

“拍照赚钱”的任务定价

摘要

随着劳务众包平台的兴起, 如何对任务进行合理定价成为平台的重要工作。本文依据题目所给数据研究现有定价规律, 并建立模型进行优化。

针对问题一, 本文首先将任务点全部在地图上标出, 发现任务集中在佛山, 广州, 深圳, 东莞四个城市, 对四个城市内会员分布进行聚类分析, 得到三个研究区域。计算区域中心点, 用于衡量任务的地理位置优劣。定义任务偏僻程度、会员密度、任务密度 3 个影响定量的变量, 建立定价的函数表达式, 用最小二乘法计算各变量的系数, 从而得到任务定价规律(见式(8))。根据已完成任务数据重新计算系数, 将未完成任务数据代入新的定价公式中, 发现有 89% 的未完成任务实际定价低于理论定价, 因此价格偏低是任务未被完成的主要原因。

针对问题二, 以提高任务完成率和控制定价成本为目标修改问题一的模型。本文利用了附件二中会员信誉值的数据这一反映其活跃程度的重要因素。考虑在某一小区域内对任务与会员相互的吸引竞争关系进行研究, 引入贝叶斯-纳什均衡理论, 建立新的定价方案(见式(16))。通过计算, 任务定价上涨 12%, 任务完成情况上涨 23%。新方案在采用较少提价的前提下可以使任务完成率有较大提高。

针对问题三, 考虑以任务点为圆心, 1.5km 为半径的圆域, 通过计算, 认为当一个任务点存在周围超过 2 个其他任务点时任务点密集, 需要打包。建立打包对象的挑选公式(见式(17)), 符合要求的每 3 个任务点将进行打包, 遍历所有任务点得到了打包点的分布图。对打包点的价格进行调整, 计算打包后任务完成率, 与问题二模型的计算结果相比提高 3.7%。

针对问题四, 首先将附件三中数据标注在地图上并对其进行聚类分析, 划分出新的区域并找出新的中心点, 利用之前建立的定价模型计算出相应的定价公式(见式(18)), 代入附件三数据计算任务定价及完成率, 平均任务定价为 69.4 元, 预计任务完成率为 93.7%。

本文借助谷歌地图进行数据分析和筛选, 提高了模型的可靠性。

【关键字】 多元回归 K-means 聚类 贝叶斯-纳什平衡



一、问题重述

1.1 问题背景

在大数据时代，传统的产品铺货率调查弊端越来越显著，“拍照赚钱”这种基于移动互联网的自助式劳务众包平台应运而生。它是基于智能手机和移动互联网下的一种自助式服务模式。用户下载相应的应用软件，注册会员，领取需要拍照的任务并赚取相应酬金。这种方式为企业提供各种商业检查和信息搜集，且助力于 o2o 大潮中的数据采集和产品推广，相比传统的市场调查方式可以大大节省调查成本，通过 GPS、系统云时间、图片等手段来控制数据质量，有效地保证了调查数据真实性，缩短了调查的周期。因此 APP 成为该平台运行的核心，而 APP 中的任务定价又是其核心要素。如果定价不合理，有的任务就会无人问津，而导致商品检查的失败。

1.2 问题提出

为了合理地进行“拍照赚钱”的任务定价，提出以下任务：

1. 研究附件一中项目的任务定价规律，分析任务未完成的原因。
2. 为附件一中的项目设计新的任务定价方案，并和原方案进行比较。
3. 实际情况下，多个任务可能因为位置比较集中，导致用户会争相选择，一种考虑是将这些任务联合在一起打包发布。在这种考虑下，如何修改前面的定价模型，对最终的任务完成情况有什么影响？
4. 对附件三中的新项目给出一个任务定价方案，并评价该方案的实施效果。

二、问题分析

2.1 问题一的分析

问题一需要根据附件一中的数据研究项目的任务定价规律，首先要找到任务定价的影响因素，分析附件一中所给数据，将会员信息和任务信息分别标注在地图上。根据会员分布划分区域，计算区域中心点，用于衡量任务的地理位置优劣。定义偏僻程度、会员密度、任务密度 3 个变量，建立定价的函数表达式，用最小二乘法计算各变量的系数，从而得到附件一的任务定价规律。比较未完成任务的



实际定价和根据已完成任务标准确定的定价，可以发现未完成任务的劣势，得到任务未完成原因。

2.2 问题二的分析

问题二是对第一问模型进行的修改。第一问中商家给出的定价策略并不十分合理，导致相当一部分任务没有被完成，同时某些任务定价过高成本过大。问题二中利用了附件二中会员信誉值的数据，考虑在某一小区域内对任务与会员相互的吸引竞争关系进行研究，引入贝叶斯-纳什均衡理论，使平台与用户在最大程度上达到共赢的效果，并通过计算与原来的模型进行比较。

2.3 问题三的分析

问题三是建立在第二问模型的基础上的。问题三要求考虑多个任务集中分布的问题建立相应的打包策略。首先仍考虑以任务点为圆心，1.5Km 半径的圆域，当一个任务点周围超过 2 个其他任务点时认为任务点密集，需要打包。建立打包对象的挑选公式，符合要求的任务点将进行打包，遍历所有任务点可得到打包点的分布。对打包点的价格进行调整，最终评估区域内的任务完成情况。

2.4 问题四的分析

问题四将前三问建立的模型进行系统化阐述，首先将附件三中数据标注在地图上并对其进行聚类分析，划分出新的区域并找出新的中心点，通过代入之前建立的定价模型计算出相应的定价方案，并计算任务完成率，评价该方案的实施效果。

三、基本假设

1. 假设所给任务难易程度均相同且都可实现，不考虑 APP 中任务描述、参考资料等对用户是否接受任务的影响。
2. 忽略天气、交通状况等不确定因素对任务定价的影响。
3. 认为附件中所有数据均真实可靠。

