文章编号: 1001-0920(2013)01-0029-07

流数据概念漂移的检测算法

张 杰, 赵 峰

(山东科技大学 经济管理学院，山东 青岛 266590)

摘 要: 鉴于流数据具有实时、连续、有序和无限等特点, 使用近似方法便可检测连续分时段的流数据序列, 基于此, 运用目标分布数据, 结合相似分布理论, 提出了利用 Tr-OEM 算法对流数据中的概念漂移现象进行检测. 该算法能够动态地判断流数据概念漂移的发生, 自适应地优化概念漂移的检测值, 适用于不同类型的流数据. 通过分析和实验仿真可以表明, 该算法在处理流数据概念漂移时具有较好的适应性.

关键词: 流数据；概念漂移；检测；数据挖掘

中图分类号: TP393.02 **文献标志码:** A

Detecting algorithm of concept drift from stream data

ZHANG Jie, *ZHAO Feng*

(School of Economic and Management，Shandong University of Science and Technology，Qingdao 266590，China. Correspondent：ZHANG Jie，E-mail：zhangjie0371@163.com)

Abstract: Based on the stream data with the characters such as real-time, continuous, orderly and unlimited, the continuous-time data sequence can be detected by using the approximate method. Based on this, making use of samples not only from the target distribution but also from similar distributions, Tr-OEM algorithm is proposed to detect the concept drift phenomenon in stream data. This algorithm dynamically estimates the occurrence of concept drift in stream data, automatically determines optimizing or reconstructing classifiers, and is applied to different types of stream data. The analysis and simulation experiments show that the proposed algorithm has better adaptability while handling the concept drift in stream data.

Key words: stream data；concept drift；detecting；data mining

0 引 言

流数据 (stream data) 是一种实时连续的数据信息序列. 目前, 计算机与信息技术得到了快速发展, 可以为越来越多的领域提供大量数据, 例如: Web 服务器、传感器网络、股票期货分析、电子商务等, 这些领域都存在着流数据, 与传统磁盘上的数据静态性不同, 这类数据具有快速、易逝和不可预测的特点, 并经常出现概念漂移现象, 即在进行数据挖掘时, 往往会有新的目标概念蕴藏在到达的流数据中. 这类目标概念会随着时间的推移而发生显著或缓慢的变化, 从而给流数据的处理和利用带来较大的障碍. 这些特性的存在给流数据概念漂移的检测带来了极大挑战, 这也是目前数据挖掘领域研究的一个难点和前沿问题. Domingos 等[1]提出了增量决策树算法 (VFDT), 该算法使用 Hoeffding 边界保证算法, 从而与批量学习的输出模型走向保持一致. 但是该算法也存在一定的缺陷性, 它假定数据是随机抽取于静态分布中的, 因而不能将数据随时间推移而变化的特性反映出来, 也无法对流数据中的概念漂移现象进行检测. Hulten 等[2]引入滑动窗口概念, 提出了 CVFDT(concept-adapting VFDT) 算法. 该算法采取生成替代子树的方法对概念进行保存, 只要替代子树具有更好的精度就会代替旧子树, 但是该算法不能对窗口内的概念差异进行处理, 仅通过优化这种不彻底的方式来处理概念漂移. Wang 等[3]提出一种集成加权分类的方法来处理概念漂移, 将窗口分成若干数据块, 各个数据块生成各自的分类器, 然后采取一定的策略演化成最优分类器. 该方法虽然较好地解决了概念漂移, 但是各个分类器会出现因训练数据不足而影响精度的情况, 从而影响最终的检测结果.

收稿日期: 2011-09-04；**修回日期:** 2012-04-01.

基金项目: 中国博士后基金项目(20100481284)；全国统计科研计划重点项目(2011LZ048)；山东省优秀中青年科学家科研奖励基金项目(BS2012SF024).

作者简介: 张杰(1975−), 男, 副教授, 博士后, 从事系统评价理论与技术的研究；赵峰(1978−), 女, 副教授, 博士后, 从事技术经济、风险管理与控制的研究.

本文在前人研究的基础上, 提出了Tr-OEM算法, 动态地解决了流数据的概念漂移检测问题, 并能适应不同类型的流数据.

1 问题的描述

1.1 问题的定义

假设流数据是形如 $z_1, z_2, \cdots, z_i, \cdots, z_n, \cdots$ 按顺序不断流入的数据元素序列, 其中每个数据元素 $z_i = (x, y)$ 由特征向量 $x \in \chi$ 和类标号 $y \in Y$ 组成. 按照时间的先后将流数据组织成固定大小的如 $S_1, S_2, \cdots, S_i, \cdots$ 的数据块序列. 基本窗口对应一个数据块 S, 记为 w, 包含在数据块中的数据数量用窗口宽度表示, 记为 $|w|$. 滑动窗口由一系列基本窗口组成, 记 $W = w_1, w_2, \cdots, w_i, \cdots, w_k$. 其中: w_i 为滑动窗口中的第 i 个基本窗口, w_k 为当前窗口保存最新的数据窗口. C_i 表示在基本窗口 w_i 上训练得到的基本分类器, $|W|$ 表示滑动窗口的宽度, 相当于多分类器的最大个数.

