计算机科学
COMPUTER SCIENCE

基于DBSCAN的动态邻域密度聚类算法

张朋, 李小林, 王李妍

引用本文

张朋, 李小林, 王李妍.基于DBSCAN的动态邻域密度聚类算法[J]. 计算机科学, 2023, 50(6A): 220400127-7.

ZHANG Peng, LI Xiaolin, WANG Liyan. Dynamic Neighborhood Density Clustering Algorithm Based on DBSCAN [J]. Computer Science, 2023, 50(6A): 220400127-7.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.220400127

基于 DBSCAN 的动态邻域密度聚类算法

张 朋 李小林 王李妍
中国矿业大学矿业工程学院 江苏 徐州 221003
(905929036@qq.com)

摘 要 传统的密度聚类算法在聚类划分时不会考虑数据点间的属性差异,它将所有数据点都看成同质化的点。对此,在 DBSCAN 算法的基础上,提出了一种动态邻域密度聚类算法 DN-DBSCAN(Dynamic Neighborhood-Density Based Spatial Clustering of Applications with Noise)。该算法在聚类时由样本点的属性决定其自身的邻域半径,因此各点的邻域半径是动态变化的,由此可将具有不同属性的点对集群产生的不一样的影响力体现在聚类结果之中,使密度聚类算法更具有现实意义。在算例分析的基础上,针对长三角城市群划分问题应用所提 DN-DBSCAN 算法进行分析求解,并对比分析 DBSCAN 算法、OPTICS 算法和 DPC 算法的求解效果。结果显示,DN-DBSCAN 算法能根据各城市属性的不同合理地划分出长三角城市群,准确率为 95%,准确率分别高于上述 3 种对比算法 85%,85%,88%,说明其具有更好的解决实际问题的能力。

关键词:动态邻域;密度聚类;动态邻域密度聚类;属性差异;划分准确率
中图法分类号 TP301

Dynamic Neighborhood Density Clustering Algorithm Based on DBSCAN

ZHANG Peng,LI Xiaolin and WANG Liyan
College of mines,China University of Mining and Technology,Xuzhou,Jiangsu 221003,China

Abstract The traditional density clustering algorithms do not consider the attribute difference between data points in the clustering process,but treat all data points as homogenous points. Based on the traditional DBSCAN algorithm,a dynamic neighborhood—density based spatial clustering of applications with noise(DN-DBSCAN) is proposed. When it is working,each point's neighborhood radius is determined by the properties of itself,so the neighborhood radius is dynamic changing. Thus,different influences on datasets produced by points with different properties is reflected in the clustering results,making the density clustering algorithm has more practical meaning and can be more reasonable to solve practical problems. On the basis of example analysis,the DN-DBSCAN algorithm is applied to solve the urban agglomeration division problem in the Yangtze river delta,and the results of DBSCAN algorithm,OPTICS algorithm and DPC algorithm are compared and analyzed. The results show that DN-DBSCAN algorithm can reasonably classify urban agglomerations in the Yangtze river delta according to the different attributes of each city with an accuracy of 95%,which is much higher than the accuracy of 85%,85% and 88% of the other three algorithms respectively,indicating that it has a better ability to solve practical problems.

Keywords Dynamic neighborhood,Density clustering,Dynamic neighborhood density clustering,Attribute differences,Division accuracy

1 引言

DBSCAN 算法是最著名也是最具代表性的密度聚类算法,其通过两个全局参数 ε 与 *MinPts* 将具有足够高密度的区域划分为簇,可在带噪声的空间数据集中发现任意形状的类簇[1]。然而,DBSCAN 算法对输入的邻域半径参数 ε 非常敏感,这会显著影响聚类的质量。对此,国内外专家提出了不同的改进方法,Cai 等[2]提出的 DBSCANCC 算法通过记录所有的簇连接信息,再参考实际的聚类结果将错误分开的子簇合并,以此来屏蔽输入参数 ε 的敏感性对聚类结果的影响。Feng 等[3]提出的 Greedy DBSCAN 算法采用贪心策略,在输入一个参数 *MinPts* 的情况下可以自适应地确定参数 ε。Chen 等[4]提出的 Improved DBSCAN 算法通过引入最小二乘法的概念,基于统计学自适应寻找参数 ε。Zhou 等[5]提出的 I-DBSCAN 算法依据数据集本身的统计特性,通过观察距离分布矩阵来得到参数 ε。Yue 等[6]提出了一种基于数据统计信息确定参数 ε 的算法,该方法可以在更广的范围内搜索参数 ε。这些对 DBSCAN 改进的算法,主要是通过调整聚类策略来减小输入参数 ε 的敏感性对聚类结果的影响,或是根据数据集本身的数据分布规律来确定邻域半径参数 ε,但在处理数据集中样本点的方式上与 DBSCAN 算法并无区别,都是把待处理的数据集视作同质化的样本点的集合,没有考虑样本点的属性差异,因此在处理实际问题时效果不佳。对此,本文在 DBSCAN 算法的基础上,进一步考虑样本点的差异信息,提出了由样本点属性决定自身邻域半径的动态邻域密度聚类算法 DN-DBSCAN,并应用改进算法解决现实生活

