赛区评阅编号（由赛区组委会填写）：

2017 高教社杯全国大学生数学建模竞赛

承　诺　书

我们仔细阅读了《全国大学生数学建模竞赛章程》和《全国大学生数学建模竞赛参赛规则》(以下简称为“竞赛章程和参赛规则”,可从全国大学生数学建模竞赛网站下载)。

我们完全明白，在竞赛开始后参赛队员不能以任何方式（包括电话、电子邮件、网上咨询等）与队外的任何人（包括指导教师）研究、讨论与赛题有关的问题。

我们知道，抄袭别人的成果是违反竞赛章程和参赛规则的，如果引用别人的成果或资料（包括网上资料），必须按照规定的参考文献的表述方式列出，并在正文引用处予以标注。在网上交流和下载他人的论文是严重违规违纪行为。

我们以中国大学生名誉和诚信郑重承诺，严格遵守竞赛章程和参赛规则，以保证竞赛的公正、公平性。如有违反竞赛章程和参赛规则的行为，我们将受到严肃处理。

我们授权全国大学生数学建模竞赛组委会，可将我们的论文以任何形式进行公开展示（包括进行网上公示，在书籍、期刊和其他媒体进行正式或非正式发表等）。

我们参赛选择的题号（从 A/B/C/D 中选择一项填写）：B

我们的报名参赛队号（12 位数字全国统一编号）：01039013

参赛学校（完整的学校全称，不含院系名）：北京工商大学

参赛队员 (打印并签名) ：1. 刘畅

2. 刘雪儿

3. 李玥桥

指导教师或指导教师组负责人 (打印并签名)：

（指导教师签名意味着对参赛队的行为和论文的真实性负责）

日期: 2017 年 08 月 22 日

(请勿改动此页内容和格式。此承诺书打印签名后作为纸质论文的封面，注意电子版论文中不得出现此页。以上内容请仔细核对，如填写错误，论文可能被取消评奖资格。)

赛区评阅编号（由赛区组委会填写）：

2017 高教社杯全国大学生数学建模竞赛

编 号 专 用 页

赛区评阅记录（可供赛区评阅时使用）：

评阅人						
备注						

送全国评阅统一编号（由赛区组委会填写）：

全国评阅随机编号（由全国组委会填写）：

（请勿改动此页内容和格式。此编号专用页仅供赛区和全国评阅使用，参赛队打印后装订到纸质论文的第二页上。注意电子版论文中不得出现此页。）

“拍照赚钱”的任务定价与任务分配的分析与探讨

摘要

本文主要研究了“拍照赚钱”APP 的任务定价问题，在不同的任务条件下，分别建立了基于密度估计的非线性回归模型，基于密度聚类最优解模型，利用 *MATLAB*、*SPSS*、*WEKA* 等软件，实现了众包分配需求中，对不同任务分配，不同定价程度的问题求解。

针对问题一，利用热力图，选取任务位置分布与会员位置分布两项因素来考虑任务定价规律。对任务定价建立基于密度估计的非线性回归模型。针对问题二，选取任务位置分布、任务完成度、会员位置分布、会员信誉值及会员最大限额来构造数学模型，采用非线性回归方程拟合的方法找出最优模型，确定新的定价方案。将两个定价方案在变量完成度相同的情况下比较任务定价总额，新方案在降低任务总酬金上具有优越性。

针对问题三，建立密度聚类算法进行分析。选任务聚类作为打包策略，在单个聚类中，用加权的方式将权重集中于一点，根据生活实际，对总报价做出一定折扣。依赖成本最低原则，构造目标函数，枚举的方法解出近似最优解。

针对问题四，观察数据可知，其数据分布及其集中，若使用基于网格的非线性回归模型和基于密度的聚类模型难以处理极高密度数据，因而采用 *k-means* 聚类和空间距离权重联合作为定价模型。

关键字：众包定价问题　基于网格化的密度估计　非线性回归　*DBSCAN*　*k-means*

目录

一、 问题重述

1.1 问题背景

“拍照赚钱”是移动互联网下的一种自助式服务模式，通过下载、注册 APP，用户领取需要拍照的任务，赚取其对任务所标定的酬金。这种 APP 成为基于互联网的自助式劳务众包平台的运行核心，而 APP 任务的合理定价又是商品成功核心要素。

1.2 问题提出

（1）研究附件一中已结束项目的任务定价规律，分析任务未完成的原因。

（2）为附件一中已结束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又有什么影响？

（4）对附件三中的新项目给出你的任务定价方案，并评价该方案的实施效果。

二、 问题分析

2.1 问题一分析

为研究项目一中的任务定价规律，我们画出任务位置分布、会员位置分布及价格分布热力图，通过图形直观得出任务定价规律；从任务定价、位置分布与会员的位置分布几方面考虑任务未完成情况的原因，利用核密度估计对数据网格化处理，建立非线性回归模型，量化任务完成度。

2.2 问题二分析

为制定新的定价方案，我们由问题一中分析的任务定价规律及决定完成度的非线性方程，初步建立影响任务定价的数学模型，再加入会员信誉值及抢单限额对模型进行进一步优化。最后根据建立的数学模型得到具体的任务定价方案，与原始方案进行固定单一变量的比对，即可以在同一完成度的情况下比较 APP 所需付给会员的酬金，从而检验新方案的可行性。

2.3 问题三分析

考虑题目的实际情况，采用密度聚类算法进行分析。选取任务聚类作为打包策略，在单个聚类中，采取缩点操作，将所有点的属性权重加成到单点中，并对总报价做出一定折扣。然后依赖成本最低原则，构造目标函数，用枚举的策略解出近似最优解。