定义 1 (虚拟错误率)[4] 假设分类器 h 关于目标函数 f 在分布 Φ 上的虚拟错误率为 p, 则有

$$p = \text{error}_{\Phi}(h) = \Pr_{x \in \Phi}[f(x) \neq h(x)]. \tag{1}$$

定义 2 (真实错误率)[4] 设 $\text{error}_s(h)$ 为分类器 h 关于目标函数 f 在样本集 S 的真实错误率, 则有

$$\text{error}_s(h) = \frac{1}{n}\sum_{x \in S}\delta(f(x), h(x)). \tag{2}$$

定义 3 (自适应时间窗口分类器) 假设 G_u 为自适应时间窗口分类器, 则 G_u 同时为在由最近的 u 个连续的基本窗口 $w_{k-u+1} \bigcup \cdots \bigcup w_k$ 所组成的自适应窗口 AW 上建立的分类器, 其中 AW 为候选窗口集合 $\{w_1 \bigcup \cdots \bigcup w_k, w_2 \bigcup \cdots \bigcup w_k, \cdots, w_k\}$ 中产生的分类器在最新训练数据上有最小期望误差的基本窗口组合. 这意味着 $w_{k-u+1}, w_{k-u+2}, \cdots, w_{k-1}$ 与 w_k 有相似的概念.

依据文献 [5], 假设位于基本窗口 w_i 内的数据均独立且其分布均服从于同一分布 $P_i(x, y)$, 随着时间的推移, 分布在滑动窗口内的各个基本窗口的位置不断发生变化. 假设在滑动窗口中最新窗口为 w_k, 算法要求只将 w_k 存入缓存, 此时不能再次访问位于先前窗口中的数据, 但是建立在其上的分类器 $C_1, C_2, \cdots, C_{k-1}$ 已经保存.

1.2 流数据概念漂移的特征

定义 4 (概念漂移)[6] 检测的目标是从训练数据集的过程中学习所蕴含的概念, 即从属性到类别的映射关系, 决策树、贝叶斯网络等类似经典算法均假定映射模型只存在一种固定形式, 但对于流数据而言, 随着数据的流动, 映射关系不断发生变化, 这种变化称为概念漂移. 概念漂移具有较强的时间性, 当数据在一定的时间内反映的只是当时的概念, 但随着时间的推移, 可能会改变数据中的概念.

当流数据逐步流入时, 流数据被分成数据段 $S_1, S_2, \cdots, S_n$, 其中 S_n 为最新时间流入的数据段. 数据段 S_n 中的实例 $X = \{A_1, A_2, \cdots, A_n, C\}$. 其中: A_j 为 X 的属性, 其值为 e_j; 类变量 $C = c_k, k = 1, 2, \cdots, m$. $C_{k,S_n} = \{X \in S_n, C(X) = c_k\}$ 在缺乏概念漂移时, 分类数据块中的实例所期望的信息[7]由下式给出:

$$\text{Info}(S_n) = -\sum_{k=1}^{m} p_k 1b(p_k). \tag{3}$$

其中: $\forall X \in S_n, p_k = P(C(X) = c_k)$, 用 $|C_{k,S_n}|/|S_n|$ 估计; $\text{Info}(S_n)$ 是识别 S_n 中实例的类标记所需要的平均信息量, 且有

$$\text{Info}_{A_j}(S_n) = \sum_{l=1}^{v}\frac{|S_{nl}|}{S_n}\text{Info}(S_{nl}) = \sum_{l=1}^{v}\frac{|S_{nl}|}{S_n}\Big(-\sum_{z=1}^{m} p_z 1b(p_z)\Big). \tag{4}$$

其中

$$S_{nl} = \{X | X = \{A_1, A_2, \cdots, e_l, \cdots, A_n, C\}, X \in S_n,\ A_j = e_l\},$$

$$v = |A_j|,\ \forall X \in S_{nl},\ p_z = P(C(X) = c_z),$$

用 $|C_{z,S_{nl}}|/|S_{nl}|$ 估计. 信息变化量为

$$\text{Change}(A_j) = \text{Info}(S_n) - \text{Info}_{A_j}(S_n) = -\sum_{k=1}^{m} p_k 1b(p_k) + \sum_{l=1}^{v}\frac{|S_{nl}|}{|S_n|}\sum_{z=1}^{m} p_z 1b(p_z). \tag{5}$$

当数据分布趋于一致且稳定时, 检测准确率应该在 $(0.01, 0.09)$ 的范围内波动; 否则, 在某个时间点, 检测精度明显下降, 表示有概念漂移发生, 即有新的概念出现在流数据中[8]. 利用平均平方正确率作为评估参数来检测概念的流动, 有

$$\theta(n) = \frac{1}{|S_n|}\sum_{X \in S_n}[\max(P(C_1|X), P(C_2|X), \cdots, P(C_m|X))]^2. \tag{6}$$

当 S_{n+1} 到达时, 通过计算 $|\theta(n+1) - \theta(n)|$ 检测是否有概念漂移发生. 若 $|\theta(n+1) - \theta(n)| \geqslant 0.09$, 即分类器的精度明显下降, 则表明有概念漂移发生, 更新特征子集, 然后更新分类器[9].