四、符号说明

变量	符号
区域	Z
任务点	P
定价	ω
权系数	C
底价	a_0
城市中心点	O
人数	n
任务书	m
分项价格	z
信誉分值	G
期望价格	V_H
最低价格	V_L

五、模型的建立与求解

5.1 问题一

5.1.1 定价影响因素

劳务众包平台“拍拍赚”与本题有相同的应用背景，从官方网站^[1]上我们了解到，其任务的定价是算法驱动的，主要考虑了任务的复杂度、时间长短、门店的位置，以及会员分布情况等因素。根据我们的基本假设，附件一所给所有任务的复杂程度相同，因此，任务位置和会员分布是影响任务定价的两个主要因素。

5.1.2 数据分析处理

将附件一任务坐标导入谷歌地图中进行标注，可以发现任务集中在广东省的

根据会员分布划分 3 个区域：蓝色区域（Z1）主要覆盖深圳市；红色区域（Z2）主要覆盖广州市和佛山市；绿色区域（Z3）主要覆盖东莞市。将三个类的质点作为三个区域中心点 O1,O2,O3.三个中心点的地理位置见下表 1。

表 1 中心点地理位置

	O1	O2	O3
经度	113.2757	114.0689	113.8294
纬度	23.1235	22.6414	22.9159

研究划分的 3 个区域，可以对任务分布数据做进一步分析。

记任务点为 P_i ，任务点所属区域 $Z_j(j=1,2,3)$.任务点距中心点的距离可以表示任务的地理位置，该距离越大，任务的地理位置也就越差，任务标价较高。

以任务点 P_i 为中心，1.5km 为半径做圆，该范围内覆盖的会员数为 n ,其他任务数为 m 。当一个任务周围的会员密度越大，任务定价越低。

5.1.3 问题一模型的建立

认为任务定价 w 由 3 部分组成：

$$w = a_0 + a + \varepsilon \quad (1)$$

a_0 为固定底价，是任务的最低价格，取附件一中的 65。 a 表示赏金，是浮动价格，受任务位置分布和会员分布的影响而变化。 ε 为受不确定因素影响（如天气、交通管制等）的价格浮动。由于数据有限认为 ε 影响很小，在此暂不讨论。

记 z_1 为任务偏僻程度。

$$z_1 = \frac{|P_i O_j|}{|P_m O_j|} = \frac{\sqrt{(x_i^2 - x_j^2) + (y_i^2 - y_j^2)}}{\sqrt{(x_m^2 - x_j^2) + (y_m^2 - y_j^2)}} \quad (2)$$

O_j 是任务点所属区域 Z_j 的中心点， P_m 是离中心点最远的任务点。即离中心点越远越偏僻。

由于任务分布集中在广东，距离较近，故近似认为任务点和会员点分布在二维平面上，以他们的经纬度作为坐标 (x_i, y_i) 。

记 z_2 为会员密度。

$$z_2 = \frac{n_m - n_i}{n_m} \quad (3)$$

n_i 为任务点 1.5km 圆域内的会员数量, n_m 是 n_i 的最大值。

记 z_3 为任务密度。

$$z_3 = \frac{m_m - m_i}{m_m} \quad (4)$$

m_i 为任务点 1.5km 圆域内的其他任务点数量, m_m 是 m_i 的最大值。 z_3 的提出是对会员密度的矫正, 考虑到一个任务点的圆域内可能存在其他任务点, 导致会员密度的相对下降。认为 m 越大, 定价越高。

$$a = C_1 z_1 + C_2 z_2 + C_3 z_3 \quad (5)$$

将(5)代入(1), 忽略 ε 即可得到任务定价公式

$$w = C_1 z_1 + C_2 z_2 + C_3 z_3 + 65 \quad (6)$$

表 2 回归方程变量含义

C_1	任务偏僻程度的系数, 预期为+
C_2	会员密度的系数, 预期为-
C_3	任务密度的系数, 预期为+

从式(6)可以看出, 任务定价由固定底价和浮动价格组成, 最终定价主要受三个变量的数值影响, 即任务偏僻程度、会员密度、任务密度三个变量。

5.1.4 问题一模型的求解

● 任务定价规律

首先将附件一中全部任务点的坐标数据代入(2)可以计算一系列对应的 z_1 。

然后根据经纬度与地表距离的换算关系, 利用 MATLAB 统计每个任务点 1.5km 圆域内的会员数量和其他任务点数量, 分别代入(3)(4)进行计算, 得到 $z_2 z_3$ 。

现在我们已知每个任务点的定价 w_0 和 $z_1 z_2 z_3$ 的具体数值, 用最小二乘法拟合的方法计算(6)中的系数 $C_1 C_2 C_3$ 。计算结果如表 3 所示。