2.4 问题四分析

考虑到高密度数据的情况，其不能套用网格化处理和密度聚类，需采用一种更加符合其数据特征的处理方案。在聚类算法中，典型的 k-means 算法能够较好的适应。

三、 模型假设和符号说明

3.1 模型假设

- 会员因素在空间分布上不随时间变化而变化，即认为其在全文建模中不产生变化；
- 对于预定任务开始时间，认为受预定人员的主观因素影响，不对分析造成干扰。

3.2 符号说明

符号	意义
P	任务定价 (元)
Y	任务完成度 (%)
ρ_1	人员密度 (人/km^2)
ρ_2	任务密度 (件/km^2)
r	会员预定任务限额 (件)
s	会员信誉值
f_n	任务完成度 (%)
C_i	第 i 个 (聚类) 任务总价 (元)
ρ_{1i}	第 i 个任务点密度 (件/km^2)
ε	密度聚类半径 (km)
$MinPts$	ε 领域的任务数 (件)

四、 模型的建立与求解

4.1 问题一模型的求解

4.1.1 数据处理

(1) 异常数据处理

我们利用 excel 排序对数据进行分析，筛除掉过于远离数据分布的位置点，避免其对整体分析造成极端影响。

(2) 位置数据处理

对会员位置经纬度数据利用 MATLAB 做出分布概率图，可确定其纬度基本分布在 22-24.5，经度基本分布在 112-115，如图 1 所示：

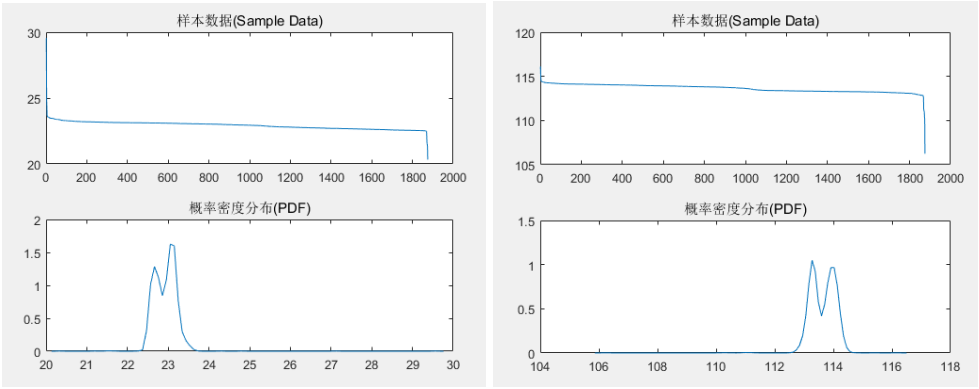


图 1　任务、会员密度图

(3) 网格化数据处理

我们依据地理学上的网格分析法，将任务位置分布的大片区域细化成网格，通过对题中所给数据运用核密度算法：

$$f_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh}\sum_{i=1}^{n} K_h(\frac{x - x_i}{h}) \tag{1}$$

其中，$x_1, x_2, ..., x_n$ 为独立同分布 F 的 n 个样本点，设其概率密度函数为 $f_h(x)$，K_h 为核函数，$h > 0$ 为平滑参数。将数据精简化来提取网格化区域特征，最后再结合实际特征，进行多次回归演算。通过该方法，得到如下核密度空间分布图 (图 2)

4.1.2 非线性多元回归

(1) 附件一中项目任务定价规律

我们先做出任务分布和会员分布的密度图（图 3），通过分析密度图和用 Excel 处理有关任务定价的数据来探究定价规律。

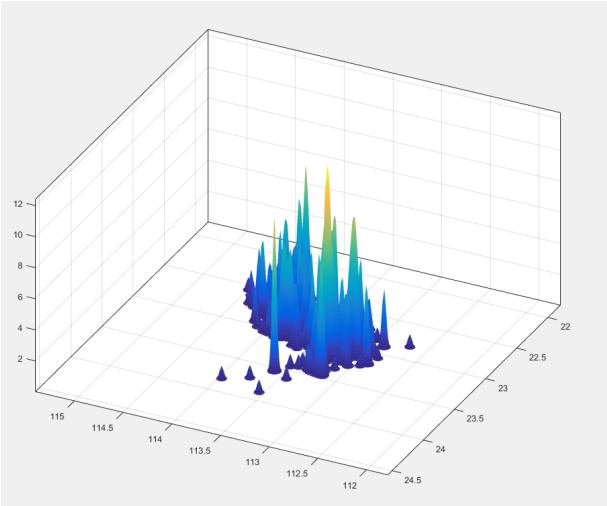


图 2　会员位置纬度、经度概率密度分布图

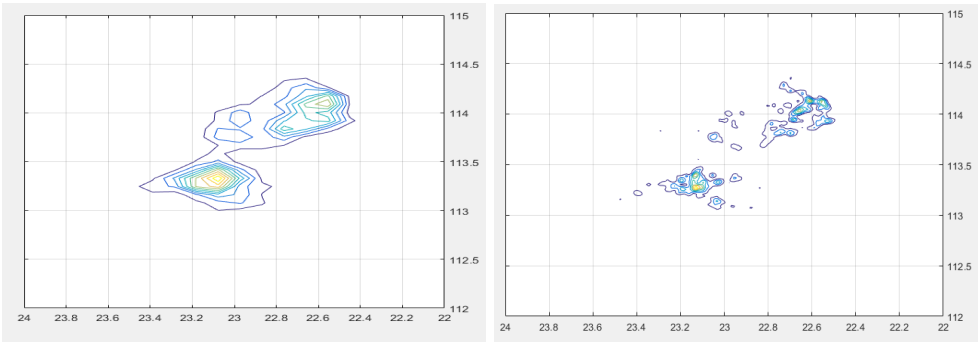


图 3　会员位置纬度、经度概率密度分布图

通过密度图可直观看到，任务位置分布较为分散，但仍有集中的几块区域，最明显的为纬度范围在 $[23, 23.4]$，经度范围在 $[113, 113.5]$，和纬度范围在 $[22.4, 22.8]$，经度在 $[113.8, 114.3]$ 的区域，而会员则更为集中的分布在纬度 $[23, 23.2]$, 经度 $[113, 113.5]$ 这一区域。

