【工程与技术研究】

基于 *t*-SNE 降维的密度峰值聚类算法

何婷霭，李　秦

（兰州交通大学 数理学院，甘肃 兰州 730070）

摘　要：为了提高密度峰值聚类（DPC）算法处理复杂高维数据的能力，提出了一种基于 *t*-SNE 降维的密度峰值聚类算法（*t*-SNE-DPC）。该算法用 *t*-SNE 算法对数据进行预处理，将高维数据点间的关系用概率分布映射到低维空间中，通过最小化相对熵最大化保留数据的本质特征，使用密度峰值聚类算法进行聚类操作。仿真实验结果表明，*t*-SNE-DPC 可以高效地对高维数据进行聚类，在 *AMI* 指标上的聚类结果可达 0.828。

关键词：聚类分析；密度峰值聚类；*t*-SNE 算法；有效性度量

中图分类号：TP 181　　**文献标识码**：A　　DOI：10.13486/j.cnki.1673－2618.2023.02.014

聚类分析是数据挖掘技术的基础与核心，它能够在无监督的条件下探索数据背后潜在的关系。依据原理的不同，将现有的聚类分为 5 类[1-5]：划分聚类、层次聚类、网格聚类、基于密度的聚类和模型聚类，每种聚类方法都有其独特的优势。

密度峰值聚类（Density Peaks Clustering，DPC）算法[6]是 2014 年由意大利学者 Alex Rodriguez 和 Alessandro Laio 提出的，该算法不仅简单易懂、参数少，而且不需要迭代，能够对任意形状的数据集进行高效聚类。正是基于这些优势，DPC 算法被广泛应用于机器学习、模式识别和图像处理等多个领域。但该算法也有不足之处：高维数据集聚类效果不佳；算法中的唯一参数——截断距离需人工选取，对聚类结果影响较大；不适用于大规模数据的聚类分析。

在对高维数据进行聚类研究时发现，进行降维操作可以减少数据冗余，提高聚类效率。主成分分析（Principal Component Analysis，PCA）[7]可以去除部分噪声并发现数据中的部分数据结构，但对于非线性数据，并不能很好地发现数据的隐含信息；线性判别分析（Linear Discriminant Analysis，LDA）[8]是一种基于监督学习的数据降维方法，但可能过度拟合数据；等度量映射（Isometric Mapping，Isomap）[9]使用测地线距离计算数据点间的距离，但对噪声敏感且它的拓扑结构不稳定。基于上述表述，将 *t* -分布随机近邻嵌入（*t*-distributed Stochastic Neighbor Embedding，*t*-SNE）这一降维方法引入 DPC 算法中，提出了一种基于 *t*-SNE 降维的密度峰值聚类算法（*t*-SNE-DPC）。*t*-SNE-DPC 将高维数据点通过概率分布映射到低维空间中，随后用传统的 DPC 算法将其进行聚类，通过实验验证，*t*-SNE-DPC 对高维数据有很强的实用性。

1　DPC 算法

密度峰值聚类算法是基于密度的聚类算法，它的核心在于对聚类中心的刻画。该算法适用于聚类中

收稿日期：2022－04－12

基金项目：国家自然科学基金地区科学基金项目（11262009）

第一作者简介：何婷霭（1997—），女，甘肃白银人，硕士研究生，主要从事机器学习研究。E-mail：h2398385335@163.com

心点的局部密度较高，且比其密度高的点之间的距离相对较远的情况。假设有数据集合 $S=\{x_1,x_2,\cdots,x_n\}$，$I_s=\{1,2,\cdots,n\}$是其指标集，$d_{ij}=\mathrm{dist}(x_i,x_j)$是数据点 x_i 与 x_j 之间的欧氏距离[10]。定义局部密度 ρ_i 为

$$\rho_i=\sum_{j\neq i}\chi(d_{ij}-d_c),\chi(x)=\begin{cases}1,x<0,\\0,x\geqslant 0。\end{cases} \tag{1}$$

其中，d_c 是截断距离，需要人为指定，一般选取整个数据集的 1%～2%作为截断阈值。$\chi(x)$是指示函数。定义相对距离 δ_i 为

$$\delta_i=\min_{j:\rho_j>\rho_i}(d_{ij}), \tag{2}$$

当数据点 x_i 的局部密度达到最大时，相对距离变为 $\delta_i=\max_j(d_{ij})$。

局部密度和相对距离确定完成后，通过绘制决策图选取聚类中心，一般选择决策图右上方的点，这些点既有较高的局部密度，也有较大的相对距离。最后采用一步分配策略进行非聚类中心点的分配：若数据点 x_j 不是中心点，则将其归入密度比 x_j 大且距离 x_j 最近的数据点 x_i 所在的类，该过程只执行一次，不用进行迭代更新[10]。DPC 算法的具体步骤如下：(Ⅰ)确定截断参数 d_c，根据公式(1)计算每个数据点的局部密度 ρ_i；(Ⅱ)根据公式(2)计算每个数据点的相对距离 δ_i；(Ⅲ)以 ρ 为横坐标，δ 为纵坐标绘制决策图；(Ⅳ)根据决策图选取聚类中心，分配非聚类中心数据点，完成聚类。