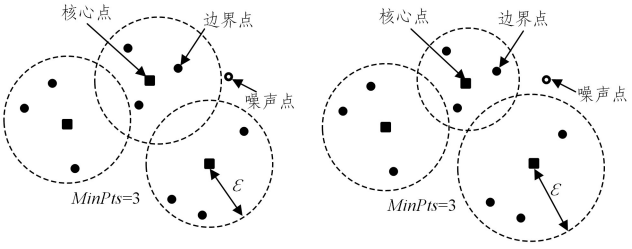
基金项目:国家自然科学基金(71401164)
This work was supported by the National Natural Science Foundation of China(71401164).
通信作者:李小林(xlli@cumt.edu.cn)

中典型的城市群划分问题，对比分析表明所提算法的划分结果准确度远远高于基础的 DBSCAN 算法以及另外两种经典密度聚类算法 OPTICS 算法和 DPC 算法。

2 DN-DBSCAN 算法设计

2.1 算法介绍及基本定义

DN-DBSCAN 是一种基于密度的聚类算法，其假定类别可以由样本分布的紧密程度决定，同一类别的样本之间是紧密相连的，即在该类别任意样本周围近距离内一定有同类别的样本存在。通过将紧密相连的样本划为一类，可以得到一个聚类类别；通过将各组紧密相连的样本划为不同的类别，可以得到所有的聚类类别。如图 1 所示，当某样本点 ε 邻域内包含的样本点数量达到设定的阈值 *MinPts* 时，则称之为核心点；核心点 ε 邻域内的点称之为边界点；既不是核心点也不是边界点的称之为噪声点。DN-DBSCAN 与 DBSCAN 算法的区别在于其邻域 ε 不是固定不变的，而是依据样本点的属性不同而动态变化的，具体到某一个点的邻域半径如何，只有当算法运行到该点时才能由其自身属性确定。这样处理的意义在于，现实环境下群体中的点普遍是有差异的，对群体的影响力是不同的，DN-DBSCAN 算法会在聚类的过程中考虑到这种差异性，并将其体现在聚类结果中。因此，DN-DBSCAN 算法在解决现实问题时会比传统的 DBSCAN 算法有更好的效果；同时算法的初始输入参数也由 2 个变为了 1 个，只需要输入密度阈值 *MinPts* 即可。



(a)DBSCAN 算法图解　　(b)DN-DBSCAN 算法图解

图 1　DBSCAN 与 DN-DBSCAN 对比图

Fig. 1　Comparison between DBSCAN and DN-DBSCAN

ε 邻域：对任意一个点 p，其 ε 邻域定义为 $N_\varepsilon(p)=\{q\in D \mid dist(p,q)\leqslant\varepsilon\}$。

密度：设 $x\in X$，则 $\rho(x)=|N_\varepsilon(x)|$ 为 x 的密度。

核心点(Core point)：设 $x\in X$，若 $\rho(x)\geqslant MinPts$，则称 x 为 X 中的核心点，中心点构成的集合为 X_c。

边界点(Border point)：设 $x\in X\backslash X_c$，且 x 落在某个核心点的 ε 邻域内。一个边界点可能落在多个核心点的 ε 邻域内。

噪声点(Noise)：既不是核心点也不是边界点的点。

直接密度可达：设 $x,y\in X$，若满足 $y\in N_\varepsilon(q)$ 且 $|N_\varepsilon(q)|\geqslant MinPts$，则称 y 从 x 直接密度可达。

密度可达：假设存在一串点 $p_1,p_2,\cdots,p_n,p_1=q,p_n=p$，使得 p_{i+1} 从 p_i 是直接密度可达的，那么就认为 p 从 q 密度可达。

密度相连：假设存在点 o,p,q，其中 p,q 均从 o 密度可达，那么 p 和 q 密度相连。密度相连具有对称性。

类簇：设非空集合 $C\subset X$，若满足：$\forall p,q$，

(1)$p\in C$，且 q 从 p 密度可达，那么 $q\in C$。

(2)p 和 q 密度相连。

则称 C 构成一个类簇。

2.2 DN-DBSCAN 算法描述及伪代码

DN-DBSCAN 的算法描述及伪代码如算法 1 所示。

算法 1　动态邻域密度聚类算法

输入：一个包含 n 个对象的数据集 D、邻域密度阈值 MinPts

输出：基于密度的簇的集合

算法描述：

1. 任意选取 D 中一个未访问的样本点 P，以 P 为中心，由 P 自身的属性决定其邻域半径 ε；
2. 如果 p 的 ε-邻域内至少有 MinPts 个对象，则点 P 为核心点，遍历其邻域内的点，如果已有点属于某一现有簇 B，则将该核心点 P 归入 B 中，如果邻域内没有任何点属于某一现有簇，则创建一个新簇 C 并将核心点 P 归入其中，然后将邻域内的所有点归入核心点所在的簇(B or C)中；
3. 遍历簇 B 或簇 C 中未被访问的样本点，若有核心点，将核心点邻域内的点也放入对应核心点所在的簇 B 或簇 C 中；
4. 重复步骤 3 的操作，直到簇 B 或簇 C 中不再加入新的点；
5. 重复步骤 1—4 的操作，直到 D 中所有对象都被放入簇内或标记为噪声。