利用 *Excel* 对任务定价进行降序处理，并没有发现明显规律，但可看出任务定价在 $[65, 85]$ 区间，再筛选出任务定价低于 70 的任务数据，分析这些任务位置，发现任务定价低的任务位置处于任务集中地和会员集中地共同的那一区域，可见，在这一区域，任务定价偏低。

由上述分析，我们可得附件一项目中的任务定价规律：

- 任务定价主要受任务分布位置和会员分布位置两方面的影响，在任务位置集中分布区，会员在预定任务限额内完成多个任务较为容易，所以，任务定价相较于任务分散区较低；
- 在人员集中分布区，由于用户竞争压力大，完成任务效率高，任务被完成概率大。所以，任务定价会相较于人员疏松区定价低。

(2) 项目任务未完成原因

a) 任务完成情况与任务位置分析

我们利用地图定位，在地图上分类标出已完成、未完成任务点及会员位置点，并对它们之间的位置关系进行粗略分析，如图 4。

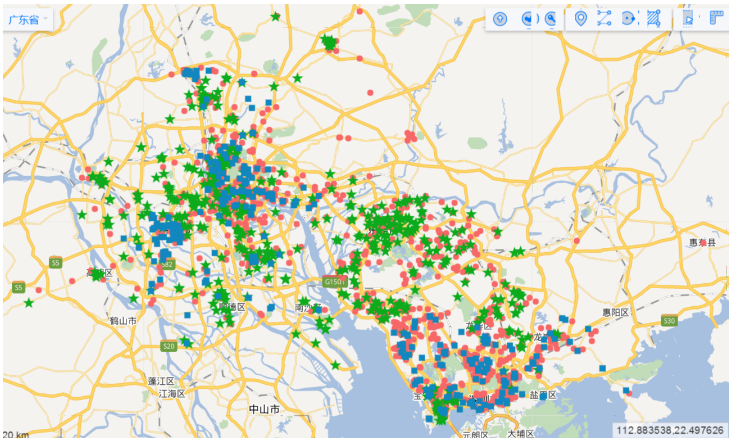


图 4　会员、任务位置分布图

分析地图，可明显看到，已完成任务点分布较为分散，而未完成任务主要集中在以佛山，广州和深圳市为中心的三大区域，而这三个区域会员分布密度相对较大，任务完成情况可能与会员分布密度呈负相关，而这三个地区主要为高新技术产区和商业区的聚集区，所以我们猜测这地区的人们，利用 *SPSS* 对变量进行相关性分析得到表 6，证实了我们的猜想。

表 1　任务完成度与人员密度相关性分析

	B	$S.E.$	$Wald$	df	显著性	$Exp(B)$
人员密度	-0.351	0.137	6.552	1	0.010	0.704
常数	1.541	0.374	17.005	1	0.000	4.670

由表 1 人员密度与任务完成度显著性为 $0.01 < 0.05$，$B = -3.51$，可得任务完成度与人员密度有关联，且呈负相关。

b) 任务完成情况与任务定价分析

由表 2 显著性＝ 0.000 确定两者有关联。而在相同难度的情况下，标价高的任务总是会被优先完成，因此我们认为任务完成情况与任务标价呈正相关，分析如表 3 所示。

c) 建立数学模型

使用上述量化分析后的会员密度、任务密度与任务定价对完成度进行非线性拟合，通过多组模型的对比，得出最理想解，参数评估见表 4 。

表 2　变异数同质性及变异数分析

Lenvene 统计资料	*df1*	*df2*	显著性
11.351	22	812	0.000

	平方和	*df*	平均值平方	*F*	显著性
群组之间	17.300	22	0.786	3580	0.000
在群组之内	178.372	812	0.20		
总计	195.672	834			

表 3　任务完成度与任务定价相关性分析

	B	$S.E.$	$Wald$	df	显著性	$Exp(B)$
平均价格	0.202	0.084	5.747	1	0.017	1.223
常数	-13.420	5.924	5.132	1	0.023	0.000

$$Y = a_0 * (P - 60) + a_1 * \rho_1^2 + a_2 * \rho_2 \tag{2}$$

表 4　完成度的回归参数

参数	估计	标准错误	95% 信赖区间	
			下限	上限
a_0	0.057	0.004	0.048	0.066
a_1	0.026	0.018	0.010	0.063
a_2	-0.002	0.002	0.005	0.002

因原始最低价格为 65，而在满足实际情况下，价格不会下降过多，因此选取价格与 60 的差值作为变量。由此，可得任务完成情况与价格和任务密度成正相关，而与人员密度呈负相关的。因此，我们认为，在会员集中的地区，由于商业区及高新产业的影响，会员反而不会主动去完成任务，导致任务完成度降低。

