第 56 卷　第 4 期
2020 年 7 月

南京大学学报(自然科学)
JOURNAL OF NANJING UNIVERSITY
(NATURAL SCIENCE)

Vol. 56, No. 4
July, 2020

DOI:10.13232/j.cnki.jnju.2020.04.009

基于迁移学习的软子空间聚类算法

王丽娟[1,2],丁世飞[1*],丁　玲[1]

(1.中国矿业大学计算机科学与技术学院,徐州,221116;2.徐州工业职业技术学院信息与电气工程学院,徐州,221400)

摘　要:随着大数据时代的到来,大量的高维数据在生活中无处不在.聚类是分析描述数据并按照某种相似性将数据归类的一项技术.传统聚类算法在面对高维数据时,往往无法进行有效的聚类处理.软子空间聚类是通过分配权重,描述样本隶属于不同簇的不确定性来进行聚类,然而,当数据残缺或信息不准时,现有的软子空间聚类的准确度和效率会受到很大的影响.从软子空间聚类面临的问题出发,提出一种改进的软子空间聚类算法;同时针对数据残缺不足的问题,引入迁移学习来削弱数据量不足对聚类分析的影响;通过引入信息熵的概念,用信息熵确定高维数据权重.实验证明,通过结合迁移学习和信息熵,有效地提高了软子空间聚类算法精确度和准确度.

关键词:子空间聚类,迁移学习,信息熵,高维数据

中图分类号:TP301　　**文献标识码:**A

Soft subspace clustering algorithm based on transfer learning

Wang Lijuan[1,2], Ding Shifei[1*], Ding Ling[1]

(1. School of Computer Science and Technology, China University of Mining and Technology, Xuzhou, 221116, China;
2. School of Information and Electrical Engineering, Xuzhou College of Industrial Technology, Xuzhou, 221400, China)

Abstract: With the advent of the era of big data, a large number of high-dimensional data have become very common. Clustering is a technique of analyzing, describing and classifying data according to some similarity. When faced with high-dimensional data, the traditional clustering algorithms are often unable to carry out effective clustering processing. Soft subspace clustering is based on the distribution of weights to describe the uncertainty of samples belonging to different clusters. However, the accuracy and efficiency of existing soft subspace clustering will be significantly affected when the data is incomplete or the information is not timely. Starting from the problems faced by soft subspace clustering, this paper proposes an improved soft subspace clustering algorithm. At the same time, aiming at the problem of insufficient data, we introduce migration learning to reduce the impact of insufficient data on clustering analysis. By introducing the concept of information entropy, we use information entropy to determine the weight of high-dimensional data. By combining migration learning and information entropy, the accuracy and accuracy of soft subspace clustering algorithm are effectively improved.

Key words: subspace clustering, transfer learning, information entropy, high dimensional data

大数据时代无时无刻地进行着海量的数据和信息交换,如何从海量的高维数据中挖掘提取有价值的信息是近年讨论的热点问题.数据聚类分析是数据挖掘的有效工具之一,是数据挖掘领域研究的重点和热点[1-3].聚类分析是一种通过算法自动分析数据对象之间的相似性或者相异性、自动地将数据集中未标记的数据分到不同的簇之中的方法.每个簇中的数据在某个标准下具有一

基金项目:国家自然科学基金(61672522,61976216),2020年江苏省高校"青蓝工程"
收稿日期:2020－06－20
*通讯联系人,E-mail:dingsf@cumt.edu.cn

定的相似性,而簇间的数据在这一标准下的相似性则很低[4]. 这种方法的用途是对原始的数据集合进行处理,得到一种聚类处理结果,再通过对聚类结果的分析,提取人们需要的有价值的信息. 目前,聚类分析已被广泛应用到各个领域:在商业领域,聚类分析可以被用来发现不同的客户群,研究不同客户的消费行为,寻找潜在市场来制定不同的销售方案[5-6];在生物医学领域,聚类分析能够对基因进行分类,从而研究不同的种群结构,分析与各种疾病之间的潜在联系[7];在电子商务类行业,聚类分析能从网站建设的数据中挖掘分析出各个客户的相似习惯,达到优化服务的目的.

