第53卷第5期 南京航空航天大学学报 Vol. 53 No. 5
2021年10月 Journal of Nanjing University of Aeronautics & Astronautics Oct. 2021

DOI:10.16356/j.1005-2615.2021.05.015

峰值点非负矩阵分解聚类算法

徐晓华,方 威,何 萍,仁 祥,姜玉麟,葛方毅
(扬州大学信息工程学院,扬州 225000)

摘要: 非负矩阵分解模型是一种常见的数据降维方法。在现有非负矩阵分解算法用于聚类的研究中,每个类别一般仅由一个或者指定多个中心点表示,然而这种表示方式往往无法准确描述其类别的特征和结构,从而影响聚类效果。为了解决这个问题,本文提出了峰值点非负矩阵分解算法。该算法首先为数据集找到多个密度峰值点,并构建密度峰值点和样本点的二部图,然后利用二部图完成聚类。此外该算法引入流形图正则化项来充分利用数据间的流形结构信息,并给出了算法的迭代更新规则。在大量真实数据集上的实验结果表明,该方法可以更加有效地利用数据本身的结构信息,从而提高聚类效果。

关键词: 非负矩阵分解;降维;密度峰值;图正则;聚类分析

中图分类号: TP391.4 **文献标志码:** A **文章编号:** 1005-2615(2021)05-0772-08

Clustering Algorithm for Peaks Non-negative Matrix Factorization

XU Xiaohua, FANG Wei, HE Ping, REN Xiang, JIANG Yulin, GE Fangyi
(College of Information Engineering, Yangzhou University, Yangzhou 225000, China)

Abstract: The non-negative matrix factorization model is a common data dimensionality reduction method. In the existing non-negative matrix factorization algorithm for clustering research, each category is generally represented by only one or more designated center points. However, this type of representation often fails to accurately describe the characteristics and structure of its category, which affects the clustering performance. In order to solve this problem, we proposed the peaks non-negative matrix factorization (PNMF) algorithm. The algorithm first finds multiple density peak points for the dataset, constructs a bipartite graph of the density peak points and sample points, then uses the bipartite graph to complete the clustering. In addition, the algorithm introduces a manifold regularization term to make full use of the manifold structure information between the data, and gives the iterative update rules of the algorithm. Sufficient experiments on real-world datasets demonstrate that the proposed method can effectively utilize the structural information of data and improve the clustering performance.

Key words: non-negative matrix factorization; dimensionality reduction; density peaks; graph regularization; clustering analysis

随着互联网和计算机技术的发展,人类已经获取了大量的高维数据。在信号处理[1]、模式识别[2]和计算机视觉[3]领域,如何将高维数据转换为更有效的低维表示是至关重要的问题。在大多数情况下,数据被组织成矩阵或张量,因此一些线性模型,例如主成分分析(Principal component analysis,

基金项目: 国家自然科学基金(61402395)资助项目;江苏省自然科学基金(BK20201430,BK20151314,BK20140492)资助项目。

收稿日期: 2020-09-25;**修订日期:** 2020-11-09

通信作者: 何萍,女,副教授, E-mail:angeletx@gmail.com。

引用格式: 徐晓华,方威,何萍,等. 峰值点非负矩阵分解聚类算法[J]. 南京航空航天大学学报,2021,53(5):772-779.
XU Xiaohua, FANG Wei, HE Ping, et al. Clustering algorithm for peaks non-negative matrix factorization[J]. Journal of Nanjing University of Aeronautics & Astronautics,2021, 53(5):772-779.

PCA)[4]，局部线性嵌入(Locally linear embedding，LLE)[5]和线性判别分析(Linear discriminant analysis，LDA)[6]可以很好地工作。与上述方法不同，Lee and Seung 在 Nature 上提出了非负矩阵分解[7](Non-negative matrix factorization，NMF)方法，它要求分解原始矩阵和通过分解得到的两个矩阵都是非负的，并实现线性降维。非负矩阵分解及其改进算法已成功应用于文本聚类[8]、图像去噪[9]以及人脸识别[10]等领域。

然而传统 NMF 方法的单一非负约束束不能满足各个领域的需求，因此仍存在一些缺陷和局限性。为了挖掘高维数据间潜在的流形结构信息，Cai 等[11]基于数据点之间的相似性构造一个邻居图和一个加权邻接矩阵提出了图正则非负矩阵分解(Graph-regularized non-negative matrix factorization，GNMF)，而考虑到 NMF 和 GNMF 中单个聚类中心不足以描述原始数据的复杂结构，Gao 等[12]采用多个中心点来表示样本的类别从而提出了局部中心结构非负矩阵分解(Local centroids structured non-negative matrix factorization，LCSNMF)。为了自适应学习局部流形结构，Huang 等[13]提出自适应邻域的概念，为每个数据点自适应分配邻居从而提出了具有自适应领域的非负矩阵分解(Non-negative matrix factorization with adaptive neighbor，NMFAN)。一般来说，簇中心是由一些局部密度较低的点所围绕，且这些点距离其他高密度的点的距离都比较远，针对簇中心的该特性，文献[14]中提出了密度峰值算法，该算法通过计算最近邻的距离，并依据密度大小进行排列得到数据的多个峰值点，从而得到聚类中心以实现数据的高效聚类。

然而 GNMF 所构造的近邻图是基于传统的欧几里得距离，在处理复杂数据结构时有时并不能准确地描绘出样本间的真实距离。此外，LCSNMF 模型中对每个簇指定了相同的中心点数，而在实际应用中不同簇的结构都存在差异，这样的描述显然是有缺陷的。针对上述两个算法中存在的问题，本文提出了峰值点非负矩阵分解算法(Peaks non-negative matrix factorization，PNMF)。该算法通过找到数据的多个密度峰值点，并将其峰值点与样本点构造二部图，再通过构造基于测地线距离的数据近邻图，并将其融入非负矩阵分解模型。在利用多个密度峰值点表示样本的类别的同时，也考虑了数据内在的流形结构。在多种类型的数据集上的实验结果表明，该方法在特征提取和数据聚类等方面优于其他同类的算法。

