基于特征偏好的聚类研究

方 玲 陈松灿
(南京航空航天大学计算机科学与技术学院 南京 210016)

摘 要 传统的聚类方法,如k均值和模糊c均值,通常并不区分数据特征对聚类的不同贡献或重要度,因此在面对高维数据聚类时,常会导致偏低的聚类性能,这归咎于聚类时未考虑高维数据特征间所存在的高度相关性或冗余。而通过在聚类时为每一特征引入权重并通过聚类目标的优化,不仅能自动获得对应的权重,而且也获得了聚类性能的提升。尽管如此,但无监督获取的特征权重未必吻合用户所期望的特征间的相对重要性(或偏好)。因此尝试利用用户给定的实际偏好设计出能反映特征偏好的聚类方法,其将现有独立于个体聚类的全局加权型偏好聚类方法拓展至聚类依赖的局部特征加权型方法,由此弥补了前者的不足,提升了偏好聚类算法的性能。

关键词 聚类分析,特征偏好,特征权重,聚类依赖,二次规划
中图法分类号 TP391.4 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.5.012

Research on Clustering with Feature Preferences

FANG Ling CHEN Song-can
(Department of Computer Science & Engineering, Nanjing University of Aeronautics & Astronautics, Nanjing 210016, China)

Abstract Traditional clustering methods, such as k-means and fuzzy c-means, do not generally distinguish different contributions or importance of data features to individual clusters, thus when facing high dimensional data, they often lead to lower clustering performance due to hardly considering the presence of high correlation or redundancy between features. In order to mitigate such adversity, with the introduction of the feature weights for each cluster in the clustering objective, we could automatically obtain not only the cluster-dependent weights but also the enhanced clustering performance. Though so, the feature weights obtained by an unsupervised clustering algorithm do not necessarily match the relative importance (or preferences) between the features as users expect. Thus this paper attempted to take advantage of actual preferences from users to design a clustering method which can reflect the feature preference. As a result, the proposed method not only extends the existing clustering methods with globally-weighted cluster-independent features to the one with locally-weighted cluster-dependent features but alos improves the clustering performance for feature preferences.

Keywords Clustering analysis, Feature preferences, Feature weighting, Cluster-dependent, Quadratic programming

1 引言

聚类是挖掘数据结构的有效工具之一,其被广泛地应用于诸如分类学、图像处理[14]、文本处理[2]、生物信息学[27]、数据挖掘、信息检索等众多领域。其目的是将一组数据对象划分成多个簇,使得同一簇内的对象比其它簇内的对象更相似。至今已有众多聚类算法相继被提出和应用,其中最常见和流行的当属k均值聚类[6,9]、模糊c均值聚类算法[10]和高斯混合算法(GMM)[19]。本文将立足k均值开展研究,但所发展出的方法能够用于诸如模糊c均值等其他聚类算法。

k均值聚类算法尝试将给定的一组数据划分到预先设定的k个聚类中,使得所有聚类的类内均方误差最小。其实现简单、易理解且结果具有可接受的有效性。目前围绕k均值聚类算法,已产生了一系列研究成果,其中包括对其鲁棒化改造[20]以实现对离群数据的免疫能力,以及分布式[21]和并行化设计[22]实现对大规模数据的聚类等。

k均值聚类算法通常采用欧氏距离度量进行聚类,结果使每个特征在聚类过程中被视为同等重要。而特征常作为对象属性,代表了其不同的性质。在实际问题中每个特征的重要性往往不同,如人脸识别中,特征包括头发、眼睛、鼻子、嘴和脸等,在识别过程中它们并不起着同等重要的作用。一些特征比另一些特征重要,如人的头发对人脸的识别就显得不那么重要。因此特征在提高聚类算法的性能方面起着重要的作用,不是所有的特征都是有用的,它们之间常存在相关或者冗余。这不仅影响聚类质量,而且在高维场景[28]下易导致过拟合。为了缓和该问题,已有各种反映特征权重的聚类算法相继被提出,这些算法的一个基本实现思想是将大的权重赋予重要或相关的特征,而将小的权重赋予不相关或者冗余的

到稿日期:2014-01-13 返修日期:2014-03-30
方 玲(1989—),女,硕士生,主要研究方向为机器学习,E-mail:fangling1105@126.com;陈松灿(1962—),男,教授,博士生导师,主要研究方向为机器学习、模式识别、数据挖掘。

特征。特征加权实际可视为特征选择方法[3,7,29]的推广，前者将不同的特征赋予了[0,1]之间的值来表明每个特征的重要性，而后者通过权重0来消除对应特征而权重1则保持特征。不同的最优化策略可以被应用到特定的特征权重的目标函数中，通过对特征重要性不同的刻画来提高聚类算法的性能。