算法伪代码：

```
1. 标记所有对象为 unvisited
2. 随机选择一个 unvisited 对象 p
3. 标记 p 为 visited
4. 由 p 的属性确定自身的 ε-邻域
5. If p 的 ε-邻域至少有 MinPts 个对象
6.     令 N 为 p 的 ε-邻域中的对象集合
7.     For N 中每个点 p*
8.       If p* 已归入某个簇 B，则把 p 也归入簇 B
9.       Else 创建一个新簇 C，并把 p 添加到簇 C
10.    End for
11.    For N 中每个点 p*
12.      If p* 是 unvisited，标记 p* 为 visited
13.        由 p* 的属性确定自身的 ε-邻域
14.        If p* 的 ε-邻域至少有 MinPts 个对象，把这些对象添加到 N
15.        将 p* 添加到 p 所在的簇(B or C)
16.    End for
17.      输出 B or C
18. Else 标记 p 为噪声
19. Until 没有标记为 unvisited 的对象
```

3 算例研究

此算例数据集共有 14 个二维数据点，每个点除了有 X，Y 坐标外，还有一个重量属性，具体数值如表 1 所列。

表 1　算例数据集

Table 1　Example datasets

No.	X	Y	重量
1	0.0	0.0	1.0
2	0.0	1.0	0.5
3	1.0	0.0	0.5
4	1.0	1.0	0.9
5	1.1	1.1	0.1
6	0.8	0.8	0.3
7	2.0	2.0	0.1
8	4.0	4.0	1.0
9	5.0	5.0	2.0
10	4.0	5.0	1.0
11	5.0	4.0	1.0
12	5.1	5.1	0.1
13	4.8	4.8	0.3
14	6.0	6.0	1.0

将这些点呈现在二维图上，如图 2 所示。

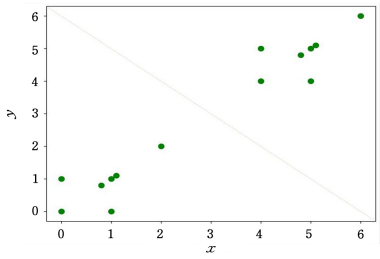
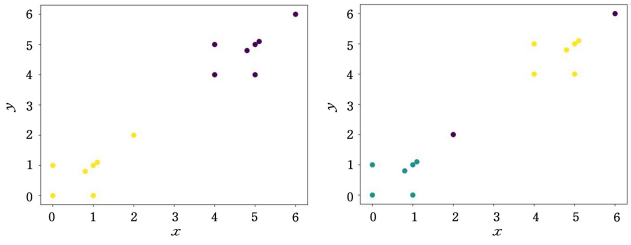


图 2　数据点二维图

Fig. 2　Two-dimensional graph of data points

可以看到，以虚线为界这些点被划分为两群，两群点的相对位置分布是一样的。点与点之间的距离度量选用二维欧几里得距离。若使用 DBSCAN 聚类，此时算法只会考虑点的 X，Y 位置坐标，不会考虑点的重量，两群点的聚类结果会是相同的。如图 3 所示，随着参数 ε 的调整，两群点会同时被聚为各自的一簇，如图 3(a)所示；或同时出现一样数量和位置的噪声，如图 3(b)所示。



(a)无噪声的两个簇　　(b)两个簇各有一个噪声

图 3　DBSCAN 算法聚类结果

Fig. 3　Clustering results of DBSCAN

实际上这两群点只是位置关系一致，并不是所有属性完全一样的点(现实环境中的对象往往如此)，此时 DBSCAN 算法无法有区别地进行聚类划分，而 DN-DBSCAN 算法则不同。此算例将每个点的重量数值作为 DN-DBSCAN 算法中的动态邻域半径，如图 4 所示，两群点就会依据自身的属性不同而出现不一样的聚类结果——点群①被划分为两个簇(每个簇包含 3 个点)和 1 个噪声点；点群②整体归入一个簇内。

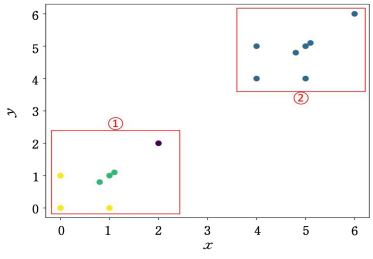


图 4　DN-DBSCAN 算法聚类结果

Fig. 4　Clustering results of DN-DBSCAN

可见，DN-DBSCAN 算法在处理属性差异化样本点的聚类问题时是行之有效的。鉴于此，下文给出了运用 DN-DBSCAN 算法解决实际问题的一个案例。

4　DN-DBSCAN 算法用于城市群的划分

城市群是在城镇化过程中，在特定的城镇化水平较高的地域空间里，以区域网络化组织为纽带，由若干个密集分布的不同等级的城市及其腹地通过空间相互作用而形成的城市-区域系统[7]。城市群的出现是生产力不断发展、生产要素逐步优化组合的产物，每个城市群一般以一个或两个(有少数的城市群是多核心的例外)经济比较发达、具有较强辐射带动功能的中心城市为核心，由若干个空间距离较近、经济联系密切、功能互补、等级有序的周边城市共同组成。发展城市群可在更大范围内实现资源的优化配置，增强辐射带动作用，同时促进城市群内部各城市自身的发展[8-9]。

