

蚁群算法与DBSCAN融合的仓库选址智能算法研究

何 彭^{1*}, 刘长春¹, 王莹莹², 王梦旭², 杨 胤², 周林华^{2#}

¹长春理工大学计算机科学技术学院, 吉林 长春

²长春理工大学数学与统计学院, 吉林 长春

收稿日期: 2023年11月11日; 录用日期: 2023年12月4日; 发布日期: 2023年12月14日

摘 要

正确的仓库选址, 不仅能够提高运输效率, 还能降低运输成本, 对货物运输有着重要的作用。本文通过初始化蚁群算法距离矩阵, 迭代计算不可直达客户点之间的最短距离, 实现对蚁群算法的改进; 相较于传统的DBSCAN聚类算法, 我们将最短距离矩阵作为算法的输入, 提出了基于Distance自适应的DBSCAN算法(Distance_DBSCAN), 对核心点的选择策略进行修改, 实现对DBSCAN算法的改进; 针对聚类结果再次利用蚁群算法, 给出每一个类别的仓库选址位置和最短路径策略。实验表明, 本文提出的Distance_DBSCAN算法可以正确分离噪声点, 有着较好的ARI指数, 且时间复杂度低, 可以有效缩短聚类总路程, 更重要的是此算法更加符合货物运输仓库选址的实际意义。

关键词

仓库选址, 蚁群算法, 自适应的DBSCAN算法, 最短路径策略

Research on Intelligent Algorithm of Warehouse Location Based on Ant Colony Algorithm and DBSCAN

Peng He^{1*}, Changchun Liu¹, Yingying Wang², Mengxu Wang², Yin Yang², Linhua Zhou^{2#}

¹School of Computer Science and Technology, Changchun University of Science and Technology, Changchun Jilin

²School of Mathematics and Statistics, Changchun University of Science and Technology, Changchun Jilin

Received: Nov. 11th, 2023; accepted: Dec. 4th, 2023; published: Dec. 14th, 2023

*第一作者。

#通讯作者。

文章引用: 何彭, 刘长春, 王莹莹, 王梦旭, 杨胤, 周林华. 蚁群算法与 DBSCAN 融合的仓库选址智能算法研究[J]. 应用数学进展, 2023, 12(12): 5027-5038. DOI: [10.12677/aam.2023.1212494](https://doi.org/10.12677/aam.2023.1212494)

Abstract

The correct location of the warehouse can not only improve the transportation efficiency, but also reduce the transportation cost, which has an important role in the transportation of goods. By initializing the distance matrix of ant colony algorithm and iteratively calculating the shortest distance between inaccessible customer points, this paper realizes the improvement of ant colony algorithm. Compared with the traditional DBSCAN clustering algorithm, we take the shortest Distance matrix as the input of the algorithm, and propose an Adaptive DBSCAN algorithm based on distance (Distance_DBSCAN), modify the core point selection strategy, and realize the improvement of DBSCAN algorithm. According to the clustering results, the ant colony algorithm is used again to give the warehouse location and the shortest path strategy for each category. Experiments show that the Distance_DBSCAN algorithm proposed in this paper can correctly separate noise points, have a good ARI index and low time complexity, and can effectively shorten the total clustering distance. More importantly, this algorithm is more in line with the practical significance of cargo transportation warehouse location.

Keywords

Location of the Warehouse, Ant Colony Algorithm, Adaptive DBSCAN Algorithm, The Shortest Path Strategy

Copyright © 2023 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

随着网络技术的发展,电商平台在近20年间呈现出了爆炸式发展,并迅速达到饱和阶段[1]。在竞争方向上,商品价格以及服务人群已经很难再有所突破,而面对如此庞大的物流体系,如何有效降低运输成本是目前所有电商企业、快递业务的一个重要发展方向。

1959年,Dantzig和Ramser首次提出了车辆路径(VRP)问题[2],如何优化运输路径达到成本最低从此成为相关领域的研究热点。1996年Martin Ester等提出了DBSCAN算法[3],该算法是一种无监督的密度聚类算法,通过基于数据点的密度来确定簇,对噪声数据也具有较好的鲁棒性,广泛应用于实际问题。同年Marco Dorigo在文献[4]中首次提出了使用蚁群算法(AS)来解决(VRP)问题,AS能够自适应地搜索解空间并能够有效克服局部最优解,目前已被成功地应用于TSP和VRP等组合优化问题的求解[5][6][7][8]中。VRP问题属于组合优化问题[9],是NP难问题,目前主流的解决方案有两大类:精确方法和近似方法。

精确方法[10][11][12]针对求得全局最优解;近似方法则着力于求得局部最优解,其在寻找过程使用的算法进一步分为近似算法和启发式算法。对于VRP问题的求解,启发式算法有着更小的算法复杂度并可以保证一定精度,在求解过程中有其优势[13][14]。在这些算法的优势逐渐明晰后,国内外许多学者在DBSCAN和蚁群算法上展开研究,衍生出许多求解VRP问题的方案。徐书扬等[15]利用DBSCAN聚类算法对配送点进行聚类划分,根据聚类结果初始化蚁群算法参数,实现蚁群算法的改进与应用。许芳芳[16]提出的LF-DBSCAN算法,通过蚁群对数据集进行划分,为解决DBSCAN算法不适用密度不均匀数据集这一问题提供了思路。李静[17]结合蚁群优化算法和基于密度的DBSCAN算法来优化聚类效果。赵