4.2 问题二模型的求解

4.2.1 模型初步建立

由问题一分析已知：任务定价主要与任务位置和会员位置有关，所以进行相关性分析，根据其相关性强弱，看能否建立线性回归。

表 5　会员密度、任务密度、任务定价的单因素方差分析

		人员密度	任务密度	平均价格
会员密度	皮尔森 (Pearson) 相关	1	0.480	-0.490
	显著性（双尾）		0.000	0.000
	N	74	74	74
任务密度	皮尔森 (Pearson) 相关	0.480	1	-0.444
	显著性（双尾）	0.000		0.000
	N	74	74	74
平均价格	皮尔森 (Pearson) 相关	-0.490	-0.444	1
	显著性（双尾）	0.000	0.000	
	N	74	74	74

分析得到：任务定价与任务密度和人员密度呈负相关，任务数越多越集中，会员分布越密集，则任务定价要相应地降低，因此，我们建立线性回归，结果如表 5：

表 6　任务定价与任务、会员密度的相关性分析

模型	非标准化参数		标准化参数	T	显著性
	B	标准错误	β		
常数	73.463	0.546		134.428	0.000
会员密度	-0.561	0.178	-0.359	-3.162	0.002
任务密度	-0.462	0.193	-0.272	-2.397	0.019

因整体参数的 *sig* 均远小于 0.05，可认为该模型有实际统计意义。我们可初步得到关于价格的回归方程：

$$P = -0.561 * \rho_1 - 0.462 * \rho_2 \tag{3}$$

4.2.2 模型优化

由于会员本身对任务定价也存在影响，因此，将任务定价与会员预定任务限额和信誉值之间进行相关性分析，如表 7：

表 7　任务定价、会员预定限额及信誉值的相关性分析

		任务定价	限额	信誉值
任务定价	皮尔森 (Pearson) 相关	1	-0.033	-0.204
	显著性（双尾）		0.780	0.081
	N	74	74	74
限额	皮尔森 (Pearson) 相关	-0.033	1	0.519
	显著性（双尾）	0.780		0.000
	N	74	74	74
信誉值	皮尔森 (Pearson) 相关	-0.204	0.519	1
	显著性（双尾）	0.081	0.000	
	N	74	74	74

由表中数据得，价格与会员并无明显的相关关系，因此，建立多变量的非线性回归模型进行预测分析, 通过多次拟合，得到最优化非线性回归方程。

$$P = \frac{(Y + a_3 * \rho_1^2 + a_4 * r * s)}{a_5} + C \qquad (C = 50) \tag{4}$$

表中三种变量与平均价格间的误差分别为 0.002、0.002、0.005 均小于 0.05，因此有很强相关性；而考虑原始最低价格和实际可能情况找出最优常数 50。

4.2.3 方案对比

统一两种方案的完成度，分别求出 APP 所需付给会员的总酬金额，并画出折线图（图 5）：

表 8　平均价格的回归参数

参数	估计	标准错误	95% 信赖区间	
			下限	上限
a_3	0.002	0.002	-0.002	0.005
a_4	0.001	0.002	-0.002	0.004
a_5	0.077	0.005	0.067	0.088

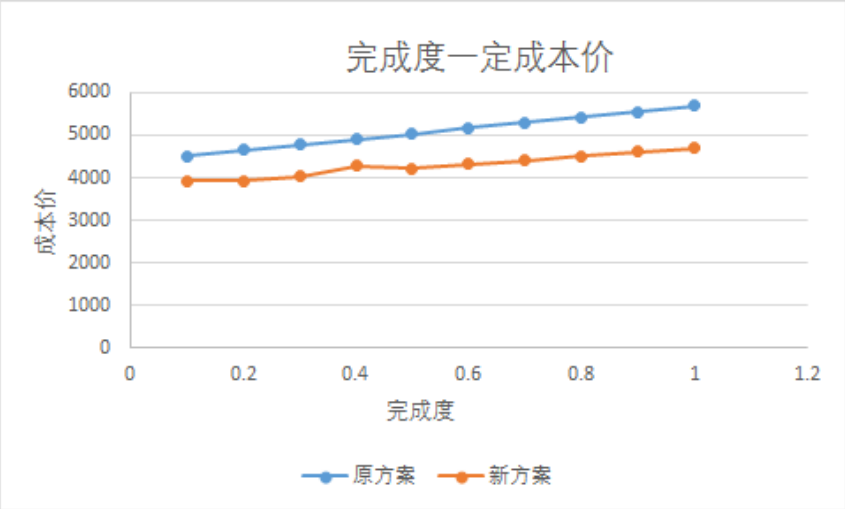


图 5　两方案成本价对比

由图可知：完成同样的任务，新方案所需付酬金较低。所以，新方案较原方案由更好的获利性。

4.3 问题三模型的求解

该题特色为考虑任务的非固定性，认为在多任务的分配机制上，可以把一部分位置集中的任务进行打包处理。需考虑合理的打包方案，并对上两问定价的定价模型进行修改，从而较好的解决用户对集中任务的争抢问题。在问题一二的模型中，均是按网格划分的方法对散布的点进行分离与重组。但在该题中，需对高集中区域的点进行划分，而网格形的划分难以实现对任意形状的聚类，亟需要一种合理的划分方案。

4.3.1 基于密度的聚类算法 (DBSCAN) 分析

DBSCAN(Density-Based Spatial Clustering of Application with Noise)，是一种基于聚类的分析算法，该算法具有足够密度的区域划分为簇，在给定的某一空间 θ，若存在点