2 基于 OEM 与 Tr-OEM 算法的流数据概念漂移的检测分析

2.1 OEM 算法

静态流数据检测估计的最新工作是边界元法 (BEM)[10]. 文献 [11] 提出了对象交换模型 (OEM) 算法, 通过其对等级漂移估计进行检测, 该算法也是对

BEM 算法的一个延伸, 检测漂移的 OEM 算法表述为

$$\hat{P}(\omega_i|x_k)=\frac{P(x_k|\omega_i\hat{P}_{k-1}(\omega_i))}{\sum\limits_{\omega_j\in\Omega}P(x_k|\omega_j)\hat{P}_{k-1}(\omega_j)},$$

$$\hat{P}_0(\omega_i)=P_{\rm ini}(\omega_i),$$

$$\hat{P}_k(\omega_i)=(1-\alpha)\hat{P}_{k-1}(\omega_i)+\alpha\hat{P}(\omega_i|x_k). \tag{7}$$

其中: $\hat{P}(\omega|x_k)$ 为对第 k 个样例后验几率的估计, $\hat{P}_k(\omega_i)$ 为升级的流数据检测概率分布, α 为经验上的遗忘因子. 初时 $t=0$, 得到测试数据的检测初始为 $\hat{P}_0(\omega_i)=P_{\rm ini}(\omega_i)$, 初始分类器在独立测试集 $\hat{P}_0(\omega_i)$ 上进行. 在 BEM 算法的基础上, 文献 [11] 证明了 ω_i 等级的后验概率可以用来检测漂移 $P(\omega_i)$ 的估计量. 当 $t=k-1$ 时, 得到 $\hat{P}_{k-1}(\omega_i)$ 的漂移估计 $P_{k-1}(\omega i)$; 当 $t=k$ 时, 在测试数据 (x_k,y_k) 中, 生成分类器会计算出后验几率估计 $\hat{P}(\omega_i|x)$, 如方程组 (7) 中第 2 式所示; 当 $t=k$ 时, 数据检测依据指数遗忘规律得到了一定程度的调整, 如方程组 (7) 中第 3 式所示.

相对于 BEM 算法, 当流数据检测出现动态变化时, OEM 算法在虚拟数据集和真实数据集方面均表现出一种有效的状态, 但此时漂移检测能力还有上升的空间. 当概念漂移率出现提升时, 检测效果会有所下降, 造成此现象的原因有 3 点: 1) 概念漂移里的检测不是一个小样例参数估计问题; 2) OEM 算法较差的检测能力; 3) 遗忘因子 α 是控制升级率的重要参数, 其值由经验确定, 特别是针对快速漂移问题, 而 OEM 算法的效果对值非常敏感.

2.2　Tr-OEM 算法

概念漂移里的检测是一个典型性的大量样例分类估计问题, 所以, 在综合考虑检测算法与 OEM 分类算法的基础上, 可以得到一个新的 OEM 算法对概念的漂移进行检测, 即 Tr-OEM 算法, 描述为

$$\begin{cases}\hat{P}_0(\omega_i)=P_{\rm ini}(\omega_i),\\ \hat{P}_0(\omega_i|x_k)=\dfrac{P(x_k|\omega_i\hat{P}_{k-1}(\omega_i))}{\sum\limits_{\omega_j\in\Omega}P(x_k|\omega_j)\hat{P}_{k-1}(\omega_j)},\\ \hat{P}_{tr}(\omega_i)=\dfrac{1}{n}\sum\limits_{l=0}^{n-1}\tilde{P}(\omega_i|x_{k+l}),\\ \hat{P}_k(\omega_i)=(1-\alpha)\hat{P}_{k-1}(\omega_i)+\alpha\hat{P}_{tr}(\omega_i).\end{cases} \tag{8}$$

其中: $\hat{P}_{tr}(\omega_i)$ 为分类检测值, n 为 $\hat{P}_{tr}(\omega_i)$ 使用的样例数量.

2.3　流数据概念漂移的检测

受转移学习原理的启发, 为了解决流数据概念漂移问题, 在 OEM 和 Tr-OEM 算法的基础上, 提出了具体的流数据概念漂移的检测原理. 若 x_{t+k} 为 $P_{t+k}(x)$ 分布中 X_t 的相邻样例, $P_{t+k}(x)$ 与 $P_t(x)$ 是不同的, 且 $P_{t+k}(x)$ 和 $P_t(x)$ 有着相同的内等级分布, 混合分布的格式也相同, 则二者高度相似, 利用 x_{t+k} 中包含的信息对 $P_t(\omega_i)$ 中的信息进行检测. 在学习协变量转变问题时, 最简单有效的方法是对 $P_{t+k}(\omega_i)$ 的加权估计值 $\tilde{P}_{t+k}(\omega_i)$ 进行总结, 并进行更新得到一个标为 $\tilde{P}_{t+k}(\omega_i)$ 的新的检测值

$$\tilde{P}_{tr}(\omega_i)=\sum_{k=0}^{n-1}\alpha_k\tilde{P}_{t+k}(\omega_i),\ \sum_{k=0}^{n-1}\alpha_k=1, \tag{9}$$

其中: α_k 为加权系数, $n-1$ 为相邻两个样例的数量.

如何对加权系数 α_k 进行选择应视具体情况进行分析, 首先考虑概念漂移的速度, 但实际生活中获得关于概念漂移的信息较为困难, 所以, 本文采用比较常用的方法, 即将 α_k 的值定为 $1/n$, 式 (9) 可改写为

$$\tilde{P}_{tr}(\omega_i)=\frac{1}{n}\sum_{k=0}^{n-1}\tilde{P}_{t+k}(\omega_i). \tag{10}$$

与检测量 $\tilde{P}_{[t,\cdots,t+n-1]}(\omega_i)$ 不同, 检测值 $\tilde{P}_{tr}(\omega_i)$ 不仅对位于目标分布 $P_t(x)$ 内的样例进行了使用, 而且对与 $P_t(x)$ 密切相关分布中的样例也予以采用, 将该方法称为概念漂移的检测方法, 并记 $\tilde{P}_{tr}(\omega_i)$ 为概念漂移的检测值. 对相关分布的样例予以采用会在某种程度上使检测值的平方偏差得到提高, 但过多使用信息也会使估计方差得到降低, 接下来论证概念漂移通常会给出更好的效果.