特征权重的无监督聚类算法在聚类过程中能够自动学得权重，如最优化变量加权法(OVW)[23]、特征权重的k均值聚类算法(w-k-means)[24]以及特征权重自我调节方法(FW-SA)[25]等。此类方法的不足在于权重自动学习后，未必吻合用户所期望的特征间的相对重要性(如后文实验结果所示)。基于此，研究者借助问题的先验知识提出了特征偏好[4,12]概念，用于刻画相对问题而言的特征之间所存在的某种重要性关系。特征偏好常用成对约束[12]来体现，如两个特征是强相关或不相关等，本文采用了特征权重的差值来度量两者的相对重要性。结合特征偏好的聚类被称为基于偏好的聚类或偏好聚类。

偏好聚类的研究相对较少，据我们所知，仅有Sun等人[4]的一项工作。他们提出了一个以特征偏好作为约束并结合基于Bregman散度的聚类算法CFP，但该算法仅考虑了权重的全局性，独立于聚类，即所有聚类共享了与给定偏好吻合的共同权重。而在实际中，常常存在聚类依赖现象，即对于不同的聚类，关联特征的权重往往不同，如在人脸识别中，胡子这一特征在男女两个类别中的重要程度差别就十分明显。因此，本文尝试利用用户给定的实际偏好，将现有独立于个体聚类的全局加权型偏好聚类方法拓展至聚类依赖的局部特征加权型方法，对数据集中各个类别赋予不同的特征权重，以此来体现聚类过程中各特征对不同类别贡献的大小，同时结合特征偏好的约束，使聚类过程中获得的权重能更好地遵守特征之间的先验关系。

本文第2节主要介绍了基本的算法模型；第3节给出了算法中的公式推导；第4节通过实验比较了不同的算法模型；最后进行了总结与展望。

2 算法模型

2.1 相关记号定义

文中用$\mathbb{F}\in\mathbb{R}$表示输入空间。所有的n个样本点表示为$x_1,x_2,\cdots,x_n$。每一个样本点$x_i=[x_{i_1},x_{i_2},\cdots,x_{i_d}]^{\mathrm{T}}$。因此整个输入数据矩阵为$X=[x_1,x_2,\cdots,x_n]^{\mathrm{T}}\subseteq\mathbb{R}^{n\times d}$。

在数据集中，假设聚类数为k。最终获得的最优划分为$\{\pi_c\}_{c=1}^{k}$，其中π_c表示第c个聚类，$n_c=|\pi_c|$则表示第c个聚类中样本的个数。$Z=[z_1,z_2,\cdots,z_k]^{\mathrm{T}}\subseteq\mathbb{R}^{k\times d}$为聚类中心集，其中$z_i=[z_{i1},z_{i2},\cdots,z_{id}]$表示第$i$个聚类的中心。全局特征权重$w=[w_1,w_2,\cdots,w_d]^{\mathrm{T}}\subseteq\mathbb{R}^d$，局部特征权重$W=[w_1,w_2,\cdots,w_d]^{\mathrm{T}}\subseteq\mathbb{R}^{d\times k}$，其中$w_j=[w_{j1},w_{j2},\cdots,w_{jk}]$。此外，$1_d\in\mathbb{R}^d$表示$d$维的全1列向量，$I_d\in\mathbb{R}^{d\times d}$是单位阵。$\mathbb{R}_+$表示非负集合。$\Delta_d$表示集合$\Delta_d=\{w\in\mathbb{R}^d_+\mid w^{\mathrm{T}}1_d=1\}$，且$\Delta_{d\times k}=\{W1_k\in\mathbb{R}^d_+\mid W^{\mathrm{T}}1_d=1_k\}$。$e_a^b$表示$b$维列向量，其中除第$a$个元素为1外，其余元素均为0。

2.2 聚类依赖的局部偏好聚类模型

2.2.1 目标函数构建

实际中不同的聚类常常拥有不同重要性的特征。用$w=[w_1,w_2,\cdots,w_d]^{\mathrm{T}}\subseteq\mathbb{R}^d$来表示全局特征权重，$w_j(1\leqslant j\leqslant d)$表明特征$j$的重要程度，且$w\in\Delta_d$。$W=[w_1,w_2,\cdots,w_d]^{\mathrm{T}}\subseteq\mathbb{R}^{d\times k}$表示局部特征权重，其中$w_j=[w_{j1},w_{j2},\cdots,w_{jk}]$表示不同聚类相对特征$j$的重要程度。$W$满足每个聚类所有特征权重和为1的约束，即$\sum_{j=1}^{d}w_{jc}=1(c=1,2,\cdots,k)$。在算法实现中，假设存在一组预先给定的特征偏好以反映问题的先验知识。偏好通过一个元组(s,t,δ)表示，特征s与特征t之间的权重满足：$w_s-w_t\geqslant\delta$，δ是一个大于0的数，其值可在实验中自动调整或预定。所有的m个特征偏好元组形成集合$P=\{(s_i,t_i,\delta_i)\}_{i=1}^{m}$。

