基于可变加权的高维数据子空间聚类算法研究

邓 莹，杨双远，刘 菡
(厦门大学 软件学院，福建 厦门 361005)

摘 要：高维数据的稀疏性和“维灾”问题使得多数传统聚类算法失去作用，因此研究高维数据集的聚类算法已成为当前的一个热点。子空间聚类算法是实现高维数据集聚类的有效方法之一。介绍并实现了基于可变加权的高维数据子空间聚类算法SCAD和EWKM，并分别对人造数据、现实数据等数据集进行测试，根据测试结果进行分析，对比两种算法的性能及适用场合。

关键词：高维数据；稀疏；子空间聚类；精确率；熵
中图分类号：TP311 **文献标识码**：A

Study of subspace clustering algorithm of high dimensional data based on variable weighting methods

DENG Ying，YANG ShuangYuan，Liu Han
(Software School，Xiamen University，Xiamen 361005，China)

Abstract：The sparsity and the problem of the curse of dimensionality of high-dimensional data，make the most of traditional clustering algorithms lose their action in high-dimensional space. Therefore，clustering of data in a high-dimensional space becomes a hot research area. Subspace clustering algorithm is one of the effective ways to handle problems of high-dimensional data clustering. This paper introduces and realizes two algorithms (SCAD and EWKM) that discover clusters in subspaces spanned by different combinations of dimensions via local weightings of features. We experiment these algorithms using synthetic datasets and real datasets，then analyze the results and contrast their performance and applicable occasions.

Key words：high dimensional data; sparsity; subspace clustering; precision; entropy

聚类是一种无监督分类，即按照事物的某些属性，将数据划分成有意义或有用的类(或称为“簇”)，使类间的相似性尽可能小，类内相似性尽可能大。聚类在模式识别、数据分析、图像处理和市场研究等领域都有着重要的应用。在这些应用中，经常会碰到一些高维数据，比如购物篮数据、文档数据、多媒体数据等[1]。然而在高维空间中，传统聚类算法的性能直接受到维度的影响：一是经常存在一些干扰特征误导聚类算法的执行；二是高维空间的数据分布大多比较稀疏，在这些数据中基于距离的聚类结构很难被区分开。为了克服这一困难，可以使用特征(或属性)变换和特征(或属性)选择技术[2]。

传统的特征选择算法可用来确定相关维，然而在高维空间中并不是所有的维都与给定的类有关，不同的类可能对应不同的子空间，并且每个子空间的维数也可能不同。因此不可能在一个子空间中发现所有的类。为了解决这个问题，对全空间聚类问题进行了推广，称为“子空间聚类”或“投影聚类”，意在发现数据集中所有的类以及它所蕴涵的子空间。子空间聚类是特征子集选择的一种扩展，它在高维数据聚类方面显示出了优势。

1 软子空间聚类

区别于传统的聚类方法，子空间聚类的主要挑战在于要同时测定目标的类成员和每个类的子空间。类成员是由关于子空间的目标的相似性度量来决定。根据决定类的子空间的方法，子空间聚类方法可分为硬子空间聚类和软子空间聚类两种类型。软子空间聚类是在整个数据空间对目标数据聚类。但在聚类过程中，根据这些维对相应类的重要性，对类的不同维指定不同的加权值。在一次聚类中，每一维对每一个类

都有贡献，但是具有较大权值的维构成聚类的维度子空间。一些可变加权方法的扩展，可以实现软子空间聚类的功能。它们只是在传统的聚类过程的每一次循环迭代中，另外附加一步权重值的计算，从而获得不同分类的不同权重变量集合。接下来介绍并实现2种分别基于k-means和FCM的高维数据可变加权聚类算法。

2 基于可变加权的高维数据软子空间聚类算法

定义加权聚类[3]：考虑一组在某一个D维空间的数据点集，例如加权类C是一个数据点子集，加权向量$w=(w_1, w_2, \cdots, w_D)$，因此根据使用$w$的$L_2$标准加权距离$C$中的点是紧密聚集的。元素$w_j$用来度量特征$j$对类$C$的参与程度。如果$C$中的点在特征$j$上都很好地聚集，$w_j$的值就大，反之它的取值较小。因此现在的问题就变为在数据集中如何为每一个分类估计权重向量w。