标记 1　θ_k 为分布 $P(\theta_k)$ 的真实值参数, $k=1,2,\cdots,n$; $\theta_{[1,2,\cdots,n]}$ 为 θ_1, θ_2, $\cdots$, θ_n 的缩写, 平均值记为 $\bar{\theta}=1/n\sum\limits_{k=1}^{n}\theta_k$; 参数 θ_k 的估计值记为 $\tilde{\theta}_k$, 概念漂移检测估计值记为 $\tilde{\theta}_{\rm tr}$, $\tilde{\theta}=(1/n)\sum\limits_{k=1}^{n}\tilde{\theta}_k$.

定义 5　$\tilde{\theta}_{\rm tr}$ 为一个整体优于 $\tilde{\theta}_{[1,2,\cdots,n]}$ 的检测估计量, 满足

$$\sum_{k=1}^{n}E(\tilde{\theta}_{\rm tr}-\theta_k)^2\leqslant\sum_{k=1}^{n}nE(\tilde{\theta}_k-\theta_k)^2. \tag{11}$$

定义 5 是建立在平均平方误差原则基础上的, 其中 $\sum\limits_{k=1}^{n}nE(\tilde{\theta}_{\rm tr}-\theta_k)^2$ 为 $\tilde{\theta}_{\rm tr}$ 的总体分类效果, $\sum\limits_{k=1}^{n}nE(\tilde{\theta}_k-\theta_k)^2$ 为 $\tilde{\theta}_{[1,2,\cdots,n]}$ 的总体检测效果, 它定义了一个检测概念漂移的计算方法. 在标记 1 和定义 5 的基础上, 概念漂移检测值的属性包含在下列原理 1 和原理 2 中.

原理 1　假设检测估计量 $\tilde{\theta}_{[1,2,\cdots,n]}$ 是无偏的[12], 则当且仅当下列条件满足时, $\tilde{\theta}_{\rm tr}$ 是一个较好的检测估计值:

$$\sum_{k=1}^{n}(\theta_k^2-\bar{\theta}^2)\leqslant\frac{n-1}{n}\sum_{k=1}^{n}{\rm Var}(\tilde{\theta}_k), \tag{12}$$

其中 ${\rm Var}(\tilde{\theta}_k)$ 为 $\tilde{\theta}_k(k=1,2,\cdots,n)$ 的方差.

若检测估计量 $\tilde{\theta}_{[1,2,\cdots,n]}$ 是无偏的, 则需要考虑分

类偏差的影响, 如原理2所述.

原理2 下列条件满足时, $\tilde{\theta}_{\rm tr}$ 是一个较好的检测估计值[12]:

$$\sum_{k=1}^{n}(\theta_k^2-\bar{\theta}^2)+\frac{2}{n}\sum_{k=1}^{n}\{b(\tilde{\theta}_k)(\bar{\theta}-\theta_k)\}\leqslant \frac{n-1}{n}\sum_{k=1}^{n}{\rm Var}(\tilde{\theta}_k), \tag{13}$$

其中 $b(\tilde{\theta}_k)$ 和 ${\rm Var}(\tilde{\theta}_k)$ 分别为 $\tilde{\theta}_k$ 的偏差和方差, $k=1,2,\cdots,n$.

原理1和原理2讨论了概念漂移检测值在不同情况下的属性. 在原理1中, 式(12)给出了当检测估计量 $\tilde{\theta}_{[1,2,\cdots,n]}$ 无偏时, 确保概念漂移检测估计值 $\tilde{\theta}_{\rm tr}$ 的结果为良好的必要和充分条件, $\sum_{k=1}^{n}(\theta_k^2-\bar{\theta}^2)$ 区分了检测估计量 $\tilde{\theta}_{[1,2,\cdots,n]}$ 的不同, 它测量了引入概念漂移检测估计值带来的增加的平方偏差. 如何定义概念漂移里的检测估计比率很重要, 但没有明确的方法. $\sum_{k=1}^{n}(\theta_k^2-\bar{\theta}^2)$ 可以视为概念漂移对 $\tilde{\theta}_{\rm tr}$ 检测效果的影响, 因此, $\sum_{k=1}^{n}(\theta_k^2-\bar{\theta}^2)\Big/n$ 平均不同可以用来定量区分检测率, 其中 n 为所用检测估计值的样例数量. 在式(12)中, $\sum_{k=1}^{n}{\rm Var}(\tilde{\theta}_k)$ 为方差的总和, 它体现检测估计量 $\tilde{\theta}_{[1,2,\cdots,n]}$ 的整体效果. $(n-1)\Big/n\sum_{k=1}^{n}{\rm Var}(\tilde{\theta}_k)$ 量化了概念漂移检测值后的减少方差, 当检测估计量 $\tilde{\theta}_{[1,2,\cdots,n]}$ 无偏时, 原理2提供了一个充分条件, 与式(12)相比, 式(13)提出了一个额外的术语 $2\Big/n\sum_{k=1}^{n}\{b(\tilde{\theta}_k)(\bar{\theta}-\theta_k)\}$, 其反映了与参数检测连在一起的分类偏差的影响, 但是, 当值相对较大时, 这种影响会很弱. 总之, 概念漂移检测值的优越性主要取决于 $\sum_{k=1}^{n}(\theta_k^2-\bar{\theta}^2)$(即检测参数影响)和 $(n-1)\Big/n\sum_{k=1}^{n}{\rm Var}(\tilde{\theta}_k)$(即检测量方差的影响)之间的比较, 当 $\sum_{k=1}^{n}(\theta_k^2-\bar{\theta}^2)\leqslant(n-1)\Big/n\sum_{k=1}^{n}{\rm Var}(\tilde{\theta}_k)$ 条件得到满足时, 流数据中的概念漂移便能得到较好的检测.