传统k均值聚类算法在计算各样本点与聚类中心的相异度时由于没有反映先验知识，无法区分数据特征对聚类的不同贡献而导致算法的性能偏低。本文所提出的聚类算法能够结合特征偏好的约束在聚类过程中学习权重，并通过给定的特征偏好对权重进行修正，从而提高算法的性能。

文中基于特征偏好聚类算法的目标函数由3部分组成：1)距离度量，用于刻画样本间的相异性；2)对于偏好违反的惩罚项；3)正则化项，用于保证权重的一致性。下面对各部分给予详细描述。

(1)距离度量

距离度量是聚类算法的关键所在，它用来度量不同样本点与聚类中心的差别，因而不同的度量刻画了不同的聚类结构。依据“没有免费午餐”定理[26]，结合问题相关知识的度量期望产生更优的学习性能，本文工作便是一个尝试。现假设每个聚类π_c的中心为$z_c\in\mathbb{F}$，该类的内部失真度为$\sum_{x_i\in\pi_c}\varphi^2(x_i,z_c)$。这里$\varphi^2(x_i,z_c)$表示两个数据点之间的失真度量。因此对于所有的聚类$\{\pi_c\}_{c=1}^{k}$，其总体表现性能可用如下形式表示：

$$J=\sum_{c=1}^{k}\sum_{x_i\in\pi_c}\varphi^2(x_i,z_c)\tag{1}$$

(2)基于特征偏好的权重惩罚

为了使聚类过程中获得的权重尽可能服从给定的特征偏好P，在上述目标函数中增加一惩罚项以反映对偏好可能的违反。因此对于每个特征偏好元组$p=(s,t,\delta)\in P$，其惩罚项定义为$\max(\delta-(w_s-w_t),0)$。现针对m个偏好，引入相应的m个辅助变量构成向量$\xi=[\xi_p]\in\mathbb{R}^m$且$p\in P$，由此形成如下的特征偏好惩罚项：

$$\begin{aligned}&\min\ \sum_{p\in P}\xi_p+\text{other terms}\\&\text{s.t. } w\in\Delta_d\\&\quad w_s-w_t\geqslant\delta-\xi_p\ \text{for all}\ p=(s,t,\delta)\in P\\&\quad\xi_p\geqslant0\end{aligned}\tag{2}$$

(3)熵项

除了给定的特征偏好先验外，并不希望对特征权重的其他方面作无根据的假设以造成过拟合，因此在目标函数中引入(负)熵项[11]$-\hat{H}(w)$，并使之最小化来保证权重尽可能的光滑或均衡。权重越均衡，$\hat{H}(w)$的值越大，反之则越小。为计算方便，我们采用的熵为$\hat{H}(w)=1-w^{\mathrm{T}}w$，对熵的最大化等价于最小化$w^{\mathrm{T}}w$。

通过折中上述3部分，最终形成基于特征偏好的聚类算法的总体目标函数如下：

$$\min_{(w,\xi),\{\pi_c\}_{c=1}^k,\{z_c\}_{c=1}^k} \sum_{c=1}^{k}\sum_{x_i\in\pi_c}\varphi^2(x_i,z_c)+\lambda_1\sum_{p\in P}\xi_p+\lambda_2 w^{\mathrm{T}}w$$

$$\text{s. t. } w\in\Delta_d \quad (3)$$

$$w_s-w_t\geqslant\delta-\xi_p \text{ for all } p=(s,t,\delta)\in P$$

$$\xi_p\geqslant 0 \text{ for all } p\in P$$

其中，λ_1 和 λ_2 是对应特征偏好项和熵项的非负惩罚参数。在实验中，通过参数值的合理设置，可获得更优的聚类性能。

2.2.2 聚类依赖的局部权重建模

为建立特征偏好的局部权重聚类算法，将 2.2.1 节中的距离度量取作

$$\varphi^2(x_i,z_c)=\sum_{j=1}^{d}w_{jc}D_{ic}^j \quad (4)$$

其中，D_{ic}^j 表示样本点 x_i 与聚类中心 z_c 在第 j 个特征上的距离。聚类依赖的权重满足 $\forall j,c,w_{jc}\in[0,1]$ 及 $\sum_{j=1}^{d}w_{jc}=1$。w_{jc} 是对应聚类 c 的特征 j 的权重。实际上距离度量可根据数据集的不同特性采用不同的定义，本文采用了常用的欧氏距离，即 $D_{ic}^j=(x_{ij}-z_{cj})^2$。