2 *t*-SNE 算法

t-SNE 算法是基于流形学习的非线性降维方法。文献[11]提出了随机近邻嵌入(Stochastic Neighbor Embedding，SNE)算法，文献[12]在 SNE 算法的基础上进行了优化改进，提出了 *t*－SNE 算法[12]。*t*-SNE 算法在低维空间中用 *t*－分布代替原来的高斯分布表示两点间的相似度，独有的长尾特征有效地解决了 SNE 算法存在的拥挤问题。

假设：高维数据点集合为 $X=\{x_1,x_2,\cdots,x_n\}$，映射到低维空间后的数据点集合为 $Y=\{y_1,y_2,\cdots,y_n\}$；$\boldsymbol{P}$ 与 $\boldsymbol{Q}$ 均为 $n\times n$ 矩阵，分别表示高维空间和低维空间中的概率分布，p_{ij} 与 q_{ij} 是其各自的矩阵元素[13]。在 *t*-SNE 方法中，联合概率分布 p_{ij} 是对称条件概率

$$p_{ij}=\frac{p_{j|i}+p_{i|j}}{2n}, \tag{3}$$

$p_{j|i}$是点 i 相对于点 j 的概率分布

$$p_{j|i}=\frac{\exp(-\frac{\|x_i-x_j\|^2}{2\sigma_i^2})}{\sum_{k\neq j}\exp(-\frac{\|x_i-x_k\|^2}{2\sigma_i^2})}。 \tag{4}$$

其中，σ_i 是以 x_i 为中心的高斯分布的方差，它反映了中心 x_i 周围的样本分布密度，此参数的值由复杂度因子通过执行二元搜索来确定，复杂度因子定义为

$$\mathrm{perp}(p_i)=2^{H(p_i)}, \tag{5}$$

$H(p_i)$ 是分布 p_i 的熵

$$H(p_i)=-\sum_j p_{j|i}\log_2 p_{j|i}。 \tag{6}$$

定义低维空间中 y_i 与 y_j 之间的联合概率分布为

$$q_{ij}=\frac{(1+\|y_i-y_j\|^2)^{-1}}{\sum_{k\neq l}(1+\|y_k-y_l\|^2)^{-1}}。 \tag{7}$$

设两个分布间的相对熵为代价函数 C，采用梯度下降法来最小化 C，其计算过程为

$$C = KL(P \parallel Q) = \sum_i \sum_{j \neq i} p_{ij} \log \frac{p_{ij}}{q_{ij}}, \tag{8}$$

$$\frac{\delta C}{\delta y_i} = 4 \sum_j (p_{ij} - q_{ij})(y_i - y_j)(1 + \parallel y_i - y_j \parallel^2)^{-1}。 \tag{9}$$

为了避免优化过程陷入局部最优，在进行梯度计算时加入一个相对较大的动量项 $\alpha(t)$，则此时梯度更新为

$$Y^{(t)} = Y^{(t-1)} + \eta \frac{\delta C}{\delta Y} + \alpha(t)(Y^{(t-1)} - Y^{(t-2)}), \tag{10}$$

其中，$Y^{(t)}$ 表示第 t 次的迭代解，η 表示学习率。

综上所述，t-SNE 算法的主要步骤如下：(Ⅰ)根据式(5)和式(6)计算复杂度因子 perp(p_i)；(Ⅱ)设置参数迭代次数 T，学习率 η，动量项 $\alpha(t)$；(Ⅲ)根据式(3)和式(4)计算 p_{ij}，$p_{j|i}$，并得到初始化结果 $Y^{(0)} = \{y_1, y_2, \cdots, y_n\}$；(Ⅳ)从 $t=1$ 到 T 进行迭代更新，根据式(7)计算 q_{ij}，根据式(8)和式(9)计算梯度，根据式(10)进行梯度更新。

