

文章编号: 1000-5862(2014)01-0089-06

# 基于互信息的模糊决策树及其增量学习

张 巍, 聂 进, 滕少华

(广东工业大学计算机学院 广东 广州 510006)

**摘要:** 将互信息引进模糊决策树, 用于确定决策树的候选分类属性, 进而构建模糊决策树. 通过增量学习来修正决策树分类模型, 以修正分类效果, 并用实验验证了该方法的有效性.

**关键词:** 模糊决策树; 信息熵; 互信息; 增量学习

**中图分类号:** TP 311

**文献标志码:** A

## 0 引言

在数据挖掘中, 分类是常用的一种方法, 而决策树是最流行的分类模型之一, 它能方便地通过图形化属性结构表示挖掘结构. 通过决策树归纳学习产生规则, 是知识获取过程中常用而有效的方法. Quinlan 于 1986 年提出了 ID3 算法, 它基于分类信息熵来选取扩展属性. 后来人们又相继提出了 C4.5、PUBLIC、CART、SLIQ、SPRING 等算法<sup>[1-3]</sup>. 由于在现实世界中, 人们的认识存在模糊性和不确定性, 这要求人们从模糊环境中获取知识, 因而人们引进了模糊度量<sup>[4]</sup>, 并将模糊概念引入决策树归纳算法中, 即模糊决策树归纳算法. 常用的模糊决策树算法有 Quinlan 的 Fuzzy\_ID3 算法和 Yuan 的 Min\_Ambiguity 算法<sup>[5-6]</sup>. 对模糊决策树归纳学习的研究已经成为当前的一个研究热点<sup>[7-9]</sup>.

在实际应用中, 一方面有些数据随着时间的增长逐步产生, 这造成不能一次性获取整个需要分析的数据集; 另一方面, 有些数据由于时间的推移, 需要移出原有的分析模型中, 使得已构建的学习模型难以适应新的数据与环境要求, 上述 2 个方面要求分析模型具备更新与学习的能力. 增量式学习<sup>[10-12]</sup>成为研究的热点; Carpenter 等于 1991 年提出了一种基于自适应共振理论的方法 ARTMAP; Kasabov 于 2001 年提出了进化模糊神经网络 (EFuNNS) 的方

法; 决策树的 ID3 算法和 C4.5 算法等. 上述算法均是基于单个模型的方法, 他们通过改变模型的结构、学习新的知识, 由此获得适应实际要求的分析模型. 近年来, 一些学者研究了基于集成学习的增量式学习方法.

由于已有的 ID3 及其推广算法在选择分类属性时, 没有考虑属性之间的相关性对分类的影响, 本文基于后验熵和互信息理论来选择属性, 使构建的模型更合理. 另外, 通过对连续型数据进行模糊化处理, 提取模糊规则, 并且通过集成学习来完成对数据集的增量学习.

## 1 数据预处理及数据模糊化

在数据分析之前, 需要将原始数据集进行预处理, 包括数据清洗、数据集成、数据转换和数据归约, 以删除噪声数据、解决数据不一致的现象等, 并将预处理后的数据进行模糊化. 由于本文采用 UCI 标准数据, 因而此处讨论数据的模糊化.

模糊决策树归纳算法要求首先把数据模糊化, 模糊化实际上是一个概念化的过程. 本文利用层次聚类获聚类个数  $k$ , 然后利用密度聚类算法来获取聚类中心点, 利用聚类中心点构造隶属函数, 从而模糊化数据.

假设属性  $A$  有连续型值  $x$ , 希望将  $x$  聚类到  $k$  个语义项  $T_i, i = 1, 2, \dots, k$ , 得到  $k$  个最优聚类中心点

收稿日期: 2013-11-12

基金项目: 教育部重点实验室基金(110411), 广东省自然科学基金(10451009001004804, 9151009001000007) 和广东省科技计划(2012B091000173) 资助项目.

作者简介: 张 巍(1964-), 女, 江西南昌人, 副教授, 主要从事大数据、数据挖掘和协同计算方面的研究.