在这一方面，类的概念不仅仅取决于数据点而且包含一个加权距离指标，即类形成于由w转化的空间。每一个类有它自己的w，它反应了在类中的数据点的相关性。w的作用是变换距离从而使相关的类被重塑成一个数据点密集的超球体以和其他的数据分离。

2.1 同步聚类和特征识别算法

2.1.1 变量描述

同步聚类和属性识别SCAD(Simultaneous Clustering and Attribute Discrimination)算法[3]是设计用来同时地搜索最优聚类中心C和最优特征权重集W。每一个类i有它自己的特征权重集$W_i=[w_{i1}, w_{i2}, \cdots, w_{iD}]$。其目标函数定义如下：

$$J(C,U,W;x)=\sum_{i=1}^{C}\sum_{j=1}^{N}u_{ij}^{m}\sum_{k=1}^{D}w_{ik}(x_{jk}-c_{ik})^2+\sum_{i=1}^{C}\delta_i\sum_{k=1}^{D}w_{ik}^2 \quad (1)$$

其中：$\begin{cases} u_{ij}\in[0,1],\forall i \\ 0<\sum_{j=1}^{N}u_{ij}<N,\forall i,j \\ \sum_{i=1}^{C}u_{ij}=1,\forall j \end{cases}$

且$w_{ik}\in[0,1],\forall i,k;\sum_{k=1}^{D}w_{ik}=1,\forall i$

对于w_{ik}，有：

$$w_{ik}=\frac{1}{n}+\frac{1}{2\delta_i}\sum_{j=1}^{N}(u_{ij})^m[\frac{\|x_j-c_i\|^2}{n}-(x_{jk}-c_{ik})^2] \quad (2)$$

δ_i计算公式为

$$\delta_i=K\frac{\sum_{j=1}^{N}(u_{ij})^m\sum_{k=1}^{D}w_{ik}(x_{jk}-c_{ik})^2}{\sum_{k=1}^{D}(w_{ik})^2} \quad (3)$$

修正的隶属度等式类似于FCM的隶属度函数，即，

$$u_{ij}=\frac{1}{\sum_{k=1}^{C}(\frac{\bar{d}_{ij}^2}{\bar{d}_{kj}^2})^{\frac{1}{m-1}}} \quad (4)$$

关于聚类中心，可表达为：

$$c_{ik}=\begin{cases}0, & w_{ik}=0\\ \frac{\sum_{j=1}^{N}(u_{ij})^m x_{jk}}{\sum_{j=1}^{N}(u_{ij})^m}, & w_{ik}>0\end{cases} \quad (5)$$

2.1.2 算法描述

输入：N个点$x\in R^D$，聚类数目c、模糊度指数$m(m\in[1, \infty])$，停机误差$\varepsilon(\varepsilon>0)$，最大循环次数MAXITER；

(1)初始化聚类中心$C^{(0)}$和模糊分割矩阵$U^{(0)}$；

(2)用公式(2)更新特征权重集$W^{(t)}$；

(3)用公式(4)更新分割矩阵$U^{(t)}$；

(4)用公式(5)更新聚类中心$C^{(t)}$；

(5)用公式(3)更新$\delta_i^{(t)}$；

(6)计算误差$E^{(t)}=\sum_{i=1}^{c}\|C_i^{(t)}-C_i^{(t-1)}\|^2$；

(7)若$E^{(t)}<\varepsilon$，算法终止，否则重复执行(2)~(7)。

2.2 高维稀疏数据子空间聚类的K-Means熵加权算法

2.2.1 变量描述

在高维稀疏数据软子空间聚类的K-Means熵加权算法EWKM(An Entropy Weighting K-Means Algorithm for Subspace Clustering of High-Dimensional Sparse Data)[4]中，一个类中的某一维的权重代表该维对构成这一类的贡献概率。这一维权重的熵代表该维在这一类的识别中的可能性。因此，修改目标函数，在其中添加权重熵项，可以同时得到类内分散度的最小值和负的权重熵的最大值，以刺激更多的维对类的识别做出贡献。本方法可以避免只由稀疏数据中的几个维来识别聚类的问题。其目标函数定义如下：

$$F(U,C,W)=\sum_{l=1}^{k}[\sum_{j=1}^{N}\sum_{i=1}^{D}u_{ij}w_{li}(c_{li}-x_{ji})^2+\gamma\sum_{i=1}^{D}w_{li}\log w_{li}] \quad (6)$$