近年来,在各个应用领域的实际数据都呈现维度剧增的趋势,数据呈现高维化发展的态势并因此爆发了"维度灾难"[8-10]. 高维数据比低维有更多的难以处理的特性,比如在高维数据中,判断数据样本之间的相似性非常困难,因为数据样本之间的距离几乎一致,这是数据在高维空间的分布越来越稀疏导致的;其次,高维数据有大量的子属性,这些子属性中存在一些与特定簇无关或者冗余的属性,导致进行聚类时不同的子空间可能发现不同的簇的问题[11];并且,随着维数的不断增加,每个维度的取值将会呈现指数级别的增长,很难完全枚举所有的子空间. 因此在高维数据领域,传统的聚类方法的表现并不理想. Agrawal et al[12]在SIGMOD会议上提出子空间聚类的概念. 子空间聚类是对传统聚类的扩展,能从高维数据集中发现隐藏在不同低维子空间中的簇类. 子空间聚类将原始数据集划分成不同的簇并同时搜索各个簇的子空间,对各个簇中关联的各个属性赋予不同的权重,从而研究属性与簇的关联程度. 子空间聚类算法又分硬子空间(Hard subspace)和软子空间(Soft subspace)[13-16]. 硬子空间是采用自底向上或者自顶向下的搜索策略,按照一定的标准在源数据集的所有特征集中选取精确的特征子集组成子空间并进行聚类. 对高维数据的聚类算法就是从硬子空间开始研究的,并且已经取得了很大成果,所以硬子空间聚类已经相对成熟,如CLIQUE算法[17]、PROCLUS算法[18]等. 软子空间聚类则是在硬子空间聚类之后慢慢发展起来的,因为其在面对高维数据时有更好的适应性,因而引起国内外学者的广泛关注[19-20]. 软子空间算法为簇类各个特征赋予不同的权值,从而获知簇类与全特征空间中哪些特征具有相关性,并且反应各个特征与簇的相关程度与差异,为每个簇寻找一个模糊子空间. 与硬子空间相比,面对高维数据时软子空间有更好的适应性与灵活性.

迁移学习是一种在已有的环境中认知和学习到的信息被应用到新的任务和环境下的能力. 迁移学习作为一种能利用其他相似领域上学到的知识来辅助当前任务的一种方法,被广泛运用于各个邻域中[21]. 根据源任务和目标任务之间的差异,可将迁移学习大致分为归纳式迁移学习、直推式迁移学习和无监督学习[22]. 在聚类分析算法的过程中,需要大量已知数据支持,而实际情况下,很多时候会出现已知数据样本不足、数据残缺或者信息不准确的情况[23]. 因此本文引入熵的概念,根据信息熵来确定权重,并将迁移学习与子空间聚类算法结合,利用迁移学习改进优化软子空间聚类算法的聚类性能.

1 基础理论

1.1 子空间聚类算法

1.1.1 软子空间聚类算法 在传统的软子空间聚类算法中,所有的簇类共享相同的子空间和权重向量,例如WK-Means(Weights K-Means)算法和WFCM(Weighting Fuzzy C-Means)算法.

WFCM算法的全称为样本加权模糊C均值算法,它是对FCM((Fuzzy C-means))算法的改进[24]. FCM算法基于传统欧式距离,每个数据样本对聚类的贡献几乎相同,然而实际上在高维领域,每个数据样本都会对聚类产生不同的程度的影响. 用传统的FCM算法无法体现噪声点或者偏远数据样本集体对聚类的影响,所以WFCM引入一种点密度函数来作为样本点的加权系数计算方法,对于每个样本点,点密度函数计算方式为:

$$z_i = \sum_{j=1,j\neq i}^{n} \frac{1}{di_j} \tag{1}$$

$$d_{ij} = \left\| x_i - x_j \right\|, 1 \leqslant i \leqslant n, 1 \leqslant j \leqslant n \tag{2}$$

其中,d_{ij}表示两个样本点之间的欧式距离,若数据样本点周围点越多,则z的值越大. 用W_i表示第i个样本X_i对分类的影响程度:

$$W_i=\frac{z_i}{\sum_{j=1}^{n}z_j},1\leqslant i\leqslant n \tag{3}$$

将 W_i 引入到FCM的目标函数中得到新的目标函数公式:

$$J(u,v,w)=\sum_{i=1}^{c}\sum_{j=1}^{n}w_j u_{ij}^m d_{ij}^2 \tag{4}$$

聚类中心点 v_i 和模糊隶属度 u_{ij} 的更新公式如下:

$$v_i=\frac{\sum_{j=1}^{n}w_j u_{ij}^m x_j}{\sum_{j=1}^{n}w_j u_{ij}^m},1\leqslant i\leqslant c \tag{5}$$

$$u_{ij}=\left(\sum_{k=1}^{c}\left(\frac{\left\|v_i-x_j\right\|}{\left\|v_k-x_j\right\|}\right)^{\frac{2}{m-1}}\right)^{-1},1\leqslant i\leqslant c,1\leqslant j\leqslant n \tag{6}$$

WFCM的算法流程如图1所示.