参数,这些参数将属性用模糊变量表示,并作为三角形隶属度函数的参数,参与计算实值属性数据的隶属度.采用三角隶属度函数来模糊数据,设数据模糊化后产生  $k$  个中心点  $C_1, C_2, \dots, C_k$  (记为  $C = [C_1, C_2, \dots, C_k]$ ) 则当  $i = 1$  时,有

$$T_1(x) = \begin{cases} 1 & x \leq c_1, \\ (c_2 - x) / (c_2 - c_1) & c_1 < x < c_2, \\ 0 & x \geq c_2, \end{cases}$$

当  $1 < i < k$  时,有

$$T_i(x) = \begin{cases} 0 & x \geq c_{i+1}, \\ (c_{i+1} - x) / (c_{i+1} - c_i) & c_i < x < c_{i+1}, \\ (x - c_{i-1}) / (c_i - c_{i-1}) & c_{i-1} < x < c_i, \\ 0 & x \leq c_{i-1}. \end{cases} \quad (1)$$

当  $i = k$  时,有

$$T_k(x) = \begin{cases} 1 & x \geq c_{k+1}, \\ (x - c_{k-1}) / (c_k - c_{k-1}) & c_{k-1} < x < c_{k+1}, \\ 0 & x \leq c_{k-1}. \end{cases}$$

对 UCI 中的 Iris 数据集<sup>[12]</sup>,取数据集中 SpelLength、SpelWidth、PetalLength 和 PetalWidth 属性,按层次聚类<sup>[8]</sup>,得到  $K = 3$ ,即每个属性可以模糊化为 3 个高层语义概念,然后通过 K-means 聚类<sup>[9]</sup>获得

3 个高层语义概念的聚类中心点,结果见表 1. 随机选取 Iris 数据集中 12 个实例数据来作为一个示例数据集说明本文算法,具体数据集见表 2.

表 1 各属性高层语义聚类中心点

属性	MIN	MAX	Cluster1	Cluster2	Cluster3
SpelLength	4.3	7.9	5.072 3	6.186 2	7.235 0
SpelWidth	2.0	4.4	2.640 4	3.178 7	3.844 4
PetalLength	1.0	6.9	1.464 0	4.339 7	5.688 1
PetalWidth	0.1	2.5	0.244 0	1.337 0	2.073 9

表 2 Iris 示例原始数据

ID	SpelLength	SpelWidth	PetalLength	PetalWidth	Class
1	5.1	3.5	1.4	0.2	setosa
2	4.6	3.1	1.5	0.2	setosa
3	4.9	3.1	1.5	0.1	setosa
4	5.8	4.0	1.2	0.2	setosa
5	7.0	3.2	4.7	1.4	versicolor
6	5.2	2.7	3.9	1.4	versicolor
7	6.0	2.9	4.5	1.5	versicolor
8	5.5	2.4	3.8	1.1	versicolor
9	6.3	3.3	6.0	2.5	virginica
10	7.7	3.8	6.7	2.2	virginica
11	5.6	2.8	4.9	2.0	virginica
12	6.7	3.3	5.7	2.1	virginica

由表 1 获得的高层语义聚类中心,用公式(1)可以将表 2 中的数据模糊化为表 3,模糊化后的数据见表 3.

表 3 Iris 模糊化后的示例数据

ID	SpelLength			SpelWidth			PetalLength			PetalWidth			Class		
	Spel- Length1	Spel- Length2	Spel- Length3	Spel- Width1	Spel- Width2	Spel- Width3	Petal- Length1	Petal- Length2	Petal- Length3	Petal- Width1	Petal- Width2	Petal- Width3	setosa	versico- lor	virginica
1	0.97	0.03	0	0	0.52	0.48	1	0	0	1	0	0	1	0	0
2	1	0	0	0.15	0.85	0	0.99	0.01	0	1	0	0	1	0	0
3	1	0	0	0.15	0.85	0	0.99	0.01	0	1	0	0	1	0	0
4	0.35	0.65	0	0	0	1	1	0	0	1	0	0	1	0	0
5	0	0.22	0.78	0	0.9	0.1	0	0.73	0.27	0	0.91	0.09	0	1	0
6	0.85	0.15	0	0.89	0.11	0	0.15	0.85	0	0	0.91	0.09	0	1	0
7	0.17	0.83	0	0.48	0.52	0	0	0.88	0.12	0	0.78	0.22	0	1	0
8	0.39	0.61	0	1	0	0	0.19	0.81	0	0.22	0.78	0	0	1	0
9	0	0.9	0.1	0	0.82	0.18	0	0	1	0	0	1	0	0	1
10	0	0	1	0	0.09	0.91	0	0	1	0	0	1	0	0	1
11	0	0.47	0.53	0.7	0.3	0	0	0.42	0.58	0	0.1	0.9	0	0	1
12	0	0.5	0.5	0	0.82	0.18	0	0	1	0	0	1	0	0	1