振强等[18]通过改进 DBSCAN 算法和 K-means 二次聚类, 确定前置仓建设数量, 用多重心法实现前置仓的选址。

根据本文的研究发现, 大部分的实验利用 DBSCAN 算法中原始的距离计算方法, 即计算两点之间的欧氏距离[19], 而在实际问题中, 欧氏距离并不能很好地代表两地之间的远近。本文提出一种基于 Distance 自适应的 DBSCAN 算法(Distance_DBSCAN), 根据中国部分省会的经纬坐标作为客户点建立配送模型, 并通过初始化点集、输入所有相邻点之间的路程, 通过蚁群算法找到任意两点之间的最短路程, 然后使用 Distance_DBSCAN 算法迭代判断中心点是否满足目标条件, 再利用实际路程进行密度聚类, 得到更贴近实际情况的聚类结果。通过此算法与普通的基于密度的 DBSCAN 聚类方法以及 K-means 聚类方法的比较, 此算法的聚类效果可以得到更短的配送路程以及更合理的配送路线结果。

2. 背景简介和算法描述

2.1. 问题背景

2.1.1. VRP 问题

Versatile Routing Platform (VRP)问题即为车辆路线问题, 是运筹学领域重要的研究问题之一。该问题可简述为: 货物配送中心需要为一定数量, 各自拥有不同的货物需求客户配送货物, 并且由一个车队负责运送货物, 组织适当的行车路线。而货物配送中心的目标是既能满足客户的需求, 又能在一定的约束下, 达到诸如路程最短、成本最小、耗时间最少等目的。其原理图如图 1:

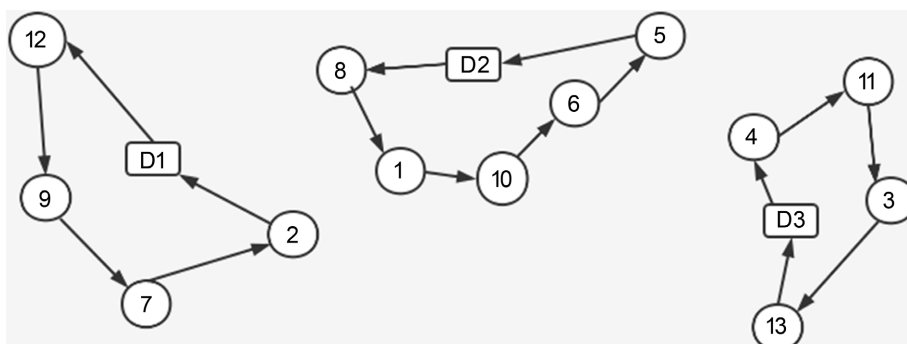


Figure 1. VRP diagram

图 1. VRP 示意图

图 1 中 D1、D2、D3 为货物配送仓库中心, 1~13 为客户位置, 通过不同的路径选择, 进行各自的货物运输, 达到最优目的。

2.1.2. 仓库选址问题

仓库选址问题是运筹学中经典的问题之一[20]。该问题可描述为在确定选址空间、选址区域和成本函数等约束条件的前提下, 以总物流运输成本最低或总服务水平最优或社会效益达到最大化为目标, 通过仓库位置的选择, 依次来确定物流配送方案中物流节点的数量、配送顺序等信息, 从而合理规划物流网络结构。

2.2. Distance_DBSCAN 算法(DBSCAN)描述

2.2.1. 蚁群算法和 DBSCAN 算法描述

蚁群算法(Ant Clony Optimization, ACO)是一种群智能优化算法, 该算法主要是由一群蚂蚁(轻微智能

个体, Agent)通过相互之间的协作(Collaboration) (通过信息素浓度(Pheromone Concentration)的积累和衰减)而表现出智能行为, 从而为求解复杂优化问题提供了一个新的可能性。

DBSCAN 算法是一种著名的密度聚类算法, 该算法不会受到距离的影响, 能够通过样本分布的紧密程度(密度)直接确定聚类结构, 从样本的密度的角度来考察样本之间的相关性, 并基于相关样本不断扩展聚类簇最终获得聚类结果。DBSCAN 算法不仅可以处理各种分布的数据并对簇进行判定, 而且还可以通过两个基本参数识别出数据集中的每个簇与噪声点。

2.2.2. DDBSCAN 算法设计

(a) 改进的蚁群算法初始化距离矩阵

蚁群算法进行矩阵初始化时使用的是欧式距离(European Distance)进行直接计算, 而实际货物运输过程中, 直接使用欧式距离并不能代表实际问题, 在改进过程中, 本文通过寻找出直接相连的客户之间的实际公路路程得到邻接矩阵, 并将该矩阵作为初始化矩阵输入蚁群算法中, 设置蚁群算法参数使其只能通过已有的客户点, 并找出不可直达的客户的最短路径, 不断迭代以完善最短距离矩阵。用这一方法计算得到的最短距离矩阵, 能够充分代表任意两个客户之间的实际距离。