2.4 检测标准

所谓的标准平均变化距离(AVD)即为非真实数据序列的真实概念漂移, 在对非真实数据集进行实验时, 通过对漂移概率 $\hat{P}_t(\omega_i)$ 和真实漂移概率 $P_t(\omega_i)$ 之间的散度进行检测来评估各种算法的漂移检测效果. 在概率理论和统计中, 利用较多篇幅介绍了漂移和真实漂移两种概率分布之间的散度度量, 其中变化距离是一个最直接的度量[9]. 按下式计算位于 $\hat{P}_t(\omega_i)$ 和 $P_t(\omega_i)$ 之间的变化距离:

$$V(\hat{P}_t,P_t;\Omega)=\sum_{\omega_i\in\Omega}|\hat{P}_t(\omega_i)-P_t(\omega_i)|. \tag{14}$$

其中: t 为测试样例指数, $\hat{P}_t(\omega_i)$ 为估计漂移概率, $P_t(\omega_i)$ 为真实漂移概率. 根据变化距离, 定义平均变化距离为

$${\rm AVD}(k)=\Big(\sum_{t=1}^{k}V(\hat{P}_t,P_t;\Omega)\Big)\Big/k. \tag{15}$$

本文对分类效果进行评估以分类精度(ACC)为标准, 在整体数据集上, 它表示比较不同算法的整体效果. 假设有 n 个样例包含在数据集中, 被正确检测新目标样例有 m 个, 此时可以将该数据集的分类精度定义为 ${\rm ACC}(k)=m/n$. 为了体现分类精度随着数据序列变化这一特征, 需要对各种算法的检测效果进行动态估计, 因此将累计精度 ${\rm AACC}(k)$ 定义为测试序列中前 k 个样例的分类精度(ACC). 若正确分类了 k 个样例中所包含的 m_k 个样例, 则 ${\rm AACC}(k)=m_k/k$.

3 实验结果与分析

为了论证流数据概念漂移检测算法的有效性, 对虚拟和现实数据集进行仿真实验. 所有的数据集在文献[11-12]中部分使用过, 在这些数据集上测试Tr-

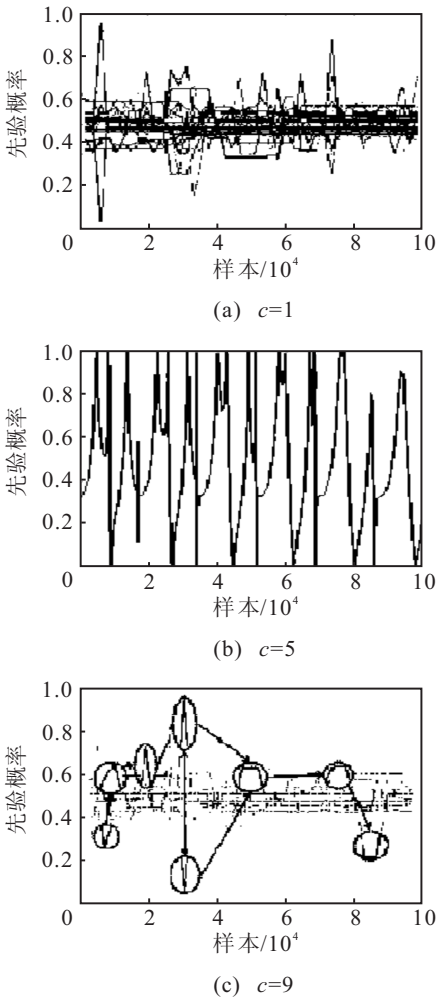

图1 不同调整参数 c 下对1D2CV数据集 $P_t(\omega_i)$ 的估计

OEM 算法的效果, 仿真结果表明该算法对流数据概念漂移具有较好的检测效果.

3.1 虚拟数据

综合数据集“1D2CV”是一个有显著概念漂移的二级一元数据序列, 在 1D2CV 数据集里, 两个极分别由高斯分布 $N(0,1)$ 和 $N(1,1.5)$ 单独得到, 每个级生成 500 个样例, 测试序列里的样例根据指定的不同检测分布得出, 检测漂移的生成如下:

$$\tilde{P}_t(\omega_i) = \left| i + 5it/L + 10\sin(4itc/L) \right|, \tag{16}$$

$$P_t(\omega_i) = \tilde{P}_t(\omega_i) \Big/ \sum_{\omega_j \in \Omega} \tilde{P}_t(\omega_j). \tag{17}$$

其中: t 为样例指数, $i \in [1,2]$, c 为调整参数, 测试序列长度为 $l = 100\,000$, c 为控制检测漂移的检测率. 如图 1 所示, 当 c 增长时, 检测概念漂移得更快.