1　相关工作

1.1　非负矩阵分解

对于非负数据矩阵 $\boldsymbol{X}\in\mathbf{R}^{m\times n}$，NMF[7]算法可以将其分解为两个非负矩阵 $\boldsymbol{F}\in\mathbf{R}^{m\times k}$ 和 $\boldsymbol{G}\in\mathbf{R}^{n\times k}$ 的乘积形式，其中 k 为样本簇数。$\boldsymbol{F}$ 中的每个列向量可以看作是每个簇的聚类中心，$\boldsymbol{G}$ 中的每个行向量是每个样本点与中心点之间的相关度。最初的 NMF 提议采用 Frobenius 范数来最大程度地减少重构误差，旨在解决以下问题

$$\min_{\boldsymbol{F}\geqslant 0,\boldsymbol{G}\geqslant 0}\left\|\boldsymbol{X}-\boldsymbol{F}\boldsymbol{G}^{\mathrm{T}}\right\|_{\mathrm{F}}^{2} \tag{1}$$

式中 $\|\cdot\|_{\mathrm{F}}$ 表示 Frobenius 范数。为了优化该问题，一种迭代的乘法更新方案被提出，即

$$\boldsymbol{F}_{ij}\leftarrow\boldsymbol{F}_{ij}\left(\frac{\boldsymbol{X}\boldsymbol{G}}{\boldsymbol{F}\boldsymbol{G}^{\mathrm{T}}\boldsymbol{G}}\right)_{ij},\ \boldsymbol{G}_{ij}\leftarrow\boldsymbol{G}_{ij}\left(\frac{\boldsymbol{X}^{\mathrm{T}}\boldsymbol{F}}{\boldsymbol{G}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F}}\right)_{ij} \tag{2}$$

1.2　图正则非负矩阵分解

由于 NMF 是学习欧氏空间中输入数据的低维表示的线性方法，因此它无法发现输入数据的固有几何结构。因此，Cai 等研究了一种 GNMF 的算法，该算法将基于图的正则化器引入非负矩阵分解中，以在矩阵分解过程中保留数据的固有几何结构[11]。该问题的算法模型表示为

$$\min_{\boldsymbol{F}\geqslant 0,\boldsymbol{G}\geqslant 0}\left\|\boldsymbol{X}-\boldsymbol{F}\boldsymbol{G}^{\mathrm{T}}\right\|_{\mathrm{F}}^{2}+\lambda\mathrm{tr}(\boldsymbol{G}^{\mathrm{T}}\boldsymbol{L}\boldsymbol{G}) \tag{3}$$

式中：λ 为平衡参数；$\boldsymbol{L}$ 为拉普拉斯矩阵。

1.3　局部中心结构非负矩阵分解

在 NMF 和 GNMF 等算法中每个类别仅由一个中心点表示，然而这种表示由于缺少类别的结构信息往往是模糊且粗糙的。针对上述问题，Gao 等[12]提出了 LCSNMF，该算法将非负数据矩阵 $\boldsymbol{X}\in\mathbf{R}^{m\times n}$ 分解成两个矩阵 $\boldsymbol{F}\in\mathbf{R}^{m\times k}$ 和 $\boldsymbol{G}\in\mathbf{R}^{n\times k}$，这里 $k=ac$ 且 a 为每个簇中的中心点数，且每个簇都由 a 个中心点来表示。LCSNMF 的优化问题可以记为

$$\begin{aligned}&\min_{\boldsymbol{F}\geqslant 0,\boldsymbol{G}\geqslant 0}\left\|\boldsymbol{X}-\boldsymbol{F}\boldsymbol{G}^{\mathrm{T}}\right\|_{\mathrm{F}}^{2}\\&\text{s.t.}\quad\left\|\boldsymbol{g}^{i}\right\|_{0}\leqslant s\quad i=1,2,\cdots,n\end{aligned} \tag{4}$$

式中：$\boldsymbol{g}^{i}$ 为矩阵 $\boldsymbol{G}$ 的第 i 个行向量；$s>0$ 为超参数；$\left\|\boldsymbol{g}^{i}\right\|_{0}\leqslant s$ 表示一个样本点最多可关联的中心点的个数为 s。

因为每个簇有多个中心点，所以在分解后需要利用 K-means[15]算法对系数矩阵 $\boldsymbol{G}$ 进一步聚类，但是因为考虑到 K-means 算法对初始值较敏感，可能

无法获得最优解，为了解决这一问题，LCSNMF构造了一个由中心点和样本点组成的二部图，其相似度矩阵$\boldsymbol{S}$为

$$\boldsymbol{S}=\begin{bmatrix}0 & \boldsymbol{G}\\ \boldsymbol{G}^{\mathrm{T}} & 0\end{bmatrix} \tag{5}$$

该图中同类的样本点和中心点会构成一个连通分量，将该连通分量的个数设置为类数c，可以利用图分割算法得到c个类的聚类结果。因此该二部图的连通分量的数目等于拉普拉斯矩阵$\boldsymbol{L}_S$中特征值为0的个数，即$\sum_{i=1}^{c}\sigma_i(\boldsymbol{L}_S)=0$，$\sigma_i(\boldsymbol{L}_S)$为拉普拉斯矩阵$\boldsymbol{L}_S$中第$i$小的特征值，且$\boldsymbol{L}_S=\boldsymbol{D}(\boldsymbol{S})-\boldsymbol{S}$，有

$$\sum_{i=1}^{c}\sigma_i(\boldsymbol{L}_S)=\min_{\boldsymbol{P}\in\mathbf{R}^{(n+k)\times c},\boldsymbol{P}^{\mathrm{T}}\boldsymbol{P}=\boldsymbol{I}}\mathrm{tr}(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{L}_S\boldsymbol{P}) \tag{6}$$