假设 t 时刻客户 i 和客户 j 的距离为 d_{ij} , 路径上的信息素为 $\Upsilon_{ij}(t)$ 。初始时刻, 各个客户直接路径上信息素相同, 设为 $\Upsilon_{ij}(0) = \Upsilon_0$, 设 $p_{ij}^k(t)$ 为蚂蚁 k 在 t 时刻从客户 i 到客户 j 的概率, 如公式(1)所示:

$$p_{ij}^k(t) = \begin{cases} \frac{[\Upsilon_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta}{\sum_{j \in J_k(i)} [\Upsilon_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta}, & \text{if } j \in J_k(i) \\ 0, & \text{other} \end{cases} \quad (1)$$

其中 α 、 β 分别为信息素和启发式因子的相对重要程度, $\eta_{ij}(t)$ 为 t 时刻的启发函数。

在蚂蚁寻优系统中, 路径上的信息素浓度会随着蚂蚁离开的时间慢慢消散, 也会随着下一个到来的蚂蚁释放自己的信息素而积累, 令 $\rho(0 < \rho < 1)$ 为信息素挥发程度, m 为蚂蚁总数, 其信息更新表达式如公式(2)所示:

$$\begin{cases} \Upsilon_{ij}(t+n) = (1-\rho) \times \Upsilon_{ij}(t) + \Delta\Upsilon_{ij} \\ \Delta\Upsilon_{ij} = \sum_{k=1}^m \Delta\Upsilon_{ij}^k \end{cases} \quad (2)$$

初始化客户距离矩阵的蚁群算法运算过程:

步骤一, 初始化算法参数。如蚂蚁的数量 m 、信息素因子 α 、信息素挥发因子 ρ 、启发函数因子 β 、最大迭代次数 Maxiter、客户坐标、初始直达矩阵(Direct to the Matrix)等;

步骤二, 进行构建解空间。即把每个蚂蚁 k 放置于同一个出发点, 然后随机选择一个当前客户点未能直达的客户点, 直到所有蚂蚁遍历完所有点, 并最终到达目标点停止;

步骤三, 进行信息素的更新。更新前, 计算出每个蚂蚁走过的路径长度 L , 并记录当前的最优解 bestrode, 更新信息素浓度;

步骤四, 判断是否终止算法。如果迭代次数少于最大迭代次数, 或还存在客户之间不能连通, 则清除蚂蚁走过的路径记录, 迭代次数加一($iter = iter + 1$), 返回步骤二, 否则停止计算, 输出最优解;

步骤五: 输出程序结果。输出寻优过程中的相关指标, 如运行时间、收敛迭代次数等, 并返回最短距离矩阵(Shortest Distance Matrix)。

(b) DDBSCAN 聚类

传统的 DBSCAN 聚类算法, 在进行核心点选择时, 一般是通过比较客户矩阵的距离是否小于 Eps,

进而直接求得核心点坐标的索引, 这样算法时间较长, 效率低。采用 DDBSCAN 算法, 对核心点的选择策略进行修改, 通过直接对迭代过程中产生的核心点进行判断, 避免核心点被重复计算, 有效地降低了核心点的寻找和判断时间, 提升了算法效率。

DBSCAN 聚类算法参数 Eps (邻域半径) 为常数, 其值被赋为点与点之间的欧式距离。而在仓库选址问题中, 直接使用欧式距离进行距离代表两地间的距离并不符合实际, 本文改进的蚁群算法可以得到最短距离矩阵, 并将其使用在 DDBSCAN 聚类算法的距离选择中, 通过对符合实际情况的距离的迭代比较, 进行聚类划分。

DBSCAN 算法原理: 以 $Eps = 2$, $MinPts = 6$ 为假设条件, 图 2(a) 中, 初始化 P1 为第一个簇核心点; 图 2(b) 中, 以 P1 为核心, Eps 为距离阈值, 判断在最远 Eps 距离内可达客户点数是否满足数量大于等于 6, 以判断是否满足核心点条件; 图 2(c) 中, 在 P1 为核心的圆内, 以每一个点为核心画圆, 重复检验过程, 发现 P2 满足条件, 说明 P1 可达 P2; 图 2(d) 中, P2 可达 P3, 而 Q1 在验证过程中发现不能满足 $MinPts \geq 6$ 的条件, 则判定为不可达。经过以上过程本文可以得到 P1 通过 P2 可达 P3 的结果。