3.2 真实数据

为了估计非虚拟数据集不同算法的效果, 真实数据需要对“森林覆盖”的大量真实世界数据序列进行实验[12]. 森林覆盖型数据集是一个多级问题, 它具有不同的分类漂移, 该数据集共含有 581 012 个样例, 这些样例分属于 7 个等级, 各样例由 54 个象征组成, 具有数量变量 10 个, 二进制自然保护区 4 个, 二进制土壤类型变量 40 个, 样例中的前 15 120 个被选中作为测试样例, 其余用作建立测试序列. 在真实的数据集中, 事先并不知道数据序列的真实概念漂移情况, 为了便于对样例漂移进行检测, 采用相邻的两个真实标签来计算用作目标检测漂移的近似价值的频率, 本文用到 2 000 个近邻. 为了达到节省空间的目的, 图 2 只显示近似分类漂移 1 级和 2 两级, 可以看到森林覆盖

表 1　对 1D2CV 数据集的 OEM 算法和 Tr-OEM 算法得到的 $\mathrm{AVD}(k)$ 的比较

参数	AVD/%	$\alpha=0.05$	$\alpha=0.10$	$\alpha=0.15$	$\alpha=0.20$	$\alpha=0.25$	$\alpha=0.30$	$\alpha=0.35$	$\alpha=0.40$	$\alpha=0.45$	$\alpha=0.50$
$c=1$	OEM	14.88	17.96	21.70	25.31	28.81	31.80	35.73	39.02	41.94	46.55
	Tr-OEM	13.85	15.23	16.71	17.83	18.64	19.24	19.71	20.09	20.39	20.64
$c=5$	OEM	24.22	22.75	24.89	28.04	31.56	35.37	39.32	43.47	47.71	52.17
	Tr-OEM	20.20	17.07	17.14	17.86	18.60	19.28	19.88	20.40	20.83	21.19
$c=9$	OEM	30.84	28.11	29.09	31.57	34.72	38.07	41.43	44.98	48.67	52.27
	Tr-OEM	25.32	20.44	19.78	20.26	20.97	21.65	22.25	22.77	23.21	23.60
$c=13$	OEM	30.93	28.35	28.82	30.47	32.75	35.45	38.51	41.91	45.54	49.02
	Tr-OEM	25.08	20.12	19.28	19.64	20.30	20.97	21.60	22.16	22.63	23.05
$c=17$	OEM	32.38	28.22	27.75	28.96	30.99	33.48	36.29	39.26	42.44	45.90
	Tr-OEM	25.67	19.89	19.04	19.48	20.34	21.24	22.07	22.79	23.40	23.93
$c=21$	OEM	35.37	32.92	32.62	33.66	35.57	38.06	41.02	44.33	47.73	51.05
	Tr-OEM	27.98	21.95	20.18	20.16	20.63	21.24	21.86	22.46	23.02	23.52
$c=25$	OEM	37.32	35.27	34.70	35.50	35.17	39.43	42.11	45.10	48.36	51.73
	Tr-OEM	30.90	25.10	25.84	25.79	23.43	24.29	25.15	25.93	26.62	27.24

表 2　对 1D2CV 数据集的 OEM 算法和 Tr-OEM 算法得到的 $\mathrm{ACC}(k)$ 的比较

参数	ACC/%	$\alpha=0.05$	$\alpha=0.10$	$\alpha=0.15$	$\alpha=0.20$	$\alpha=0.25$	$\alpha=0.30$	$\alpha=0.35$	$\alpha=0.40$	$\alpha=0.45$	$\alpha=0.50$
$c=1$	OEM	79.03	78.85	78.80	78.51	78.37	78.35	78.22	78.12	78.00	78.03
	Tr-OEM	79.03	78.85	78.80	78.51	78.37	78.35	78.22	78.12	78.00	78.03
$c=5$	OEM	74.03	74.55	74.48	73.90	73.42	72.71	71.91	71.01	69.08	68.28
	Tr-OEM	75.00	75.75	75.92	75.88	75.67	75.51	75.48	75.36	75.37	75.28
$c=9$	OEM	72.54	73.66	73.53	73.04	72.67	72.15	71.28	70.51	69.39	68.46
	Tr-OEM	74.38	75.88	76.05	75.91	75.85	75.79	75.67	75.83	75.75	75.74
$c=13$	OEM	71.72	73.37	73.79	73.54	73.25	72.61	72.09	71.54	70.55	69.48
	Tr-OEM	73.35	75.17	75.58	75.30	75.00	75.13	74.77	74.67	74.65	74.68
$c=17$	OEM	71.48	72.78	73.23	73.05	72.72	75.30	71.82	71.37	70.77	69.97
	Tr-OEM	73.25	74.90	75.66	75.77	75.68	75.48	75.40	75.24	75.40	75.10
$c=21$	OEM	69.35	70.24	71.02	71.02	71.31	70.91	70.27	69.71	69.03	67.53
	Tr-OEM	71.53	73.61	73.93	74.30	74.35	74.23	74.18	74.16	74.06	74.00
$c=25$	OEM	70.07	71.18	71.60	71.60	71.32	71.01	70.29	69.64	68.71	67.56
	Tr-OEM	70.07	71.18	71.60	71.60	71.32	71.01	70.29	69.64	68.71	67.56

型数据集中的明显概念漂移现象, 如近似分类漂移1级在序列初始是明显的(达到了80%), 而在序列的中间其命题降到了30%[9].