令$\boldsymbol{P}$表示二部图的聚类指示矩阵，则LCSNMF的优化模型为

$$\begin{aligned}&\min_{\boldsymbol{F}\geqslant 0,\boldsymbol{G}\geqslant 0}\left\|\boldsymbol{X}-\boldsymbol{F}\boldsymbol{G}^{\mathrm{T}}\right\|_{\mathrm{F}}^{2}+\lambda\mathrm{tr}\left(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{L}_S\boldsymbol{P}\right)\\ &\text{s.t. }\boldsymbol{P}^{\mathrm{T}}\boldsymbol{P}=\boldsymbol{I},\left\|\boldsymbol{g}^{i}\right\|_{0}\leqslant s\quad i=1,2,\cdots,n\end{aligned} \tag{7}$$

2 峰值点非负矩阵分解

虽然LCSNMF算法采用了利用多个中心点来表示一个簇中样本点的方法，但实际应用中每个簇的结构不尽相同，对不同的簇指定相同的中心点数量显然是不合理的，对于结构复杂的数据无法得到最优聚类结果。针对于该问题，本文提出了PNMF，本算法先通过密度峰值算法为数据集找到多个密度峰值点，再利用密度峰值点的线性组合得到簇中心点进行聚类，此外利用测地线距离构建流形近邻图正则项融入NMF框架。

2.1 测地线距离

在很多研究中，一般都会使用样本点之间距离作为相似性度量。常用的距离度量包括欧氏距离、曼哈顿距离等。为了更好地利用复杂结构的数据中的流形结构信息，采用测地线距离[16]作为本文的距离度量标准。首先为原始数据中的所有样本点构造一个加权无向图$\boldsymbol{H}=<\boldsymbol{V},\boldsymbol{E}>$，每个样本点都是图$\boldsymbol{H}$中的一个顶点，边的集合表示为$\boldsymbol{E}=\{e_{ij}\}$，即样本点$\boldsymbol{x}_i$和$\boldsymbol{x}_j$之间的欧氏距离。令$q$表示样本点$\boldsymbol{x}_i$到$\boldsymbol{x}_j$的路径，$\boldsymbol{Q}_{ij}=\{q_1,q_2,\cdots\}$表示所有样本点$\boldsymbol{x}_i$到$\boldsymbol{x}_j$的路径的集合，则样本点$\boldsymbol{x}_i$和$\boldsymbol{x}_j$间的测地线距离为

$$d_{ij}=\min_{q\in\boldsymbol{Q}_{ij}}\mathrm{Dijkstra}(q) \tag{8}$$

2.2 密度峰值

假设数据集$\boldsymbol{X}\in\mathbf{R}^{m\times n}$中样本点$\boldsymbol{x}_i$和$\boldsymbol{x}_j$之间的测地线距离为$d_{ij}$，假如将样本点$\boldsymbol{x}_i$的邻域定义为以样本点$\boldsymbol{x}_i$为中心，截断距离$d_{\mathrm{cut}}$为半径的范围，那该邻域内样本点$\boldsymbol{x}_i$的局部密度就可以定义为

$$\rho_i=\sum\nolimits_j\chi(d_{ij}-d_{\mathrm{cut}}) \tag{9}$$

式中截断距离d_{cut}的值取太大会使得每个数据点都被归为一类以致区分度不高，d_{cut}的值取太小会使得每个数据点都被单独分为一个类。根据文献[14]中的经验，在实验中对于d_{cut}的选取，使平均每个点的邻居数为所有点的1%。其中$\chi(a)$为比较函数，且如果$a<0$值为1；否则为0。

另外，从局部密度比$\boldsymbol{x}_i$大的样本点中选取与最接近$\boldsymbol{x}_i$的样本点，并将它们之间的距离表示为

$$\delta_i=\begin{cases}\max\left\{d_{ij}|\rho_i>\max\left(\rho_j\right)\right\} & \rho_i=\max(p)\\ \min\left\{d_{ij}|\rho_j>\rho_i\right\} & \text{其他}\end{cases} \tag{10}$$

当有局部密度更大的样本点时，将δ_i定义为从最接近$\boldsymbol{x}_i$的样本点到$\boldsymbol{x}_i$的距离；如果$\boldsymbol{x}_i$已经是局部密度最大的样本点时，δ_i定义为数据集中离$\boldsymbol{x}_i$最远的样本点到$\boldsymbol{x}_i$的距离。

因此对于密度峰值点的选取，综合考虑样本点局部密度和与密度中心的距离。在实际应用中，不同类中样本的个数相差较大，密度也不尽相同，这样会使得选取的峰值点分布不均，首先将所有样本点作为密度峰值点的候补集合，再考虑每个样本点的局部密度和与密度中心的距离按从大到小的顺序依次选出一个样本点，并从此前的候补集中去除以该点为中心、半径为d_{cut}的领域内的样本点，直到剩下的密度峰值点的数目为k，然后可以得到峰值矩阵$\boldsymbol{X}_{\mathrm{dp}}\in\mathbf{R}^{m\times k}$。

在人造数据集Twomoons中，将提出的PNMF通过密度峰值算法在Twomoons数据集中找到多个密度峰值点，如图1所示，密度峰值[17]算法通过考虑样本点之间的距离和密度得到选取的密度峰值点能更好地获得数据本身的流形结构。

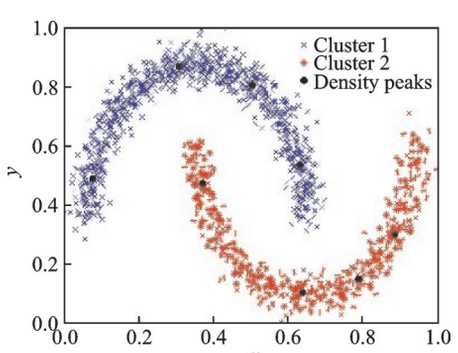

图1 Twomoons数据集上的密度峰值点

Fig.1 Peaks on the Twomoons dataset

2.3　峰值点非负矩阵分解

在得到数据集中的密度峰值点后，本文提出的PNMF算法将原矩阵分解为

$$\boldsymbol{X}\approx(\boldsymbol{X}_{\mathrm{dp}}\boldsymbol{F})\boldsymbol{G}^{\mathrm{T}} \tag{11}$$

