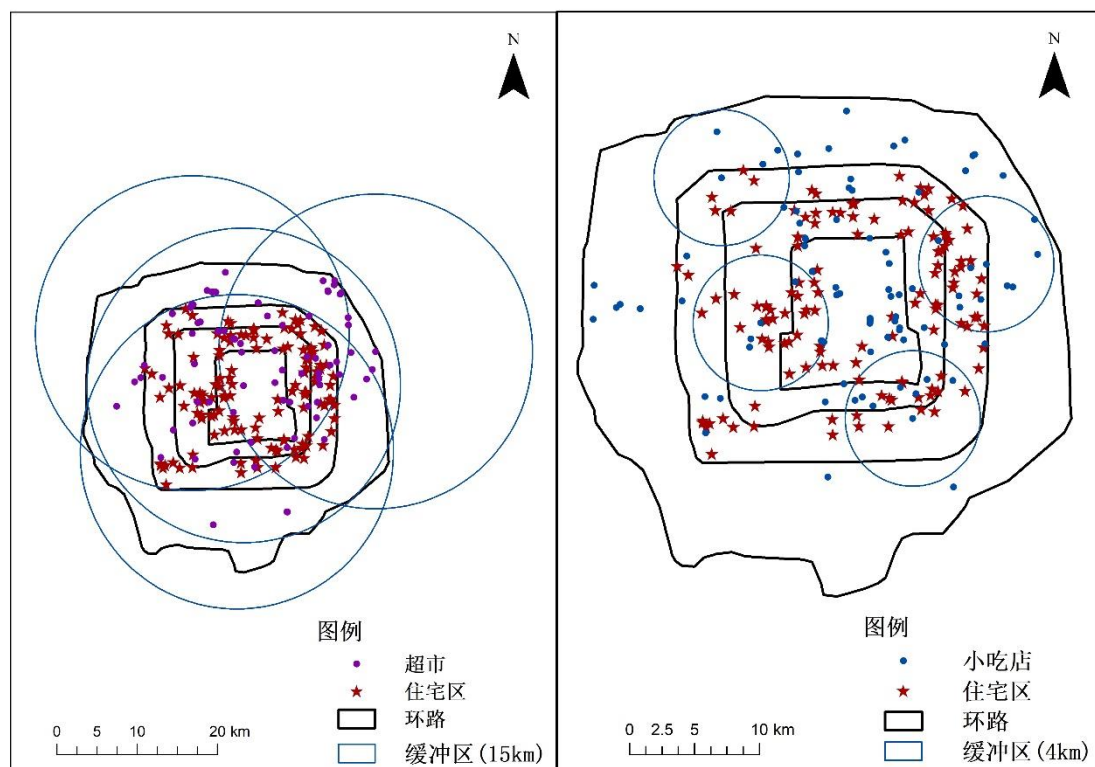


图 3-5-2 不同类别商家配送区域(以北京为例)

注：为保证视觉效果，图中仅随机挑选了 4 个地点进行了可视化制图。

四、模型的评价与优化

4.1 模型的优点

在模型建立的过程中，充分考虑到了利用地理空间数据，使用了 python 爬虫、matlab 等软件进行数据处理，精确把握地理空间信息。同时考虑到了不同天气情况、早晚高峰的路况信息，并且权衡多方利益相关者的利益。除此之外，模型较为简单，方便理解和计算，对于一些简单问题可以快速学习并应用。

4.2 模型的不足

建立模型时，一些标准较为主观，且进行评估的数据方面较少，导致在送餐时间和服务质量上有些片面和不准确。同时建立的模型中未考虑政府政策（比如物价等）、平台优惠、外卖骑手接单时间、路上突发意外等方面对于本模型的约束与影响，过于简单，并不贴近于现实生活。

4.3 模型的优化

对于上述模型的建立，下一步要充分了解平台运营和商家成本等基础信息，通过对外卖骑手、商家和消费者等大量数据统计（如问卷调查等），以及对政策、物价、天气状况等进行更加充分的了解，以得到更加贴近实际的评估和策略。

参考文献

- [1]时亚星. 我国网络订餐食品安全的监管博弈及其实证分析[D]. 东北财经大学, 2017.
- [2]韩中庚. 数学建模方法及其应用[M]. 北京. 李晓鹏, 刘春萍. 2017. 12. 458-461
- [3]司守奎, 孙兆亮. 数学建模算法与应用[M]. 北京. 周刚, 仲维杰. 2015. 375-380
- [4]盛骤, 谢式千, 潘承毅. 概率论与数理统计[M]. 北京. 李蕊, 朱惠芳. 2008. 6. 43-47
- [5]吕靖烨, 范欣雅. 个人碳账户建设的三方演化博弈分析——以“蚂蚁森林”为例[J]. 技术与创新管理, 2021, 42(02):198-204.
- [6]彭亚楠. 外卖商家选择的影响因素分析[J]. 中国市场, 2021(14):126-128.
- [7]邓智平. “接单游戏”与平台经济中的劳动实践——以外卖骑手为例[J]. 求索, 2021(03):108-117.
- [8]郎唯群. 平台经济的公平与效率——以外卖骑手为例[J]. 社会科学动态, 2021(04):40-48.