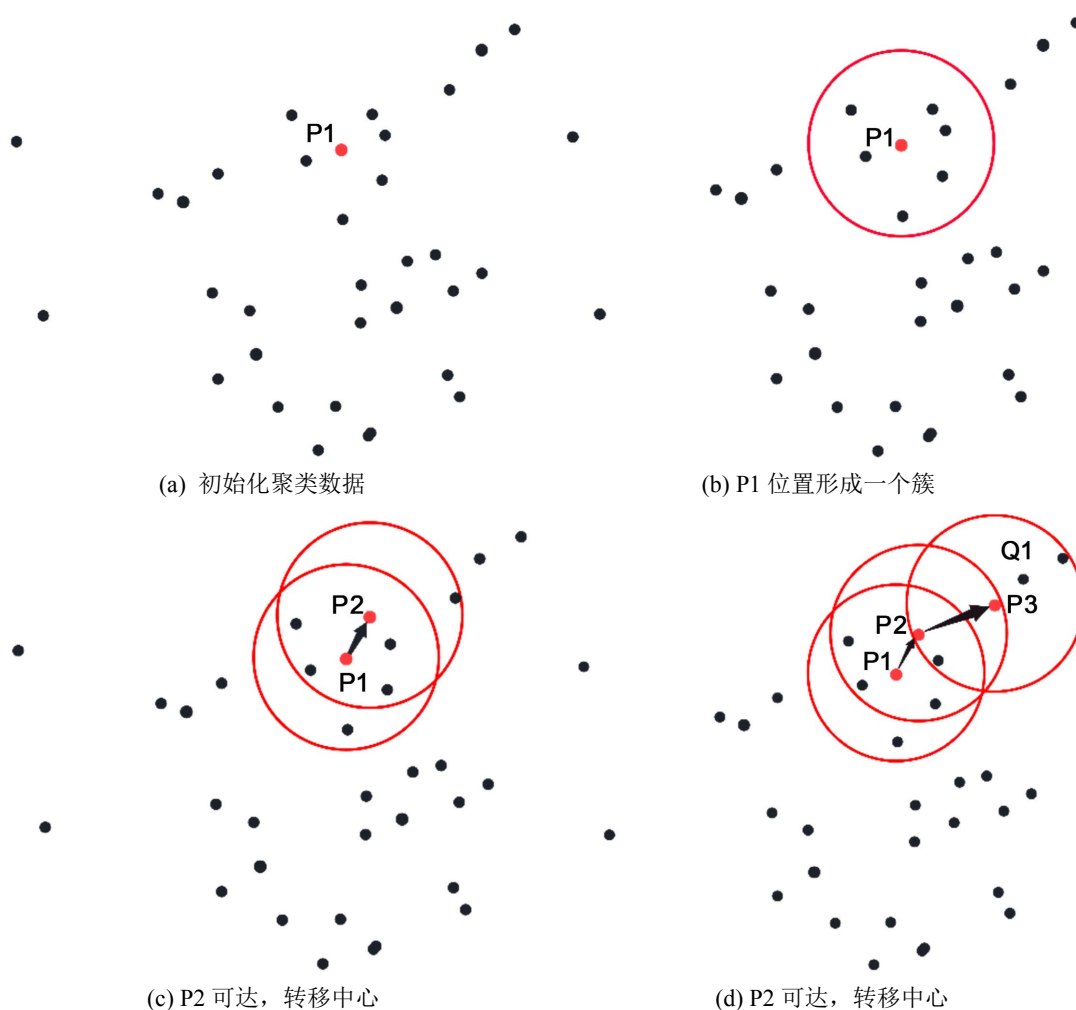


Figure 2. DBSCAN schematic diagram

图 2. DBSCAN 原理图

基于蚁群算法的 DDBSCAN 聚类算法流程如图 3。

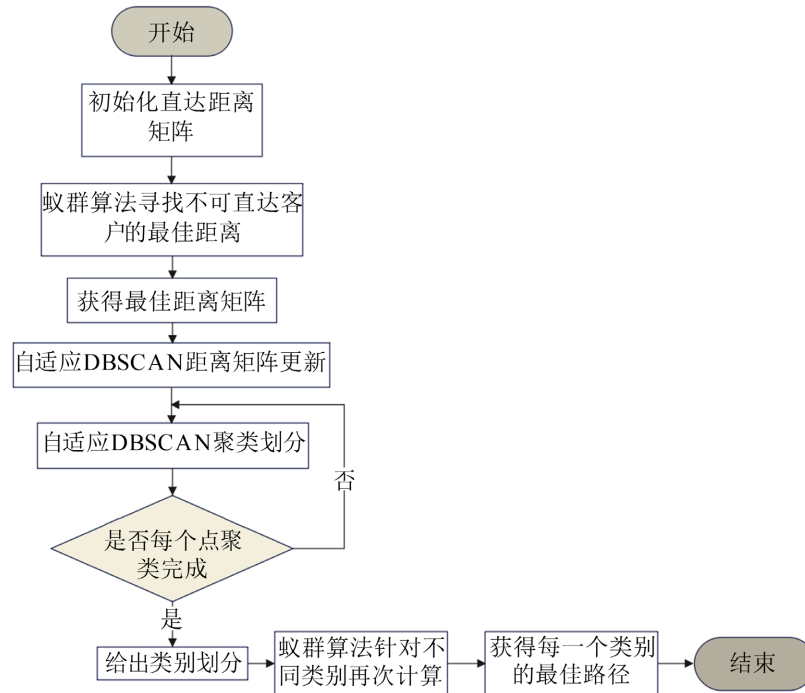


Figure 3. DDBSCAN clustering algorithm flow based on ant colony algorithm
图 3. 基于蚁群算法的 DDBSCAN 聚类算法流程