自城市群的概念被提出以来，国内外研究城市群的专家学者和政府机构越来越多，对城市群的理解也逐渐深入。但对城市群空间范围的界定目前还没有固定的标准和精确的模型可用[10]。Yao 等提出城市群应满足的基本要求：1)有超大或特大城市作为核心城市；2)具有不同性质与规模的一定数量的城市；3)城市间交通运输与经济产业紧密联系[11-12]。Zhou 等提出的中国城市群界定的 5 个标准为：1)有两个人口大于百万的特大城市；2)有规模大、技术先进的海港和空港，并有多条定期国际航空线运营；3)有便利的交通干线作为交通走廊，连接核心城市与腹地区域；4)总人口规模达到 2 500 万以上，人口密度达到每平方公里 700 人以上；5)各城市之间有紧密的社会经济联系[13]。此外 Huang[14] 还提出城市群内各城市要有共同的自然、历史、文化相似性和地域认同感。国外学者 Gottmann[15] 将通过集聚作用形成的城市群称为“大都市连绵区”，并给出其标准：1)区域内有密集的城市；2)核心城市与腹地地区有密切的社会经济联系；3)城市间交通十分便利；4)总人口规模在 2500 万以上；5)具有国际影响力。日本界定大都市区的标准主要有：1)有一个或几个大城市作为核心城市；2)人口大于 3000 万、核心城市的 GDP 占都市区总量的 1/3 以上[14]。以上划分城市群的方式要么是靠定性的描述，要么是定性的描述和定量的指标相结合，操作起来并不容易，且划分出的城市群边界也比较模糊，效果并不理想。散布的城市聚集为城市群，可以看作是一种聚类问题。下文将以划分长三角城市群的过程为例，验证 DN-DBSCAN 聚类算法的有效性。

4.1　动态邻域 ε 的确定

在 DN-DBSCAN 算法中，动态邻域 ε 是样本点综合影响力的体现，决定因素往往有多个，这与现实场景下的真实情境也一致，具体可以通过权重系数法得到。假设影响动态邻域 ε 的多种因素 A，B，C…其数值为 a，b，c…，对应的影响系数则记为 a_i，b_i，c_i…，那么动态邻域 ε 可以由如下公式得到：

$$\varepsilon = a * a_i + b * b_i + c * c_i \cdots \tag{1}$$

其中，影响因素的数值是样本点的属性信息，可以直接从数据集中获得；影响系数则需要在具体的应用环境中综合分析得到，可以采用专家经验法或多因素统计分析法获得。式(1)在城市群划分问题下的具体应用将在下文呈现。

4.2　DN-DBSCAN 算法划分长三角城市群

位于华东地区，邻近长江三角洲的江浙沪皖三省一市是中国经济最活跃的地区之一。这里分布着大大小小的城市，既有上海这种超级大都市，也有诸如池州一类的临江小城。众多的城市之间相互影响，相互融合，慢慢形成城市群。城市群的形成可以看作一个聚类问题，一个城市就是一个样本点，由于每个城市的人口数量、经济实力、交通便利度不尽相同，

所以不能把所有城市看作同质化的数据点，这种情况下，可以使用 DN-DBSCAN 进行聚类来划分出城市群。

首先依据经纬度数据将江浙沪皖三省一市共 41 座城市（江苏省：南京、无锡、常州、苏州、南通、盐城、扬州、镇江、泰州、淮安、宿迁、徐州、连云港；浙江省：杭州、宁波、嘉兴、湖州、绍兴、金华、舟山、台州、丽水、衢州、温州；安徽省：合肥、芜湖、马鞍山、铜陵、安庆、滁州、池州、宣城、六安、黄山、蚌埠、阜阳、亳州、淮南、淮北、宿州；上海市）的位置呈现在二维平面图上，如图 5 所示。正方形点表示国务院批准的《长江三角洲城市群发展规划》中属于长三角城市群的 26 个城市（具体城市如图 6 所示）；三角形点表示不在其中的其他 15 个城市（江苏省：淮安、宿迁、徐州、连云港；浙江省：丽水、衢州、温州；安徽省：六安、黄山、蚌埠、阜阳、亳州、淮南、淮北、宿州）[16]。

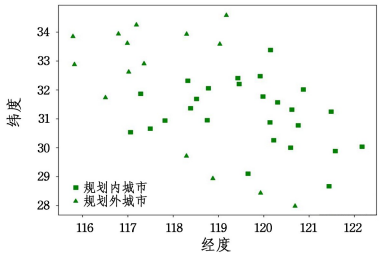


图 5 长三角区域城市分布

Fig. 5 City distribution in Yangtze river delta region

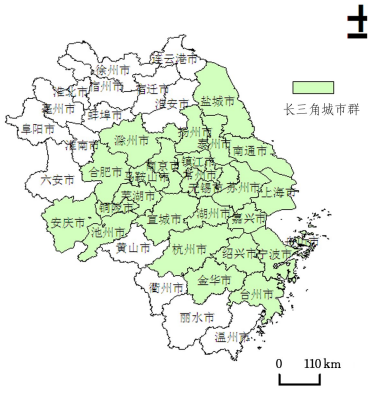


图 6 长三角城市群

Fig. 6 Urban agglomeration in Yangtze river delta

根据经纬度数据，可以计算得到任意两城市间的距离（二维空间欧氏距离）。在 DN-DBSCAN 算法运行过程中，每个样本点的邻域半径由该城市的“吸引半径”r 决定（r 即为动态邻域半径 ε），r 由城市的人口数量 p、GDP（下文中用 g 表示）以及地理优势值 s 这 3 部分决定。本文选用的是第 7 次全国人口普查的人口数据[17]和 2020 年的 GDP 数据[18]，均来自于国家统计局。城市的地理优势值由其地理位置的优劣决定，沿江靠海的城市因交通便利而更具有吸引力，地理优势值打分为 50；内陆城市相对闭塞，地理优势值打分为 5。在计算城市吸引半径 r 时，3 个因素（p，g，a）对应的系数分别为人口系数 p_i = 0.1（公里/万人）；经济系数 g_i = 0.001（公里/亿元）；地理系数 s_i = 1（公里）。由式(1)可知吸引半径 r 的具体计算式为：