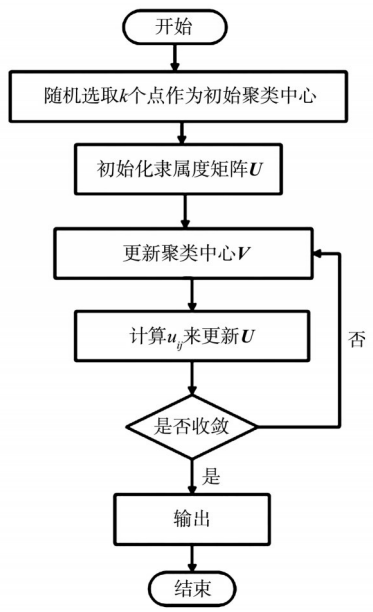

图1　WFCM的算法流程图

Fig. 1　Flow chart of WFCM algorithm

1.1.2　扩展软子空间聚类算法　扩展的软子空间聚类算法[25]通过引入新的机制来进一步优化提升传统软子空间聚类或者独立软子空间聚类算法的聚类效果,典型的有ESSC算法,ESSC算法原名Enhanced Soft Subspace Clustering,意为增强的软子空间聚类算法.该算法通过引入类间分离度的思想,其聚类效果经过实验表明明显优于之前只考虑类内相似度的算法.ESSC算法的目标函数为:

$$J_{\mathrm{ESSC}}=\sum_{i=1}^{C}\sum_{j=1}^{N}u_{ij}^m\sum_{k=1}^{D}w_{ij}\left(x_{jk}-v_{jk}\right)^2+\varphi\sum_{i=1}^{C}\sum_{k=1}^{D}w_{ik}\ln w_{ik}-\eta\sum_{i=1}^{C}\left(\sum_{j=1}^{N}u_{ij}^m\right)\sum_{k=1}^{D}w_{ik}\left(v_{ik}-v_{0k}\right)^2 \tag{7}$$

其中,全局中心点 v_{0k} 的计算如式(8)所示:

$$v_{0k}=\frac{\sum_{j=1}^{N}x_{jk}}{N} \tag{8}$$

ESSC算法引入了一个参数 η,用来调节类间分离度对聚类结果的影响.ESSC中对划分矩阵 $\boldsymbol{U}$、簇中心矩阵 $\boldsymbol{V}$ 和权值矩阵 W 的更新如式(9)和式(10)所示:

$$u_{ij}=\frac{\left(d_{ij}\right)^{\frac{-1}{m-1}}}{\sum_{i=1}^{C}\left(d_{ij}\right)^{\frac{-1}{m-1}}} \tag{9}$$

$$v_{ik}=\frac{\sum_{j=1}^{N}u_{ij}^m\left(x_{ik}-\eta v_{0k}\right)}{\sum_{j=1}^{N}u_{ij}^m\left(1-\eta\right)} \tag{10}$$

其中,d_{ij} 和 δ_{ik} 的计算如式(11)和式(12)所示:

$$d_{ij}=\sum_{k=1}^{D}w_{ik}\left(x_{jk}-v_{ik}\right)^2-\eta\sum_{k=1}^{D}w_{ik}\left(v_{jk}-v_{0k}\right)^2 \tag{11}$$

$$\delta_{ik}=\sum_{j=1}^{N}u_{ij}^m\left(x_{jk}-v_{ik}\right)^2-\eta\sum_{j=1}^{N}u_{ij}^m\left(v_{jk}-v_{0k}\right)^2 \tag{12}$$

1.2　迁移学习　迁移学习作为机器学习领域的一个新的研究方向,近年来受到越来越多的关注.传统的机器学习方法要求源领域数据和目标领域数据同分布,而迁移学习放松了这一限制,能够把已经获得的知识应用到不同但相似的领域中,解决了目标领域中可用训练样本不足的学习问题.

为了解决目标任务数据仅存在少量或无标注数据问题,通过迁移学习将某个领域或任务已具有的先验知识或模型应用到与其相关的任务或问题中,更为有效地利用有标注数据[26].通常,迁移学习主要针对两个问题展开研究:(1)小数据问题:传统机器学习算法一般假设训练数据与测试数据服从相同的数据分布规律但实际应用中往往无法满足,为了保证训练效果,通常需要重新标注

大量数据但有时会带来数据的浪费,而当训练数据过少时,还会出现严重过拟合问题;而迁移学习可从源域的小数据中抽取并迁移知识来完成新的学习任务.(2)个性化问题:当源领域过广又不够具体且研究需要专注于某一个特定目标领域时,可以通过迁移学习将源领域的预训练模型特征迁移到目标领域,从而实现个性化.