DDBSCAN 算法聚类伪代码如下所示:

ALGORITHM1: Pseudocode for the DDBSCAN algorithm

Input: DC : Dataset = (x_1, x_2, \dots, x_m)

Input: ε : Radius

Input: $\min Pts$: Density threshold

Input: $dist$: Minimum distance matrix #The shortest distance matrix obtained by the ant colony algorithm

Data: $label$: Point labels, initially undefined

Output: CD : Cluster division

Initialize: core object collection $\Omega = \phi$, number of cluster $K = 0$, unvisited sample collection $\Gamma = DC$, cluster division $C = \phi$

foreach point p **in** Γ **do** # Iterate over every point

| **if** $label(p) \neq undefined$ **then continue** # Skip processed points

| Neighbors $N \leftarrow RangeQuery(DC, dist, p, \varepsilon)$ # Find initial neighbors

| **if** $|N| < \min Pts$ **then** # Non_core points are noise

|| $label(p) \leftarrow Noise$

Continued

```

|| continue

|  $C \leftarrow$  next cluster label # Start a new cluster

|  $label(p) \leftarrow C$ 

| Seed set  $\Omega \leftarrow N/p$  # Expand neighborhood

| foreach point  $q$  in  $\Omega$  do

|| if  $label(q) = Noise$  then  $label(q) \leftarrow C$ 

|| if  $label(q) \neq undefined$  thencontinue

|| Neighbor  $N \leftarrow RangeQuery(DC, dist, q, \epsilon)$ 

||  $label(q) \leftarrow C$ 

|| if  $|N| < \min Pts$  thencontinue # Core_point check

||  $\Omega \leftarrow \Omega \cup N$ 

Output Result:  $CD = \{C_1, C_2, \dots, C_k\}$ 

```

DDBSCAN 整体算法摘要:

ALGORITHM2: Abstract DDBSCAN Algorithm

Initialize the ant colony algorithm distance matrix

Calculate the shortest distance matrix and replace it with the DBSCAN distance matrix

Computing neighbors of each point and identify core points

Joining neighboring core points into clusters

foreach non_core point do

| Adding to a neighboring core point if possible

| Otherwise, adding to noise

3. 实验内容和结果分析

3.1. 问题评价指标

为了评价聚类效果好坏，本文引入以下评价指标，用于检测聚类效果的优劣。

3.1.1. 轮廓系数[21]

轮廓系数(Silhouette Coefficient, SI)是描述聚类后各个类别的轮廓清晰度的指标。其包含有两种因素——内聚度和分离度，轮廓系数的计算方法如公式(3)所示：

$$S(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \quad (3)$$

其中 $a(i)$ 代表内聚度，可通过公式(4)计算：

$$a(i) = \frac{1}{n-1} \sum_{j \neq i}^n dis(i, j) \quad (4)$$

j 是与样本 i 在同一个类内的其他样本点， $dis(i, j)$ 代表 i 与 j 的距离。而 $b(i)$ 需要遍历其他簇得到多个值 $\{b_1(i), \dots, b_m(i)\}$ ，并从中选择最小的值作为结果。

所以， $S(i)$ 可进一步简化为公式(5)：

$$S(i) = \begin{cases} 1 - \frac{a(i)}{b(i)} & a(i) < b(i) \\ 0 & a(i) = b(i) \\ \frac{b(i)}{a(i)} - 1 & a(i) > b(i) \end{cases} \quad (5)$$

轮廓系数 $S(i) \in [-1, 1]$ ，数值越大聚类效果越好。

3.1.2. 噪声比

聚类噪声比是指在聚类过程中，噪声点占整个数据集的比例。它是评估聚类算法性能的一个重要指标之一，通常噪声比越小越好。

3.1.3. 调节的兰德指数[19]

给定 n 个对象集合， $S = \{O_1, O_2, \dots, O_n\}$ ，假设 $U = \{u_1, u_2, \dots, u_R\}$ 和 $V = \{v_1, v_2, \dots, v_C\}$ 表示 S 的连个不同划分，且满足 $\bigcup_{i=1}^R u_i = S = \bigcup_{j=1}^C v_j$ ， $u_i \cap u_{i^*} = \emptyset = v_j \cap v_{j^*}$ 其中 $1 \leq i \neq i^* \leq R$ ， $1 \leq j \neq j^* \leq C$ 。