## 2 构造模糊决策树

### 2.1 互信息

设集合  $E = \{e_1, e_2, \dots, e_m\}$ ,  $\forall e_i \in E, i = \{1, 2, \dots, N\}$ .  $e_i$  由  $n$  个特征属性  $A_1, A_2, \dots, A_n$  和一个类别属性  $C = \{c_1, c_2, \dots, c_m\}$  构成. 对于每个特征属性  $A_i$  有  $k$  个属性值,表示为  $A_i = \{A_{i1}, A_{i2}, \dots, A_{ik}\}$  ( $i =$

$1, 2, \dots, n$ )  $k$  的值因特征属性  $A_i$  而不同. Shannon 的信息理论使用熵和交互信息来测量随机变量所包含的信息量<sup>[10]</sup>. 熵所度量的是随机变量所包含的不确定性; 条件熵是指在已知某一条件后原随机变量剩下的不确定性. 2 个随机变量之间的交互信息是指 2 个随机变量所共同包含的信息量,它与 2 个随机变量之间的相关性成正比<sup>[11]</sup>.

对于上述问题,离散型随机变量  $C$  有  $m$  种取值,

其中每一种取值的概率为  $P(c_i)$   $c_i \in C$  则训练示例集  $D$  关于类的初始熵为  $H(C) = - \sum_i p(C_i) \log P(C_i)$ . 在已知属性  $A_i$  下, 类的条件熵为  $H(C | A_i) = \sum_j P(A_{ij}) H(C | A_{ij}) = - \sum_j P(A_{ij}) \sum_i P(C_i | A_{ij}) \log P(C_i | A_{ij})$ . 属性  $A_i$  和类  $C$  之间的互信息为<sup>[6]</sup>  $I(C, A_i) = H(C) - H(C | A_i)$ , 则任意属性  $A_i$  与属性  $A_j$  之间的互信息为<sup>[6]</sup>  $I(A_i, A_j) = H(A_i) - H(A_i | A_j) = I(A_j, A_i)$ .

## 2.2 扩展算法

文献[3, 10-15]给出了一种基于信息增益理论选择最佳分类属性的方法, 该算法是 ID3 算法在模糊数据集下的扩展. ID3 算法只考虑了属性带来的信息增益最大, 忽略了已选属性对信息增益的影响. 本文基于互信息理论, 扩展了互信息在模糊数据集下的最佳属性选择方法. 通过已选属性对候选属性和类别属性之间的互信息的影响来计算候选属性的选取.

**定义 1** 给定模糊信息系统  $FIS = (U, A \cup D)$ ,  $\forall A_{ij} \in A_i$  有

$$M(A_{ij}) = \sum_{x \in U} \mu_{A_{ij}}(x). \quad (2)$$

**定义 2** 对于构造决策树中任意分支第  $q$  个节点属性的选取, 已知分支上前  $q-1$  个属性序列为  $U_{q-1} = \{A_{i_{j_1}}, A_{i_{j_2}}, \dots, A_{i_{j_{q-1}}}\}$ , 在属性序列  $U_{q-1} = \{A_{i_{j_1}}, A_{i_{j_2}}, \dots, A_{i_{j_{q-1}}}\}$  下的累积信息熵  $H(B, U_{q-1})$  为

$$H(B, U_{q-1}) = - \sum_{j=1}^{m_b} P(B_j, U_{q-1}) \log_2 P(B_j, U_{q-1}), \quad (3)$$

其中  $P(B_j, U_{q-1}) = M(B_j \times U_{q-1}) / \sum_{j=1}^{m_b} M(B_j \times U_{q-1})$ .

**定义 3** 在属性序列  $U_{q-1} = \{A_{i_{j_1}}, A_{i_{j_2}}, \dots, A_{i_{j_{q-1}}}\}$  条件下, 第  $q$  个属性取  $A_{i_q}$ , 属性条件互信息熵  $H(B, A_{i_q} | U_{q-1})$  为

$$H(B, A_{i_q} | U_{q-1}) = \sum_{j_q=1}^{m_{i_q}} P(A_{i_q, j_q}) H(B, U_{q-1}), \quad (4)$$

其中  $P(A_{i_q, j_q}) = M(A_{i_q, j_q}) / \sum_{j_q=1}^{m_{i_q}} M(A_{i_q, j_q})$ .

