高技术通讯 2013 年 第23 卷 第1 期:85 ~90

doi:10.3772/j.issn.1002-0470.2013.01.014

基于最大化间隔准则和成对约束的鲁棒半监督聚类研究①

曾 洪② 宋爱国 卢 伟

(东南大学仪器科学与工程学院 南京 210096)

摘 要 针对现有半监督最大间隔聚类算法在不同类别中有不少样本非常相似的情况下难以提高聚类准确度的问题,提出了下述解决策略:首先,基于最大化间隔准则设计一种鲁棒的成对约束损失函数,即使不同类别有较多样本非常相似,该函数仍然能有效地检测不能满足成对约束的聚类结果,并提供相应的惩罚,从而能较好地提高聚类的性能。其次,基于约束凹凸过程设计一种迭代算法进行求解。进而,基于这一策略,提出了一种新的聚类算法——鲁棒的成对约束最大化间隔聚类(BPCMMC)算法。实验结果表明,该算法能有效克服现有半监督最大间隔聚类算法的不足,其聚类错误率明显低于传统的半监督聚类算法。

关键词 半监督聚类,成对约束,最大化间隔准则,鲁棒的损失函数,约束凹凸过程(CCCP)

0 引 言

为提高数据聚类的精确度,半监督聚类算法采用两种成对约束即必连(must-link,ML)约束和不连(cannot-link,CL)约束来指导聚类过程,ML 约束要求两个样本在聚类时被分配到同一个簇中,而 CL 约束要求两个样本在聚类时被分配到不同的簇中[1]。传统的半监督聚类方法[2-5]通常首先利用成对约束来学习一种新的距离测度或相似性度量,然后基于这种新的距离测度或相似性度量,再运用常规的 K 均值或谱聚类方法进行聚类。但是由于数据真实的距离测度或相似性度量可能相当复杂,事先假设的测度或相似性模型很难准确地对其进行刻画[6],因此难以有效地提高聚类准确度。近来研究人员发现,与这种首先利用成对约束学习新的距离测度再进行聚类的方法相比,直接利用成对约束指导聚类的方法更加有效[6]。因此,基于最大间隔聚类算法[7],文献[6]提出了一种半监督聚类算法,该算法直接搜索最优的分簇边界,使其与簇之间的间隔最大,并尽量满足给定成对约束的边界。尽管此半监督最大间隔聚类算法在某些数据上取得了较好的聚类效果,但是当不同类别里有不少样本非常相似时(例如两篇不同主题的文章使用了大量相同的词语从而看上去相似),该方法的成对约束损失函数在此复杂情形下无法提供鲁棒的惩罚,导致不能很好地提高聚类性能。针对现有半监督聚类算法的不足,本研究设计了能鲁棒地对不满足成对约束的聚类结果提供惩罚的新的成对约束损失函数,而且,基于约束凹凸过程(constrained concave-convex procedure,CCCP)[8]设计了一种迭代算法进行求解,在此基础上提出了一种新的半监督最大间隔聚类算法——鲁棒的成对约束最大化间隔聚类(robust pairwise constrained maximum margin clustering,RPCMMC)算法。实验证明,该新聚类算法能够在不同类别有较多样本非常相似的复杂情形下有效地提高聚类性能,在同样的成对约束条件下取得比传统半监督聚类算法更低的聚类错误率。

1 半监督最大间隔聚类问题描述

最大间隔聚类方法[7]将监督学习中的间隔最大化准则推广到无监督学习中的聚类分析,其基本思想是搜索使得簇间间隔最大的判别边界来实现对

① 国家自然科学基金(61105048,60972165,51175080),教育部博士点基金(20100092120012, 20110092120034),人事部留学人员科技活动择优资助基金(6722000008)和江苏省自然科学基金(BK2010240,BK2010423)资助项目。

② 男,1981 年生,博士,讲师;研究方向:机器学习,模式识别,信号处理;联系人, E-mail: hzeng@seu.edu.cn

(收稿日期:2012-02-16)