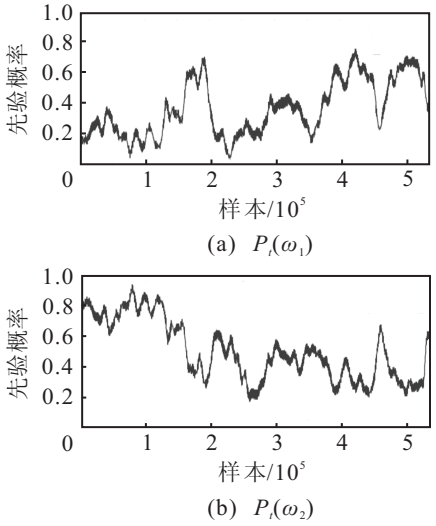

图 2 森林覆盖型数据集分类漂移估计

3.3 虚拟数据结果

在虚拟数据集中, 虚拟数据的结果存在真实的漂移$P(\omega_i)$, 通过各算法也可以计算出$P(\omega)$和估计漂移$\hat{P}(\omega_i)$两者的平均变化距离AVD, 表1列举出了在1D2CV7数据集中利用不同遗忘因子α得出的算法Tr-OEM和算法OEM的AVD. 表1中: $k=10\,000, n=50, P_{\rm ini}(w_i)=0.5$. 结果表明, 在不同的遗忘因子和调整参数下, Tr-OEM算法有助于优化检测效果, 由于概念漂移检测率得到提高, OEM算法和Tr-OEM算法的效果会有所下降. 但是, Tr-OEM算法的效果相对稳定, OEM算法的效果越差, Tr-OEM算法改进就会越大. 在现实生活中, 遗忘因子α由经验确定, OEM算法的效果随着α值发生变化, 当$\alpha \geqslant 0.5$时, OEM算法不能被接受, 结果不显示, 此时Tr-OEM算法对α值不敏感.

表2为在具有不同调整参数的1D2CV数据集中利用不同遗忘因子得到的算法OEM和算法Tr-OEM的ACC. 表2中: $k=10\,000, n=50, P_{\rm ini}(w_i)=0.5$. 实验表明, 与概念漂移真实情况相似, 在不同的遗忘因子和调整参数下, ACC能被算法Tr-OEM稳定提高; 从整体上来看, 算法OEM的效果越差, Tr-OEM算法效果越优, 这也使得Tr-OEM算法的效果对α的敏感度比OEM算法低.

图3为利用OEM算法和Tr-OEM算法得到的1D2CV数据集上$P_t(\omega_1)$的真实值和估计值. 图3中: $\hat{P}_t(\omega_1)=0.5, n=50, \alpha=0.2$, $P_t(\omega_1)$表示真实概念漂移, 由点划线表示, 估计概念漂移$\hat{P}_t(\omega_1)$由实线表示. 显然, 算法Tr-OEM比算法OEM在曲线上更准确.

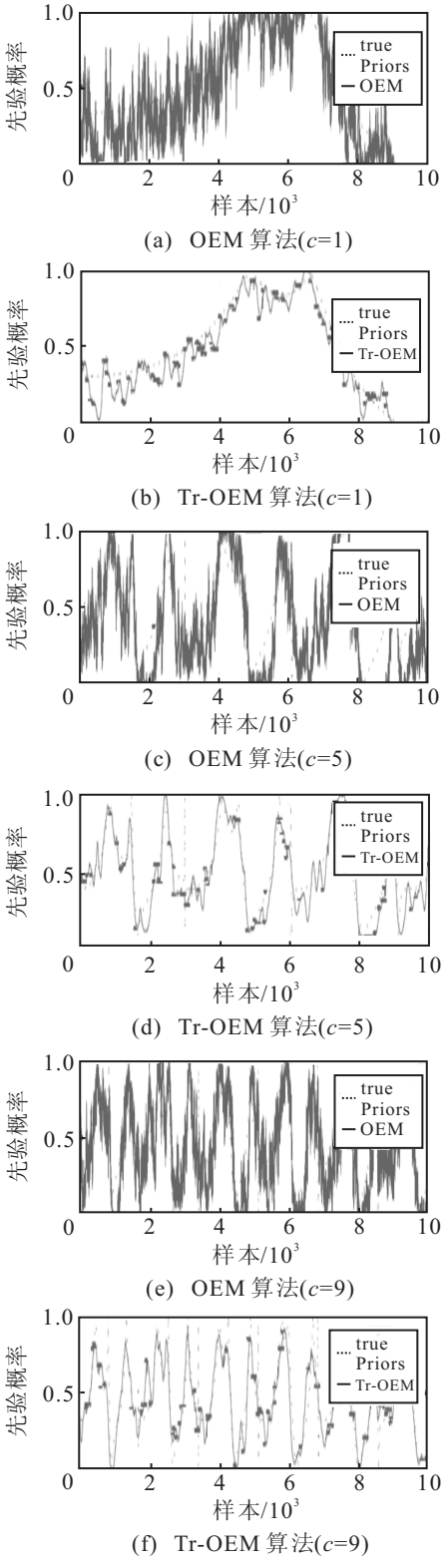

图 3 1D2CV数据集上$P_t(\omega_1)$的真实值和估计值

3.4 现实数据的结果

对"森林覆盖"的大量现实数据进行实验[11-12], 前15 120个样例作为测试样例, 检测频率为0.142 9, 用来对OEM算法和Tr-OEM算法进行初始的分类漂移. 表3为模糊控制规则, 实验表明, OEM算法的效果随着α值发生变化, 为了得到最佳的效果, 只有当$\alpha \leqslant 0.05$时, α值才会有所不同, 当$\alpha \geqslant 0.1$时,

OEM 算法的效果会不断恶化. 由于 α 值从经验中得来, 在实践中使用 OEM 算法较为不便.