因此在欧氏距离度量下，基于聚类依赖的局部权重聚类模型(CDCFP)的目标函数归结为

$$\min_{(W,\xi),\{\pi_c\}_{c=1}^k,\{z_c\}_{c=1}^k} \sum_{c=1}^{k}\sum_{x_i\in\pi_c}\sum_{j=1}^{d}w_{jc}(x_{ij}-z_{cj})^2+\lambda_1\sum_{p\in P}\xi_p+\lambda_2 tr(W^{\mathrm{T}}W)$$

$$\text{s. t. } W\in\Delta_{d\times k}$$

$$\sum_{c=1}^{k}w_{sc}-\sum_{c=1}^{k}w_{tc}\geqslant\delta-\xi_p \text{ for all } p=(s,t,\delta)\in P$$

$$\xi_p\geqslant 0 \text{ for all } p\in P \quad (5)$$

这里 $W=[w_1,w_2,\cdots,w_d]^{\mathrm{T}}\subseteq\mathbb{R}^{d\times k}$ 表示权阵，其每一列对应某一聚类的(局部)权重。如果所有 $w_1,w_2,\cdots,w_d$ 取作相等，则式(5)的局部特征权重退化为非聚类依赖的全局权重，因而全局权重的聚类模型[4]是模型 CDCFP 的一个特例，其相应目标函数为

$$\min_{(w,\xi),\{\pi_c\}_{c=1}^k,\{z_c\}_{c=1}^k} \sum_{c=1}^{k}\sum_{x_i\in\pi_c}\sum_{j=1}^{d}w_j(x_{ij}-z_{cj})^2+\lambda_1\sum_{p\in P}\xi_p+\lambda_2 w^{\mathrm{T}}w$$

$$\text{s. t. } w\in\Delta_d$$

$$w_s-w_t\geqslant\delta-\xi_p \text{ for all } p=(s,t,\delta)\in P$$

$$\xi_p\geqslant 0 \text{ for all } p\in P \quad (6)$$

3 目标优化和算法设计与分析

目标函数(5)中共有 3 组未知变量：$\{\pi_c\}_{c=1}^k$、W 和 $\{z_c\}_{c=1}^k$，若固定其中任两组，则问题关于另一组变量为凸问题，因此可通过块坐标优化[17]方式迭代地求解 $\{\pi_c\}_{c=1}^k$、W 和 $\{z_c\}_{c=1}^k$。根据 Zangwill[13]的收敛理论可证迭代收敛至一局部最优解。

3.1 目标优化

3.1.1 固定 $\{\pi_c\}_{c=1}^k$ 和 W，求解 $\{z_c\}_{c=1}^k$

固定 $\{\pi_c\}_{c=1}^k$ 和 W，即可获得各聚类的划分，将式(4)代入式(1)中并对其关于 z_{cj} 求导得 $\frac{\partial J}{\partial z_{cj}}=\sum_{x_i\in\pi_c}-2w_{jc}(x_{ij}-z_{cj})$，令其为 0 即可获得中心坐标为

$$z_{cj}=\frac{1}{n_c}\sum_{x_i\in\pi_c}x_{ij} \quad (7)$$

3.1.2 固定 $\{z_c\}_{c=1}^k$ 和 W，求解 $\{\pi_c\}_{c=1}^k$

固定 $\{z_c\}_{c=1}^k$ 和 W，问题(5)即为

$$\min_{\{\pi_c\}_{c=1}^k}\sum_{c=1}^{k}\sum_{x_i\in\pi_c}\sum_{j=1}^{d}w_{jc}(x_{ij}-z_{cj})^2 \quad (8)$$

最小化上式等价于执行如下的聚类划分：

$$\pi_c=\{x\in\{x_i\}_{i=1}^n\mid\varphi^2(x,z_c)\leqslant\varphi^2(x,z_l)\text{ for all } 1\leqslant l\leqslant k\} \quad (9)$$