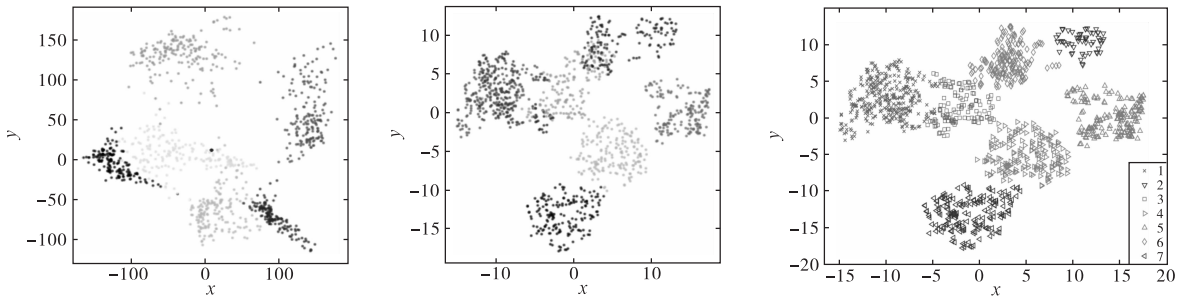
3　t-SNE-DPC

如前所述，密度峰值聚类算法在聚类高维数据集时效果不佳，基于此将 t-SNE 算法引入 DPC 算法中，需要注意的是，在进行绘制决策图选取聚类中心时，横坐标采用公式 $\gamma_i = \rho_i \cdot \delta_i$ 进行计算更新。故基于 t-SNE 的 t-SNE-DPC 设计思想如下：把待聚类的高维数据用 t-SNE 算法进行降维处理；将降维后的数据使用密度峰值聚类算法进行聚类。t-SNE-DPC 算法具体步骤如下：(Ⅰ)利用 t-SNE 算法进行高维数据预处理；(Ⅱ)确定截断距离 d_c，根据降维后得到的数据利用式(1)和式(2)计算局部密度 ρ_i 和相对距离 δ_i；(Ⅲ)根据 $\gamma_i = \rho_i \cdot \delta_i$ 绘制决策图，选取聚类中心；(Ⅳ)分配其他数据点，得到聚类结果。

4　实验验证及结果分析

4.1　实验验证

为验证提出的聚类算法 t-SNE-DPC，采用结构化数据的经典数据集——手写数字数据集 MNIST 来进行仿真实验。此次实验使用 MNIST 手写数字数据集的 train 训练集，选取手写数字 0～6，共计 1264 个数据样本，且每个样本都具有 64 个特征。图 1 为 MNIST 数据集在 Isomap、t-SNE 和 t-SNE-DPC 下的聚类可视化结果。



(a)Isomap　　(b)t-SNE　　(c)t-SNE-DPC

图 1　MNIST 数据集在三种降维算法下的聚类可视化结果

从图 1(a)(b)可以看出，Isomap 对高维数据不能进行很好的聚类，应属于不同类簇的点错误地归类到了同一类簇；t-SNE 方法相对来说效果好了很多，类簇与类簇间的分离效果可以明显肉眼识别，但仍然出现了同一类簇中混杂着其他数据点的情况。图 1(c)是使用新算法 t-SNE-DPC 进行的聚类效果图，为了能够更直观地看出其效果，采用不同的记号来刻画每一个类簇。从图中可以很明显看出，t-SNE-DPC 可以准确确定聚类个数且聚类效果很好。

4.2 算法性能评估

为验证算法的有效性，采用三种外部评价指标：准确率、调整互信息[14]和调整兰德指数[15]来进行算法性能评估。

记数据集 $X=\{x_1,x_2,\cdots,x_n\}$的真实类标记为 $U\{U_1,U_2,\cdots,U_u\}$，实验得到的类标记为 $V\{V_1,V_2,\cdots,V_v\}$。

定义 1（准确率 ACC） 正确聚类的数据样本个数 N_1 占数据样本总数 N 的比率 $ACC=N_1/N$。

定义 2（调整互信息 AMI）[14] 设函数 $F(x_1,x_2)$为 max 函数，则有

$$AMI=[MI(U,V)-E\{MI(U,V)\}]/[\max\{H(U),H(V)\}-E\{MI(U,V)\}],$$

其中，$MI(U,V)$是互信息，E 是期望，$H(U)$与 $H(V)$是熵。

定义 3（调整兰德指数 ARI）[15] 设 α 为同属于 U 与 V 的数据对个数；b 为在 U 中属在同类，在 V 中属不同类的数据对个数；c 为在 U 中属不同类，在 V 中属同类的数据对个数；d 为在 U 与 V 中都属不同类的数据对个数，则有

$$ARI=[2(ad-bc)]/[(a+b)(b+d)+(a+c)(c+d)]。$$

表 1 是 DPC 算法[6]、PCA-DPC 算法[16]以及本文所提出的 t-SNE-DPC 算法分别在三个聚类指标下对 MNIST 数据集进行的性能对比结果。从表中数据可以看出，t-SNE-DPC 的聚类效果远高于其他算法，在三个聚类指标上均有很大的提升，尤其在 AMI 指标上逼近于 1，聚类性能很好。