迁移学习中,域与任务是两个常见的基本概念.领域D(Domain)定义为由d维特征空间χ和边缘概率分布$p(x)$组成,即:

$$D=\{\chi,p(x)\},x\in\chi \tag{13}$$

迁移学习的任务T由对应某一领域的类别空间Y和模型$f(x)$组成,即:

$$T=\{Y,f(x)\},y\in Y \tag{14}$$

目标领域D_t是最终要赋予知识和标注的对象,是关注的中心.知识从源域传递到目标域就完成了迁移建立模型的领域.D_t的数据集一般分为两部分:标注样本和无标注样本,有标注数据样本往往数量少且难以建立模型.源领域D_s是可以辅助目标领域建模的相近领域,数据集一般为:

$$D_s=\{(x_i,y_i)\mid i=1,2,\cdots,n_s\} \tag{15}$$

源领域一般包含大量有标注数据,且源领域可以为一个或多个.由于D_t和D_s为不同的领域,两者的数据分布存在差异,这也导致源领域不能直接用来辅助训练,必须通过迁移学习的方法提高领域之间的相似性.若只考虑一个源域和目标域的情况,可以定义迁移学习为给定源域$D(s)$和源任务$T(s)$以及目标域$D(T)$和目标任务$T(T)$;当域或任务有一者不同时,迁移学习则通过使用源域$D(s)$和源任务$T(s)$对应的知识来改善目标域中转换函数$f(x)$性能,这一过程称为迁移学习.

迁移学习的关键在于找到源域与目标域或源任务与目标任务之间的共性,包括样本实例、网络架构或特征表示等方面,从而获得可以对目标域样本进行分类或识别的新模型,达到有效完成目标任务的目标,如图2所示.

迁移学习主要研究以下问题:(1)迁移什么和何时迁移,即源领域数据的哪些先验知识训练出新的模型应用到目标域中能够表现出优异的性能,也就是什么条件下可以迁移?(2)在无标注或少量标注数据的目标域中,如何在训练中与大量有标注的源数据结合,获得测试误差最小,即迁移学习算法的研究也就是如何迁移.目前的迁移学习技术涉及多种机器学习技术,如半监督学习、领域适配、鲁棒学习、样本选择偏置、多任务学习等.通过迁移学习的研究,不仅可以更加充分地利用现有已标签数据信息,而且可以利用模型的泛化能力和鲁棒性实现知识在新领域新应用模型中的迁移复用.

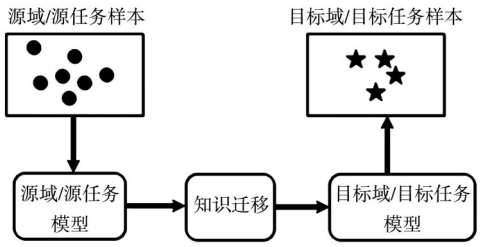

图2 知识迁移

Fig. 2 Knowledge Transfer

2 结合迁移学习的软子空间聚类算法

近年来,聚类分析在统计学、数据库领域和机器学习等领域得到广泛研究.传统聚类分析算法存在诸多限制,而子空间聚类算法能进一步提升聚类分析的性能和效果,其中软子空间聚类算法更是同时具有灵活性和适用性.目前大部分软子空间算法是基于传统k-Means/FCM框架进行聚类,而这类算法往往存在如下缺点:(1)无法为每个簇选择各自有用的特征维度,从而导致聚类精度大大降低;(2)算法在运算时需要已有的完整数据作为支撑,所以聚类效果往往不佳[27].

基于以上问题,本文将熵加权软子空间聚类算法(Entropy Weighting k-Means Algorithm for Subspace Clustering,EWKM)和迁移学习进行融合,通过引入信息熵的概念和迁移学习的思想,提出一种基于迁移学习的软子空间聚类算法(Soft Subspace Clustering Algorithm Based on Transfer Learning,TSC).

2.1 熵加权的k-Means软子空间聚类算法 熵加权的软子空间聚类算法通过引入信息熵的概

念,使数据维度的权重由信息熵来计算和确定[28],因此权重不会使每个簇拥有相同的特征子空间维度.熵加权的软子空间聚类算法和以往的其他子空间聚类算法相比,如模糊加权软子空间聚类算法等,在大数据集或高维度数据集上往往能获得更好的聚类效果.

2.1.1　算法原理　熵加权的k-Means软子空间聚类算法的目标函数为:

$$J_{\mathrm{EWKM}}(\boldsymbol{W},\boldsymbol{Z},\boldsymbol{\Lambda})=\sum_{l=1}^{k}\left[\sum_{j=1}^{n}\sum_{i=1}^{m}w_{lj}\lambda_{li}\left(z_{li}-x_{ji}\right)^2+\gamma\sum_{i=1}^{m}\lambda_{li}\lg\lambda_{li}\right] \tag{16}$$

$$\begin{cases}\sum_{l=1}^{k}w_{lj}=1,1\leqslant j\leqslant n,1\leqslant l\leqslant k,w_{lj}\in\{0,1\}\\ \sum_{l=1}^{k}\lambda_{lj}=1,1\leqslant l\leqslant k,1\leqslant j\leqslant m,0\leqslant\lambda_{lj}\leqslant 1\end{cases} \tag{17}$$

其中,$\boldsymbol{W}$表示分配矩阵,大小为$k\times n$,$\boldsymbol{Z}$表示当前聚类中心矩阵;k表示聚类总数,n表示数据集中对象个数,m表示对象的维数;λ表示每个簇所对应的权重,维度为$k\times m$,γ为大于0的参数.式(16)中矩阵$\boldsymbol{\Lambda}$为λ所对应的矩阵,求和式中的第一项为簇内分散度的总和,第二项为负熵权.正参数γ控制了聚类在更多维度上的贡献程度.$\gamma\sum_{i=1}^{m}\lambda_{li}\lg\lambda_{li}$的绝对值越大,对应目标函数的值就会越小.所以在最小化目标函数的过程中,熵项会尽量使各个权值值趋于平滑来避免某些维度权值为0的情况,起到一种平衡的作用.