式中：$\boldsymbol{X}_{\mathrm{dp}}\in\mathbf{R}^{m\times k}$为密度峰值矩阵；$\boldsymbol{F}\in\mathbf{R}^{k\times k}$为峰值点的非负线性组合；$\boldsymbol{G}\in\mathbf{R}^{n\times k}$为样本点与峰值点的关联矩阵。

根据流形假设：如果在原始空间中样本点$\boldsymbol{x}_i$和$\boldsymbol{x}_j$间的测地线d_{ij}距离相近，那么它们在子空间下的表示$\boldsymbol{g}^i$和$\boldsymbol{g}^j$间的距离也应该是相近的，因此构造图正则项$\mathrm{tr}(\boldsymbol{G}^{\mathrm{T}}\boldsymbol{L}_{\mathrm{geo}}\boldsymbol{G})$，其中$\boldsymbol{L}_{\mathrm{geo}}=\boldsymbol{D}(\boldsymbol{W})-\boldsymbol{W}$为原始空间中样本点间流形距离的拉普斯拉斯矩阵，其中$\boldsymbol{D}(\boldsymbol{W})=\mathrm{diag}(\boldsymbol{W}_1)$，$\boldsymbol{W}$定义为

$$\boldsymbol{W}=e^{-\left(\frac{D^2}{2\sigma^2}\right)} \tag{12}$$

将密度峰值矩阵融入NMF分解模型，利用密度峰值点与样本点的关联矩阵$\boldsymbol{G}$构造二部图，引入基于测地线距离的流形图正则项，最终得到PNMF的优化模型为

$$\begin{aligned}\min_{\boldsymbol{F}\geqslant 0,\boldsymbol{G}\geqslant 0,\boldsymbol{P}}\ &\frac{1}{2}\left\|\boldsymbol{X}-(\boldsymbol{X}_{\mathrm{dp}}\boldsymbol{F})\boldsymbol{G}^{\mathrm{T}}\right\|_{\mathrm{F}}^2+\lambda_1\mathrm{tr}(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{L}_S\boldsymbol{P})+\\&\frac{\lambda_2}{2}\mathrm{tr}(\boldsymbol{G}^{\mathrm{T}}\boldsymbol{L}_{\mathrm{geo}}\boldsymbol{G})\\\text{s.t }&\boldsymbol{P}^{\mathrm{T}}\boldsymbol{P}=\boldsymbol{I},\left\|\boldsymbol{g}^i\right\|_0\leqslant s\quad i=1,2,\cdots,n\end{aligned} \tag{13}$$

2.4　模型优化

目标函数式(12)中的$\boldsymbol{F},\boldsymbol{G},\boldsymbol{P}$并非同时都是凸的，因此很难找到全局最小值解，下面将介绍一种迭代算法来获取模型的局部最优解。

更新因子$\boldsymbol{P}$：先固定因子$\boldsymbol{F},\boldsymbol{G}$，求解因子$\boldsymbol{P}$，此时的优化问题为

$$\begin{aligned}\min_{\boldsymbol{P}}\ &\lambda_1\mathrm{tr}(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{L}_S\boldsymbol{P})\\\text{s.t. }&\boldsymbol{P}^{\mathrm{T}}\boldsymbol{P}=\boldsymbol{I}\end{aligned} \tag{14}$$

该问题的最优解由拉普拉斯矩阵$\boldsymbol{L}_S$前c小的特征值所对应的特征向量组成。

更新因子$\boldsymbol{F}$：先固定因子$\boldsymbol{P},\boldsymbol{G}$，求解因子$\boldsymbol{F}$。此时的优化问题为

$$\min_{\boldsymbol{F}\geqslant 0}\ \frac{1}{2}\left\|\boldsymbol{X}-(\boldsymbol{X}_{\mathrm{dp}}\boldsymbol{F})\boldsymbol{G}^{\mathrm{T}}\right\|_{\mathrm{F}}^2 \tag{15}$$

通过利用Frobenius范数与矩阵迹的关系：$\|\boldsymbol{A}\|_{\mathrm{F}}=\sqrt{\mathrm{tr}(\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}})}$可以将优化问题式(14)转化为

$$\begin{aligned}\mathcal{J}=\frac{1}{2}\mathrm{tr}(&\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}-2\boldsymbol{X}\boldsymbol{G}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}^{\mathrm{T}}+\\&\boldsymbol{G}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}\boldsymbol{F}\boldsymbol{G}^{\mathrm{T}})\end{aligned} \tag{16}$$

则式(15)关于$\boldsymbol{F}$的偏导数为

$$\frac{\partial\mathcal{J}}{\partial\boldsymbol{F}}=-\boldsymbol{X}_{\mathrm{dp}}^{\mathrm{T}}\boldsymbol{X}\boldsymbol{G}+\boldsymbol{X}_{\mathrm{dp}}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}\boldsymbol{F}\boldsymbol{G}^{\mathrm{T}}\boldsymbol{G} \tag{17}$$

因此，$\boldsymbol{F}$的更新公式为

$$\boldsymbol{F}\leftarrow\boldsymbol{F}\odot\frac{\boldsymbol{X}_{\mathrm{dp}}^{\mathrm{T}}\boldsymbol{X}\boldsymbol{G}}{\boldsymbol{X}_{\mathrm{dp}}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}\boldsymbol{F}\boldsymbol{G}^{\mathrm{T}}\boldsymbol{G}} \tag{18}$$

更新因子$\boldsymbol{G}$：先固定因子$\boldsymbol{P},\boldsymbol{F}$，求解因子$\boldsymbol{G}$。此时的优化问题为

$$\begin{aligned}\min_{\boldsymbol{G}\geqslant 0}\ &\frac{1}{2}\left\|\boldsymbol{X}-(\boldsymbol{X}_{\mathrm{dp}}\boldsymbol{F})\boldsymbol{G}^{\mathrm{T}}\right\|_{\mathrm{F}}^2+\lambda_1\mathrm{tr}(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{L}_S\boldsymbol{P})+\\&\frac{\lambda_2}{2}\mathrm{tr}(\boldsymbol{G}^{\mathrm{T}}\boldsymbol{L}_{\mathrm{geo}}\boldsymbol{G})\\\text{s.t. }&\left\|\boldsymbol{g}^i\right\|_0\leqslant s\quad i=1,2,\cdots,n\end{aligned} \tag{19}$$