即将每个数据点指派到一个使 $\varphi^2(x_i,z_c)$ 最小的聚类。

3.1.3 固定 $\{z_c\}_{c=1}^k$ 和 $\{\pi_c\}_{c=1}^k$，求解 W

现转向求 W，同上，固定 $\{z_c\}_{c=1}^k$ 和 $\{\pi_c\}_{c=1}^k$，问题(5)可重写为关于 W 的如下优化目标：

$$\min_{W\in\mathbb{R}^{d\times k},\xi\in\mathbb{R}^m} tr(WB)+\xi^{\mathrm{T}}u+\frac{1}{2}tr(W^{\mathrm{T}}W)$$

$$\text{s. t. } W\in\Delta_{d\times k} \quad (10)$$

$$AW1_k+\xi\geqslant\delta$$

$$\xi\geqslant 0$$

$B=[B_1,B_2,\cdots,B_d]\in\mathbb{R}^{k\times d}$，$B_j=\frac{1}{2\lambda_2}\sum_{c=1}^{k}\sum_{x_i\in\pi_c}e_c^k(x_{ij}-z_{cj})^2\in\mathbb{R}^k$，$u=\frac{\lambda_1}{2\lambda_2}1_m\in\mathbb{R}^m$，$A=[a_{ij}]\in\mathbb{R}^{m\times d}$ 和 $a_{ij}=\begin{cases}1, & s_i=j\\ -1, & t_i=j\\ 0, & \text{其他}\end{cases}$。$p_i=(s_i,t_i,\delta_i)$ 同上定义，表示一特征偏好元组。为求解主问题(10)，定义拉格朗日函数如下：

$$L(W,\xi,\eta,\alpha,\beta,\gamma)=tr(WB)+\xi^{\mathrm{T}}u+\frac{1}{2}tr(W^{\mathrm{T}}W)-\alpha^{\mathrm{T}}(AW1_k+\xi-\delta)-\beta^{\mathrm{T}}W1_k-\gamma^{\mathrm{T}}\xi-\eta^{\mathrm{T}}(W^{\mathrm{T}}1_d-1_k) \quad (11)$$

其中，η 和 α,β,γ 是分别对应等式和不等式约束的拉格朗日乘子，且 α,β,γ 均大于等于 0。现分别对式(11)的主变量 W、η、ξ 求导可得 $\frac{\partial L}{\partial W}=B^{\mathrm{T}}+W+A^{\mathrm{T}}\alpha 1_k^{\mathrm{T}}-\beta 1_k^{\mathrm{T}}-1_d\eta^{\mathrm{T}}$，$\frac{\partial L}{\partial\xi}=u-\alpha-\gamma$ 及 $\frac{\partial L}{\partial\eta}=W^{\mathrm{T}}1_d-1_k$，并令此 3 式均为 0，则解得 $W=A^{\mathrm{T}}\alpha 1_k^{\mathrm{T}}-(I_d-\frac{1}{d}1_d1_d^{\mathrm{T}})B^{\mathrm{T}}+(I_d-\frac{1}{d}1_d1_d^{\mathrm{T}})\beta 1_k^{\mathrm{T}}+\frac{1}{d}1_d1_k^{\mathrm{T}}$，将其代入原主问题 L 式即式(1)中，即变为求解如下对偶问题：

$$\min_{\alpha\in\mathbb{R}^m,\beta\in\mathbb{R}^d}(\alpha^{\mathrm{T}},\beta^{\mathrm{T}})\begin{pmatrix}\frac{k}{2}AA^{\mathrm{T}} & 0\\ kA^{\mathrm{T}} & \frac{k}{2}H\end{pmatrix}\begin{pmatrix}\alpha\\ \beta\end{pmatrix}-(\alpha^{\mathrm{T}},\beta^{\mathrm{T}})\begin{pmatrix}\delta+AB^{\mathrm{T}}1_k\\ HB^{\mathrm{T}}1_k-\frac{k}{d}1_d\end{pmatrix}$$

$$\text{s. t. } 0\leqslant\alpha\leqslant u$$

$$0\leqslant\beta \quad (12)$$

其中，$H=I_d-\frac{1}{d}1_d1_d^{\mathrm{T}}$ 且半正定，从而优化问题(12)是一个凸二次优化规划[15]，可用许多现成软件包来解之。在得到对偶问题(12)中 α,β 的最优解后，则主问题(10)的最优解为

$$W=A^{\mathrm{T}}\alpha 1_k^{\mathrm{T}}-HB^{\mathrm{T}}+H\beta 1_k^{\mathrm{T}}+\frac{1}{d}1_d1_k^{\mathrm{T}} \quad (13)$$

3.2 算法设计

通过上述结果，现给出算法 CDCFP 的具体实现步骤如下：

输入：数据集$\{x_i\}_{i=1}^n$，聚类数 k，特征顺序偏好 P，参数 λ_1 和 λ_2。

输出：聚类划分$\{\pi_c\}_{c=1}^k$。

Step1　初始化：初始化聚类数 k，聚类中心$\{z_c\}_{c=1}^k$及特征权重 $W=[\frac{1}{d},\cdots,\frac{1}{d}]\in\mathbb{R}^{d\times k}$。