其中兰德系数(R):

$$R = \frac{a+b}{C_n^2} \quad (6)$$

其中 n 为样本总数。

调节的兰德指数(Adjusted Rand Index, ARI)又称调整兰德系数，调整兰德系数假设模型的超分布为随机模型，即 U 和 V 的划分为随机的，那么各类别和各簇的数据点数目是固定的，不会随聚类标签变化而变化，计算方法如公式(7)所示：

$$ARI = \frac{RI - E(RI)}{\max(RI) - E(RI)} \quad (7)$$

其中 ARI 取值范围为 $[-1, 1]$ ，值越接近 1，聚类效果越好。

3.2. 初始数据和算法参数

初始数据集属性见表 1，在本文的数据集中，共有 30 个数据点，划分为 5 类。

Table 1. Properties of the dataset

表 1. 数据集属性

数据集名称	规模	簇数目	维度
全国主要 4S 店位置数据	30	5	2

蚁群算法部分参数与初始化数据见表 2，根据客户数量和距离矩阵，设置相应的参数，其中*为可变参数，其余为实验建议参数。

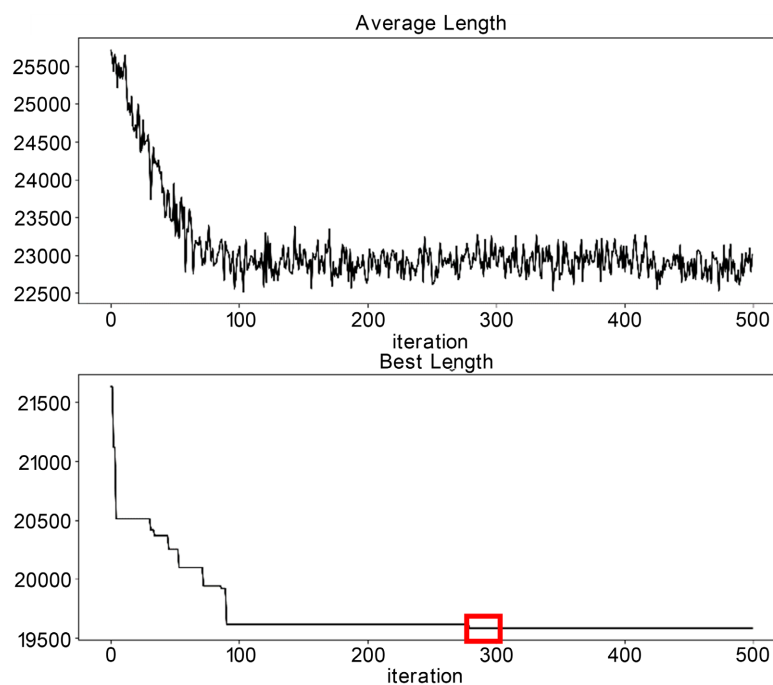
Table 2. Parameters of the ant colony algorithm

表 2. 蚁群算法参数

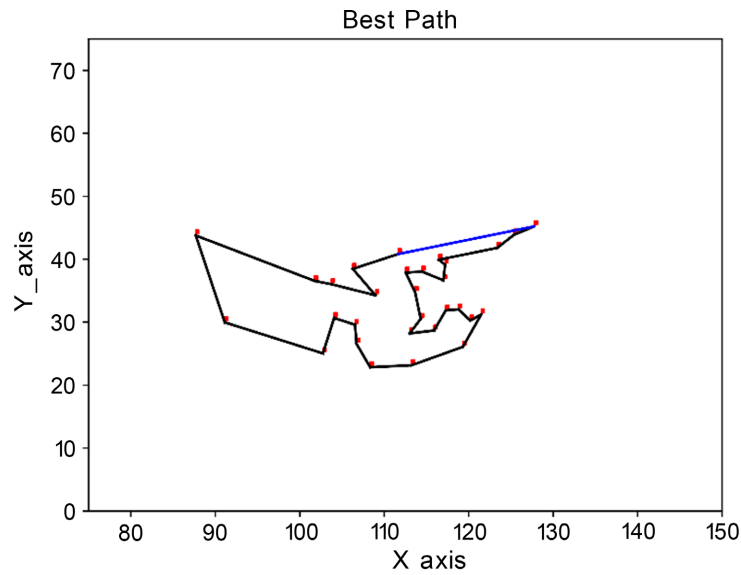
参数	值	含义
distmat	30 × 30 的距离矩阵	所有结点(包括仓库地址)之间的最短距离矩阵
numant*	45	蚂蚁数量
numcity	30	客户数量
alpha	1	信息素重要程度因子
beta	5	启发函数重要程度因子
rho	0.1	信息素的挥发速度
Q	1	信息素释放总量
iter	0	循环次数
Itermax*	500	最大循环次数
etatable	$1.0/(\text{distmat} + \text{np.diag}([1e10] * \text{numcity}))$	启发函数矩阵，表示蚂蚁从客户 i 转移到矩阵 j 的期望程度

3.3. DDBSCAN 算法结果

图 4(a)中，使用实际距离的初始化矩阵，迭代 500 次后，平均路径长度和最佳路径长度如图所示，可以看出算法迭代到约 270 轮左右达到最佳，最佳路程为 19,587；图 4(b)为全局最短路径示意图。