表 3 系数计算结果

C_1	C_2	C_3	σ^2
8.2233	-0.1959	3.7487	0.78

根据方差的计算结果可看出，系数的计算结果比较可靠。

给出附件一中的任务定价规律如下：

$$w = 8.2233z_1 - 0.1959z_2 + 3.7487z_3 + 65 \quad (7)$$

这也符合之前定价规律的探究，即偏僻程度、任务密度与定价成正相关，会员密度与定价成负相关。且偏僻程度对定价产生主要影响作用。

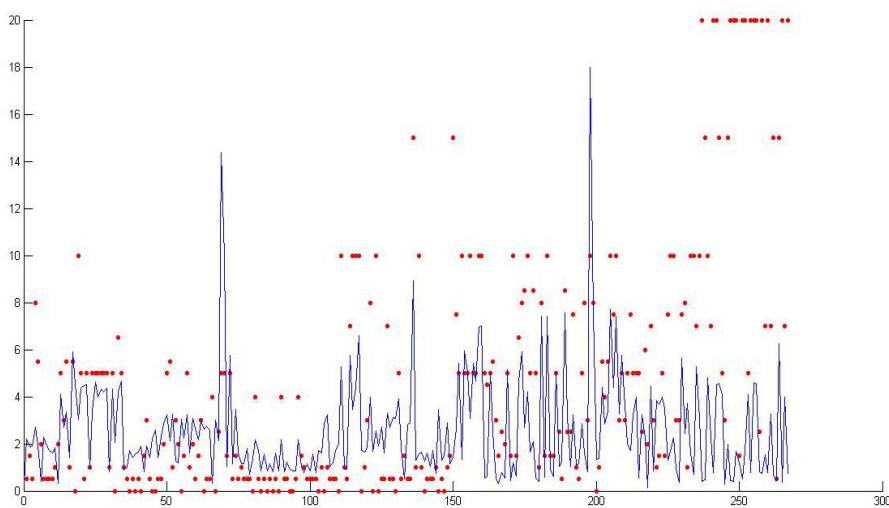


图 3 定价模型拟合与实际定价对比图

图 3 表示了(7)的拟合情况，从杂乱的任务点定价散点中找出最贴合实际的一般规律。

● 任务未完成原因

观察图 1 可知，Z3 的任务基本都已完成，而 Z1 任务基本都未完成，这两个区域的任务完成情况比较特殊，故主要研究 Z3 区域。

首先研究 Z3 中已完成任务的特点。根据 Z3 中已完成任务点的数据计算相应的 $z_1 z_2 z_3$ ，用最小二乘法拟合的方法重新计算(6)中的系数，得到新的定价规律：

$$w = 8.9122z_1 - 0.1589z_2 + 2.9427z_3 + 65 \quad (8)$$

根据未完成任务点的数据计算 $z_1' z_2' z_3'$ ，代入上述式(8)中计算 w ，然后计算每个未完成任务点的计算定价与实际的差值 $w - w_0$ 。认为当这一差值大于 0 时，

定价偏低，小于 0 时定价偏高。

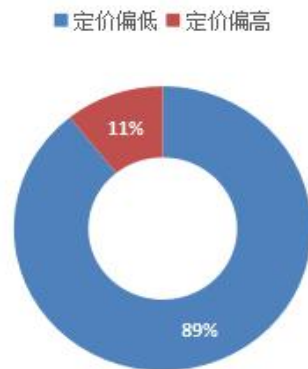


图 4 未完成任务的定价评估状况

我们发现 89%的未完成任务实际定价均小于计算定价，相比于已完成的任务点，他们的定价偏低。

因此我们得出如下结论：

- 1.任务未完成的主要原因是定价偏低，使会员积极性下降。
- 2.存在地域特殊性，如深圳地区未完成任务情况普遍。
- 3.可能存在不确定因素的影响，如天气不好、道路施工、交通堵塞、会员个人因素等，但不在本文的研究范围之内。

5.2 问题二

5.2.1 问题二模型的建立

问题一中的模型主要利用了附件二中会员的位置信息，没有考虑到会员信誉值的影响，在问题二中将主要讨论该因素与定价的关系。

基于第一问的数学模型。同时考虑到会员的信誉值反映会员的活跃程度，建立新的模型。

对会员活跃的程度数据进行打分处理如下表 3：

表 4 会员打分情况

信誉值	大于 10000	1000~10000	100~1000
打分	1	0.8	0.6
信誉值	10~100	1~10	小于 1
打分	0.4	0.2	0.8

记 z_4 为区域内会员信誉分数密度。

$$z_4 = \frac{1}{n} \sum_{i=1}^n G_i$$

n 为圆域内会员总数， G_i 为第 i 名会员的信誉分数。

基于第一问中的模型，加入新的影响变量得到如下表达式：

$$w = C_1 z_1 + C_2 z_2 + C_3 z_3 + C_4 z_4 + 65 \quad (9)$$

同样使用最小二乘法求系数，得到如下结果：

表 5 系数计算结果

C_1	C_2	C_3	C_4
6.3248	-0.1395	2.3974	1.875

即有

$$w = 6.3248 z_1 - 0.1395 z_2 + 2.3947 z_3 + 1.875 z_4 + 65 \quad (10)$$

某一区域内任务的定价反映此任务对会员的吸引程度。在某一小区域内，若某一任务的定价略高于其他任务的定价，则其对会员具有较大的吸引力。同时此区域内信誉值较高的会员更倾向于接受任务。

这种设计定价方案的问题类似于博弈问题。可以借助相关经济学理论进行研究^{[2][3]}。

首先利用式(9)代入附件一中全部完成任务的数据计算出的价格，认定为会员的期望价格，设为 V_H ，某一区域内最低定价为会员可能接受的最低价格，设为 V_L 。则最优定价应在二者之间。

假定某一任务平台的定价为 $M_L(V_L)$ ，即当平台认定商品的价格为 V_L 时，平台给出的定价为 $M_L(V_L)$ ；能使顾客接受的最低定价为 $M_H(V_H)$ ，即此任务通过模型一中全部实现数据给出的定价为 V_H 时，能使顾客接受的最低定价为 $M_H(V_H)$ 。

该交易成交则对平台和会员的效用分别为 $V_H - M_L(V_L)$ 及 $M_L(V_L) - V_L$ ，如果

不成交则双方的效用均为 0。平台和用户都希望得到最大化的期望效用。

对于任意给定的 $V_L \in [0,20]$ ，平台的报价应使其期望利润最大化。因为只 $M_L(V_L) > M_S(V_S)$ 时才能成交，完成后平台的利润为 $(M_L(V_L) + M_S(V_S))/2 - V_H$ ，而未完成是利润为 0，所以 $M_L(V_L)$ 应满足

$$\max_{M_L} \left\{ \frac{M_L + E[M_S(V_S) | M_S(V_S) \geq M_L]}{2} - V_L \right\} P\{M_S(V_S) \geq M_L\} \quad (11)$$