其中正则项$\mathrm{tr}(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{L}_S\boldsymbol{P})$可以写成

$$\begin{aligned}\mathrm{tr}(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{L}_S\boldsymbol{P})&=\frac{1}{2}\mathrm{tr}(\boldsymbol{S}\boldsymbol{B})=\\&\frac{1}{2}\mathrm{tr}\left(\begin{bmatrix}0&\boldsymbol{G}\\\boldsymbol{G}^{\mathrm{T}}&0\end{bmatrix}\begin{bmatrix}\boldsymbol{B}_{11}&\boldsymbol{B}_{12}\\\boldsymbol{B}_{21}&\boldsymbol{B}_{22}\end{bmatrix}\right)=\\&\frac{1}{2}\mathrm{tr}\left(\begin{bmatrix}\boldsymbol{G}\boldsymbol{B}_{21}&\boldsymbol{G}\boldsymbol{B}_{22}\\\boldsymbol{G}^{\mathrm{T}}\boldsymbol{B}_{11}&\boldsymbol{G}^{\mathrm{T}}\boldsymbol{B}_{12}\end{bmatrix}\right)=\\&\frac{1}{2}\mathrm{tr}(\boldsymbol{G}^{\mathrm{T}}(\boldsymbol{B}_{21}{}^{\mathrm{T}}+\boldsymbol{B}_{12}))=\\&\frac{1}{2}\mathrm{tr}(\boldsymbol{G}^{\mathrm{T}}\boldsymbol{A})\end{aligned} \tag{20}$$

式中：$\boldsymbol{B}_{ij}=\left\|\boldsymbol{p}^i-\boldsymbol{p}^j\right\|_F^2$；$\boldsymbol{p}^i$为矩阵$\boldsymbol{P}$的第$i$行。因为矩阵$\boldsymbol{B}$是对称的，可以得到$\boldsymbol{A}=\boldsymbol{B}_{21}{}^{\mathrm{T}}+\boldsymbol{B}_{12}=2\boldsymbol{B}_{12}$。因此式(18)的优化问题可以写为

$$\begin{aligned}\mathcal{J}=\frac{1}{2}\mathrm{tr}(&-2\boldsymbol{X}\boldsymbol{G}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}^{\mathrm{T}}+\boldsymbol{G}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}\boldsymbol{F}\boldsymbol{G}^{\mathrm{T}})+\\&\lambda_1\mathrm{tr}(\boldsymbol{G}^{\mathrm{T}}\boldsymbol{B}_{12})+\frac{\lambda_2}{2}\mathrm{tr}(\boldsymbol{G}^{\mathrm{T}}\boldsymbol{L}_{\mathrm{geo}}\boldsymbol{G})\end{aligned} \tag{21}$$

则式(20)关于$\boldsymbol{G}$的偏导数为

$$\begin{aligned}\frac{\partial\mathcal{J}}{\partial\boldsymbol{G}}=&-\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}\boldsymbol{F}+\boldsymbol{G}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}\boldsymbol{F}+\\&\lambda_1\boldsymbol{B}_{12}+\lambda_2\boldsymbol{L}_{\mathrm{geo}}\boldsymbol{G}\end{aligned} \tag{22}$$

因此，$\boldsymbol{G}$的更新公式为

$$\boldsymbol{G}\leftarrow\boldsymbol{G}\odot\frac{\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}\boldsymbol{F}+\lambda_2\boldsymbol{W}\boldsymbol{G}}{\boldsymbol{G}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}\boldsymbol{F}+\lambda_1\boldsymbol{B}_{12}+\lambda_2\boldsymbol{D}(\boldsymbol{W})\boldsymbol{G}} \tag{23}$$

2.5　复杂度分析

可以预先知道$\boldsymbol{X}_{\mathrm{dp}}^{\mathrm{T}}\boldsymbol{X}$，$\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}$和$\boldsymbol{X}_{\mathrm{dp}}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}$在迭代过程中是固定的，因此可以提前计算$\boldsymbol{X}_{\mathrm{dp}}^{\mathrm{T}}\boldsymbol{X}$，$\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}$的时间复杂度均为$O(mnk)$，$\boldsymbol{X}_{\mathrm{dp}}^{\mathrm{T}}\boldsymbol{X}_{\mathrm{dp}}$的复杂度为$O(mk^2)$。由于$n>>k$，可以得到预计算的时间复杂度为$O(mnk)+O(mk^2)=O(mnk)$。

计算更新$\boldsymbol{P}$的复杂度为$\boldsymbol{L}_S$特征值分解需要$O((n+k)^3)$。$\boldsymbol{F}$的更新中分子的复杂度为$O(mnk)$，分母的复杂度为$O(mnk)$，因此更新$\boldsymbol{F}$的复杂度为$O(mnk)$。在$\boldsymbol{G}$的更新中分子的复杂度为$O(mnk)$，分母的复杂度为$O(mnk)+$

$O((m+n)c)+O(mnk)=O(mnk)$，因此更新 F 的复杂度为 $O(mnk)$。

综上，因为 $k\leqslant\min\{m,n\}$，迭代更新一次 PNMF 算法需要复杂度为 $O(mnk)+O((n+k)^3)+O(mnk)+O(mnk)=O(n^3)$。如果更新迭代 t 次，则算法复杂度为 $O(n^3t)$。

3 实验分析

为了验证所提出的 PNMF 算法的有效性，分别在 3 个常见的面部数据集（Yale，ORL，COIL20）、1 个文本数据集 TDT2 以及 1 个声音数据集 ISOLET 上进行聚类实验，并选取 NMF、GNMF、LSCNMF 和 NMFAN 为比较算法。本节将给出数据集、评价指标和实验分析等内容，此外每次实验独立随机，重复 20 次取平均和标准差作为最后实验结果。