数据集的分组。半监督最大间隔聚类方法则进一步利用已知的成对约束指导最优判别边界的搜索。具体来说,对于给定的无类别标记的样本 $\boldsymbol{x}_i(i=1,\cdots,m)$,以及已知带成对约束的样本 $C_j=(\boldsymbol{x}_{j1},\boldsymbol{x}_{j2},l_j)(j=1,\cdots,n$,其中 $l_j=1$ 代表 $\boldsymbol{x}_{j1}$ 和 $\boldsymbol{x}_{j2}$ 必连,$l_j=-1$ 代表 $\boldsymbol{x}_{j1}$ 和 $\boldsymbol{x}_{j2}$ 不连),半监督最大间隔聚类的目标是在该样本集上所有可能的二值标记 y_i, y_{j1}, $y_{j2}\in\{\pm1\}(i=1,\cdots,m,j=1,\cdots,n)$ 设置下,寻找到各簇间隔最大的判别边界。记判别函数 $f(\boldsymbol{x})=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b$,文献[6]指出半监督最大间隔聚类问题等价于求解如下的问题:

$$\min_f \frac{1}{2}\|\boldsymbol{w}\|^2+\lambda\sum_{i=1}^{m}L(|f(\boldsymbol{x}_i)|)+\lambda\sum_{j=1}^{n}L'(f_{j1},f_{j2},l_j) \tag{1}$$

其中 $\|\boldsymbol{w}\|^2$ 是约束搜索空间的规则化项,λ 是大于 0 的平衡常数。$L(|f(\boldsymbol{x}_i)|)$ 是针对无类别标记样本违反间隔约束的广义损失函数[6](其中 $L(z)=\max\{0,1-z\}$),以保持它们与分界面之间的距离。易见,只要样本点在间隔区域以外,无论位于判别边界的哪一边,就不会引入损失。f_{j1} 和 f_{j2} 分别是 $f(\boldsymbol{x}_{j1})$ 和 $f(\boldsymbol{x}_{j2})$ 的简记。$L'(f_{j1},f_{j2},l_j)$ 是违反成对约束 C_j 的损失函数。求解式(1)所示的问题后,样本 $\boldsymbol{x}_i$ 的簇类别由 $f(\boldsymbol{x}_i)$ 的符号来决定。

已有的半监督最大间隔聚类算法[6]采用 $L'(f_{j1},f_{j2},l_j)=|f_{j1}-l_jf_{j2}|$,其有效性将通过图 1 中所示两种不同难易程度的二维数据聚类任务进行考察。图 1 中不同类别的样本分别用○和△表示,判别边界用粗黑线表示,椭圆代表在该判别边界下对应的簇,此时的聚类结果均不满足给定的成对约束。