**定义 4** 由(3)式和(4)式可得对于决策树分支中第  $q$  个节点的选取在属性序列  $U_{q-1} = \{A_{i_{j_1}}, A_{i_{j_2}}, \dots, A_{i_{j_{q-1}}}\}$  条件下, 第  $q$  个节点选取属性  $A_{i_q}$  的

互信息  $I(U_{q-1}, A_{i_q})$  为

$$I(B, A_{i_q} | U_{q-1}) = H(B, U_{q-1}) - H(B, A_{i_q} | U_{q-1}). \quad (5)$$

求属性  $A_{i_q}$  使  $\text{MAX}(I(B, A_{i_q} | U_{q-1}))$ , 那么  $A_{i_q}$  为决策树在该分支下第  $q$  个节点选择的属性.

算法步骤如下:

函数: buildFDT( $U, A \cup B, U_{q-1}, A_{\text{Attributes}}$ )

输入:  $U$  为数据集,  $A$  为条件属性,  $B$  为分类属性,  $U_{q-1}$  为该扩展节点的父类属性取值序列, 初始时  $U_{q-1} = \emptyset$ ,  $A_{\text{Attributes}}$  为候选属性集合.

输出: 分类类别  $B_i$ .

(i) 对于决策树中的每一个扩展节点, 在该分支父类已选条件序列  $U_{q-1} = \{A_{i_{j_1}}, A_{i_{j_2}}, \dots, A_{i_{j_{q-1}}}\}$  下, 对于候选属性  $A_{\text{num}}, A_{\text{num}} \in A_{\text{Attributes}}$  的条件互信息  $I(B, U_{q-1} | A_{\text{num}})$ ;

(ii) 选择使互信息最大的属性  $A_{\text{num}}$ ;

(iii) 把  $A_{\text{num}}$  作为构建子树的节点, 并从候选属性集合中删去  $A_{\text{num}}, A_{\text{Attributes}} = A_{\text{Attributes}} - A_{\text{num}}$ ;

(iv)  $q++$ ,  $U_{q-1} = U_{q-1} \cup A_{\text{num}}$  得到新的条件属性序列  $U_{q-1}$ ;

(v) 对于  $A_{\text{num}}$  每个属性值, 检查是否是叶子节点. 若不是, 递归调用 buildFDT( $U, A \cup B, U_{q-1}, A_{\text{Attributes}}$ ).

## 2.3 实例分析

利用表 3 中 Iris 模糊化后的示例数据, 将此数据集作为训练集, 利用本文算法构造模糊决策树.

算法步骤如下:

(i) 对于初始根节点  $U_{q-1} = \emptyset$  根据(3)式可以得到  $H(\text{Class})$

$$\begin{aligned} H(\text{Class}) &= - \sum_{j=1}^3 P(\text{Class}_j) \log(P(\text{Class}_j)) = \\ &= - \sum_{j=1}^3 \frac{M(\text{Class}_j)}{\sum_{i=1}^3 M(\text{Class}_i)} \log \left( \frac{M(\text{Class}_j)}{\sum_{i=1}^3 M(\text{Class}_i)} \right) = \\ &= - \frac{1}{3} \log\left(\frac{1}{3}\right) - \frac{1}{3} \log\left(\frac{1}{3}\right) - \frac{1}{3} \log\left(\frac{1}{3}\right) = 0.4771. \end{aligned}$$

(ii) 从候选属性中选取一个候选属性, 由(4)式, 计算  $H(B | A_i)$ , 那么对于属性  $\text{SpellLength}$  有  $H(\text{Class}, \text{SpellLength} | U_{q-1}) =$

$$\begin{aligned} & \sum_{i=1}^3 P(\text{SpellLength}_i) H(\text{Class}, \text{SpellLength}_i) = \\ & \sum_{i=1}^3 P(\text{SpellLength}_i) \sum_{j=1}^3 P(\text{Class}_j | \text{SpellLength}_i) \end{aligned}$$

$\log(P(\text{Class}_j, \text{SpellLength}_i)) = 0.3261$ , 其中  $P(\text{SpellLength}_i) = M(\text{SpellLength}_i) / \sum M(\text{SpellLength}_i) P(\text{Class}_j, \text{SpellLength}_i) = M(\text{Class}_j \times \text{SpellLength}_i) / M(\text{SpellLength}_i)$ . 同理  $H(\text{Class} | \text{SpellWidth} | U_{q-1}) = 0.2650$   $H(\text{Class} | \text{PetalLength} | U_{q-1}) = 0.1411$   $H(\text{Class} | \text{PetalWidth} | U_{q-1}) = 0.0958$ .