2.1.2　EWKM算法流程

输入:聚类中心数k,正参数γ.随机选取k个数据点作为聚类中心,初始化所有权重为$1/m$.

重复:

Step1. 更新分配矩阵;

Step2. 更新聚类中心矩阵;

Step3. 更新特征权重矩阵.

直到:目标函数得到其局部最小值.

2.2　TSC算法　虽然EWKM算法能很好地解决数据分散在稀疏的高维子空间的问题,但其和以往的软子空间聚类算法一样,优点是建立在数据样本充足并且没有大量残缺数据信息的条件下.而当样本数据量不足或者存在信息缺失时,软子空间聚类的性能将大幅下降.为此,从熵加权软子空间聚类算法的基础上,尝试引入迁移学习来改善数据样本不足或信息缺失的问题.这种基于迁移学习的熵加权软子空间聚类算法的关键是如何用以往的数据信息作为辅助数据来弥补数据样本不足或信息缺失的缺点,从而得到更好的聚类效果.

TSC算法通过从历史数据中获得的聚类中心$\hat{z}$作为一种可以使用的知识,用来指导算法对目标域数据样本的聚类分析.

2.2.1　算法原理　TSC算法在进行计算时,其目标函数可以描述为:

$$J_{\mathrm{TSC}}\left(\boldsymbol{W},\boldsymbol{Z},\hat{\boldsymbol{Z}},\boldsymbol{\Lambda}\right)=J_{\mathrm{EWKM}}(\boldsymbol{W},\boldsymbol{Z},\boldsymbol{\Lambda})+J_{\mathrm{Transfer}}\left(\boldsymbol{W},\boldsymbol{Z},\hat{\boldsymbol{Z}}\right) \tag{18}$$

$$J_{\mathrm{EWKM}}(\boldsymbol{W},\boldsymbol{Z},\boldsymbol{\Lambda})=\sum_{l=1}^{k}\left[\sum_{j=1}^{n}\sum_{i=1}^{m}w_{lj}\lambda_{li}\left(z_{li}-x_{ji}\right)^2+\gamma\sum_{i=1}^{m}\lambda_{li}\lg\lambda_{li}\right] \tag{19}$$

$$J_{\mathrm{Transfer}}\left(\boldsymbol{W},\boldsymbol{Z},\hat{\boldsymbol{Z}}\right)=\beta_1\sum_{l=1}^{k}\sum_{j=1}^{n}\sum_{i=1}^{m}w_{lj}\lambda_{li}\left(\hat{z}_{li}-x_{ji}\right)^2+\beta_2\sum_{l=1}^{k}\sum_{j=1}^{n}\sum_{i=1}^{m}w_{lj}\lambda_{li}\left(\hat{z}_{li}-z_{li}\right)^2 \tag{20}$$

$$\begin{cases}\sum_{l=1}^{k}w_{lj}=1,1\leqslant j\leqslant n,1\leqslant l\leqslant k,w_{lj}\in\{0,1\}\\ \sum_{l=1}^{k}\lambda_{lj}=1,1\leqslant l\leqslant k,1\leqslant i\leqslant m,0\leqslant\lambda_{li}\leqslant 1\end{cases} \tag{21}$$

其中,n表示数据样本总数,m表示每个数据样本所含维数,k表示簇的个数;$\boldsymbol{W}$表示分配矩阵,大小为$k\times n$;$\boldsymbol{\Lambda}$表示权重矩阵,维度为$k\times m$,γ为大于0的参数;$\boldsymbol{Z}$表示当前聚类中心矩阵,$\hat{\boldsymbol{Z}}$表示从历史信息中获得的聚类中心矩阵;β_1用来控制当前聚类的权重,β_2用来平衡历史数据的应用.