Step2　迭代求解

2.1　先由$\{z_c\}_{c=1}^k$和 W 通过式(9)得出聚类划分$\{\pi_c\}_{c=1}^k$；

2.2　再由$\{\pi_c\}_{c=1}^k$和 W 通过式(8)得出聚类中心$\{z_c\}_{c=1}^k$；

2.3　最后由$\{z_c\}_{c=1}^k$和$\{\pi_c\}_{c=1}^k$通过式(13)更新权重 W。

重复步骤 2.1－2.3 直到算法的前后两次聚类目标值小于一给定阈值(实验中设定为 10^{-3})。

Step3　输出聚类划分$\{\pi_c\}_{c=1}^k$。

3.3　算法复杂性分析

本节分析算法 CDCFP 的时间复杂度。在 Step2 中，步骤 2.1 与 2.2 的时间复杂度分别为 $O(nkd)$与$O(nd)$。对于步骤 2.3，其时间复杂度则是 $O(L_2(m+d)^2)$，其中 L_2 是优化 α,β 时的迭代次数。若 L_1 是整个算法的迭代次数，则总体的时间复杂度为 $O(L_1(nkd+L_2(m+d)^2))$。

4　实验

为了验证算法 CDCFP 的有效性，进行了一系列实验，并给出了其与不加特征偏好的子空间聚类算法(FSC)[13]和基于 Bregman 散度的全局权重的特征偏好算法(CFP)[4,5]的性能比较。

4.1　实验设置

实际中，特征偏好通常可由用户先验给出或通过数据本身估算获得[4]。我们采用后者，即首先计算出各特征的类内方差 $\Theta_j=\sum_{c=1}^{k}\sum_{x_i\in\pi_c}(x_{ij}-z_{cj})^2$，据此获得逆畸变(inverse distortion) $\Gamma_j=\frac{\sum_{l\neq j}\Theta_j}{\Theta_j}$[4]，进而估计出最优权重$\widetilde{w}_j=\frac{\Gamma_j}{\sum_{l=1}^{d}\Gamma_l}$。在选择特征偏好对$(s,t)$时，我们遵循了文献[4]的设置方式，即从最大的$\lfloor\frac{d}{2}\rfloor$个权重中随机选取 m 个作为特征偏好中的 w_s，再从最小的$\lfloor\frac{d}{2}\rfloor$个特征中随机选取 m 个作为特征偏好的 w_t，形成一个偏好元组$(s,t,\widetilde{w}_s-\widetilde{w}_t)$。实验中 λ_1 和 λ_2 的值分别置为$\frac{d}{m}$和 d。

4.2　评价标准

本文采用两个常用的指标对聚类算法进行性能评估：1)归一化互信息(NMI)[4,16]，2)聚类精度(ACC)[4,18]。

已知聚类划分 C 和真实的分类 B，其中的簇(cluster)和类数均为 k。设 n_i 是第 i 个簇中的样本数，n_j'是第 j 个类中的样本数，n_{ij}是既在第 i 个簇又在第 j 个类中的样本数。则 NMI 可由如下方式计算：

$$NMI(C,B)=\frac{\sum_{i=1}^{k}\sum_{j=1}^{k}n_{ij}\log\frac{n\cdot n_{ij}}{n_i\cdot n_j'}}{\sqrt{\sum_{i=1}^{k}n_i\log\frac{n_i}{n}\sum_{j=1}^{k}n_j'\log\frac{n_j'}{n}}}\tag{14}$$

ACC 是指簇与类之间一一对应的正确关系。设函数 $map(i):\{i\}_{i=1}^k\to\{j\}_{j=1}^k$是将每一个簇映射到一个类的映射，则 C 和 B 之间的 ACC 定义为

$$AAC(C,B)=\frac{\max(\sum_{i=1}^{k}n_i,map(i))}{n}\tag{15}$$

ACC 和 NMI 的值越大，表明聚类性能越好。

4.3　实验结果

实验中使用 UCI[1]中的 6 个数据集，如表 1 所列。其中 n 为样本数，p 为特征数(维度)，k 为聚类数。

表 1　数据集

数据集	n	p	k
Iris	150	4	3
wdbc	569	30	2
Breast tissue	106	9	6
Wine	178	13	3
image segment	2100	16	7
vert. column	310	6	3

CDCFP、FCS 和 CFP 聚类性能的比较结果如表 2、表 3 所列。由表可见，加有特征偏好的 CFP 和 CDCFP 的聚类性能优于无偏好约束的 FSC；且随着偏好数 m 的增大，聚类性能越好；最后当每个数据集的 m 增至最大的 d 时，获得了最好的聚类性能，这与直觉吻合。另外在大部分情况下，CDCFP 的性能要优于 CFP，表明基于聚类依赖的局部权重算法性能要优于基于全局权重的聚类算法。