表 1 三种聚类算法在三个聚类指标上的计算结果

算法	ACC	AMI	ARI
DPC	0.355	0.273	0.163
PCA-DPC	0.249	0.151	0.039
t-SNE-DPC	0.744	0.828	0.715

5 结语

针对密度峰值聚类算法对高维数据聚类效果不理想这一问题，提出了一种基于 t-SNE 降维的密度峰值聚类算法 t-SNE-DPC，此算法把数据间的度量方式改用概率分布来表示，通过最小化相对熵将高维空间的数据映射到低维空间，再而进行聚类操作。最后将算法应用到 MNIST 数据集上进行实验，并对算法进行了有效性度量，实验结果表明新算法可以对高维复杂数据进行高效聚类，且在 AMI 指标上结果逼近于 1，证实了新算法的有效性与可用性。

参 考 文 献：

[1] HAN J，PEI J，TONG H. Data mining：concepts and techniques[M]. San Francisco：Morgan Kaufmann Publishers Inc，2022.

[2] XU R，WUNSCH D. Survey of clustering algorithms[J]. IEEE Transactions on neural networks，2005，16(3)：645－678.

[3] 段明秀. 层次聚类算法的研究及应用[D]. 长沙：中南大学，2009.

[4] XU D，TIAN Y. A comprehensive survey of clustering algorithms[J]. Annals of data science，2015，2(2)：165－193.

[5] 张敏，于剑. 基于划分的模糊聚类算法[J]. 软件学报，2004，15(6)：858－868.

[6] RODRIGUEZ A，LAIO A. Clustering by fast search and find of density peaks[J]. Science，2014，344(6191)：1492－1496.

[7] WOLD S，ESBENSEN K，GELADI P. Principal component analysis[J]. Chemometrics and intelligent laboratory systems，1987，2(1)：37－52.

[8] BALAKRISHNAMA S，GANAPATHIRAJU A. Linear discriminant analysis：a brief tutorial[J]. Institute for signal and information processing，1998，18：1－8.

[9] TENENBAUM J B，SILVA V，LANGFORD J C. A global geometric framework for nonlinear dimensionality reduction[J]. Science，2000，290(5500)：2319－2323.

[10] 陈俊芬，张明，赵佳成. 复杂高维数据的密度峰值快速搜索聚类算法[J]. 计算机科学，2020，47(3)：79－86.

[11] HINTON G E，ROWEIS S. Stochastic neighbor embedding[J]. Advances in neural information processing systems，2002(15)：857－864.

[12] VAN DER MAATEN L，HINTON G. Visualizing data using *t*-SNE[J]. Journal of machine learning research，2008，9(11)：2579－2605.

[13] 徐秀芳，徐森，花小朋，等. 一种基于 *t*-分布随机近邻嵌入的文本聚类方法[J]. 南京大学学报(自然科学版)，2019，55(2)：264－271.

[14] AMELIO A，PIZZUTI C. Correction for closeness：adjusting normalized mutual information measure for clustering comparison[J]. Computational intelligence，2017，33(3)：579－601.

[15] STEINLEY D. Properties of the Hubert-Arable adjusted rand index[J]. Psychological methods，2004，9(3)：386－396.

[16] 杜明晶. 密度峰值聚类算法研究[D]. 徐州：中国矿业大学，2018.

Density Peak Clustering Algorithm Based on *t*-SNE Dimensionality Reduction

HE Ting-ai，LI Qin

(*School of Mathematics and Physics*，*Lanzhou Jiaotong University*，*Lanzhou* 730070，*China*)

Abstract：In order to improve the ability of Density Peak Clustering (DPC) algorithm to deal with complex high-dimensional data，a density peak clustering algorithm is proposed based on *t*-SNE dimensionality reduction (*t*-SNE-DPC). The algorithm uses the *t*-SNE algorithm to pre-process the data，maps the relationship between high-dimensional data points to the low-dimensional space with probability distribution，maximizes the retention of the essential characteristics of the data by minimizing the relative entropy，and finally uses the density peak clustering. The algorithm performs clustering operations. The simulation results show that *t*-SNE-DPC can efficiently cluster high-dimensional data，and the clustering results of *t*-SNE-DPC on the *AMI* index can be as high as 0.828.

Keywords：cluster analysis；density peak clustering；*t*-SNE algorithm；effectiveness measurement

(责任编辑：贾晶晶)