(a) 改进的蚁群算法迭代过程



(b) 改进的蚁群算法总数据路径

Figure 4. Iterative process of the improved ACO algorithm
图 4. 改进的蚁群算法迭代过程

DDBSCAN 对 30 个客户最终聚类结果如图 5 所示:

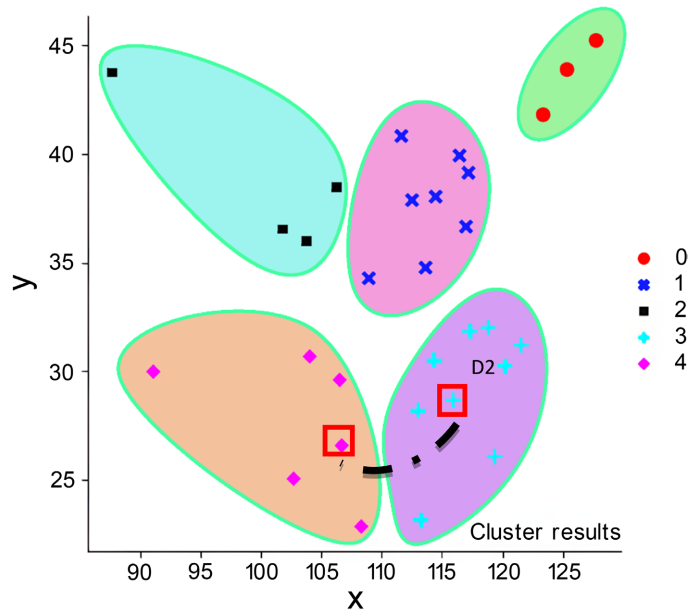


Figure 5. DDBSCAN clustering results
图 5. DDBSCAN 聚类结果

使用 DDBSCAN 进行聚类, 并将噪声点进行归类, 完成仓库聚类选址问题, 在图中一共分为 5 类, 仓库位置可选择在同一个聚类点的任何位置, 为了最优化上一层运输到仓库问题, 可选择类与类之间相对较近的点最为该类的仓库点, 如图 5 中的 D1 和 D2。

图 6 更进一步展示 DDBSCAN 聚类距离效果, 使用蚁群算法重新进行类间距离寻找, 路线如图 6 所示, 最后计算得到总路程为 19,030。

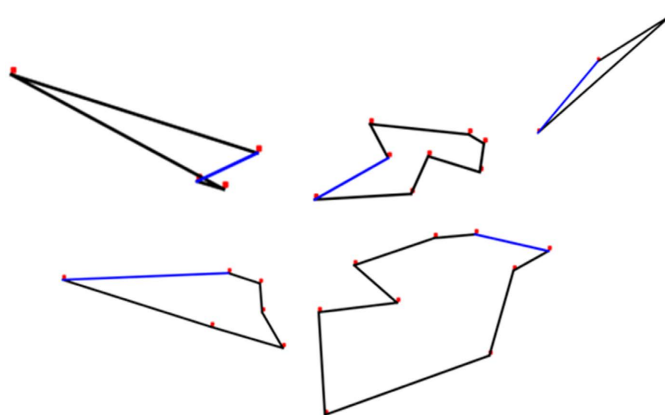


Figure 6. Shortest path obtained by ant colony algorithm after DDBSCAN clustering
图 6. DDBSCAN 聚类后蚁群算法获得最短路径

K-means 算法在对数据集进行聚类时，将所有数据点作为计算对象，在聚类结果中不存在噪声点，在一个类内部的类内元素紧密程度小，内聚度小；而在 DBSCAN 算法和 DDBSCAN 算法计算轮廓系数时，噪声点并不在任何一个类内部，使这两种算法的内聚度大；则 K-means 有较好的轮廓系数。而在 ARI 分数上，DDBSCAN 算法得到的聚类结果与理想的结果最为接近，分数相较于 K-means 算法有较大的提升，同时，DDBSCAN 算法的时间复杂度较低，详细数据见表 3。

Table 3. Comparison of algorithm performance
表 3. 算法性能对比

算法	轮廓系数	噪声比	ARI	总运行时间
K-means	0.305	/	0.590	0.01
DBSCAN	0.295	0.2	0.7122	0.01
DDBSCAN	0.295	0.2	0.7122	0.00099

蚁群算法通过启发式寻找优化出总路程为 19,587, 经过 DDBSCAN 算法聚类得到的总路程为 19,030, 相较于直接使用蚁群算法总路程减少 557, 且每一个类中的路径总路程均小于或等于 K-means 算法结果, 见表 4。综合表 3 和表 4 可以说明该算法具有一定的优势。

Table 4. Shortest path distance comparison
表 4. 最短路径路程对比

算法	类别 1	类别 2	类别 3	类别 4	类别 5	总路程
蚁群算法	/	/	/	/	/	19,587
K-means	1116	4890	6434	4344	3748	20,532
DDBSCAN	1116	4779	6060	3881	3194	19,030