由式(18)可知,算法的目标函数中第一项是熵加权k-Means软子空间聚类算法,主要用来处理当前数据集;第二项为迁移学习项,主要作用是利用历史聚类中心来指导当前聚类任务.该算法的主要思想就是利用历史数据化指导目标数据聚类分析来强化熵加权软子空间聚类,弥补数据样本不足或信息残缺的问题.同样,使用拉格朗日乘子法可得到分配矩阵更新公式为:

$$\begin{cases}w_{lj}=1 & \text{if } d_{lj}\leqslant d_{rj}\\ w_{lj}=0 & \text{otherwise}\end{cases} \tag{22}$$

$$d_{lj}=\sum_{i=1}^{m}\lambda_{li}\left(z_{li}-x_{ji}\right)^2+\beta_1\sum_{i=1}^{m}\lambda_{li}\left(\hat{z}_{li}-x_{ji}\right)^2+\beta_2\sum_{i=1}^{m}\lambda_{li}\left(\hat{z}_{li}-z_{li}\right)^2 \tag{23}$$

聚类中心更新公式为:

$$z_{li}=\frac{\sum_{j=1}^{n}w_{lj}x_{ji}+\beta_2\sum_{j=1}^{n}w_{lj}\hat{z}_{li}}{\sum_{j=1}^{n}w_{lj}+\beta_2\sum_{j=1}^{n}w_{lj}},1\leqslant l\leqslant k,1\leqslant i\leqslant m \tag{24}$$

权重矩阵更新公式为:

$$\lambda_{lt}=\frac{\exp\left(\frac{-D_{lt}}{\gamma}\right)}{\sum_{i=1}^{M}\exp\left(\frac{-D_{li}}{\gamma}\right)} \tag{25}$$

$$D_{lt}=\sum_{j=1}^{n}w_{lj}\left(z_{lt}-x_{jt}\right)^2+\beta_1\sum_{j=1}^{n}w_{lj}\left(\hat{z}_{lt}-x_{jt}\right)^2+\beta_2\sum_{j=1}^{n}w_{lj}\left(\hat{z}_{lt}-z_{lt}\right)^2 \tag{26}$$

2.2.2 TSC算法流程

输入:聚类中心数k,正参数γ,β_1和β_2. 随机选取k个数据点作为聚类中心,初始化所有权重为$1/m$.

重复:

Step1. 通过式(22)更新分配矩阵W.

Step2. 通过式(18)更新聚类中心矩阵$\boldsymbol{Z}$.

Step3. 通过式(25)更新特征权重矩阵$\boldsymbol{\Lambda}$.

直到:满足迭代终止条件.

3 实验与分析

用MATLAB R2019a进行仿真实验,选取UCI标准数据集中的Iriss、Wine、Vehicle和Australian这四个典型的数据集进行测试,并与以往的典型软子空间聚类分析算法EWKM,ESSC和FSC进行比较. 本文设计的实验中,最大迭代次数iterations=100为算法终止条件,设置$\gamma=50$,$m=\frac{\min(N,D-1)}{\min\left(N,(D-1)-2\right)}$,$\beta_1=1$,$\beta_2=1$.

3.1 UCI数据集 为了测试算法的性能和有效性,本文选用三个UCI标准数据集,它们都是在聚类分析算法评测中广泛使用的典型的数据集. 数据集的详细信息如表1所示.

表1 UCI数据集详细信息

Table 1 The details of UCI datasets

序号	名称	样本数N	维度D	聚类数目C
1	Iris	150	4	3
2	Wine	178	13	3
3	Vehicle	208	18	4
4	Australian	690	14	2

3.2 聚类评价指标 评测聚类分析算法的有效性需要有有效的评价指标,本文采用大多数研究文献中的评价标准,即兰德指数(RI)和标准化互信息(NMI)作为评价指标:

$$RI=\frac{f_{00}+f_{11}}{N(N-1)/2} \tag{27}$$

$$NMI=\frac{\sum_{i=1}^{K}\sum_{j=1}^{C}n_{ij}\log_2\frac{N\times n_{ij}}{n_i\times n_j}}{\sqrt{\left(\sum_{i=1}^{K}n_i\log_2\frac{n_i}{N}\right)\left(\sum_{i=1}^{C}n_j\log_2\frac{n_j}{N}\right)}} \tag{28}$$

其中,N表示整个数据集样本数,C为簇的数目,K是数据集实际簇数;f_{00}表示属于不同簇的具有不同标签的数据样本对数,f_{11}表示属于相同簇且具有相同标签的数据样本对数;n_i表示实际属于第i簇的样本点数,n_j表示实验得出的属于第j簇的样本点数,n_{ij}表示分错类的样本点数,其中$i\neq j$.

RI和NMI的评测值均在$[0,1]$,得分越高表示聚类效果越好;RI或NMI值为1则表示聚类结果完全匹配,准确度为100%;RI或NMI的值为0则表示聚类结果和实际情况完全不匹配.

3.3 实验结果分析 为了验证本文提出的基于迁移学习的软子空间算法的性能,将各个数据集中前70% 的数据作为历史数据信息X_{history},剩下的30% 作为当下需要聚类数据集X_{current}. 又将X_{current}分为两类:一类包含全部类别的数据样本$X_{\text{current-all}}$,模拟数据样本不足的情况;一类缺失一种类别的数据样本$X_{\text{current-lost}}$,模拟信息缺失的情况. 实验时,先将三种传统算法在X_{history}数据集上运行,得到的聚类评测结果如表2和表3所示.