表 2　NMI 评价标准下不同模型的实验对比结果

	NMI				
	FSC		m=d/4	m=d/2	m=d/1
Iris	0.7473	CFP	0.7381	0.8265	**0.8642**
		CDCFP	0.8023	0.8028	0.8038
wdbc	0.5698	CFP	0.6113	0.6182	0.6276
		CDCFP	0.6086	0.6296	**0.6503**
Breast tissue	0.5372	CFP	0.4906	0.5236	**0.5598**
		CDCFP	0.5171	0.4901	0.5227
Wine	0.7497	CFP	0.7647	0.7658	0.7699
		CDCFP	0.7964	0.7926	**0.7930**
image segment	0.5770	CFP	0.5769	0.5977	0.6145
		CDCFP	0.5960	0.6072	**0.6188**
vert. column	0.2733	CFP	0.3022	0.3147	**0.3287**
		CDCFP	0.2513	0.2631	0.2823

表 3　ACC 评价标准下不同模型的实验对比结果

	ACC				
	FSC		m=d/4	m=d/2	m=d/1
Iris	0.8203	CFP	0.8913	0.9371	**0.9600**
		CDCFP	0.9326	0.9328	0.9333
wdbc	0.8901	CFP	0.9225	0.9237	0.9255
		CDCFP	0.9192	0.9315	**0.9350**
Breast tissue	0.5184	CFP	0.5226	0.5326	0.5617
		CDCFP	0.5660	0.5755	**0.5943**
Wine	0.8867	CFP	0.9213	0.9227	0.9263
		CDCFP	0.9313	0.9364	**0.9377**
image segment	0.5255	CFP	0.5754	0.5867	0.6040
		CDCFP	0.5989	0.6059	**0.6134**
vert. column	0.4805	CFP	0.5863	0.5897	0.5930
		CDCFP	0.6355	0.6710	**0.6742**

表 4 和表 5 分别列举了在数据集 vert. column 上，算法 FSC 和 CDCFP 的对应特征权重。

表 4　由 FSC 在 vert. column 上生成的特征权重

特征	权重
x_1	0.5742
x_2	0.3261
x_3	0.4514
x_4	0.5294
x_5	0.3043
x_6	0.8146

表 5　由 CDCFP 在 vert. column 上生成的特征权重

特征	k=1	k=2	k=3	sum
x_1	0.1449	0.2316	0.1971	0.5736
x_2	0.3011	0.1046	0.2825	0.6882
x_3	0.1857	0.1663	0.1641	0.5161
x_4	0.1374	0.2382	0.1880	0.5363
x_5	0.1845	0.0791	0.0413	0.3049
x_6	0.0465	0.1802	0.1269	0.3537

下面我们考察如果不施加偏好约束，聚类是否会自动产生与实际相吻合的结果。根据 4.1 节中的偏好生成方式，我们通过计算获得了 vert. column 的特征偏好阵为 $A=\begin{bmatrix}0 & 0 & 1 & 0 & -1 & 0\\ 0 & 1 & 0 & 0 & 0 & -1\\ 1 & 0 & 0 & 0 & 0 & -1\end{bmatrix}$，即偏好为 $x_3>x_5$，$x_2>x_6$，$x_1>x_6$。再由表 4 可知，在无特征偏好约束的 FSC 算法中，获得的特征权重并不满足所有的偏好要求，如 $x_2>x_6$，$x_1>x_6$。可见仅仅通过加权的聚类无法保证所形成的偏好能产生期望的效果。由表 5 进一步发现，特征 x_5 和 x_6 在不同的聚类中表现出了不同的重要程度，因而说明特征是聚类依赖的，仅仅限定全局的权重不足以反映出数据内在结构的多样性和真实性。

总体而言，所提算法 CDCFP 在实验所用的数据集上展示出了更优的聚类性能。

结束语　本文提出了一个基于特征偏好的聚类算法 CDCFP。相比现有的全局特征偏好算法，CDCFP 在权重方面考虑了更为实际的特征与聚类间的依赖关系，在距离度量时增加了聚类依赖的局部权重，使改进的算法获得了更好的性能。在实验中，由于在聚类前无法获得具体的聚类依赖的特征间的偏好关系，因此在实现时仍采用了全局的特征偏好，故探索如何先验地确定聚类依赖的特征偏好是我们下一步的研究内容之一。同时进一步将 CDCFP 应用于高维数据的特征偏好学习也是留待需做的工作。