$$r = p * p_i + g * g_i + s * s_i \tag{2}$$

将各城市数据输入式(2)计算可得各自的吸引半径，如表 2 所列。

表 2 城市吸引半径

Table 2 City attraction radius

城市	人口/万	GDP/亿	地理优势值	吸引半径/km
上海	2487.09	38700.58	50	337.40958
合肥	936.99	10045.72	5	108.74472
安庆	416.53	2467.70	50	94.12070
蚌埠	329.64	2082.73	5	40.04673
亳州	499.68	1806.01	5	56.77401
池州	134.28	868.90	50	64.29690
滁州	398.71	3032.10	5	47.90310
阜阳	820.03	2805.20	5	89.80820
淮北	197.03	1119.10	5	25.82210
淮南	303.35	1337.20	5	36.67220
黄山	133.06	850.40	5	19.15640
六安	439.37	1669.50	5	50.60650
马鞍山	215.99	2186.90	50	73.78590
宿州	532.45	1978.75	5	60.22375
铜陵	131.17	1003.70	50	64.12070
芜湖	364.44	3753.02	50	90.19702
宣城	250.01	1607.50	5	31.60850
南京	931.47	14817.95	50	157.96495
常州	527.81	7805.30	50	110.58630
淮安	455.62	4025.37	5	54.58737
连云港	459.94	3277.07	50	99.27107
南通	772.66	10036.31	50	137.30231
苏州	1274.83	20170.45	50	197.65345
宿迁	498.62	3262.37	5	58.12437
泰州	451.28	5312.77	50	100.44077
无锡	746.21	12370.48	50	136.99148
徐州	908.38	7319.77	5	103.15777
盐城	670.96	5953.38	50	123.04938
扬州	455.98	6048.33	50	101.64633
镇江	321.04	4220.09	50	86.32409
杭州	1193.6	16106.00	50	185.46600
湖州	336.76	3201.40	5	41.87740
嘉兴	540.09	5509.52	5	64.51852
金华	705.07	4703.95	5	80.21095
丽水	250.74	1540.02	5	31.61402
宁波	940.43	12408.70	50	156.45170
衢州	227.62	1639.12	5	29.40112
绍兴	527.1	6001.00	5	63.71100
台州	662.29	5262.70	50	121.49170
温州	957.29	6870.90	50	152.59990
舟山	115.78	1512.11	50	63.09011

DN-DBSCAN 算法中，参数 $MinPts$ 的确定依然沿用 DBSCAN 算法中的启发式方法，即 $MinPts \approx \ln n$[19]，这里 n 是数据集包含的样本数量。本案例中城市总数量 $n = 41$，那么计算可得一个城市群至少应包含的城市数目 $MinPts = 4$。用 DN-DBSCAN 算法进行城市群聚类分析时，算法的输入数据集为 41 座城市的经纬度和表 2 中各城市的吸引半径 r，一个参数 $MinPts = 4$。聚类结果如图 7 所示。

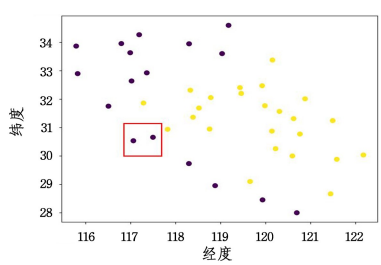


图 7 城市群聚类结果（电子版为彩图）

Fig. 7 Clustering result of urban agglomerations

图中 24 个黄色点为一簇，对应的城市分别为上海、南京、

无锡、常州、苏州、南通、盐城、扬州、镇江、泰州、杭州、宁波、嘉兴、湖州、绍兴、金华、舟山、台州、合肥、芜湖、马鞍山、铜陵、滁州、宣城；紫色点为 17 个噪声点，对应的城市分别为淮安、宿迁、徐州、连云港、丽水、衢州、温州、六安、黄山、蚌埠、阜阳、亳州、淮南、淮北、宿州、安庆、池州。对比图 7 和图 5 可以发现，黄色的点和正方形的点是基本重合的，紫色的点和三角形的点是基本重合的。具体来说只有安庆、池州（图 7 红框处）两个事实上属于长三角城市群的城市没有被归入簇内，对照图 6 可以发现这两座城市已处于长三角城市群沿长江向内陆延展的末端，位置的特殊性导致其未被归入簇内。至此，DN-DBSCAN 算法发现的类簇准确识别了 26 个长三角城市群中的 24 个城市，且没有把不相关的城市划入簇内。用 Acc 表示算法划分长三角城市群的准确率；q 表示样本点总数；w 表示被误判的样本点数（包括属于长三角城市群内的城市样本点没有被划入簇内和不属于长三角城市群内的城市样本点被划入簇内这两种情况）。则准确率的计算公式如下：

$$Acc=\frac{q-w}{q} \tag{3}$$

案例中总的城市数量 $q=41$，误判的样本点数 $w=2$（安庆和池州本属于长三角城市群的城市没有被划入簇内），计算可得 $Acc=95\%$。可见 DN-DBSCAN 算法对长三角城市群的聚类效果是很好的，具有优良的解决现实问题的能力。