这里 $E[\]$ 表示的是条件 $M_S(V_S) \geq M_L$ 下 $M_S(V_S)$ 的条件期望， $P\{\}$ 表示事件的概率。

类似的，对于任意给定的 $V_H \in [0,20]$ ，会员的最低接受价格 $M_H(V_H)$ 应该使其期望获利最大，完成后会员得到的利益为 $V_H - (M_L(V_L) + M_S(V_S))/2$ ，未完成时获利为 0，所以 $M_H(V_H)$ 应满足

$$\max_{M_H} \left\{ V_H - \frac{M_S + E[M_L(V_L) | M_L(V_L) \geq M_S]}{2} \right\} P\{M_S(V_S) \geq M_L\} \quad (12)$$

如果平台和会员组合 $(M_S(V_S), M_L(V_L))$ 同时满足(9)和(10)，则是双方的一个均衡。

博弈问题存在许多的均衡，本文采取线性价格均衡法，假定平台和会员的报价分别是任务对二者价值的线性函数，表示为：

$$M_S(V_S) = A_S + C_S V_S$$

$$M_L(V_L) = A_L + C_L V_L$$

寻找 A_S ， C_S ， A_L ， C_L 使得这个组合同时满足(9)和(10)，即构成一个均衡。通过文献^[4]求得最优解

$$M_S(V_S) = \frac{2}{3} V_S + \frac{1}{4} \quad (13)$$

$$M_S(V_S) = \frac{2}{3} V_S + \frac{1}{12} \quad (14)$$

此时任务完成概率 η 为

$$\eta = \frac{\int_0^1 \int_0^{V_H - \frac{1}{4}} (V_H - V_L) dV_H dV_L}{\int_0^1 \int_0^{V_b} (V_H - V_L) dV_H dV_L} \quad (15)$$

如图 4 所示:

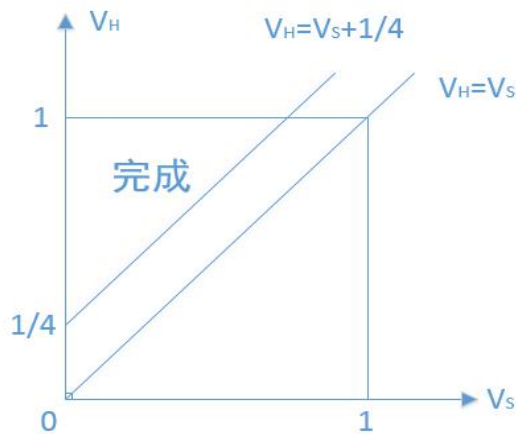


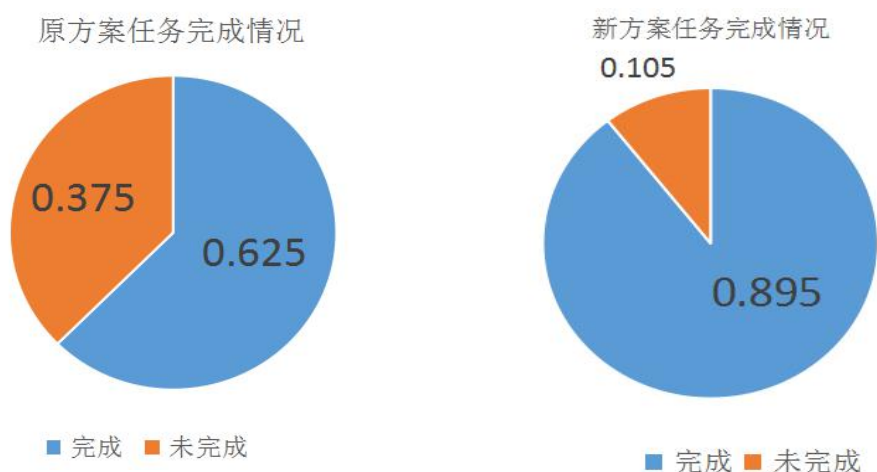
图 5 线性价格战略的交易效率

设新任务定价:

$$\omega_1 = M_H(w) = \frac{2}{3}w + \frac{1}{4} \quad (16)$$

该式因加入博弈论的经济学思想使模型更加完善。

5.2.2 问题二模型的求解



利用上述模型对任务重新进行定价, 并代入公式(15)得到结果如下图 6

图 6 新旧方案任务完成情况对比图

以任务完成概率进行估计，新方案预期任务完成率为 89.5%，相比与原方案有较大改善。

● 新方案与原方案的比较

通过计算，新方案总价为 64631 元，原方案总价为 57707 元。任务定价上涨 12%，任务完成情况上涨 23%。由此可见，新方案在采用较少提价的前提下可以使任务完成率有较大提高，从而减小商品检查失败带来的损失，使利润大大增加，与旧方案相比有较大改进。

5.3 问题三

5.3.1 问题三模型的建立

● 打包策略

在问题一中，我们定义 m_i 为任务点 1.5km 圆域内的其他任务点数量，通过计算附件一各任务点的 m_i 数值，得到 $\bar{m} = 2.2$ ，即在一个任务点周围平均存在 2 个其他任务点，于是认为以 1.5km 为半径的圆域内平均存在 3 个任务。

因此给出如下打包策略：

1. 首先选取一个任务点，若 $m_i \geq 2$ 则该点符合打包条件，该任务点将与其他两个任务点进行打包，否则按独立任务点计算。其他两个任务点的挑选原则：选取的 3 个任务点的平均任务完成概率大于给定指标即进行打包。已打包的任务点不再参与其他打包。
2. 选取下一个任务点，忽略下一个任务点圆域内已打包点，进行 1 中的打包判断流程。
3. 当全部任务点判断、打包完毕后进行重新定价。独立任务点定价不变，打包任务点定价进行适当调整。