附 录

附录 1: Python 爬虫代码:

```

from requests import get # 引入 requests 库
from json import loads, dump
import json, xlwt
from openpyxl import load_workbook

params = {
    "q": "小吃",          # 类别
    "region": "北京",     # 地点
    "output": "json",     # 输出成 json 字符串
    "page_size": "20",    # 单页信息量(最多 20 条/页)
    "page_num": "0",      # 页码(从 0 起)
    "ak": "FcoUg5FxphmmtxflBuNHvqacFMgFWnLv" # 百度 API 的个人 AK
}

url = "http://api.map.baidu.com/place/v2/search" # 接口的地址(url)
reponse = get(url, params=params) # 发起请求
data = loads(reponse.text)["results"]

with open("data.json", 'w') as f: # 存储数据
    dump(data, f)
    print("Already", len(data), "json data")

def read_data(jsonfile): # 定义 json 数据写入函数
    with open(jsonfile, encoding='utf-8') as f:
        data_all = json.load(f)

    addr = "data.xlsx" # 创建 excel 文档
    wb = load_workbook(addr) # 写入该 excel 文档
    ws = wb.create_sheet() # 创建一张新的工作表
    ws.append(['code', 'Name', 'Latitude', 'Longitude', 'Address'])

    for k in data_all: # 开始循环写入 json 数据
        id = k["uid"]
        name = k['name']
        lat = k["location"]["lat"]
        lng = k["location"]["lng"]
        address = k["address"]
        ws.append(
            [id, name, lat, lng, address])

```

```
wb.save(addr)
```

```
read_data('data.json')
```

Matlab 绘图代码:

```
clear
```

```
clc
```

```
x1=0:0.01:2
```

```
x2=2:0.01:4;
```

```
y1=0;
```

```
y2=2*x2-2;
```

```
set(0,'defaultfigurecolor','w')
```

```
plot(x1,y1,'r');hold on;
```

```
grid on;
```

```
plot(x2,y2,'b');hold on;
```

```
grid on;
```

```
clear
```

```
clc
```

```
x=4:0.01:10
```

```
y1=log(9)*exp((x-4)./3)+5;
```

```
set(0,'defaultfigurecolor','w')
```

```
plot(x,y1,'r');hold on;
```

```
grid on;
```

附录 2：餐饮地点到住宅区距离（完整见附件 2）

以下展示部分住宅区到餐饮地点的距离，全部数据请查阅“附件 2.xls”，对于餐饮地点的全部编号请查阅“餐饮地点爬虫数据.xls”，对于住宅区的全部编号请查阅“住宅区爬虫数据.xls”。

序号	住宅区编号	餐饮地点编号	距离(单位: Km)
1	0	10	3.322
2	0	6	1.201
3	0	4	3.343
4	0	19	3.088

5	0	17	3.019
6	1	3	2.103
7	1	13	1.804
8	1	11	3.171
9	1	2	2.283
10	1	24	2.810
11	1	18	2.869
12	1	15	2.381
13	1	14	2.381
14	2	10	1.267
15	2	6	1.943
16	2	4	1.765
17	2	19	2.555
18	2	17	1.776
19	3	10	2.794
20	3	6	0.318
21	3	30	2.853
22	3	19	1.613
23	3	17	1.805
24	4	13	2.682
25	4	11	1.445
26	4	2	3.457
27	4	7	2.515
28	4	25	2.682
29	4	24	2.495
30	4	21	2.175
31	4	18	1.583
32	4	15	1.410
33	4	14	1.410

34	4	23	3.529
35	4	22	3.123
36	4	16	2.795
37	5	11	1.525
38	5	12	2.650
39	5	9	2.067
40	5	8	1.724
41	5	7	0.562
42	5	25	2.562
43	5	21	2.357
44	5	18	2.390
45	5	15	2.701
46	5	14	2.701
47	5	26	3.236
48	5	23	2.232
49	5	22	2.073
50	5	20	2.577
51	5	16	1.499
52	6	10	1.226
53	6	6	3.365
54	6	4	0.587
55	6	17	2.760
56	7	10	2.506
57	7	6	0.583
58	7	4	3.418
59	7	30	2.697
60	7	19	1.447
61	7	17	1.514
62	8	10	2.937

63	8	6	0.584
64	8	30	2.598
65	8	19	1.382
66	8	17	1.763
67	9	10	2.924
68	9	6	0.679
69	9	30	2.500
70	9	19	1.287
71	9	17	1.702
72	10	3	2.618
73	10	13	1.032
74	10	11	2.904
75	10	2	3.091
76	10	35	3.393
77	10	24	2.050
78	10	21	3.133
79	10	18	2.402
80	10	15	1.864
81	10	14	1.864
82	11	10	1.538
83	11	6	1.436
84	11	4	2.593
85	11	30	2.820
86	11	19	1.704
87	11	17	1.071
88	12	13	2.305
89	12	11	1.214
90	12	7	2.661
91	12	37	3.498

92	12	36	3.498
93	12	35	3.322
94	12	25	2.250
95	12	24	1.895
96	12	21	1.721
97	12	18	1.045
98	12	15	0.798
99	12	14	0.798
100	12	23	3.352
101	12	22	2.902
102	12	16	2.681
103	13	12	0.947
104	13	9	1.213
105	13	8	1.556
106	13	7	2.764
107	13	29	1.763
108	13	26	1.922
109	13	23	2.690
110	13	22	3.121
111	13	20	1.587
112	13	16	2.909
113	13	34	3.342
114	13	31	2.806
115	13	27	2.073
116	14	11	3.346
117	14	12	0.344
118	14	9	0.492
119	14	8	0.767