3.1 数据集

为了进一步验证 PNMF 算法的有效性，选择的数据集有：Yale、ORL、COIL20、TDT2 和 ISOLET。

Yale 人脸数据集包含来自 15 个主题的 165 张图像，每个人有 11 张图像。图像显示了在不同照明条件下（左灯，中央灯和右灯）、面部表情（正常，快乐，悲伤，困倦，惊讶和眨眼）以及戴着或不戴眼镜的变化。

ORL 数据集具有 40 个不同主题中的每个主题的 10 个不同图像。一些图像是在不同的时间拍摄的，它们具有不同的照明，面部表情（睁开/闭合的眼睛，微笑/没有笑容）和面部细节（有眼镜/无眼镜）。

COIL20 图像数据集包含不同角度观看的 20 个对象的 32×32 灰度面部图像，每个对象有 72 个图像。

TDT2 文本数据集来自 NIST 主题检测与跟踪语料库。TDT2 包括 1998 年上半年收集的数据，来自 6 个来源，包括 2 个新闻通讯社（APW、NYT）、2 个广播节目（美国之音、PRI）和 2 个电视节目（CNN、ABC）。它包含 11 201 个主题文档，分为 96 个语义类别。实验中选择其子集包括 1 319 个主题文档，分为 5 个语义类别。

ISOLET 声音数据集来自 UCI 机器学习资料库，它包括 150 名受试者说出字母表中每个字母的名字两次，因此每个人有 52 个样本。选取原始数据集的子集，共包括 2 098 个样本。

在实验中将所有图像均压缩成 32×32 大小的灰度图，将其每列相连构成大小为 1 024 维的向量，其中 TDT2 文本数据集维度为 14 964，ISOLET 声音数据集维度为 617，所有数据集都进行归一化处理。图 2 给出了 3 个面部数据库的一些样本示例。

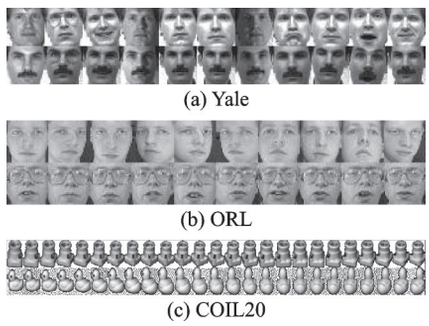

图 2 实验数据集示例

Fig.2 Instances of experimental datasets

3.2 聚类评价指标

为了更好地评估每个数据集上每种算法的聚类性能，使用了 3 个常用的聚类评估指标：ACC、NMI 和 Rand Index[18]。

聚类准确率（ACC）：它查找真实类与聚类结果之间的一对一关系，并从相应类中获取每个聚类所具有的数据样本，定义为

$$\mathrm{ACC}=\frac{\sum_{i=1}^{n}\delta(\mathrm{map}(r_i),l_i)}{n} \tag{24}$$

式中：r_i 表示 x_i 的聚类结果；l_i 表示数据 x_i 的真实标签；n 为整体的样本数量；$\mathrm{map}(r_i)$ 表示最佳映射函数，并使用 Kuhn-Munkres 算法确定最佳映射。此外 $\delta(a,b)$ 表示 Delta 函数，且如果 $a=b$ 值为 1，否则为 0。

标准互信息（NMI）：NMI 使用互信息函数和熵函数来评估聚类结果，定义如下

$$\mathrm{NMI}=\frac{\sum_{i=1}^{c}\sum_{j=1}^{c}n_{i,j}\log\frac{n_{i,j}}{n_i\hat{n}_j}}{\sqrt{\left(\sum_{i=1}^{c}n_i\log\frac{n_i}{n}\right)\left(\sum_{j=1}^{c}\hat{n}_j\log\frac{\hat{n}_j}{n}\right)}} \tag{25}$$

式中：c 表示类别数；n_i 表示属于簇 $C_i(1\leqslant i\leqslant c)$ 的数据数量；$\hat{n}_j$ 为包含在簇 $C_j(1\leqslant j\leqslant c)$ 中的数据数，并且 $n_{i,j}$ 代表簇 C_j 和簇 C_i 之间的重叠数据数。

Rand Index：它将聚类结果与数据的真实类别进行比较，计算正确聚类结果的比例。Rand Index 值越大，聚类效果越好。

3.3 实验结果与分析

在实验中，对于 NMF、GNMF 和 NMFAN 算

法，将分解的规模 k 默认设为数据集中簇的个数；对于LCSNMF算法和PNMF算法，将簇平均样本点个数的1%～10%作为每个簇中心点的个数 m，并且每个簇中的每个样本点与 s 个中心点相关，且 m 是从{1、2、3、4、5、6、7}中选择的，s 是从{1、2、3、4}中选择的。此外，对于PNMF算法模型中的正则化参数，在{1、10、100、1 000、10 000}的范围内选择参数 λ_1，但是基于测地距离的图正则项的参数 λ_2 和GNMF一致，均设置固定值为100。与谱聚类等其他聚类算法相比，K-means因其有效性和效率高而得到广泛运用，实验中和文献[19-20]中一样使用K-means应用于矩阵分解后的表示矩阵 $\boldsymbol{G}$，就可以得到最终的聚类结果。

表1～5分别显示了面部、文本和人脸数据集上的聚类性能。从表1～5可以看出，在大多数情况下，PNMF聚类评估指数的结果更好。从实验结果来看，构造基于欧几里得距离的近邻图的GNMF和NMFAN不如构造基于流形距离近邻图的PNMF的聚类性能好，说明了传统的欧几里得距离在面对更加复杂高维的数据时，并不能很好且准确地表示数据间的真实距离。LCSNMF由于其簇中心选取的局限性，效果也不如PNMF，而传统的NMF聚类因为缺乏约束，性能不是很好。从实验结果来看，本文提出的利用多个峰值点与样本点构造二部图的方法可以更好地捕获复杂数据的内部几何结构，从而提高聚类效果。