● 打包算法

对任务点进行打包的算法流程图如下图 7 所示：

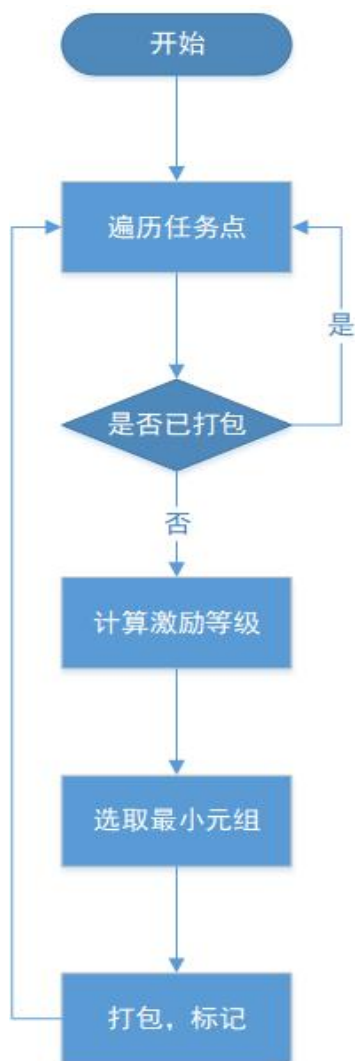


图 7 打包任务点的算法流程图

圆域内打包任务点的挑选公式如下：

$$Act = a_1(\omega_i + \omega'_i) + a_2(\eta_i + \eta'_i) + a_3d - k \quad (17)$$

式中 a_1 为定价的系数， a_2 为任务完成概率的系数， a_3 为任务点距区域中心点距离的加权系数， k 为修正系数； ω_i, ω'_i 分别代表当前任务点和被挑选任务点的定价， η_i, η'_i 分别代表当前任务点和被挑选任务点的任务完成概率， d 为两任务点之间的距离。

当 Act 的数值越接近 0 越适合被挑选打包。这样我们就可以确定被打包的任务点。

5.3.2 问题三模型的求解

利用 MATLAB 进行求解，得到下图结果。

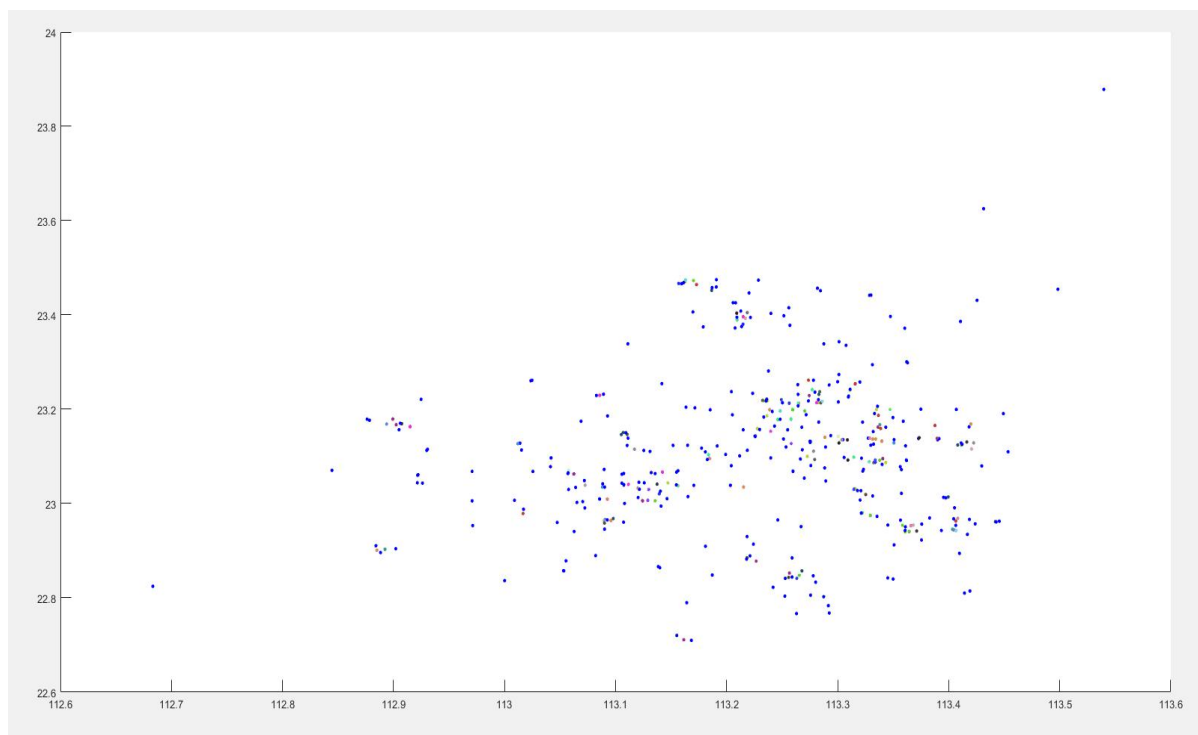


图 8 打包点的分布图

图 8 中，彩色点表示打包点，蓝色点为独立的任务点。

设打包点内的 3 个任务定价平均值为 \bar{w}_i ，设打包点的新定价为 $3\bar{w}_i - 10$ ，这样一个信誉值稍差的用户受任务限额的限制变小，会员可以一次性完成大额任务，会员积极性提高，同时分散会员参与其他项目，理论上任务的完成程度将有进一步提高。