集 $\delta(\delta \in \theta)$, 若给定一个点，算法能把附近的点分成一组，并标记出低密度区域的局外点。

而 DBSCAN 聚类，所有的点被分为核心点 θ_1(core points)、可达点 θ_2(reachable point), 局外点 θ_2(outliers)。如图 8 所示。$(A \in \theta_1 \quad A, B \in \theta_2 \quad N \in \theta_3)$

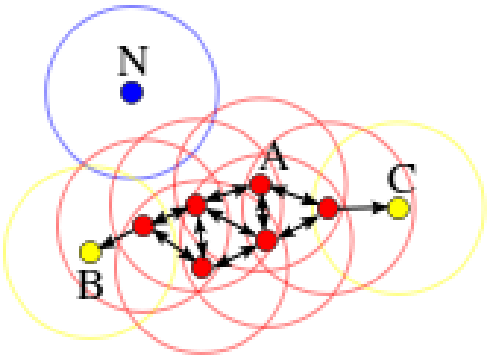


图 6　DBSCAN 算法模型图样

与传统聚类算法相比，其不需要人为主观确定聚类个数，并可以发现任意形状的聚类簇。针对于问题三中，有任务分布聚类不清晰，难以处理噪音（散乱点）的问题，而 DBSCAN 的优势可以充分发挥，确定多个高密度区域来表征打包区域，并能够避免一些较为离散点的对聚类的影响。

DBSCAN 算法所涉及的有两点基本定义有：

(1) ε 领域：给定对象半径 ε 内的区域

(2) 核心对象：如果给定 ε 领域内的样本数 $\tau \geq MinPts$，则为核心对象

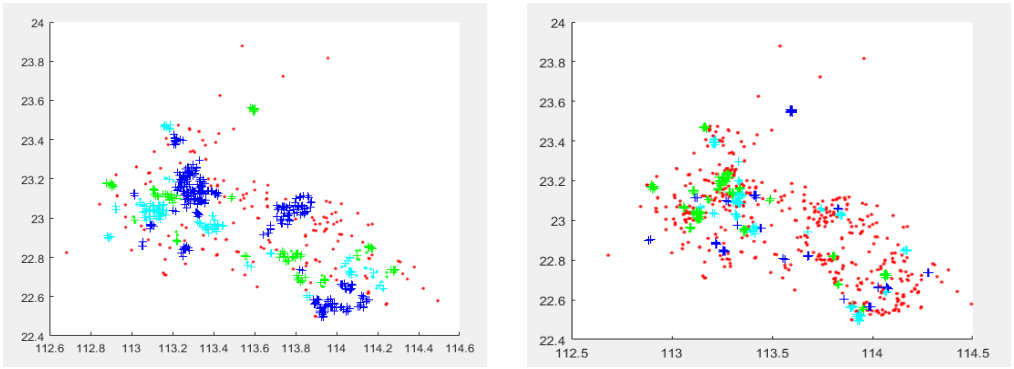
根据以上定义，我们需确定一组合理 $(\varepsilon, MinPts)$，使任务结果最优，4 组不同 $(\varepsilon, MinPts)$ 见图 7。

4.3.2 基于成本最低原则的近似最优建模

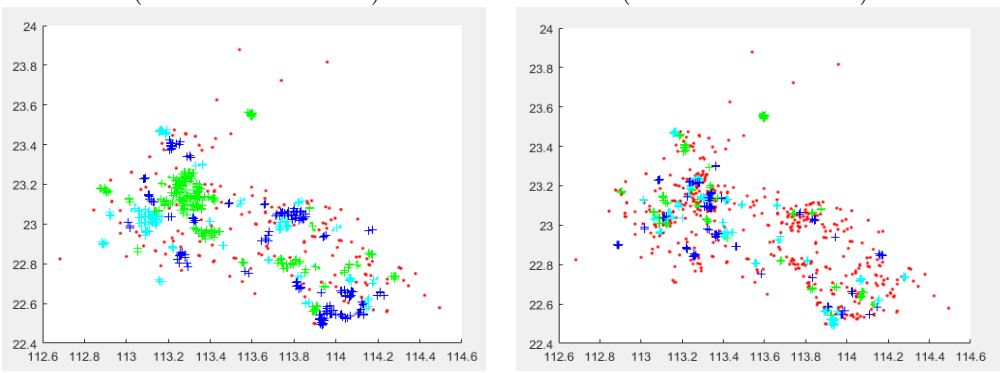
接上文所述, 我们现需定义目标函数 $f(x)$ 以及相关的约束方程, 通过对极值的求解, 来确定所需要的参数值 $(\varepsilon, MinPts)$。该题题目中, 存在明显的盈利目的, 因而运用微观经济学中成本最小化原则并结合密度聚类算法, 对问题三进行建模分析。

$$st: f(x) = min(f_n * \sum_{i=1}^{n} a_i) \tag{5}$$

$$\begin{cases} D(\varepsilon * MinPts) \propto \dfrac{1}{\rho_i} \\ min \sum_{i=0}^{n} \sum_{j=0}^{n} D(\varepsilon * MinPts) \end{cases} \tag{6}$$



($\varepsilon = 0.02$ $MinPts = 3$)　　($\varepsilon = 0.01$ $MinPts = 3$)



($\varepsilon = 0.02$ $MinPts = 2$)　　($\varepsilon = 0.01$ $MinPts = 2$)

图 7　附件三数据核密度分布

因为 $D(\varepsilon, MinPts)$ 存在缩点的操作, 即是把在第 i 个区域的所有点缩为一个点, 该点的权值为全点权值的总和 * 折扣 (80%)。除此之外, 点的减少, 对 ρ_i 有负相关作用。针对该模型解法采用枚举求解的方式, 具体流程图见图 8。