(iii) 根据(5)式计算互信息  $I(B | A_i | U_{q-1})$ , 有  $\text{Max}(I(B | A_i | U_{q-1}))$  选取候选属性  $\text{PetalWidth}$  作为根节点属性.

(iv) 以  $\text{PetalWidth}$  作为根节点, 则  $\text{PetalWidth}$  分别取值  $\text{PetalWidth1}$ ,  $\text{PetalWidth2}$  和  $\text{PetalWidth3}$  建立分支. 由于以  $\text{PetalWidth1}$ ,  $\text{PetalWidth2}$  和  $\text{PetalWidth3}$  为模糊划分分割下的频度  $f$  小于设定的阈值, 所以终止分裂, 下一节点为叶子节点, 如图1所示.

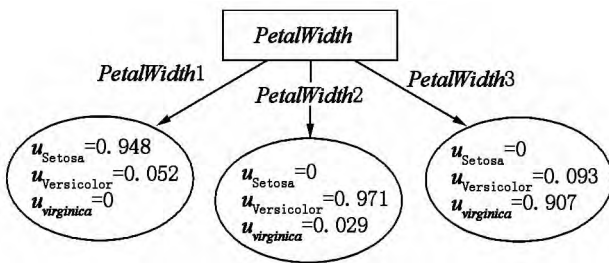


图1 Iris 示例数据决策树

### 3 增量学习

#### 3.1 基于 $Q$ 统计量的相似度计算

相似度计算是评估分类器之间差异的一种重要方法,  $Q$  统计量是度量两个分类器差异度的一种有效方法<sup>[16]</sup>.

$D = \{D_1, D_2, \dots, D_K\}$  表示  $K$  个不同的分类器的

集合  $\mathcal{C} = \{C_1, C_2, \dots, C_m\}$  表示分类类别集合.  $x \in \mathbf{R}^n$  是  $n$  维输入特征向量  $\rho = \{e_1, e_2, \dots, e_N\}$  表示输入的  $N$  个样本集合  $\rho \subseteq \mathbf{R}^n \times \mathcal{C}$ . 任意分类器  $D_i$  在样本集合  $e$  上的输出  $y_i$  用一个  $N$  维二值向量表示, 则  $y_i = \{y_{1i}, y_{2i}, \dots, y_{Ni}\}^T$ . 如果  $D_i$  能正确分类样本  $e_j$ , 则  $y_{ji} = 1$ , 否则  $y_{ji} = 0$ .

表4 一对分类器的关系表

	$D_k \text{ correct}(1)$	$D_k \text{ wrong}(0)$
$D_i \text{ correct}(1)$	$N^{11}$	$N^{10}$
$D_i \text{ wrong}(0)$	$N^{01}$	$N^{00}$

任意2个分类器  $D_i$  和  $D_k$  的  $Q$  统计量  $Q_{ik}$  定义如公式<sup>[16]</sup>

$$Q_{ik} = (N^{11}N^{00} - N^{01}N^{10}) / (N^{11}N^{00} + N^{01}N^{10}), \quad (6)$$

其中  $N^{ab}$  表示分类数据集  $E$  中的  $N$  个样本在分类器  $D_i$  和  $D_k$  值对, 同时满足  $y_{ji} = a$  且  $y_{jk} = b$  的样本总数目, 其中  $a, b \in \{0, 1\}$ ,  $j = 1, 2, \dots, N$ . 如果2个分类器相互独立, 那么它们的  $Q$  统计量的值为0.  $Q$  统计量值越大, 表示2个分类器越相似.

#### 3.2 增量学习算法

本文提出了一个具备增量学习功能的分类模型(见图2).

通过  $Q$  统计量来计算2个规则集之间的差异度, 当相似度大于阈值  $\lambda$  时, 定义为2个规则集相似. 本文采用加权投票方法<sup>[16-17]</sup>集成图中获得的2个规则集, 产生一个新的规则集; 当相似度小于阈值  $\lambda$  时, 定义为2个规则集不相似, 这时把数据集  $D$  和数据集  $I$  合并生成新数据集, 然后通过训练构造分类模型, 产生一个新的分类规则集.

集成式增量学习算法描述如下:

(i) 原始数据集  $S = S_{\text{训练集}} + S_{\text{测试集}}$  将本文算法在  $S_{\text{训练集}}$  上训练, 产生一个规则库  $R_D$ .