4.3　对比 DBSCAN 算法的划分结果

为进一步证明 DN-DBSCAN 算法的优良性能，现使用 DBSCAN 算法进行长三角城市群的划分，对比两种算法划分结果的准确性。DBSCAN 算法中第一个参数 $MinPts$ 同 DN-DBSCAN 算法一样取 4，另一个参数邻域半径 ε 是需要预先给定的固定值，参照表 2 中 DN-DBSCAN 算法的动态邻域半径 ε 取多个值对比分析聚类的效果。将表 2 中各城市的吸引半径值倒序排列，分别取其首位数、1/4 分位数、中位数、3/4 分位数和末位数作为 DBSCAN 算法中的邻域半径 ε，得到表 3 中的 5 种参数组合。

表 3　参数组合

Table 3　Parameter combination

组合	$MinPts$	邻域半径 ε	备注
1	4	337.4096	首位数
2	4	110.5863	1/4 分位数
3	4	80.2109	中位数
4	4	54.5874	3/4 分位数
5	4	19.1564	末位数

将数据集（41 座城市对应的经纬度）和表 3 中的参数组合输入 DBSCAN 算法，可以得到如图 8 所示的结果，黄色点簇表示算法划分出的长三角城市群，紫色噪声点表示没有被划入长三角城市群的其余城市。其中对照组 1 是理想状态的聚类结果（黄色点类簇正好包含所有长三角城市群内的城市），对照组 2 是 DN-DBSCAN 算法的聚类结果（见图 7），参数组合 1-5 是 DBSCAN 算法在不同参数组合（见表 3）下的聚类结果。

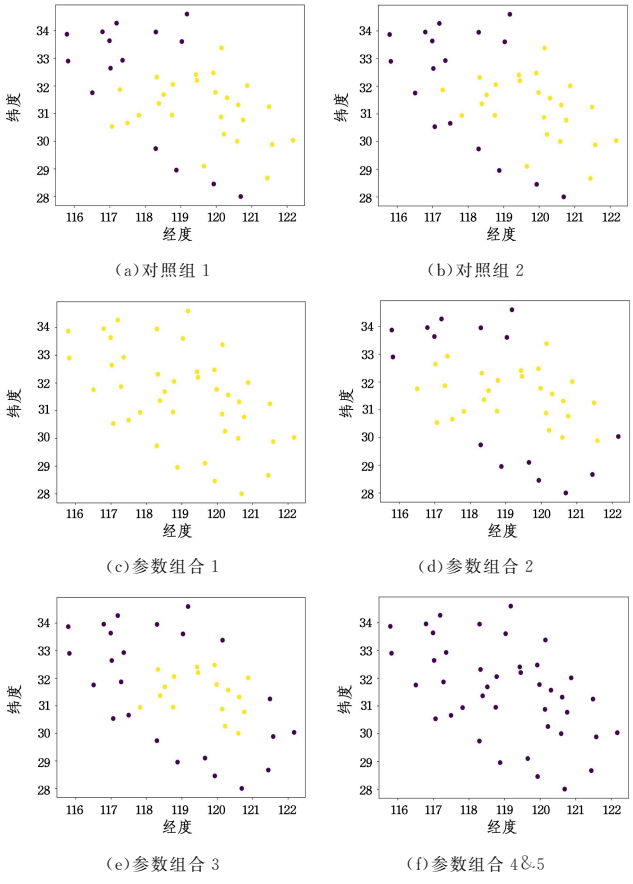


图 8　聚类结果对比（电子版为彩图）

Fig. 8　Comparison of clustering results

按照上文中聚类结果准确率的计算式（3），统计了图 8 中各组别的聚类结果准确率，如表 4 所列。可以看到 DN-DBSCAN 算法对长三角城市群划分的准确率远高于 DBSCAN 算法，即使选取多种不同的参数组合也是如此。因为问题的本质就已经决定了无法通过一个固定的邻域半径 ε 来聚类划分出符合现实场景的结果，必须依据样本点属性的不同而动态地确定聚类算法的邻域半径 ε，才能体现出属性差异化样本点对群体集聚效应不同的影响力，这正是现实情景的真实反映。通过此部分对比，证明了 DN-DBSCAN 算法在解决现实问题时相比传统 DBSCAN 算法有更好的效果。

表 4　DBSCAN 算法聚类结果准确率

Table 4　Accuracy of DBSCAN algorithm clustering results

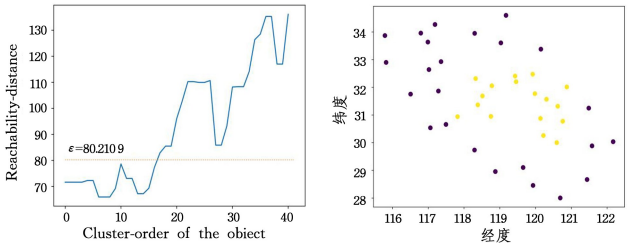
组别	对照组 1	对照组 2	参数组合 1	参数组合 2	参数组合 3	参数组合 4	参数组合 5
q	41	41	41	41	41	41	41
w	0	2	15	6	9	26	26
准确率/%	100	95	63	85	78	37	37