通过计算，得到如下结果：

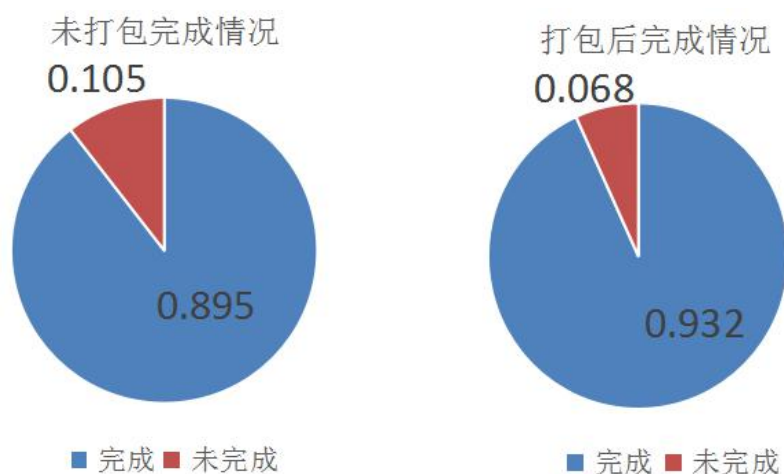


图 9 打包前后任务完成情况对比图

通过对比可以发现，打包后任务完成率可以提高 3.7%，对任务完成情况有

所改善。

5.4 问题四

问题四利用本文建立的模型对新任务进行定价并评价此方案的实施效果。

将附件三中的任务点标记在谷歌地图上进行分析，如图 10。

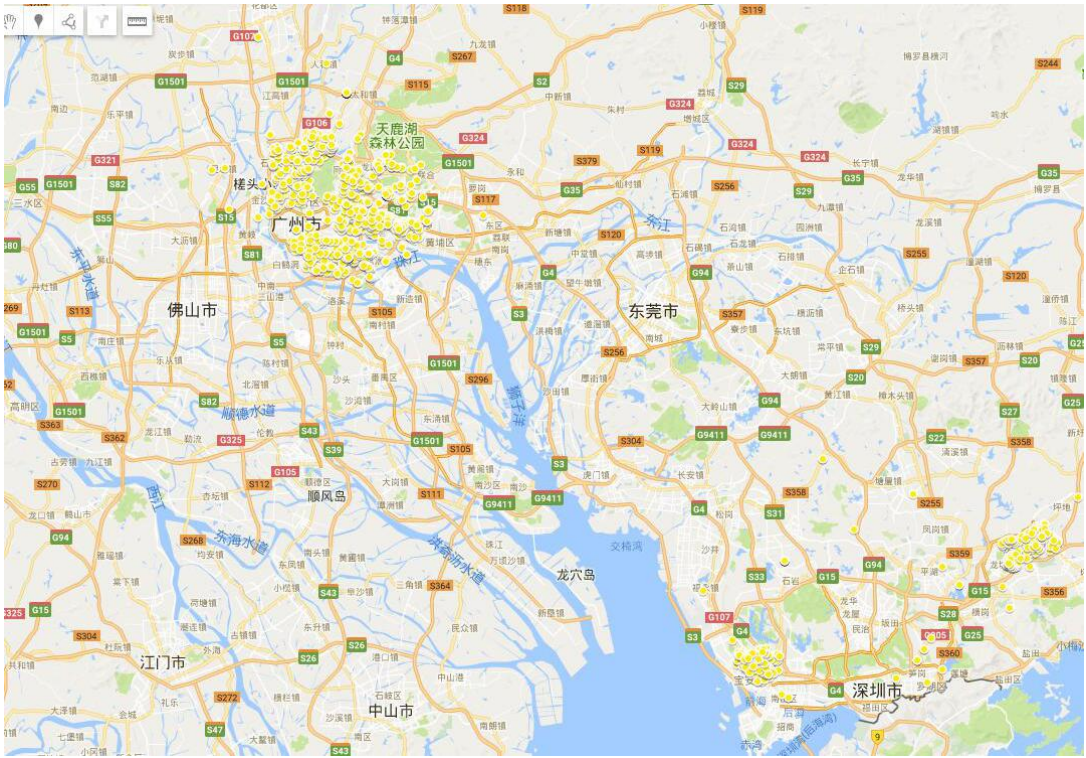


图 10 附件三任务点分布情况

从图中可以看出，任务点主要分布在广州市和深圳市。

对任务点进行聚类，结果如图 11 所示.

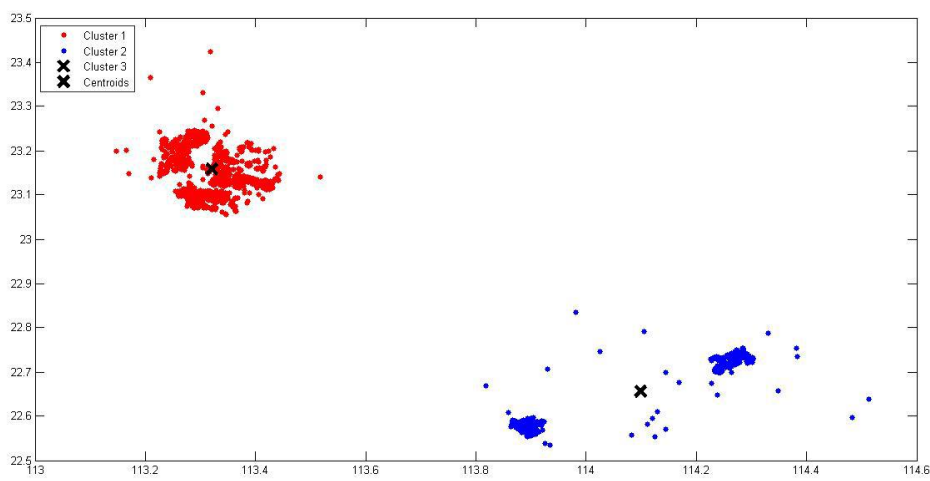


图 11 新任务聚类分析图

通过聚类将附件三数据分为两个区域并确定出两个区域的中心点。

定价方案如下所示：

第一步：计算下列数据

1. 各任务据市中心的距离
2. 各任务以 1.5km 为半径圆内的任务数
3. 各任务以 1.5km 为半径圆内的会员数
4. 各任务以 1.5km 为半径圆内的会员荣誉值