4. 结语

针对实际仓库选址问题，本文提出基于 Distance 自适应的 DBSCAN 算法(DDBSCAN)，通过提供客户点两两直达的初始路程矩阵，针对蚁群算法的初始信息矩阵进行更新，进一步求解出初始客户之间未能两两直达的距离，通过不断迭代，获得最短距离全连接矩阵，随后使用 DDBSCAN 聚类算法。首先对

算法的距离矩阵进行替换,随后在聚类的过程中,更改以往的聚类核心点选择策略,从而提高算法聚类效率,最后将噪声点归类,获得最终聚类结果并再次利用蚁群算法计算每一类的总路程路程,通过比较各个算法的性能和路程,证明了本文的 DDBSCAN 算法更突出,可以正确分离噪声点,时间复杂度低,对聚类结果、总路程均有改善,且更加符合实际仓库选址问题。考虑到数据点数量问题,本文只针对 30 个全国抽象出来的 4S 店进行测试实验,在后期的处理过程中可收集更多的仓库数据点,其次在蚁群算法迭代过程中,可进行多次参数调节,以获得最佳参数。

基金项目

吉林省自然科学基金自由探索重点项目(YDZJ202201ZYTS585);
吉林省创新能力建设项目(2022C047-2)。

参考文献

- [1] 李恒宇. 基于产品价值提升的电商平台自有品牌生产策略研究[D]: [博士学位论文]. 成都: 电子科技大学, 2022.
- [2] Yue, S.H., Li, P., Guo, J.D., et al. (2005) A Statistical Information-Based Clustering Approach in Distance Space. *Journal of Zhejiang University Science A*, **6**, 71-78.
- [3] Ester, M. (1996) A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, Portland, 2-4 August 1996, 226-231.
- [4] Dorigo, M., Maniezzo, V. and Colomi, A. (1996) Ant System: Optimization by a Colony of Cooperating Agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, **26**, 29-41. <https://doi.org/10.1109/3477.484436>
- [5] 郭城成, 田立勤, 武文星. 蚁群算法在求解旅行商问题中的应用综述[J]. 计算机系统应用, 2023, 32(3): 1-14.
- [6] 张泽群. 基于蚁群算法的 TSP 问题研究[J]. 中国科技期刊数据库工业 A, 2022(4): 290-294.
- [7] 文宗川, 王慧. 基于蚁群算法在 VRP 中的应用研究——以呼和浩特 A 物流公司为例[J]. 物流工程与管理, 2022, 44(2): 26-28, 39.
- [8] 付宇卓, 孙晓奇, 刘婷. 一种针对 VRP 问题的混合蚁群算法及其实现系统[P]. 中国专利, CN201810074125.7. 2018-06-19.
- [9] 李凯文, 张涛, 王锐, 等. 基于深度强化学习的组合优化研究进展[J]. 自动化学报, 2021, 47(11): 2521-2537.
- [10] 于胜龙, 薄煜明, 陈志敏, 等. 基于混沌粒子群优化的新型 VRP 求解算法[J]. 计算机工程与科学, 2012, 34(12): 164-168.
- [11] 魏洁, 刘畅, 郑迎迎. 混合改进人工鱼群算法逆向回收车辆路径问题的研究[J]. 信息与管理研究, 2022, 7(4): 59-72.
- [12] 王晓东, 张永强, 薛红, 等. 基于改进蚁群算法对 VRP 线路优化[J]. 吉林大学学报(信息科学版), 2017, 35(2): 198-203.
- [13] 刘维怡, 王卓, 陈振武, 等. 面向出行服务的车辆路径问题求解器应用研究[C]//中国智能交通协会. 第十五届中国智能交通年会科技论文集(1). 北京: 中国工信出版集团, 电子工业出版社, 2020: 433-444.
- [14] 朱外明, 梁培培, 刘根节, 等. 无人机与物流柜协同配送最短路径问题启发式算法[J]. 无线电工程, 2022, 52(7): 1231-1237.
- [15] 徐书扬, 俞鸿烽, 潘华铮, 等. DBSCAN 聚类处理的改进蚁群算法在车辆路径问题中的应用[J]. 电脑知识与技术, 2020, 16(19): 182-186.
- [16] 许芳芳. 一种结合蚁群聚类算法的 DBSCAN 算法[J]. 池州学院学报, 2014, 28(6): 33-36.
- [17] 李静. 结合蚁群算法与基于划分的 DBSCAN 聚类算法的研究[D]: [硕士学位论文]. 长春: 东北师范大学, 2011.
- [18] 赵振强, 张立涛, 王文钊, 等. 基于顾客需求量分布的生鲜农产品前置仓选址研究[J]. 计算机应用与软件, 2021, 38(10): 107-113, 124.
- [19] 周志华. 机器学习[M]. 北京: 清华大学出版社, 2016: 33-37.
- [20] 黎煜. 带次模特性的仓库选址问题研究[D]: [博士学位论文]. 北京: 北京交通大学, 2012.
- [21] 李珏. 数据驱动的小学数学素养评价方法研究[D]: [博士学位论文]. 武汉: 华中师范大学, 2020.