4.4　对比 OPTICS 算法的划分结果

OPTICS（Ordering Points to Identify the Clustering Structure）算法是对 DBSCAN 的一个扩展算法，因此 OPTICS 算法也是一种基于密度的聚类算法。OPTICS 算法不显式地生成数据聚类，而是通过计算样本点的核心距离和可达距离对数据集合中的对象进行排序，得到一个有序的对象列表，这个排序列表代表了各样本点基于密度的聚类结构。它包含的信息等价于从一个广泛的参数设置所获得的基于密度的

聚类，换句话说，从这个排序中可以得到基于任何参数 ε 和 *MinPts* 的聚类结果。

OPTICS 算法的输入参数同 DBSCAN 一样为 ε 和 *MinPts*，输出结果为点序列可达距离图以及最后的聚类结果。这里以表 3 中参数组合 3 为例，具体展示在邻域半径 ε=80.2109，*MinPts*=4 时 OPTICS 算法的聚类结果，如图 9 所示。从左侧点序列可达距离图中可以看出此时只有一个类簇，对应右侧城市群划分结果图中的黄色点簇，其余紫色点则为噪声。



(a)点序列可达距离图 (b)城市群划分结果图

图 9 OPTICS 算法聚类结果(电子版为彩图)

Fig. 9 Clustering results of OPTICS algorithm

将数据集以及表 3 中的参数组合依次带入 OPTICS 算法中运行，结果如表 5 所列。对比表 4，可以发现在参数组合相同的情况下，OPTICS 算法的聚类结果准确率和 DBSCAN 算法是一样的，都远低于本文提出的 DN-DBSCAN 算法。首先，相较于 DBSCAN 算法，OPTICS 算法的优良之处在于可以有效解决密度差异大的数据集聚类效果不好的问题，但是从图 5 可以看出城市样本点的分布并没有明显的密度差异。其次，相较于 DN-DBSCAN 算法，OPTICS 算法依然同 DBSCAN 算法一样，需要输入固定的邻域半径 ε 值。以上两点可以从理论上解释 OPTICS 算法在此问题上表现不佳的原因。

表 5 OPTICS 算法聚类结果准确率

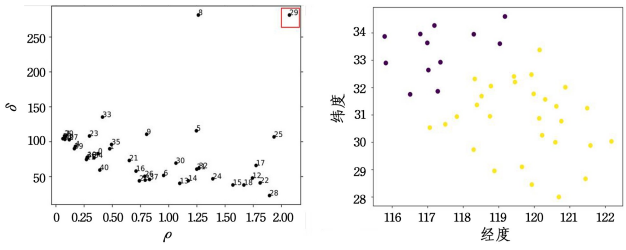
Table 5 Accuracy of OPTICS algorithm clustering results

组别	参数组合 1	参数组合 2	参数组合 3	参数组合 4	参数组合 5
q	41	41	41	41	41
w	15	6	9	26	26
准确率/%	63	85	78	37	37

4.5 对比 DPC 算法的划分结果

密度峰值聚类算法全称为基于快速搜索和发现密度峰值的聚类算法(clustering by fast search and find of density peaks，DPC)。基于密度的聚类方法的主要思想是寻找被低密度区域分离的高密度区域。同样的，DPC 算法也基于这样的假设：1)聚类中心点的密度大于周围邻居点的密度；2)聚类中心点与更高密度点之间的距离相对较大。因此在整个的算法运行过程中对每个样本点计算两个参数：1)样本点的局部密度 ρ；2)样本点到密度比其大的点的最小距离 δ，两者都大的点即是聚类中心。

将 41 座城市经纬度数据集代入 DPC 算法中运行，结果如图 10 所示。从左侧点密度-最小距离图中可以发现 ρ 和 δ 两个参数都最大的点为 29 号点，则以其为聚类中心得到右侧城市群划分结果图，黄色点为一个类簇，紫色点为噪声。依据式(3)，此时 $q=41$，$w=5$，计算得到准确率 $Acc=88\%$。

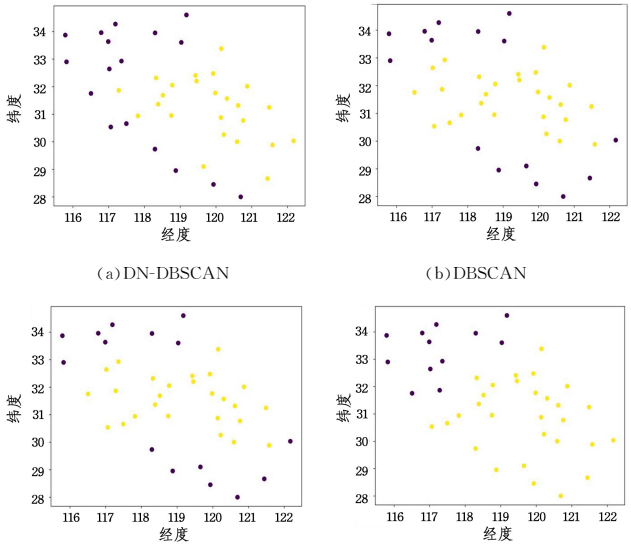


(a)密度-最小距离图 (b)城市群划分结果图

图 10 OPTICS 算法聚类结果(电子版为彩图)

Fig. 10 Clustering results of OPTICS algorithm

综上，将 DN-DBSCAN，DBSCAN，OPTICS 以及 DPC 这 4 种密度聚类算法针对城市群划分问题的聚类结果和结果准确率进行比较得到图 11 和表 6，其中 DBSCAN 和 OPTICS 算法选取的是其最佳参数组合下的结果。对比结果表明，4 种密度聚类算法中 DN-DBSCAN 效果最好，划分出的长三角城市群准确率最高。