参 考 文 献

[1] Asuncion A, Newman D. UCI machine learning repository [Z]. 2007

[2] Wang J, Wang S T, Deng Z H. A novel text clustering algorithm based on feature weighting distance and soft subspace learning [J]. Jisuanji Xuebao (Chinese Journal of Computers), 2012, 35(8): 1655-1665

[3] Andrews J L, McNicholas P D. Variable Selection for Clustering and Classification [J]. Journal of classification, 2014, 31(2): 136-153

[4] Sun J, Zhao W, Xue J, et al. Clustering with feature order preferences[J]. Intelligent Data Analysis, 2010, 14(4): 479-495

[5] Chen X, Ye Y, Xu X, et al. A feature group weighting method for subspace clustering of high-dimensional data [J]. Pattern Recognition, 2012, 45(1): 434-446

[6] Jain A K, Dubes R C. Algorithms for clustering data [M]. Prentice-Hall, Inc., 1988

[7] Witten D M, Tibshirani R. A framework for feature selection in clustering [J]. Journal of the American Statistical Association, 2010, 105(490)

[8] Banerjee A, Merugu S, Dhillon I S, et al. Clustering with Bregman divergences [J]. The Journal of Machine Learning Research, 2005, 6: 1705-1749

[9] Jain A K. Data clustering: 50 years beyond K-means [J]. Pattern Recognition Letters, 2010, 31(8): 651-666

[10] Bezdek J C. Pattern recognition with fuzzy objective function algorithms[M]. Kluwer Academic Publishers, 1981

[11] Luo P, Zhan G, He Q, et al. On defining partition entropy by inequalities[J]. IEEE Transactions on Information Theory, 2007, 53(9): 3233-3239

[12] Liu Y, Jin R, Jain A K. Boostcluster: Boosting clustering by pairwise constraints[C]// Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2007: 450-459

[13] Gan G, Wu J. A convergence theorem for the fuzzy subspace clustering (FSC) algorithm [J]. Pattern Recognition, 2008, 41(6): 1939-1947

[14] Shi J, Malik J. Normalized cuts and image segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(8): 888-905

[15] Boyd S P, Vandenberghe L. Convex optimization [M]. Cambridge university press, 2004

[16] Wu M, Schölkopf B. A local learning approach for clustering[C]// Advances in Neural Information Processing Systems. 2006: 1529-1536

[17] Bertsekas D P. Nonlinear programming(2nd Edition)[M]. 1999

[18] Strehl A, Ghosh J. Cluster ensembles—a knowledge reuse framework for combining multiple partitions[J]. The Journal of Machine Learning Research, 2003, 3: 583-617

[19] Reynolds D. Gaussian mixture models[M]// Encyclopedia of Biometrics. Springer US, 2009: 659-663

[20] McLachlan G J, Peel D. Robust cluster analysis via mixtures of multivariate t-distributions [C]// Advances in pattern recognition. Springer Berlin Heidelberg, 1998: 658-666

[21] Reed J W, Potok T E, Patton R M. A multi-agent system for distributed cluster analysis[C]// Proceedings of Third International Workshop on Software Engineering for Large-Scale Multi-Agent Systems (SELMAS'04) Workshop in conjunction with the 26th International Conference on Software Engineering Edinburgh. Scotland, UK: IEEE, 2004: 152-155

[22] Coddington P D, Baillie C F. Parallel cluster algorithms [J]. Nuclear Physics B-Proceedings Supplements, 1991, 20: 76-79

[23] Makarenkov V, Legendre P. Optimal variable weighting for ultrametric and additive trees and K-means partitioning: Methods and software [J]. Journal of Classification, 2001, 18(2): 245-271

[24] Huang J Z, Ng M K, Rong H, et al. Automated variable weighting in k-means type clustering [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, 27(5): 657-668

[25] Tsai C Y, Chiu C C. Developing a feature weight self-adjustment mechanism for a K-means clustering algorithm [J]. Computational statistics & data analysis, 2008, 52(10): 4658-4672

[26] Wolpert D H, Macready W G. No free lunch theorems for optimization [J]. IEEE Transactions on Evolutionary Computation, 1997, 1(1): 67-82

[27] Fu J, Chu S, Han Z, et al. Improved Genetic Algorithm Based on Variable Weighting FCM Clustering Algorithm[C]// Proceedings of the 9th International Symposium on Linear Drives for Industry Applications. Volume 2, Springer Berlin Heidelberg, 2014: 671-677

[28] Chen X, Ye Y, Xu X, et al. A feature group weighting method for subspace clustering of high-dimensional data [J]. Pattern Recognition, 2012, 45(1): 434-446

[29] Xiong C, Johnson D, Corso J J. Online active constraint selection for semi-supervised clustering[C]// ECAI 2012 AIL Workshop. 2012