表 3　模糊控制规则

α	0.01	0.05	0.10	0.20	0.30	0.40	0.50
OEM /%	63.54	62.38	60.86	59.29	58.92	58.67	58.41
Tr-OEM/%	62.94	64.15	65.19	67.54	66.37	64.76	64.25

图 4 是基于自适应时间窗口分类器, Tr-OEM 算法和 OEM 算法在超平面数据集上的比较. 图 5 是 Tr-OEM 算法和 OEM 算法在真实数据集上的比较. 超平面数据集选择属性个数为 20, 真实数据集与前文所用的数据集相同. 从图 4 和图 5 均可以看出 Tr-OEM 算法比 OEM 算法的精度高, 且适应概念漂移快.

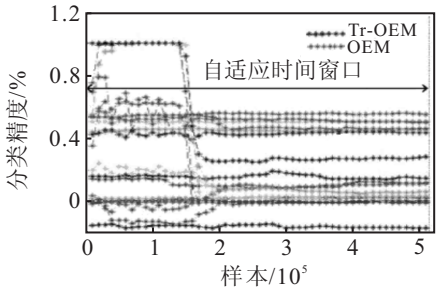

图 4　OEM 和 Tr-OEM 算法在超平面数据集上的比较

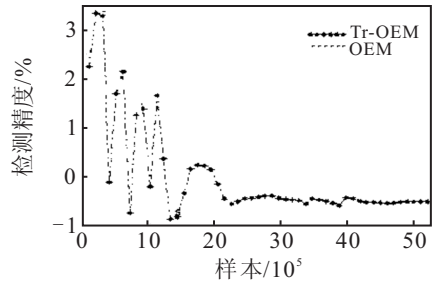

图 5　OEM 和 Tr-OEM 算法在真实数据集上的检测效果

4　结　论

流数据中对于适应概念漂移挖掘算法的研究是当前数据挖掘领域的一个热点问题, 其中检测概念漂移是流数据处理过程中的难点. 为了对流数据中的概念漂移进行检测, 本文在 OEM 算法的基础上构建 Tr-OEM 算法, 并将该算法与流数据概念漂移的检测融合在一起. 对虚拟和现实数据集的实验表明, 在检测流数据概念漂移上, Tr-OEM 算法均优于 OEM 算法. 同时, 通过分析和实验表明, 该算法在流数据概念漂移检测方面具有较好的适应性.

参考文献(References)

[1] Domingos P, Hulten G. Mining high-speed data streams[C]. Proc of ACM Sigkdd Int Conf Knowledge Discovery in Databases. Boston: ACM Press, 2000: 71-80.

[2] Hulten G, Spencer L, Domingos P. Mining time-changing data streams[C]. Proc of ACM Sigkdd Int Conf Knowledge Discovery in Databases. San Francisco: ACM Press, 2001: 97-106.

[3] Wang H, Fan W, Yu P, et al. Mining concept drifting data streams using ensemble classifiers[C]. The 9th ACM Int Conf on Knowledge Discovery and Data Mining. Washington: ACM Press, 2003: 226-235.

[4] Tom Mitchell. Machine learning[M]. McGraw Hill, 1997: 123-126.

[5] Zico Kolter J, Marcus A Maloof. Dynamic weighted majority: An ensemble method for drifting concepts[J]. J of Machine Learning Research, 2007, 8(8): 2755-2790.

[6] 陈照阳, 黄上腾. 流数据分类中的概念漂移问题研究[J]. 计算机应用与软件, 2009, 2(2): 254-256.
(Chen Z Y, Huang S T. On concept drift in stream data classification[J]. Computer Applications and Software, 2009, 2(2): 254-256.)

[7] Li C Q, Ling T W, Hu M. Efficient processing of updates in dynamic XML data[C]. Proc of the 22nd Int Conf on Data Engineering. Washington DC: IEEE Computer Society, 2006: 13-22.

[8] Li C Q, Ling T W, Hu M. Efficient updates in dynamic XML data: From binary string to quaternary string[J]. The Very Large Data Bases J, 2008, 17(3): 573-601.

[9] 李敏, 王勇, 蔡立军. 数据流分类中的增量特征选择算法[J]. 计算机应用, 2010, 30(9): 2321-2323.
(Li M, Wang Y, Cai L J. Incremental feature selection algorithm for data stream classification[J]. J of Computer Applications, 2010, 30(9): 2321-2323.)

[10] Saerens M, Latinne P, Decaestecker C. Adjusting the outputs of a classifier to new a priori probabilities: A simple procedure[J]. Neural Computation, 2002, 14(1): 21-41.

[11] Yang C, Zhou J. Non-stationary data sequence classification using online class priors estimation[J]. Pattern Recognition, 2008, 41(8): 2656-2664.

[12] Zhang Zhi-hao, Zhou Jie. Transfer estimation of evolving class priors in stream data classification[J]. Pattern Recognition, 2010, 43(9): 3151-3161.

[13] Csiszar I. Information-type measures of difference of probability distributions and indirect observations[J]. Studia Scientiarum Mathematicarum Hungarica, 1967, 2(2): 299-318.

[14] Lin J. Divergence measures based on the Shannon entropy[J]. IEEE Trans on Information Theory, 1991, 37(1): 145-151.