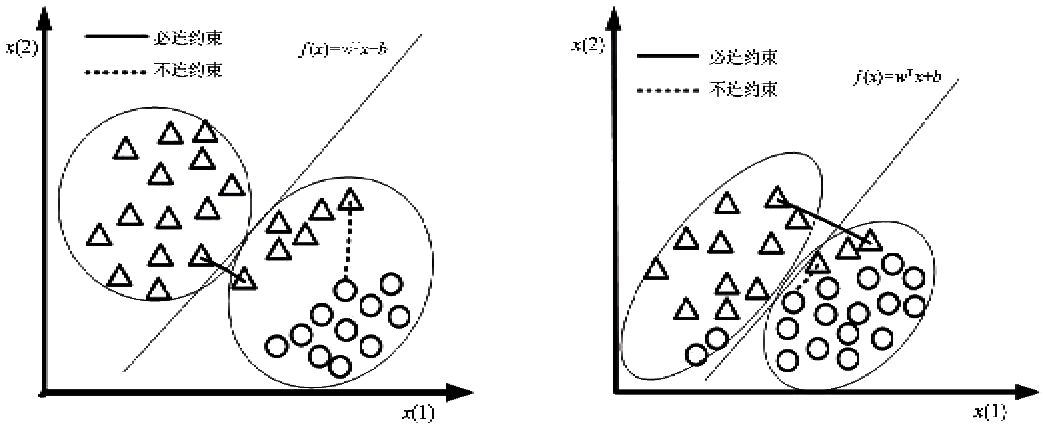

图 1 两种不同难易程度的二维数据聚类任务示意图

当不同类别的样本不相似(如图 1(a)所示)时,该损失函数的确能够对违反成对约束的聚类结果提供恰当的惩罚。比如,图 1(a)中具有不连约束的两个样本被当前的判别边界放在同一边,因而具有相同的符号,即此时约束未被满足。同时由于这两个样本点都远离边界,因此 $|f_{j1}+f_{j2}|$ 将导致形成一个较大的惩罚值,从而促使算法继续搜索一个更理想的判别边界。但是当不同类别中有部分样本非常相似(如图 1(b)所示)时,该成对约束损失函数在此复杂情形下,将不能有效地检测和惩罚约束的违反。比如图 1(b)中当前的判别边界穿越两个簇有不少样本相似的区域,并且违反了在该区域中的一对不连约束,则此时具有该不连约束的两个样本必然靠近该非最优的判别边界。从而这两个样本的判别值 f_{j1}, f_{j2} 将非常接近于 0,则 $|f_{j1}+f_{j2}|$ 必趋近 0,即该结果不恰当地表明此不连约束"并未被违反"。然而这种复杂情形在现实应用中比图 1(a)所示的简单情形更加普遍,例如在文本聚类中,两篇文档尽管拥有大量共同的词语,却描述不同的主题,现有的半监督聚类方法[6]难以提高此类数据上的聚类性能。

2 新的半监督最大间隔聚类方法——RPCMMC 算法

鉴于以上分析,本文提出了一种新的半监督最大间隔聚类方法,即鲁棒的成对约束最大化间隔聚类(RPCMMC)算法。该方法基于一种鲁棒的违反成对约束损失函数提高聚类性能,用基于约束凹凸过程设计的迭代算法求解半监督聚类问题。

2.1　鲁棒的违反成对约束的损失函数

对于约束 C_j,所提的违反成对约束的损失函数如下:

$$L'(f_{j1},f_{j2},l_j)=H(f_{j1},f_{j2})\triangleq min\{L(f_{j1})+L(l_jf_{j2}),L(-f_{j1})+L(-l_jf_{j2})\} \quad (2)$$

其有两个重要性质。

性质 1:当 C_j 被违反时,$H(f_{j1},f_{j2})\geqslant 1$。

证明:设 C_j 是不连约束($l_j=-1$),且当前聚类结果不满足 C_j。则此时 f_{j1},f_{j2} 必同号,而已知 $L(z)=\max\{0,1-z\}$ 是一个非负的单调递减函数,因此有 $\begin{cases}L(f_{j1})+L(-f_{j2})\geqslant L(0)=1\\L(-f_{j1})+L(f_{j2})\geqslant L(0)=1\end{cases}$。类似地,可证明当必连约束($l_j=1$)被违反时,亦有 $H(f_{j1},f_{j2})\geqslant 1$。故该性质成立。

由该性质可见,即使在簇里不少样本非常相似的复杂情形下(如图 1(b)所示),如果 C_j 被违反,$H(f_{j1},f_{j2})$ 仍然能提供不小于 1 的惩罚值。而此时现有半监督最大间隔聚类算法[6]中违反成对约束损失函数的值接近于 0,导致不能有效检测成对约束的违反。因此,$H(f_{j1},f_{j2})$ 能克服文献[6]中算法的不足,其将有效地促使具有不连约束的样本分布于判别边界的两侧。

性质 2:当 C_j 未被违反时,随着 $|f_{j1}|$ 和 $|f_{j2}|$ 均趋近于无穷大,$H(f_{j1},f_{j2})$ 将趋近于 0。

证明:设 C_j 是不连约束($l_j=-1$),且当前聚类结果满足 C_j,则此时 f_{j1},f_{j2} 必不同号。不失一般性,设 $f_{j1}>0$,$f_{j2}<0$,则根据 $L(z)=\max\{0,1-z\}$,当 $|f_{j1}|$ 和 $|f_{j2}|$ 均趋近于无穷大时有:

$$\begin{aligned}&\min\{L(f_{j1})+L(-f_{j2}),L(-f_{j1})+L(f_{j2})\}\\&=\min\{\max\{0,1-f_{j1}\}+\max\{0,1+f_{j2}\},\\&\quad\max\{0,1+f_{j1}\}+\max\{0,1-f_{j2}\}\}\\&=\max\{0,1-f_{j1}\}+\max\{0,1+f_{j2}\}=0\end{aligned}$$