在如式 7 的情况，（0.0437 值为 DBSCAN 在该任务点密集程度的默认值），最后得到一个 10*10 的二维矩阵值（表 9）。观察可知，在 $\varepsilon = 0.052, MinPts = 3$ 的情况下，模型有最优成本解 992.7，该解可作为近似最优成本解。

$$\begin{cases} \varepsilon_{min} = \dfrac{0.0437}{10} \quad \varepsilon_{max} = 0.0437 \quad \varepsilon_{batch} = \dfrac{0.0437}{10} \\ MinPts_{min} = 1 \quad MinPts_{max} = 10 \quad MinPts_{batch} = 1 \end{cases} \tag{7}$$

4.3.3 模型评价与改进

该模型仅仅是一个最优解的近似解, 在 $\varepsilon, MinPts$ 的不断细分值的情况下, 我们能够得到一个更为精确的解。但模型仅存在一个相对相关性, 没有明确的数值关系, 完全求解该模型应该采取一个更为明确的解法, 如采用粒子群或者模拟退火算法。

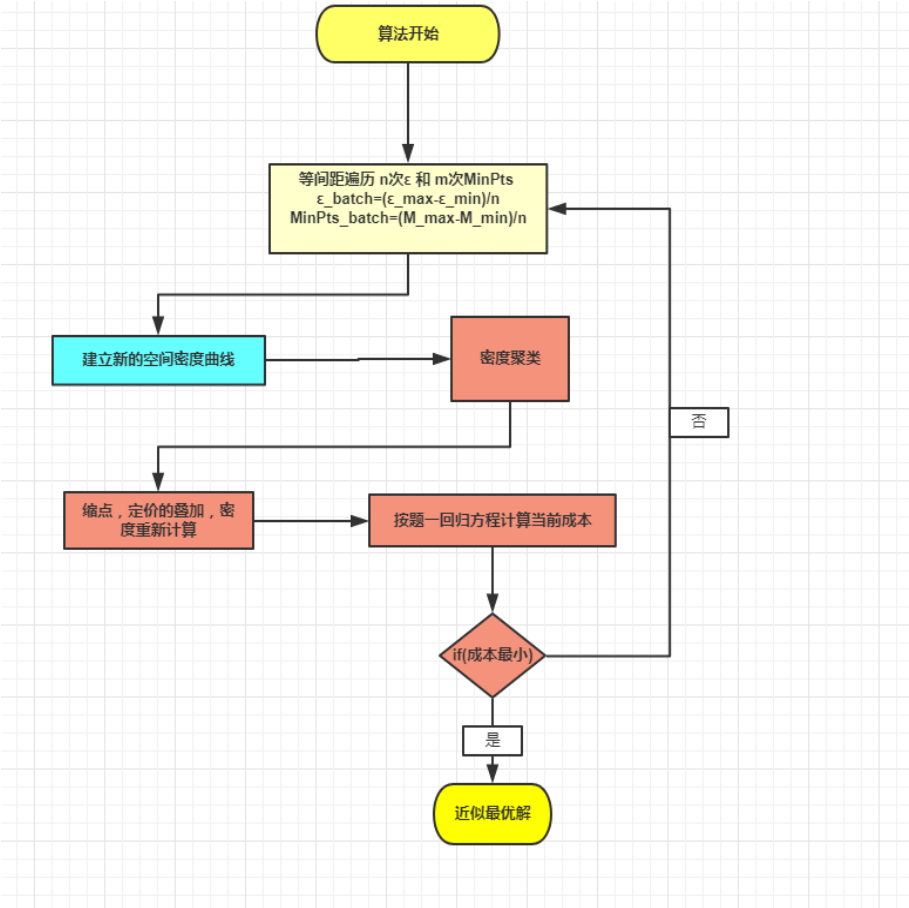


图 8　近似最优解的求解流程图

表 9　密度聚类模型枚举结果汇总

	$MinPts_{min}$									$MinPts_{max}$
ε_{min}	1574.9	1276.1	1126.8	1036.0	1016.8	1012.7	1012.7	1012.7	1012.7	1012.7
	2051.3	1716.7	1390.6	1221.5	1091.3	1031.0	1031.0	1031.0	1031.0	1012.7
	2318.5	2301.5	**992.7**	1660.5	1370.6	1283.7	1215.4	1033.4	1033.4	1033.4
	3241.4	3031.0	2414.6	2063.3	1729.1	1661.2	1561.5	1442.5	1261.6	1185.9
	4664.3	3737.0	2924.4	2164.1	1786.3	2305.7	1947.3	1779.8	1745.8	1687.5
	6298.5	5209.6	4101.2	3341.6	2866.4	3228.9	3026.3	2643.3	2144.0	2021.4
	7799.0	6448.3	5220.2	4376.0	3531.8	3291.8	3291.8	2861.4	2947.6	2999.1
	9573.8	7881.9	6654.8	5687.9	4651.3	3960.0	3960.0	3633.8	3692.0	3443.7
	11765.9	9629.7	8677.7	7753.0	5935.3	5382.2	5188.6	4724.0	4634.9	4634.9
ε_{max}	14304.0	11693.6	10129.4	9558.3	8844.2	6737.4	6737.4	6737.4	6621.6	6182.7