满足如下约束条件：

$$\begin{cases}\sum_{l=1}^{k}u_{li}=1,1\le j\le N,1\le l\le k,u_{lj}\in\{0,1\}\\ \sum_{i=1}^{D}w_{li}=1,1\le l\le k,1\le i\le D,0\le w_{li}\le 1\end{cases}$$

类似于k-means算法，分割矩阵U可用下式更新：

$$\begin{cases}u_{lj}=1, if\sum_{i=1}^{D}w_{li}(c_{li}-x_{ji})^2\le\sum_{i=1}^{D}w_{ri}(c_{ri}-x_{ji})^2, 1\leqslant r\leqslant k\\ u_{lj}=0, \quad \text{stherwise}\end{cases} \quad (7)$$

聚类中心C的更新公式为：

$$c_{li}=\frac{\sum_{j=1}^{n}u_{lj}x_{ji}}{\sum_{j=1}^{n}u_{lj}}, \quad 1\leqslant l\leqslant k且1\leqslant i\leqslant D \quad (8)$$

权重集W的计算公式为：

$$w_{lt}=\frac{\exp(\frac{-D_{lt}}{\gamma})}{\sum_{\omega}^{D}\exp(\frac{-D_{li}}{\omega})},\quad (9)$$

其中，$D_{lt}=\sum_{j=1}^{n}u_{lj}(c_{lt}-x_{jt})^2$

2.2.2 算法描述

输入：N个点$x\in R^D$，聚类数目k和参数$\gamma(\gamma>0)$，停机误差$\varepsilon(\varepsilon>0)$，最大循环次数MAXITER。

(1)初始化聚类中心$C^{(0)}$，并设初始权重为$1/m$；

(2)用公式(7)更新分割矩阵$U^{(t)}$；

(3)用公式(8)更新聚类中心$C^{(t)}$；

(4)用公式(9)更新特征权重集$W^{(t)}$；

(5)计算误差$E^{(t)}=\sum_{i=1}^{c}\|C_i^{(t)}-C_i^{(t-1)}\|^2$；

(6)若$E^{(t)}<\varepsilon$，算法终止，否则重复执行步骤(2)~(6)。

3 仿真分析

3.1 聚类效果评价标准

为了衡量聚类算法的有效性，用到2个评价标准：精确率和熵[5]。

3.1.1 精确率

设对于给定人类判定$PT=\{S_1\cdots, S_2\cdots, S_i, \cdots, S_K\}$和聚类算法结果$PT^c=\{PT_1, PT_2, \cdots, PT_r, \cdots, PT_R\}$，$PT_{ri}=PT_r\cap S_i$,$n_{ri}$、$n_r$、$n_i$分别是$PT_{ri}$、$PT_r$和$PT_i$的集合的大小，则$PT_r$对于标准判定结果的查准率$P$定义为：

$$P(PT_r,S_i)=\frac{n_{ri}}{n_r}$$

定义查全率为：

$$R(PT_r,S_i)=\frac{n_{ri}}{n_i}$$

综合查全率和查准率衡量，得到聚类结果F度量(F measure)

$$F(PT_r,S_i)-\frac{2\times P(PT_r,S_i)\times R(PT_r,S_i)}{P(PT_r,S_i)+R(PT_r,S_i)}$$

由此定义聚类算法所获的类PT_r的$FScore$ 如下。

$$FScore(PT_r)=\max_{S_i\in PT}F(PT_r,S_i)$$

由此可得知，判定一个类的所属类别，是看它与标准判定中交集最大的那个类所属的类别。这样获得了聚类结果的评分：

$$FScore=\sum_{r=1}^{R}\frac{n_r}{N}FScore(PT_i)\quad (10)$$

公式(10)有时也被称为精确率(Precision)。它表达了聚类结果与人类判定结果的接近程度，聚类结果越接近人类判定，其取值越大，容易得出其最大值为1，此时聚类结果与人类判定一致。

3.1.2 熵

另一个比较有效的评价函数是熵(Entropy)值函数。沿用上面的符号和含义，定义一个聚类PTr 的熵值为式(4–2)。

$$E(PT_r)=-\frac{1}{\log K}\sum_{k=1}^{K}\frac{n_{ri}}{n_r}\log\frac{n_{ri}}{n_r}\quad (11)$$

$$E=\sum_{r=1}^{R}\frac{1}{R}E(PT_r)\quad (12)$$

这样一个聚类结果PT^c的熵值表达为公式(12)。熵值标准刻画了聚类结果的杂乱程度，如果一个在人类判定下属于同一个簇的数据对象被聚类算法划分在很多的不同簇中时，聚类结果的杂乱程度高，熵值也会很大。当聚类结果与给定的人类判定完全一致时，其熵值为0。