类似地,可证明必连约束未被违反时,随着 $|f_{j1}|$ 和 $|f_{j2}|$ 均趋近于无穷大,$H(f_{j1},f_{j2})$ 亦趋近于 0。

结合性质 1 和性质 2,易见所提的违反成对约束的损失函数不仅能根据 f_{j1} 和 f_{j2} 的符号检测约束的违反,还基于最大化间隔准则,对具有成对约束的两个样本施加间隔限制以保持其与分界面之间的距离,从而能提供鲁棒的惩罚。于是得到如下的半监督最大间隔聚类问题:

$$\min_f \frac{1}{2}\|\boldsymbol{w}\|^2+\lambda\sum_{i=1}^{m}L(|f(\boldsymbol{x}_i)|)+\lambda\sum_{j=1}^{n}H(f_{j1},f_{j2}) \quad (3)$$

2.2　迭代算法

由于 $H(f_{j1},f_{j2})$ 中有min操作,因而很难直接对式(3)所示的优化问题进行求解。本文将寻求与式(3)所示的问题等价的且易于求解的优化问题,为此引入一个二值变量向量 $\boldsymbol{d}=(d_1,d_2,\cdots,d_n)\in\{0,1\}^n$,并对于约束 C_j,介绍以下有用的形式:

$$\begin{aligned}&H^{(1)}(f_{j1},f_{j2})\triangleq L(f_{j1})+L(l_jf_{j2})\\&H^{(2)}(f_{j1},f_{j2})\triangleq L(-f_{j1})+L(-l_jf_{j2})\\&H(f_{j1},f_{j2},d_j)\triangleq d_jH^{(1)}(f_{j1},f_{j2})+(1-d_j)H^{(2)}(f_{j1},f_{j2})\end{aligned} \quad (4)$$

其中 $d_j\in[0,1]$。易见

$$d_j^*=\arg\min_{d_j\in[0,1]}H(f_{j1},f_{j2},d_j) \quad (5)$$

只能取 0 或 1,则其可视为从 $\{H^{(1)}(f_{j1},f_{j2}),H^{(2)}(f_{j1},f_{j2})\}$ 中选择其一作为 $H(f_{j1},f_{j2})$ 的二值变量。由此式(3)所示的问题可转化成如下等价问题:

$$\min_{w,b,d}\frac{1}{2}\|\boldsymbol{w}\|^2+\lambda\sum_{i=1}^{m}L(|f(\boldsymbol{x}_i)|)+\lambda\sum_{j=1}^{n}H(f_{j1},f_{j2},d_j) \quad (6)$$

对于该优化问题,本文提出一种交替优化的迭代算法进行求解。在第 t 次迭代中,进行如下两个步骤:

(I)令 $f=f^{(t-1)}$,求解如下问题以得到 $\boldsymbol{d}^{(t)}$:

$$\min_d\frac{1}{2}\|\boldsymbol{w}^{(t-1)}\|^2+\lambda\sum_{i=1}^{m}L(|f^{(t-1)}(\boldsymbol{x}_i)|)+\lambda\sum_{j=1}^{n}H(f_{j1}^{(t-1)},f_{j2}^{(t-1)},d_j) \quad (7)$$

根据式(4)(5)可得

$$d_j^{(t)}=\arg\min_{d_j\in\{0,1\}}H(f_{j1}^{(t-1)},f_{j2}^{(t-1)},d_j) \quad (8)$$

即

$$d_j^{(t)}=\begin{cases}1, & H^{(1)}(f_{j1}^{(t-1)},f_{j2}^{(t-1)})<H^{(2)}(f_{j1}^{(t-1)},f_{j2}^{(t-1)})\\0, & \text{其他}\end{cases} \quad (9)$$