4.4 问题四模型的求解

4.4.1 数据特征分析

在附件三中, 一共包含 2066 个有效任务值, 按照前三问的方法, 通过观测其核密度曲线, 发现其数据及其集中, 基本分布于三个中心点, 在该种情况下, 我们考虑到基于密度的聚类和基于网格的多元回归模型已经难以实现定价。因此, 拟采用 *k-means* 做标准的聚类模型。

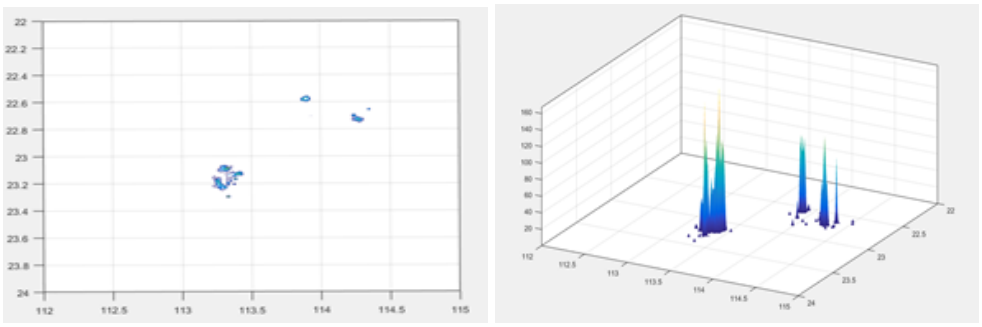


图 9　附件三数据核密度分布

4.4.2 k-means 空间加权建模

对于高密度数据，前三问所涉及到的模型难以对其进行合理化的解释。此时，可以采用以下算法:

(1) 根据问题二的任务数据在第 k 个聚类中，采取 k 周围 n 个点的均值作为第 k 个点的任务定价。

(2) 对于其他点，根据与 k 个聚类点的空间距离 r 关系，其定价 $P=\sum_{i=1}^{k}\frac{C_i}{r_i^2}$

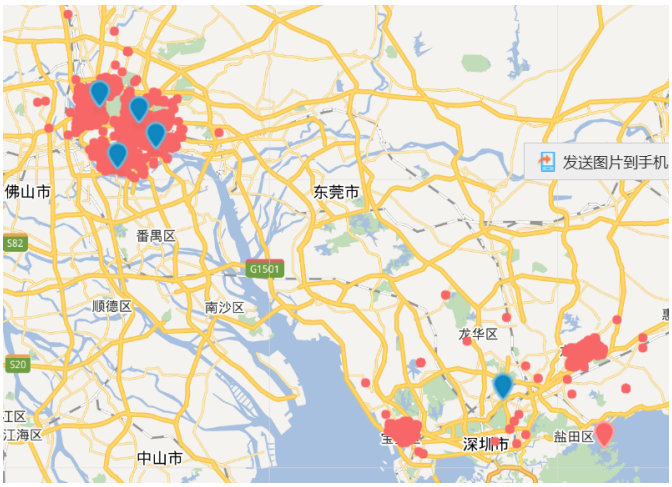


图 10　k-means (k=4) 关系图

五、 结论

5.1 模型优点

对于众包类平台发布的任务情况，本文基于不同的情况给了三种模型，其简单易懂、计算简便，对实际情况具有一定的指导意义。

5.2 模型缺点

基于网格的核密度非线性回归模型，其在参数调节具有较大的偶然性，且变量系数可能存在与现实相违背的情况。基于密度聚类的最优化求解模型，我们将一个聚类中的所有点划归为一个点，这种操作虽然能够降低整体的点密度，便于数据计算，但由于其与生活实际有相出入的地方，我们应该更加考虑到一个聚类中点的空间分布对结果的影响。而在 *k-means* 模型中简单的空间距离权重可能也难以适应复杂的实际情况，需对模型进行矫正，考虑其他因素的影响。

六、 参考文献

参考文献

[1] 徐群. 非线性回归分析的方法研究 [D][D]. 合肥: 合肥工业大学, 2009.

[2] 陈永胜. 基于 MATLAB 和 SPSS 的非线性回归分析 [J]. 牡丹江大学学报, 2009 (5): 101-103.

[3] 冯维. 基于粒子群算法求解多目标函数优化 [D]. 吉林大学, 2010.

[4] 金天坤. 多目标最优化方法及应用 [D]. 长春: 吉林大学, 2009.

[5] 夏鲁宁, 荆继武. SA-DBSCAN: 一种自适应基于密度聚类算法 [J]. 中国科学院大学学报, 2009, 26(4): 530-538.

[6] 刘淑芬, 孟冬雪, 王晓燕. 基于网格单元的 DBSCAN 算法 [J]. 吉林大学学报: 工学版, 2014 (4): 1135-1139.