3.2 仿真设计

如表1所示，仿真实验采用了一个人造高斯聚类数据集d_4[5]和来自于[6]的三个真实数据集。人造高斯聚类数据集采用4维40个数据，共2类，其中每一个聚类各有2个不相关的特征，采用这个数据集来验证两种算法的聚类和识别相关特征的能力。IRIS[6]数据集两类间存在交迭。这对验证两种算法的聚类准确度提供了一定的依据。

表1 数据集

数据集	样本点数n	维度d	聚类数k
d_4	40	4	2
IRIS	150	4	3
Wine	178	13	3
Musk	476	168	2

3.3 仿真结果及分析

对于人造数据集和真实数据集，2种算法的实验结果如表2和图1~图4所示。

表2 各数据集聚类结果

	SCAD		EWKM	
	精确率	熵	精确率	熵
d_4	0.90	0.33	0.95	0.22
IRIS	0.98	0.10	0.92	0.15
Wine	0.97	0.11	0.87	0.20
Musk	0.939	0.27	0.977	0.12

由表2可知，对于人造数据集，EWKM算法效果最好，数据划分精确度高且杂乱度最小，并且能识别出不同特征的相关性。对于IRIS数据集，SCAD算法在划分效果方面是最佳的，精确率高且杂乱度最小，这主要是因为SCAD算法采用了模糊隶属度函数，对于IRIS这种有交迭情况的数据集能够比较好地处理。对于wine数据集SCAD算法的精确率比EWKM算法更高，而对于Musk这样较大的数据集，基于K–Mean的EWKM算法具有相对可伸缩性和效率高的特点，因此聚类效果最好且效率高，而基于FCM的SCAD算法则执行代价较高。