(II)令 $\boldsymbol{d}=\boldsymbol{d}^{(t)}$,求解如下问题以得到 $f^{(t)}$:

$$\begin{aligned}&\min_{f,\{\xi\}}\frac{1}{2}\|\boldsymbol{w}\|^2+\lambda\sum_{i=1}^{m}\xi_i+\lambda\sum_{j=1}^{n}H(f_{j1},f_{j2},d_j^{(t)})\\&s.t.\ \forall i\in\{1,\cdots,m\},|f(\boldsymbol{x}_i)|\geqslant 1-\xi_i,\xi_i\geqslant 0\end{aligned} \quad (10)$$

重复步骤(I)和(II)直至收敛,再根据 $f(\boldsymbol{x}_i)$ 的符号来判别 $\boldsymbol{x}_i$ 所归簇。但由于步骤(II)中优化问题的约束中存在绝对值函数,使得式(10)成为一个非凸优化问题。不过该非凸约束优化问题可视为一个凸函数与一个凹函数的和,本文利用约束凹凸过程(CCCP)[8]优化技术对其进行求解。一般地,

CCCP 通过迭代算法对如下优化问题求解：

$$\min_z \ g_0(z) - h_0(z)$$
$$s.t. \ g_i(z) - h_i(z) \leqslant c_i, \ i = 1,\cdots,m \quad (11)$$

其中 g,h 为可微凸函数。在每一次的迭代中，CCCP 用 $h(z)$ 在当前解 z_r 处的一阶泰勒展开近似 $h(z)$，并求解下列优化问题：

$$\min_z \ g_0(z) - [h_0(z) + \nabla h_0(z_r)(z - z_r)]$$
$$\text{s.t.} \ \forall i = 1,\cdots,m$$
$$g_i(z) - [h_i(z) + \nabla h_i(z_r)(z - z_r)] \leqslant c_i \quad (12)$$

文献[8]证明式(12)所示的是一个凸优化问题，CCCP 将收敛到式(11)所示问题的局部最优解。对于式(10)所示的问题，由于 $|f(\boldsymbol{x}_i)| = |\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b|$ 是 $(\boldsymbol{w},b)$ 的非光滑函数，因此在进行泰勒展开时，需要用次梯度代替梯度。$|f(\boldsymbol{x}_i)|$ 在 $(\boldsymbol{w}_r,b_r)$ 处的一阶泰勒展开为：$|f_r(\boldsymbol{x}_i)| + \mathrm{sgn}(f_r(\boldsymbol{x}_i))[\boldsymbol{x}_i^{\mathrm{T}}(\boldsymbol{w} - \boldsymbol{w}_r) + (b - b_r)] = \mathrm{sgn}(f_r(\boldsymbol{x}_i))[\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b] = \mathrm{sgn}(f_r(\boldsymbol{x}_i))f(\boldsymbol{x}_i)$。将 $|f(\boldsymbol{x}_i)|$ 的一阶泰勒展开带入式(10)，经过整理，得到如下优化问题：

$$\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2 + \lambda\sum_{i=1}^{m} L(\mathrm{sgn}(f_r(\boldsymbol{x}_i))f(\boldsymbol{x}_i)) + \lambda\sum_{j=1}^{n} H(f_{j1}, f_{j2}, d_j^{(t)}) \quad (13)$$

CCCP 从 f_0 开始，求解式(13)以得到 $f_{r+1}(r = 0, 1, 2\cdots)$，第 R 轮迭代收敛时的解即为式(10)的解，即 $f^{(t)} = f_R$。为求解式(13)，令

$$\hat{y}_i \triangleq \mathrm{sgn}(f_r(\boldsymbol{x}_i))$$

$$\hat{y}_{j1} \triangleq \begin{cases} 1, & C_j \in \mathrm{ML} \ \text{and} \ d_j^{(t)} = 1 \\ -1, & C_j \in \mathrm{ML} \ \text{and} \ d_j^{(t)} = 0 \\ 1, & C_j \in \mathrm{CL} \ \text{and} \ d_j^{(t)} = 1 \\ -1 & C_j \in \mathrm{CL} \ \text{and} \ d_j^{(t)} = 0 \end{cases}$$

$$\hat{y}_{j2} \triangleq \begin{cases} 1, & C_j \in \mathrm{ML} \ \text{and} \ d_j^{(t)} = 1 \\ -1, & C_j \in \mathrm{ML} \ \text{and} \ d_j^{(t)} = 0 \\ -1 & C_j \in \mathrm{CL} \ \text{and} \ d_j^{(t)} = 1 \\ 1 & C_j \in \mathrm{CL} \ \text{and} \ d_j^{(t)} = 0 \end{cases} \quad (14)$$