将各数据进行标准化处理得 z_1, z_2, z_3, z_4 。

第二步：

利用下列公式对任务进行初步打分

$$w = 6.3248z_1 - 0.1395z_2 + 2.3947z_3 + 1.875z_4 + 65$$

第三步：

利用均衡理论对小区域内数据进行改进，公式如下：

$$\omega_1 = M_H(V_H) = \frac{2}{3}V_H + \frac{1}{4} \quad (18)$$

第四步：

对符合条件的密集任务点进行打包

具体定价结果见附件一，平均任务定价为 69.4 元，预计任务完成率为 93.7%。

六、模型的评价与推广

6.1 模型优点

本文的模型是在对大量数据进行认真分析和处理后建立的，通过将数据点显示在谷歌地图可以方便直观地观察分析，帮助排除特殊情况，寻找普遍适用规律，使模型建立的数据信息更加可靠，更加贴近实际。

通过对模型的层层优化，使模型适应更加复杂的实际情况，模型简洁实用，可移植性强。

问题二中利用会员期望与任务定价之间的影响，利用贝叶斯-纳什平衡理论重新定价。使得新定价与原定价相比有更好的效果。

6.2 模型缺点

本文使用附件中的数据，数据来源有限，模型准确度无法进行进一步的评估，模型可能具有一定的局限性。

6.3 模型推广

本文模型的应用背景是基于智能手机和移动互联网的劳务众包平台，在相似背景下的应用众多，如外卖应用，滴滴打车，快递跑腿服务平台等都涉及到商品定价与地理位置、会员积极程度的关系。

七、参考文献

- [1]拍拍赚科技. 关于建模竞赛的问与答[EB/OL]. <http://web.ppznet.com/>, 2017. 9. 15/2017. 9. 16.
- [2]徐晨. 论移动商务在企业信息化中的应用[J]. 情报科学, 2006, 24(1):144-147.
- [3]曾庆群, 移动商务竞价行为的精炼贝叶斯纳什均衡研究[J]. 武汉理工大学学报·信息与管理工程版, 2009, 31(4): 618-620.
- [4] 姜启源, 谢金星, 叶俊. 数学模型(第四版)[M]. 北京: 高等教育出版社, 2011.

附录

一、 自制地图

地址:

<https://drive.google.com/open?id=1PtKPImNp-LqKmARQzBH-VSw7RG4&usp=sharing>

二、代码

1. Kmeanstest.m

%获取经纬度

%[number, txt, raw]=xlsread('test.xls');

X = [GPS1(:,1) GPS(:,1)];

opts = statset('Display','final');

%调用 Kmeans 函数

%X N*P 的数据矩阵

%Idx N*1 的向量, 存储的是每个点的聚类标号

%Ctrs K*P 的矩阵, 存储的是 K 个聚类质心位置

%SumD 1*K 的和向量, 存储的是类间所有点与该类质心点距离之和

%D N*K 的矩阵, 存储的是每个点与所有质心的距离;

[Idx,Ctrs,SumD,D] = kmeans(X,2,'Replicates',2,'Options',opts);

%画出聚类为 1 的点。X(Idx==1,1), 为第一类的样本的第一个坐标; X(Idx==1,2) 为第二类的样本的第二个坐标

plot(X(Idx==1,1),X(Idx==1,2),'r.','MarkerSize',14)

hold on

plot(X(Idx==2,1),X(Idx==2,2),'b.','MarkerSize',14)

%hold on

%plot(X(Idx==3,1),X(Idx==3,2),'g.','MarkerSize',14)

%绘出聚类中心点,kx 表示是圆形

plot(Ctrs(:,1),Ctrs(:,2),'kx','MarkerSize',14,'LineWidth',4)

plot(Ctrs(:,1),Ctrs(:,2),'kx','MarkerSize',14,'LineWidth',4)

%plot(Ctrs(:,1),Ctrs(:,2),'kx','MarkerSize',14,'LineWidth',4)

legend('Cluster 1','Cluster 2','Cluster 3','Centroids','Location','NW')

Ctrs

SumD

```

2. makepackage.m
%打包脚本-
con(1:443,1)= 0;
distan(1:443,1:443) = -1;
a1 = 0.4;
a2 = 0.3;
a3 = 0.3;
k1 = 0.5 ;
temp =[zeros(1,443);zeros(1,443)];
%a1,a2,a3,k 分别为价格、完成情况、距离的加权系数与阈值
%con 为打包记录
%distan 为距离矩阵
for i = 1:443
    count = 0;
    list = 0;
    if con(i)
        continue;
    else
        for j =1:443
            if con(j) == 0
                %计算距离
                if distan(i,j) == -1

distan(i,j)=DF(message(i,1),message(i,2),message(j,1),message(j,2));
                end
                %判断距离，并给出在圈内的列表
                if 0<distan(i,j) && distan(i,j)< 1.5*0.009 && j~=i
                    count = count +1;
                    list = [list ;j];
                end
            if count>= 2
                %计算激励
                activ
a1.*(message(list(2:count+1,1),3)+message(i,3))./65+a2.*(message(list
(2:count+1,1),4)+message(i,4))+a3.*distan(i,list(2:count+1,1)))'-k1;
                for k = 1:2
                    current = 1;
                    for l = 1:count-1
                        if activ(current)>activ(l+1)
                            current = l+1;
                        end
                    end
                    temp(k,i) = current;
                    if temp(k,i) == 0