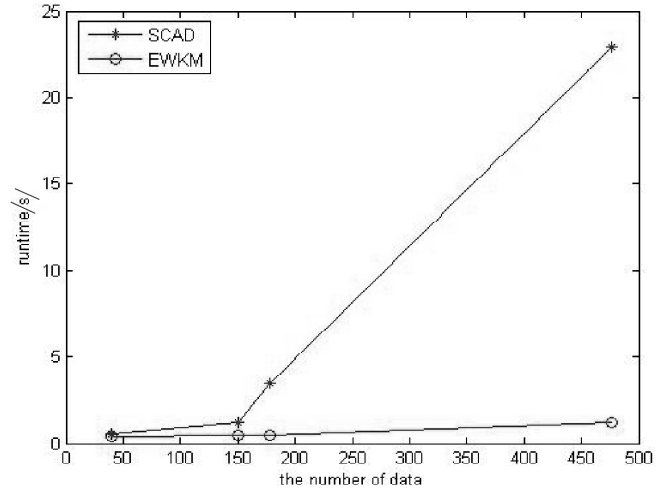

图 1　运行时间与样本数的关系

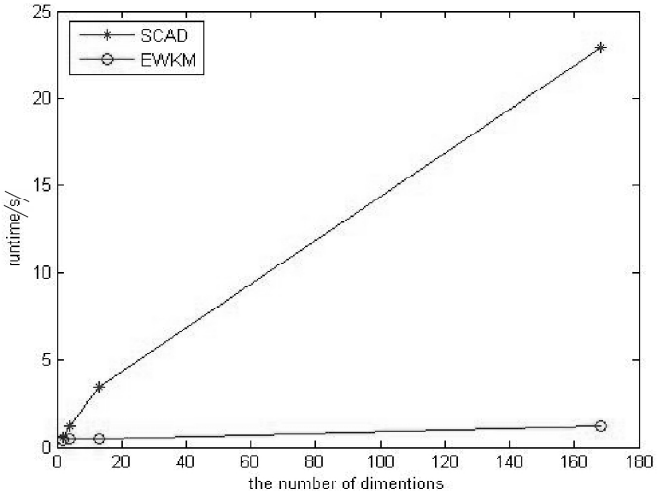

图 2　运行时间与维度的关系

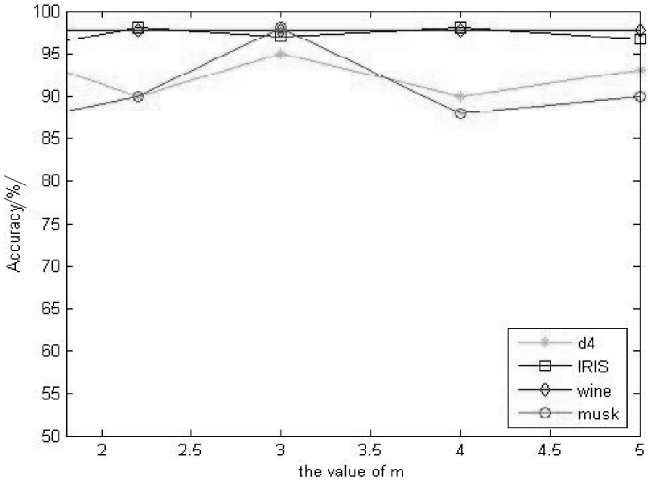

图 3　SCAD 算法精确率与参数的关系

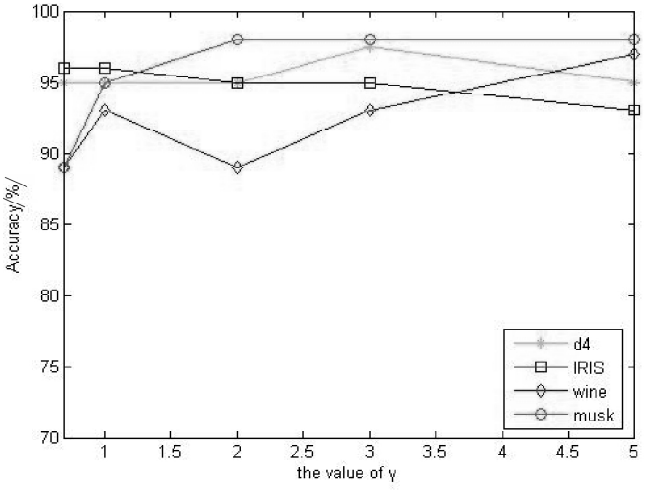

图 4　EWKM 算法精确率与参数的关系

图1和图2显示了2种算法的运行时间与样本数、维度的关系。实验结果表明，随着样本数和维度的增加，EWKM算法的运行时间呈线性增长，比较平稳。而SCAD算法由于采用了类FCM算法，运行时间变化较大。

图3和图4显示了算法精确率与参数选择的关系。实验结果表明，在5个数据集上，当参数在一个较大范围内变化时，算法的精确率改变不敏感，这说明SCAD算法和EWKM算法的鲁棒性较好。

以上介绍的2种算法都是先初始化聚类中心并在实现过程中不断更新中心点及权重向量直至收敛。通过测试实验发现这2种算法都能较好地处理高维数据的聚类问题。其中EWKM算法的运行效率最高，平均性能最好，尤其是权重结果能比较准确地反映特征与聚类中心的相关性，这对进一步进行特征选择提供了很大的方便。SCAD算法能够适应存在于数据集中的变化将它分到不同的类，因此对于维度较高且分布在整个数据空间的数据集，SCAD算法的聚类效果最好，尤其是由于SCAD算法使用模糊隶属度，对类间有重叠的数据集的聚类效果也是最佳的，但缺点是运行时间较长。当然，所讨论的算法均需预先指定聚类数目，因此未来聚类算法的研究重点将会侧重于聚类数目对聚类效果的影响，以及如何自动确定最佳聚类数目上。

参考文献

[1] 杨风召. 高维数据挖掘中若干关键问题的研究[D]. 上海:复旦大学,2003.

[2] HAN Jia Wei,MICHELINE K . 数据挖掘概念与技术[M]. 范明,孟小峰,译 . 北京:机械工业出版社,2007.

[3] FRIGUI H, NASRAOUI O, Simultaneous clustering and attribute discrimination[C]. Proceeding of the 9th IEEE International Conference on Fuzzy Systems,2000.

[4] JING L. NG M. K. and HUANG. J. Z. An Entropy Weighting K-Means algorithm for subspace clustering of high-dimensional sparse data[J]. IEEE Transactions on Knowledge and Data Engineering,2007,19(8):1-16.

[5] 赵万磊. 典型聚类算法及其应用研究[D]. 昆明:云南大学,2005.

[6] 测试数据集. http://archive. ics. uci. edu/ml/machine-learning-databases.

(收稿日期：2009-01-08)