其中 ML 和 CL 分别代表必连和不连约束集，则式(13)实际上等效于一个标准的支撑向量机(support vector machine, SVM)问题，相应的训练集是 $(\boldsymbol{x}_i, \hat{y}_i)(i = 1,\cdots,m)$，$(\boldsymbol{x}_{j1}, \hat{y}_{j1})$ 和 $(\boldsymbol{x}_{j2}, \hat{y}_{j2})$ $(j = 1,\cdots,n)$，因此本文采用成熟的 SVM 软件包 SVM-perf[9] 进行求解。由此，下面给出所提算法的详细步骤：

输入：数据，成对约束；

输出：分簇结果；

步骤 1：随机地初始化 f；

步骤 2：根据式(9)计算 $\boldsymbol{d}^{(t)}$；

步骤 3：根据式(14)计算 $\hat{y}_i$，$\hat{y}_{j1}$，$\hat{y}_{j2}$；

步骤 4：使用 SVM-perf 包求解式(13)；

步骤 5：重复步骤 3 和步骤 4 直至 CCCP 收敛，得到 $f^{(t)}$；

步骤 6：重复步骤 2 至步骤 5 直至收敛，如果 $f(\boldsymbol{x}_i) \geqslant 0$，将 $\boldsymbol{x}_i$ 分到簇 1，否则分到簇 2。

3 实验与结果分析

3.1 实验数据和评价标准

实验在 6 个现实数据集上进行。其中 Live-disorder，pima，sonar 三个数据集来源于 UCI 数据库。BCI 数据集来源于脑机接口的应用，目的是区分受试者是想像左手或右手运动[10]。文本数据集 20NewsGroup-similar2 包含两类主题非常相近(comp. os. mswindows，comp. windows. x)的文章[11]。另一个文本数据 WebKBPage 数据集则来源于被普遍认为拥有大量共同词汇的网页，其分为课程类和非课程类[12]。总体来说，这些使用的数据集中普遍存在如图 1(b)所示的复杂情形，即两个类中有不少样本非常相似，因此非常适合用于比较半监督聚类算法的性能。上述数据的相关信息如表 1 所示。

表 1 实验数据集

数据集	维数	样本数
live-disorder	6	345
pima	8	768
sonar	60	208
BCI	117	400
20NewsGroup-similar2	1864	197
WebKBPage	3000	1051

本文采用文献[2-6]中使用的聚类错误率作为评价聚类算法性能的标准。首先将数据的类别标记去掉后进行聚类，然后根据簇中大多数数据原有类别标记来设定整个簇中数据的类别标记。聚类错误率定义为

$$\text{聚类错误率} = 1 - \frac{\sum_{i=1}^{N} I(p_i = q_i)}{N}$$

其中 p_i 是由上述步骤得到的 $\boldsymbol{x}_i$ 的类别标记，q_i 是 $\boldsymbol{x}_i$ 真实的类别标记。$I(.)$ 是指示函数，成立输出 1，否