```



```

[number0,txt0,row0]=xlsread('0.xlsx');
scatter(number1(:,2),number1(:,1),10,'r','field')
hold on
scatter(number0(:,2),number0(:,1),10,'b','field')
legend('已完成任务','未完成任务');
xlabel('经度')
ylabel('纬度')

```

4. picture.m

```

for i=1:838
    if VarName8 == 1
        scatter3(Hnumber,distance1,price,10,'r','field');
    elseif VarName8 == 2
        scatter3(Hnumber,distance2,price,10,'r','field');
    else
        scatter3(Hnumber,distance3,price,10,'r','field');
    end
end
end

```

5. test.m

```

[number,txt,row]=xlsread('test.xls');
%总数
x = [0 0 0 0 0];
%成功数
a = [0 0 0 0 0];

for i = 1:835
    if (number(i,3)>80)
        x(1) = x(1) + 1;
        if (number(i,4) == 1)
            a(1)= a(1) + 1;
        end
    elseif (number(i,3)>75)
        x(2) = x(2) + 1;
        if (number(i,4) == 1)
            a(2)= a(2) + 1;
        end
    elseif (number(i,3)>70)
        x(3) = x(3) + 1;
    end
end

```



```

        if (number(i,4) == 1)
            a(3)= a(3) + 1;
        end

        elseif (number(i,3)>65)
            x(4) = x(4) + 1;
            if (number(i,4) == 1)
                a(4)= a(4) + 1;
            end

        elseif (number(i,3)>=60)
            x(5) = x(5) + 1;
            if (number(i,4) == 1)
                a(5)= a(5) + 1;
            end
        end
    end
end

stem(0:6, [0 a./x 0])
title('成功率与价格关系图')
xlabel('价格 (1 为高)')
ylabel('成功率')
axis([0 6 0 1])

```

```

6. cost1.m
cost = 0;
yb =0;
for i = 1:202
    cost = cost + (y(i)-x(i,:)*b(:,1)).^2;
    yb = [ yb ; x(i,:)*b(:,1)];
end

scatter(1:202, y, 10, 'r', 'field')
hold on
plot(0:202, yb, 'b')

```

```

7. DF.m
function [ D ] = DF( gps11,gps12,gps21,gps22 )
%DF 此处显示有关此函数的摘要
% 此处显示详细说明
D = sqrt((gps11-gps21).^2+(gps12-gps22).^2);

```

end

8. distance.m

```
temp = 0;
for i =1:2066
    count = 0;
    for j=1:1868
        D = (number(i,1) - GPS1(j,1))^2+(number(i,2) - GPS2(j,1))^2;
        if sqrt(D) < 0.009*1.5
            count = count +1;
        end
    end
    temp = [temp ; count];
end
temp2 = 0;
for i =1:2066
    count2 = 0;
    for j=1:2066
        D = (number(i,1) - number(j,1))^2+(number(i,2) - number(j,2))^2;
        if sqrt(D) < 0.009*1.5
            count2 = count2 +1;
        end
    end
    temp2 = [temp2 ; count2];
end
temp2 = temp2 - 1;
```

三、EXCEL 文件表格（部分）

1. 附件一数据及相应处理

任务号码	任务gps纬	任务gps经	任务标价	任务执行情况	任务gps纬	任务gps经
A0001	22.56614	113.9808	66	0	22.55966	113.9748
A0002	22.68621	113.9405	65.5	0	22.67972	113.9345
A0003	22.57651	113.9572	65.5	1	22.57003	113.9512
A0004	22.56484	114.2446	75	0	22.55836	114.2386
A0005	22.55889	113.9507	65.5	0	22.5524	113.9447
A0006	22.559	114.2413	75	0	22.55252	114.2353
A0007	22.549	113.9723	65.5	1	22.54252	113.9662
A0008	22.56277	113.9566	65.5	0	22.55629	113.9506
A0009	22.50001	113.8957	66	0	22.49353	113.8896

2. 附件二数据及相应处理

会员编号	会员位置(C	精度	预订任务开	信誉值	预订任务降	纬度（修
B0001	22.9471	113.68	6:30:00	67997.39	114	22.94061
B0002	22.57779	113.9665	6:30:00	37926.54	163	22.57131
B0003	23.19246	113.3473	6:30:00	27953.04	139	23.18598
B0004	23.25597	113.3188	6:30:00	25085.7	98	23.24948
B0008	23.14337	113.3763	6:42:00	14868.44	95	23.13689
B0009	23.28528	113.6518	6:36:00	13556.16	110	23.2788
B0010	23.09926	113.4889	6:36:00	13327.95	64	23.09278
B0011	23.19246	113.3473	6:42:00	11349.09	89	23.18598
B0012	22.54889	113.9554	6:33:00	10957.58	102	22.54241

3 附件三数据及相应处理

任务号码	任务GPS纬度	任务GPS经度	标记	到1的距离	到2的距离	会员
C0001	22.730041	114.24088	2	1.0303877	0.0257266	0
C0002	22.727043	114.29962	2	1.1445217	0.0454642	5
C0003	22.701311	114.2336	2	1.042494	0.0202841	9
C0004	22.732359	114.28667	2	1.1147697	0.0412056	6
C0005	22.718391	114.25755	2	1.0714651	0.0291628	5
C0006	22.753925	114.38193	2	1.2899605	0.0898743	1
C0007	22.724042	114.27218	2	1.0941566	0.0347666	7
C0008	22.719378	114.27325	2	1.1002581	0.0345253	6
C0009	22.730283	114.2305	2	1.0111783	0.0229142	0