(a)DN-DBSCAN (b)DBSCAN

(c)OPTICS (d)DPC

图 11 4 种密度聚类算法结果对比

Fig. 11 Results comparison of four density clustering algorithms

表 6 4 种算法结果准确率

Table 6 Accuracy of four algorithms clustering results

算法名称	DN-DBSCAN	DBSCAN	OPTICS	DPC
q	41	41	41	41
w	2	6	6	5
准确率/%	95	85	85	88

结束语 本文提出了一种基于 DBSCAN 算法的改进密度聚类算法 DN-DBSCAN，以实现对包含非同质化样本点的数据集进行更有效的密度聚类。相较于 DBSCAN 算法，DN-DBSCAN 算法能够根据样本点属性的不同而动态地调节邻域半径 ε，而不是将其当作一个固定的参数输入，这样做的优势可以由本文提出的算法来体现——在 DBSCAN 算法无法对属性不同的点群进行差异化聚类时，DN-DBSCAN 算法可以凭借动态邻域半径 ε 呈现出更合理、更符合现实场景的聚类效果。本文将 DN-DBSCAN 算法应用于城市群划分问题，通过对比 DBSCAN 算法、OPTICS 算法和 DPC 算法针对

相同问题的聚类结果，证明了 DN-DBSCAN 算法具有更好的效果，划分出的长三角城市群的准确率远高于其他 3 种经典密度聚类算法。同时，对于 DBSCAN 算法的另一个全局参数，即每一核心点邻域内应包含的最小点数 *MinPts*，能否像本文所述的邻域半径 ε 一样由点自身的属性确定，这有待未来进一步的研究探索。

参 考 文 献

[1] BEHARA K N S, BHASKAR A, CHUNG E. A DBSCAN-based framework to mine travel patterns from origin-destination matrices: Proof-of-concept on proxy static OD from Brisbane[J]. Transportation Research Part C: Emerging Technologies, 2021, 131: 103370.

[2] CAI Y K, XIE K Q, MA X J. An Improved DBSCAN Algorithm which is Insensitive to Input Parameters[J]. Acta Scientiarum Naturalium Universitatis Pekinensis, 2004, 40(3): 480-486.

[3] FENG Z H, QIAN X Z, ZHAO N N. Greedy DBSCAN: an improved DBSCAN algorithm on multi-density clustering[J]. Application Research of Computers, 2016, 33(9): 2693-2696, 2700.

[4] CHEN X H, XI Q G. Research and Implementation of Adaptive Clustering Algorithm based on DBSCAN[J]. Journal of Huaiyin Teachers College (Natural Science Edition), 2021, 20(3): 228-234.

[5] ZHOU H, WANG P, LI H Y. Research on Adaptive Parameters Determination in DBSCAN Algorithm[J]. Journal of Information & Computational Science, 2012, 9(7): 1967-1973.

[6] YUE S H, LI P, GUO J D, et al. A statistical information-based clustering approach in distance space[J]. Journal of Zhejiang University Science, 2005, 6(1): 71-78.

[7] WANG R M. Urban agglomeration development and housing demand: a literature review[J]. Shanghai Real Estate, 2021(9): 8-12.

[8] YU W X. Opportunities and challenges of the development of urban agglomerations empowered by technology[J]. Governance, 2021, (31): 25-29.

[9] WANG W, ZHU X C, WANG Y. Evolution and knowledge map analysis of Urban agglomeration research in China[J]. Beijing Planning Review, 2020(3): 74-79.

[10] XIAO J C. The Developing Stage of and Function Orientation of Ten Chinese Urban Cluster[J]. Reform, 2009(9): 5-23.

[11] YAO S M. Urban agglomeration in China[M]. Hefei: University of Science and Technology of China Press, 2001.

[12] YAO S M, ZHOU C S, WANG D. New theory of Urban agglomeration in China[M]. Beijing: Science Press, 2016.

[13] ZHOU Y X, XU X Q. Urban geography (2th ed)[M]. Beijing: Beijing Higher Education Press, 2009.

[14] HUANG Z X. Study on the standard of urban agglomeration definition[J]. Inquiry into Economic Issues, 2014(8): 156-164.

[15] GOTTMANN J. Megalopolis or the Urbanization of the Northeastern Seaboard[J]. Economic Geography, 2016, 33(3): 189-200.

[16] ZHANG J. Interpretation of The Development Plan of Yangtze River Delta Urban Agglomeration[J]. Education of Geography, 2017(2): 62-63.

[17] Office of the Seventh National Census Leading Group of The State Council. Key data from the seventh National Census in 2020[M]. Beijing: China Statistics Press, 2021.

[18] National Bureau of Statistics. GDP data by region in 2020[EB/OL]. (2021-01-29)[2022-01-15]. http://www.stats.gov.cn/tjsj/.

[19] BIRANT D, KUT A. ST-DBSCAN: An algorithm for clustering spatial-temporal data[J]. Data & Knowledge Engineering, 2007, 60(1): 208-221.

ZHANG Peng, born in 1991, postgraduate. His main research interests include data analysis and processing.

LI Xiaolin, born in 1986, Ph.D, professor. His main research interests include enterprise management informatization and system integration.