表1　Yale数据集上的聚类性能比较

Table 1　Comparison of clustering performance on Yale dataset

评价指标	NMF	GNMF	LCSNMF	NMFAN	PNMF
ACC	0.384 8±0.040 4	0.387 3±0.025 6	0.426 4±0.028 2	0.386 1±0.030 5	0.462 3±0.048 4
NMI	0.450 7±0.030 1	0.452 3±0.018 7	0.497 7±0.022 3	0.453 6±0.024 4	0.524 7±0.023 6
Rand Index	0.894 1±0.008 2	0.894 1±0.007 1	0.887 8±0.009 7	0.894 9±0.006 7	0.922 4±0.009 8

表2　ORL数据集上的聚类性能比较

Table 2　Comparison of clustering performance on ORL dataset

评价指标	NMF	GNMF	LCSNMF	NMFAN	PNMF
ACC	0.540 6±0.035 1	0.529 8±0.021 9	0.538 8±0.022 8	0.474 6±0.025 2	0.582 6±0.036 2
NMI	0.752 9±0.018 6	0.753 3±0.012 9	0.753 5±0.013 4	0.704 1±0.016 4	0.760 8±0.014 7
Rand Index	0.968 2±0.003 0	0.967 7±0.002 2	0.967 0±0.003 1	0.964 0±0.002 4	0.980 2±0.004 5

表3　COIL20数据集上的聚类性能比较

Table 3　Comparison of clustering performance on COIL20 dataset

评价指标	NMF	GNMF	LCSNMF	NMFAN	PNMF
ACC	0.580 3±0.041 7	0.607 7±0.056 6	0.568 0±0.058 0	0.578 9±0.040 2	0.611 8±0.078 0
NMI	0.732 7±0.018 5	0.758 9±0.028 0	0.735 1±0.029 3	0.720 9±0.022 6	0.792 8±0.031 8
Rand Index	0.948 4±0.006 7	0.950 4±0.011 9	0.941 4±0.012 8	0.947 9±0.006 3	0.939 8±0.027 1

表4　TDT2数据集上的聚类性能比较

Table 4　Comparison of clustering performance on TDT2 dataset

评价指标	NMF	GNMF	LCSNMF	NMFAN	PNMF
ACC	0.513 6±0.041 7	0.495 3±0.037 5	0.607 1±0.068 7	0.544 7±0.089 6	0.622 4±0.030 8
NMI	0.398 0±0.031 8	0.235 2±0.084 4	0.455 6±0.087 7	0.402 5±0.059 1	0.462 5±0.098 9
Rand Index	0.584 8±0.030 5	0.586 1±0.040 1	0.674 4±0.069 5	0.606 4±0.057 3	0.702 3±0.082 2

表5　ISOLET数据集上的聚类性能比较

Table 5　Comparison of clustering performance on ISOLET dataset

评价指标	NMF	GNMF	LCSNMF	NMFAN	PNMF
ACC	0.601 9±0.055 9	0.714 5±0.033 9	0.571 7±0.094 4	0.423 4±0.038 6	0.691 4±0.029 2
NMI	0.608 3±0.031 9	0.765 0±0.023 7	0.622 5±0.070 8	0.302 4±0.039 4	0.689 9±0.014 7
Rand Index	0.863 8±0.022 1	0.885 5±0.016 7	0.844 8±0.037 7	0.809 2±0.010 9	0.896 5±0.007 5

3.4　参数讨论

本节将给出PNMF在不同正则化参数设置下的聚类性能。在PNMF算法中，簇中心点数 m 设置决定了矩阵分解的大小，而样本点可以关联的中

心点数s决定了二部图的构造，因此对聚类的结果有一定影响。另外，参数λ_1和λ_2来平衡二部图的正则项和基于测地距离的近邻图正则项。在实验中，将λ_2和GNMF都设置一致为100，并讨论了参数λ_1对聚类性能的影响。

以COIL20数据集为例，将m设置为1～7，将s设置为1～4。在这种参数变化的情况下，测试了COIL20的3个聚类指标变化。测试结果如图3～5所示。从测试结果来看，当$m=5$且$s=4$时，COIL 20的聚类性能最佳。且随着s值的增加可以增强聚类性能，并且当m为4～6时，每个聚类指标的值都较高。从实验结果可以看出，聚类中心点的数量m对聚类性能影响较小，但s的值对聚类性能显得更敏感，可能因为s的值决定构造的二部图的质量，因此对结果有一定影响。

然后将讨论正则化参数对提出的PNMF模型的影响。模型具有两个正则化参数即λ_1和λ_2，它们分别来平衡二部图正则项和基于流形距离图正则项。在实验中将λ_2设置为100，然后讨论λ_1的变化对COIL20数据集的3个聚类性能指标的影响，λ_1的值选自{1、10、100、1 000、10 000}，实验结果如图6所示。