然后加入TSC算法,将四种算法在$X_{\text{current-all}}$和$X_{\text{current-lost}}$上运行,结果如表4和表5所示.

由表4和表5的评测结果可知,在多项不同类别的数据集上,TSC算法所得到的聚类结果要优

表2　$X_{history}$聚类结果(*RI*指数)

Table 2　The clustering results of $X_{history}$ (*RI* index)

数据集	EWKM	ESSC	FSC
Iris	0.8523	0.8720	0.8423
Wine	0.8415	0.8975	0.8358
Vehicle	0.3747	0.5261	0.3854
Australian	0.7552	0.7123	0.7348

表3　$X_{history}$聚类结果(*NMI*指数)

Table 3　The clustering results of $X_{history}$ (*NMI* index)

数据集	EWKM	ESSC	FSC
Iris	0.7523	0.7441	0.7105
Wine	0.7015	0.7025	0.7158
Vehicle	0.1042	0.1225	0.1156
Australian	0.4835	0.3454	0.3855

表4　$X_{current}$聚类结果(*RI*指数)

Table 4　The clustering results of $X_{current}$ (*RI* index)

数据集	$X_{current\text{-}all}$				$X_{current\text{-}lost}$			
	EWKM	ESSC	FSC	TSC	EWKM	ESSC	FSC	TSC
Iris	0.6235	0.6358	0.6135	0.7852	0.5442	0.5317	0.5423	0.7561
Wine	0.6552	0.6884	0.6451	0.7245	0.6075	0.6023	0.5997	0.8024
Vehicle	0.6578	0.6021	0.6077	0.8245	0.3871	0.3561	0.3534	0.6122
Australian	0.6021	0.5975	0.5988	0.7846	0.4223	0.4125	0.4108	0.7241

表5　$X_{current}$聚类结果(*NMI*指数)

Table 5　The clustering results of $X_{current}$ (*NMI* index)

数据集	$X_{current\text{-}all}$				$X_{current\text{-}lost}$			
	EWKM	ESSC	FSC	TSC	EWKM	ESSC	FSC	TSC
Iris	0.5247	0.5365	0.5286	0.6807	0.4275	0.4562	0.3925	0.5803
Wine	0.6218	0.6452	0.6102	0.7534	0.5842	0.6231	0.5714	0.6744
Vehicle	0.1204	0.1078	0.1107	0.1608	0.0214	0.0451	0.0168	0.1256
Australian	0.2536	0.2237	0.2496	0.4532	0.2453	0.1431	0.1087	0.3998

于其他对比算法,即能够取得相对良好的处理结果;而在对数据进行聚类时,即使面临数据样本或者数据信息缺失,TSC算法也能取得最佳的聚类效果.这是由于该算法引入了迁移学习的思想,从以往的数据中获取历史中心来指导修正当前数据不足时的聚类分析任务;而其他算法由于数据样本太少,信息不足,导致性能下降.设置好的正参数对算法性能有很大提升.

4　结　论

针对传统的软子空间聚类算法在样本数据残缺时聚类准确度不高的问题,提出一种基于迁移学习的软子空间聚类算法,通过引入迁移学习与信息熵,用熵权法确定权重处理高维数据,并将历史数据用于指导和修正当前的聚类分析,有效地提升了算法在数据样本残缺情况下的聚类效果,拓展了软子空间聚类算法的应用范围.通过实验表明,在相同的高维数据集下,与三种典型的聚类算法相比较,本文算法在两种评价指标下均取得了更高的聚类准确度,得到了更好的聚类性能.

参考文献

[1] Chan E Y, Ching W K, Ng M K, et al. An optimization algorithm for clustering using weighted dissimilarity measures. Pattern Recognition, 2004, 37(5): 943－952.

[2] Gan G J, Wu J H, Yang Z J. A fuzzy subspace algorithm for clustering high dimensional data// International Conference on Advanced Data Mining and Applications. Springer Berlin Heidelberg, 2006: 271－278.

[3] Jing L P, Ng M K, Huang J Z. An entropy weighting k-means algorithm for subspace clustering of high-dimensional sparse data. IEEE Transactions on Knowledge and Data Engineering, 2007, 19(8): 1026－1041.

[4] Domeniconi C, Gunopulos D, Ma S, et al. Locally adaptive metrics for clustering high dimensional data. Data Mining and Knowledge Discovery, 2007, 14(1): 63－97.

[5] Deng Z H, Choi K S, Chung F L, et al. Enhanced soft subspace clustering integrating within-cluster and between-cluster information. Pattern Recognition, 2010, 43(3): 767－781.

[6] Lu Y P, Wang S R, Li S Z, et al. Particle swarm optimizer for variable weighting in clustering high-dimensional data. Machine Learning, 2011, 82(1): 43－70.