则输出0。N为样本总数。聚类错误率越低则表明聚类算法的性能越好。

3.2　实验设置和结果分析

本文所提RPCMMC算法除与无监督聚类的代表算法K均值(Kmeans)方法比较外,还与目前几类具有代表性的半监督聚类算法进行了比较,比如:割平面约束最大化间隔聚类(cutting plane constrained maximum margin clustering, CPCMMC)算法(现有的半监督最大间隔聚类算法)[6];先基于成对约束通过拉普拉斯规则化度量学习(Laplacian regularized metric learning, LRML)算法获得所期望的距离度量,然后基于该度量进行K均值聚类的LRML + Kmeans算法[4];利用成对约束首先学习新的相似性度量,然后再进行谱聚类的相似性传播(affinity propagation, AffPropag)算法[5]。实验中,LRML不含参数,RPCMMC算法和CPCMMC算法的参数λ在$\{10^3, 10^2, 10^1, 1, 10^1, 10^2, 10^3\}$中选取,AffPropag算法中构造相似阵的高斯核宽度在$\{10^4, 10^3, 10^2, 10^1, 1, 10^1, 10^2, 10^3, 10^4\}$中选取,并报道此范围内最优参数对应的结果。在所有实验中,聚类的数目设定为数据集的真实类别数(所有数据集的类别数均为2)。本文采用文献[2-6]中的方法产生成对约束,即随机挑选一对样本,如果它们的类别相同,则形成必连约束,否则形成不连约束。每个数据集上测试5组数量递增的成对约束集,不同半监督聚类算法使用同样的成对约束集。实验中每组成对约束集随机地产生10套,并比较各算法在这10套成对约束集下的平均聚类错误率。各聚类算法平均聚类错误率的结果如表2所示,其中最低的平均聚类错误率已用黑体表示。

表2　各聚类算法在不同成对约束数目下运行10次的平均聚类错误率比较

数据集	约束个数	RPCMMC	CPCMMC	LRML + Kmeans	AffPropag	Kmeans
live-disorder	200	**0.3855**	0.4203	0.4180	0.4180	0.4180
live-disorder	400	**0.3899**	0.4203	0.4183	0.4055	0.4180
live-disorder	600	**0.3809**	0.4203	0.4145	0.4023	0.4180
live-disorder	800	**0.3401**	0.4203	0.3994	0.3620	0.4180
live-disorder	1000	**0.3148**	0.4203	0.3875	0.3318	0.4180
pima	100	0.3017	0.3017	0.3065	**0.2910**	0.3083
pima	200	**0.2598**	0.2924	0.2736	0.2680	0.3083
pima	300	**0.2451**	0.2884	0.2951	0.2586	0.3083
pima	400	**0.2371**	0.2932	0.2896	0.2598	0.3083
pima	500	**0.2260**	0.2862	0.2714	0.2470	0.3083
sonar	20	**0.3885**	0.3885	0.4260	0.4260	0.4476
sonar	40	**0.3404**	0.4269	0.4447	0.4659	0.4476
sonar	60	**0.3577**	0.4188	0.4505	0.4644	0.4476
sonar	80	**0.2937**	0.4303	0.4409	0.4620	0.4476
sonar	100	**0.3183**	0.3534	0.4332	0.4601	0.4476
BCI	50	**0.4262**	0.4403	0.4618	0.4713	0.4775
BCI	100	**0.4325**	0.4552	0.4557	0.4723	0.4775
BCI	150	**0.4385**	0.4663	0.4640	0.4543	0.4775
BCI	200	**0.4112**	0.4580	0.4410	0.4227	0.4775
BCI	250	**0.3293**	0.4532	0.4585	0.3750	0.4775
20NewsGroup-similar2	100	**0.2472**	0.4465	0.4482	0.4665	0.4924
20NewsGroup-similar2	200	**0.1299**	0.4609	0.4569	0.4462	0.4924
20NewsGroup-similar2	300	**0.0893**	0.4315	0.4437	0.3690	0.4924
20NewsGroup-similar2	400	**0.0533**	0.4188	0.4431	0.2650	0.4924
20NewsGroup-similar2	500	**0.0269**	0.4040	0.4472	0.0396	0.4924
WebKBPage	50	0.1030	0.1242	**0.0992**	0.2188	0.1274
WebKBPage	100	**0.0711**	0.1280	0.0845	0.2042	0.1274
WebKBPage	150	**0.0653**	0.0684	0.1855	0.1838	0.1274
WebKBPage	200	**0.0432**	0.1206	0.2002	0.0639	0.1274
WebKBPage	250	**0.0439**	0.0839	0.2188	0.0730	0.1274