从图6可以看出，PNMF对λ_1只是一点点敏感。可以发现，随着λ_1值的增加，3个性能指标呈略微上升的趋势。

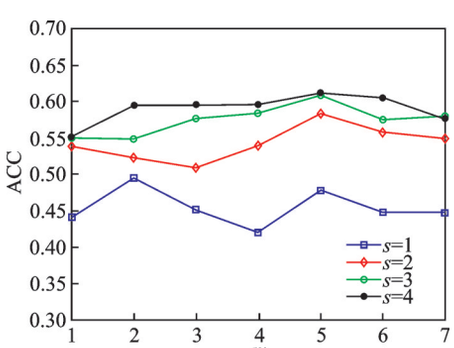

图3 COIL20在不同m及s下的ACC

Fig.3 ACC under different m and s on COIL20

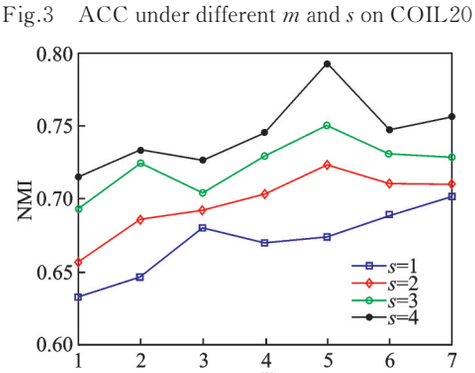

图4 COIL20在不同m及s下的NMI

Fig.4 NMI under different m and s on COIL20

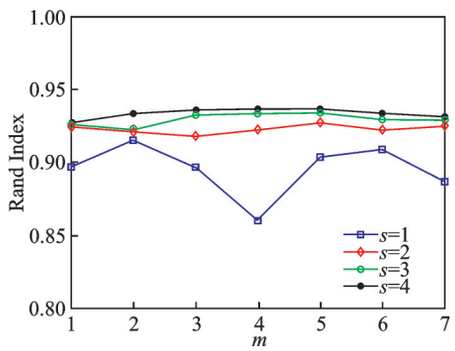

图5 COIL20在不同m及s下的Rand Index

Fig.5 Rand index under different m and s on COIL20

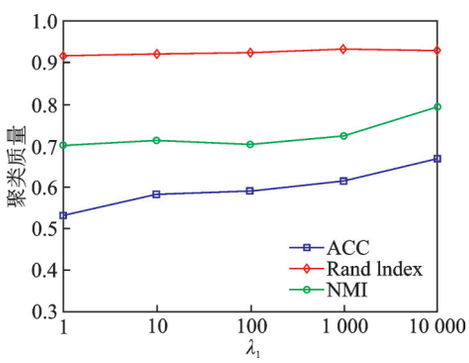

图6 COIL20在不同λ_1下的聚类结果

Fig.6 Clustering results of COIL20 under different λ_1

4 结 论

本文提出了一种新方法PNMF。首先计算每个样本点的局部密度，利用局部密度从数据集中找到多个密度峰点，它为每个簇指定多个中心点，并利用密度峰值点和样本点构造二部图。另外采用流形结构下的测地线距离，并用测地线距离构造了数据的近邻图，从而描述了局部几何关系，使得样本点之间距离更准确。为了证明该算法的有效性，本文比较了该算法在几个面部数据集以及文本、声音数据集上的聚类效果。实验结果表明，PNMF相比其他NMF算法具有更好的聚类性能。

参考文献：

[1] VAISHNAV N, TATU A. Signal processing on graphs: Structure preserving maps[J].IET Signal Processing, 2019,13(1):77-85.

[2] GILBERTO P C E, JOSÉ LISANDRO A C. Learning algorithm for the recursive pattern recognition model[J]. Applied Artificial Intelligence, 2016, 30(7): 662-678.

[3] ATHANASIOS V, NIKOLAOS D, ANASTASIOS D, et al.Deep learning for computer vision: A brief review[J]. Computational Intelligence and Neu-

roscience，2018，2018：1-13.

[4] LU Canyi，FENG Jiashi，CHEN Yudong，et al. Tensor robust principal component analysis with a new tensor nuclear norm [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence，2019，42 (4)：925-938.

[5] FANG Y，LI H，MA Y，et al. Dimensionality reduction of hyperspectral images based on robust spatial information using locally linear embedding [J]. IEEE Geoence & Remote Sensing Letters，2017，11(10)：1712-1716.

[6] AN Leilei，XING Hongjie. Linear discriminant analysis based on ZP-norm maximization [C]//Proceedings of International Conference on Information Technology & Electronic Commerce. Dalian，China：[s.n.]，2015：88-92.

[7] LEE D D，SEUNG H S. Learning the parts of objects by non-negative matrix factorization [J]. Nature，1999，401 (6755)：788-791.

[8] GONG L H，ZENG J P，ZHANG S Y，Text stream clustering algorithm based on adaptive feature selection [J]. Expert Systems with Applications，2011，38(3)：1393-1399.

[9] ZHANG K，ZUO W，CHEN Y，et al. Beyond a Gaussian denoiser：Residual learning of deep CNN for image denoising [J]. IEEE Transactions on Image Processing，2016，26(7)：3142-3155.

[10] WRIGHT J，YANG A Y，GANESH A，et al. Robust face recognition via sparse representation [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence，2009，31(2)：210-227.

[11] CAI D，HE X，HAN J，et al. Graph regularized non-negative matrix factorization for data representation [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence，2011，33(8)：1548-1560.

[12] GAO H，NIE F，HUANG H. Local centroids structured non-negative matrix factorization [C]//Proceedings of Thirty-First AAAI Conference on Artificial Intelligence. [S.l.]：AAAI，2017：1905-1911.

[13] HUANG S，XU Z，KANG Z，et al. Regularized nonnegative matrix factorization with adaptive local structure learning [J]. Neurocomputing，2020，382：196-209.

[14] RODRIGUEZ A，LAIO A. Clustering by fast search and find of density peaks [J]. Science，2016，344 (6191)：1492-1496.

[15] TANG M，AYED I B，MARIN D，et al. Secrets of GrabCut and kernel k-means [C]//Proceedings of International Conference on Computer Vision. Santiago，Chile：[s.n.]，2015：1555-1563.

[16] WANG P，ZENG G，GAN R，et al，Structure-sensitive superpixels via geodesic distance [J]. International Journal of Computer Vision，2013，103(1)：1-21.

[17] XU X，JU Y，LIANG Y，et al，Manifold density peaks clustering algorithm [C]//Proceedings of 2015 Third International Conference on Advanced Cloud and Big Data. Yangzhou，China：[s.n.]，2015：311-318.

[18] YEH C C，YANG M S. Evaluation measures for cluster ensembles based on a fuzzy generalized rand index [J]. Applied Soft Computing，2017，57：225-234.

[19] HE P，XU X，DING J，et al，Low-rank nonnegative matrix factorization on stiefel manifold [J]. Information Sciences，2020，514：131-148.

[20] SHENG Y，WANG M，WU T，et al，Adaptive local learning regularized nonnegative matrix factorization for data clustering [J]. Applied Intelligence，2019，49：2151-2168.

（编辑：刘彦东）