[7] Wang X B, Lei Z, Shi H L, et al. Co-referenced subspace clustering//2018 IEEE International Conference on Multimedia and Expo (ICME). San Diego, CA, USA: IEEE, 2018: 1－6.

[8] Elhamifar E, Vidal R. Sparse subspace clustering: algorithm, theoryand applications. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2012, 35(11):2765－2781.

[9] Dai W, Xue G R, Yang Q, et al. Transferring naive Bayes classifiers for text classification//Proceedings of the 22nd AAAI Conference on Artificial Intelligence. Vancouver, Canada: AAAI Press, 2007: 540－545.

[10] Wei F M, Zhang J P, Chu Y, et al. FSFP: transfer learning from long texts to the short. Applied Mathematics & Information Sciences, 2014, 8(4): 2033－2040.

[11] Dai W Y, Yang Q, Xue G R, et al. Boosting for transfer learning//Proceedings of the 24th international conference on Machine learning. Helsinki Finland: ACM, 2007: 193－200.

[12] Agrawal R, Gehrke J, Gunopulos D, et al. Automatic subspace clustering of high dimensional data for data mining applications//ACM SIGMOD Record. Seattle, WA, USA: ACM, 1998: 94－105.

[13] 钱鹏江,孙寿伟,蒋亦棒等. 知识迁移极大熵聚类算法. 控制与决策, 2015, 30(6): 1001－1006. (Qian P J, Sun S W, Jiang Y B, et al. Knowledge Transfer based maximum entropy clustering. Control and Decision, 2015, 30(6): 1001－1006.)

[14] Yu J, Shi H B, Huang H K, et al. Counterexamples to convergence theorem of maximum-entropy clustering algorithm. Science in China Series F: Information Sciences, 2003, 46(5): 321－326.

[15] 王熙照,安素芳. 基于极大模糊熵原理的模糊产生式规则中的权重获取方法研究. 计算机研究与发展, 2006, 43(4): 673－678. (Wang X Z, An S F. Research on learning weights of fuzzy production rules based on maximum fuzzy entropy. Journal of Computer Research and Development, 2006, 43(4): 673－678.)

[16] 邓赵红,王士同,吴锡生等. 鲁棒的极大熵聚类算法RMEC及其例外点标识. 中国工程科学, 2004, 6(9): 38－45. (Deng Z H, Wang S T, Wu X S, et al. Robust maximum entropy clustering algorithm RMEC and its outlier labeling. Engineering Science, 2004, 6(9): 38－45.)

[17] Jiang W H, Chung F L. Transfer spectral clustering// Proceedings of the 2012 European conference on Machine Learning and Knowledge Discovery in Databases. Springer Berlin Heidelberg, 2012: 790－803.

[18] Jain A K, Murty M N, Flynn P J. Data Clustering: a review. ACM Computing Surveys (CSUR), 1999, 31(3): 265－320.

[19] Guo G D, Chen S, Chen L F. Soft subspace clustering with an improved feature weight self-adjustment mechanism. International Journal of Machine Learning & Cybernetics, 2012, 3(1): 39－49.

[20] Xu Y M, Wang C D, Lai J H. Weighted multi-view clustering with feature selection. Pattern Recognition, 2016, 53: 25－35.

[21] Zhao X R, Evans N, Dugelay J L. A subspace co-training framework for multi-view clustering. Pattern Recognition Letters, 2014, 41: 73－82.

[22] Ji J C, Bai T, Zhou C G, et al. An improved k-prototypes clustering algorithm for mixed numeric and categorical data. Neurocomputing, 2013, 120: 590－596.

[23] 黄王非,黎飞,青山. 基于子空间维度加权的密度聚类算法. 计算机工程 2010, 36(9): 65－67. (Huang W F, Li F, Qing S. Density clustering algorithm based on subspace dimensional weighting. Computer Engineering, 2010, 36(9): 65－67.)

[24] Donoho D L. High-dimensional data analysis: The curses and blessings of dimensionality. American

Mathematical Society Math Challenges Lecture, 2000,1:32.

[25] 许亚骏.子空间聚类算法研究及应用.硕士学位论文.无锡:江南大学,2016.(Xu Y J. Research on subspace clustering algorithms and its applications. Master Dissertation. Wuxi: Jiangnan University, 2016.)

[26] Weiss K, Khoshgoftaar T M, Wang D D. A survey of transfer learning. Journal of Big Data, 2016, 3:9.

[27] Günnemann S, Boden B, Seidl T. DB-CSC: a density-based approach for subspace clustering in graphs with feature vectors//Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer Berlin Heidelberg, 2011:565-580.

[28] Wan S J, Wong S K M, Prusinkiewicz P. An algorithm for multidimensional data clustering. ACM Transactions on Mathematical Software, 14(2): 153-162.

(责任编辑　杨可盛)