易见,与无监督聚类 K 均值算法相比,本文提出的 RPCMMC 算法较大地提高了聚类准确度性能。同时,比较可知,在本文所用的实验数据上,本文所提方法对于聚类性能的提高程度优于 CPCMMC 算法。这是因为实验中所选用的数据大都具有图 1(b)中所示的情形,即在不同类别里有不少样本非常相似,而在这种情形下,CPCMMC 算法中不连约束的损失函数难以对违反不连约束的聚类结果进行有效的惩罚。与 LRML + Kmeans 算法和 AffPropag 算法不同,本文算法不必假设数据具有某种距离度量或相似性度量模型,表 2 显示其聚类错误率一般低于 LRML + Kmeans 和 AffPropag 算法,这也验证了本文方法利用最大化间隔准则,直接学习簇的判别边界来进行半监督聚类的优越性。

4 结 论

针对现有半监督最大间隔聚类算法的不足,本文提出一种新的半监督最大间隔聚类算法。与传统的半监督聚类方法相比,本文所提算法无需事先估计数据的距离或相似性度量,而直接学习簇的判别边界。实验结果表明,本文算法即使当不同类别中有部分样本非常相似时,也能克服现有半监督最大间隔聚类算法的不足,较好地提高聚类的性能。此外,在同样的成对约束条件下,该算法可取得比其他半监督聚类算法更低的聚类错误率。文中暂考虑仅含两类数据的聚类算法,在未来工作中将进一步研究对多类数据集进行聚类的多类半监督最大间隔聚类算法。

参考文献

[1] Zeng H, Cheung Y M. Semi-supervised maximum margin clustering with pairwise constraints. *IEEE Transactions on Knowledge and Data Engineering*, 2012, 24(5):926-939

[2] Davis J V, Kulis B, Jain P, et al. Information-theoretic metric learning. In: Proceedings of International Conference on Machine Learning, Corvallis, USA, 2007. 209-216

[3] Xing E P, Ng A Y, Jordan M I, et al. Distance metric learning with application to clustering with side-information. *Advances in Neural Information Processing Systems*, 2003, 15:521-528

[4] Hoi S C, Liu W, Chang S F. Semi-supervised distance metric learning for collaborative image retrieval and clustering. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 2010,6(3):1-26

[5] Lu Z, Carreira-Perpinan M A. Constrained spectral clustering through affinity propagation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, USA, 2008. 1-8

[6] Hu Y, Wang J, Yu N, et al. Maximum margin clustering with pairwise constraints. In: Proceedings of IEEE International Conference on Data Mining, Pisa, Italy, 2008. 253-262

[7] Wang F, Zhao B, Zhang C. Linear time maximum margin clustering. *IEEE Transactions on Neural Networks*, 2010, 21(2):319-332

[8] Collobert R, Sinz F, Weston J, et al. Large scale transductive SVMs. *Journal of Machine Learning Research*, 2006, 7:1687-1712

[9] Joachims T. A support vector method for multivariate performance measures. In: Proceedings of the International Conference on Machine Learning, Bonn, Germany, 2005. 377-384

[10] Chapelle O. The benchmark data sets. http://olivier.chapelle.cc/ssl-book/benchmarks.html, 2006

[11] Rennie J. 20 Newsgroups. http://people.csail.mit.edu/jrennie/20Newsgroups/, 2008

[12] Sindhwani V. WebKB-Page. http://people.cs.uchicago.edu/~vikass/research.html, 2005

Research on robust semi-supervised clustering algorithm based on the maximum margin principle and pairwise constraints

Zeng Hong, Song Aiguo, Lu Wei

(School of Instrument Science and Engineering, Southeast University, Nanjing 210096)

Abstract

To solve the problem that the existing semi-supervised maximum margin clustering algorithm does not work robustly when lots of very similar samples exist in different categories, this study adopted the tactics below: Firstly, design a robust loss function for violating the pairwise constraints based on the maximum margin principle, which features robust penalization to the violation of the pairwise constraints; Secondly, design an iterative algorithm based on the constrained concave-convex procedure (CCCP) to improve the clustering accuracy. Based on the tactics, a new semi-supervised clustering algorithm, the robus pairwise constrained maximum margin clusting (RPCMMC) algorithm, was put forward. The experimental results demonstrate that the proposed algorithm can overcome the drawbacks of the existing semi-supervised maximum margin clustering algorithm and outperform some representative semi-supervised clustering algorithms.

Key words: semi-supervised clustering, pairwise constraints, maximum margin principle, robust loss function, constrained concave-convex procedure (CCCP)