附录 A DBSCAN 近似最优求解–matlab 源程序

```
MinPt_min =1; MinPt_max = 10; MinPt_batch =1;
r_max = 0.013682850358461; r_min =0.013682850358461/10; r_batch =0.013682850358461/10;
MinPt_line=1;r_line =1;
result=zeros(10);
for MinPts=MinPt_min : MinPt_batch : MinPt_max
    r_line =1;
   for Eps= r_min:r_batch:r_max

       %% 导入数据集
       data = jw_ex(:,1:2);

       % 定义参数Eps和MinPts
       %MinPts = 3;
       %Eps = epsilon(data, MinPts);
       %Eps =Eps/4;
       [m,n] = size(data);%得到数据的大小

       x = [(1:m)' data];
       [m,n] = size(x);%重新计算数据集的大小
       types = zeros(1,m);%用于区分核心点1，边界点0和噪音点-1
       dealed = zeros(m,1);%用于判断该点是否处理过,0表示未处理过
       dis = calDistance(x(:,2:n));
       number = 1;%用于标记类

       %% 对每一个点进行处理
       for i = 1:m
          %找到未处理的点
          if dealed(i) == 0
              xTemp = x(i,:);
              D = dis(i,:);%取得第i个点到其他所有点的距离
              ind = find(D<=Eps);%找到半径Eps内的所有点

              %% 区分点的类型

              %边界点
              if length(ind) > 1 && length(ind) < MinPts+1
                  types(i) = 0;
                  class(i) = 0;
              end
              %噪音点
              if length(ind) == 1
                  types(i) = -1;
                  class(i) = -1;
                  dealed(i) = 1;
```

```
            end
            %核心点(此处是关键步骤)
            if length(ind) >= MinPts+1
                types(xTemp(1,1)) = 1;
                class(ind) = number;

                % 判断核心点是否密度可达
                while ~isempty(ind)
                    yTemp = x(ind(1),:);
                    dealed(ind(1)) = 1;
                    ind(1) = [];
                    D = dis(yTemp(1,1),:);%找到与ind(1)之间的距离
                    ind_1 = find(D<=Eps);

                    if length(ind_1)>1%处理非噪音点
                        class(ind_1) = number;
                        if length(ind_1) >= MinPts+1
                            types(yTemp(1,1)) = 1;
                        else
                            types(yTemp(1,1)) = 0;
                        end

                        for j=1:length(ind_1)
                           if dealed(ind_1(j)) == 0
                              dealed(ind_1(j)) = 1;
                              ind=[ind ind_1(j)];
                              class(ind_1(j))=number;
                           end
                        end
                    end
                end
                number = number + 1;
            end
        end
    end

    % 最后处理所有未分类的点 为噪音点
    ind_2 = find(class==0);
    class(ind_2) = -1;
    types(ind_2) = -1;

    %重新生成地图
    new_map_tot =0;
    new_map = zeros(835,1);
    new_map_num = zeros(835,1);
    new_map_price =zeros(835,1);
    new_map_link=zeros(835,1);
```

```
for i=1:835
    if class(1,i) == -1
      new_map_tot = new_map_tot +1;
      new_map(new_map_tot,1) = jw_task(i,1);
      new_map(new_map_tot,2) = jw_task(i,2);
      new_map_link(new_map_tot,1) = class(1,i);
      new_map_price(new_map_tot,1) = price(i,1);
    else

       if new_map_num(class(1,i),1)==0
          new_map_tot= new_map_tot +1;
          new_map(new_map_tot,1) = jw_task(i,1);
          new_map(new_map_tot,2) = jw_task(i,2);

          new_map_link(new_map_tot,1) = class(1,i);
          new_map_num(class(1,i),1) = new_map_num(class(1,i),1)+1;
          new_map_price(class(1,i),1) = new_map_price(class(1,i)) + price(i,1);
       else
          new_map_num(class(1,i),1) = new_map_num(class(1,i),1)+1;
          new_map_price(class(1,i),1) = new_map_price(class(1,i)) + price(i,1);
       end
    end
end


%重新计算密度曲线
data= new_map;
[bandwidth_task,density_task,X_task,Y_task]=kde2d(data);

%计算每一点的p(任) 和 p(人) 和 price
price_tot=0;
finishrate_mean = 0;
for ii=1:new_map_tot
    t_j_i = new_map(ii,1);
    t_w_i = new_map(ii,2);
    v_j_i = jw(ii,1);%当前人的经度
    v_w_i = jw(ii,2);%当前人的纬度
    %计算p(人)
    x =floor ( (v_j_i-112) / ((115-112)/256) );
    y =floor ( (v_w_i-22) / ((24-22)/256) );
    pp = density(y,x);

    %计算p(任)
    x =floor ( (t_j_i-112) / ((115-112)/256) );
    y =floor ( (t_w_i-22) / ((24-22)/256) );
    pt = density_task(y,x);
```

```
        %计算price
        if new_map_link(ii,1)~=-1
            price_t = new_map_price(new_map_link(ii,1))*0.7;
        else
            price_t = new_map_price(new_map_tot,1) ;
        end

        price_tot = price_tot + price_t;
        finishrate = 0.057 * (price_t - 60) - 0.002 * pp *pp + 0.026 * pt;
        finishrate_mean = ( finishrate_mean*(ii-1)+finishrate )/ ii;
    end
    %{
    %% 画出最终的聚类图
    hold on
    for i = 1:m
        if class(i) == -1
            plot(data(i,1),data(i,2),'.r');
        elseif mod(class(i), 3) == 1
            if types(i) == 1
                plot(data(i,1),data(i,2),'+b');
            else
                plot(data(i,1),data(i,2),'.b');
            end
        elseif mod(class(i), 3) == 2
            if types(i) == 1
                plot(data(i,1),data(i,2),'+g');
            else
                plot(data(i,1),data(i,2),'.g');
            end
        elseif mod(class(i), 3) == 0
            if types(i) == 1
                plot(data(i,1),data(i,2),'+c');
            else
                plot(data(i,1),data(i,2),'.c');
            end
        else
            if types(i) == 1
                plot(data(i,1),data(i,2),'+k');
            else
                plot(data(i,1),data(i,2),'.k');
            end
        end
    end
    hold off
    %}
    result(r_line,MinPt_line)= price_tot * finishrate_mean ;
    r_line =r_line+1;
```

```
    end

    MinPt_line = MinPt_line+1;
end

result =result./100
```