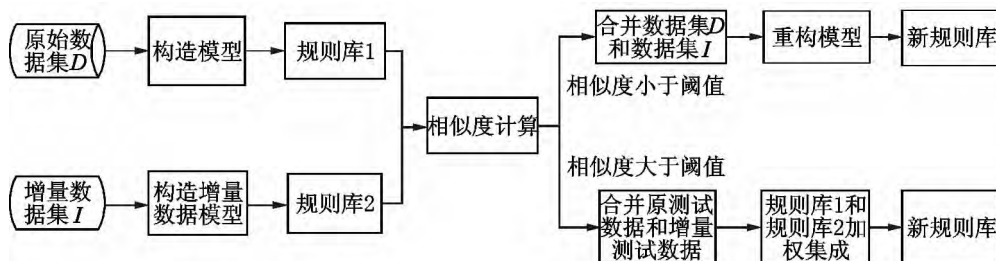


图2 增量学习的分类模型

(ii) 增量数据集  $I = I_{\text{训练集}} + I_{\text{测试集}}$  执行以下过程: (a) 将本文算法用于  $I_{\text{训练集}}$  ,产生规则库  $R_I$  , (b) 通过  $Q$  统计量计算分类规则库  $R_D$  和  $R_I$  相似度  $r$  , (c) 若  $(r > \lambda)$  则合并  $S_{\text{测试集}}$  和  $I_{\text{测试集}}$  ,生成新测试集  $D_{\text{测试集}}$  ,用  $W_i$  和  $W_j$  加权投票集成  $R_D$  和  $R_I$  ,产生新规则库  $R = w_i R_D + w_j R_I$  ;若  $r \leq \lambda$  ,则  $D = D_{\text{训练集}} + D_{\text{测试集}}$  ,  $D_{\text{测试集}} = S_{\text{测试集}} + I_{\text{测试集}}$  ,将本文算法用于  $D_{\text{训练集}}$  ;生成新规则库  $R$  .

(iii) 输出最终的分类规则库  $R$  .

4 实验及分析

本文实验数据来源于 UCI 数据库中的 Iris、Pima、Wine、Ionosphere、Glass、和 Letter 共 6 个数据集<sup>[12,14-15]</sup> ,其数据集情况如表 5 所示. 本文分别以传统贝叶斯分类算法、ID3 决策树分类算法、模糊 ID3 决策树分类算法和改进的带增量学习模糊决策树分类算法,在 Iris、Pima、Wine、Ionosphere 和 Glass 5 个数据集上验证这些算法的准确性,以 Letter 数据集进行增量学习验证,获得增量学习改进下的算法准确度.

表 5 各个数据集说明

数据集	样本数量	属性数	类别数
Iris	150	4	3
Pima	768	8	2
Wine	178	13	3
Ionosphere	351	34	2
Glass	214	10	7
Letter	20 000	16	26

在 Iris、Pima、Wine、Ionosphere 和 Glass 这 5 个数据集上,用贝叶斯分类算法、传统 ID3 决策树算法、模糊 ID3 决策树分类算法和基于条件信息熵和互信息的决策树分类算法的对比试验结果见表 6. 其准确度对比情况参见图 3.

表 6 各分类算法在 UCI5 个数据集上的分类情况

数据集	平均精度			
	贝叶斯	ID3	模糊 ID3	本文方法
Iris	0.955 1	0.921 3	0.953 1	0.967 8
Pima	0.751 9	0.692 4	0.763 6	0.740 5
Wine	0.621 4	0.579 1	0.601 7	0.653 2
Ionosphere	0.605 9	0.648 2	0.650 6	0.723 6
Glass	0.463 5	0.530 1	0.646 0	0.658 4

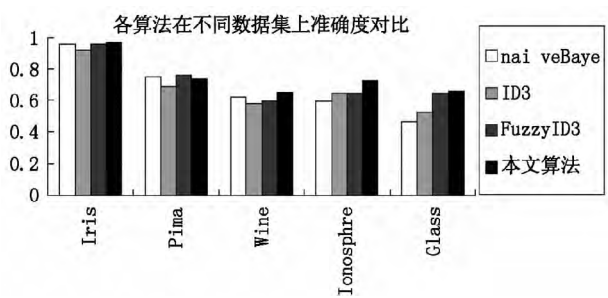


图 3 各类算法在各个数据集上的对比折线图

由表 6 和图 3 可知,在 Iris、Pima、Wine、Ionosphere 和 Glass 这 5 个数据集上,模糊决策树算法的准确度明显优于传统决策分类算法,与模糊 ID3 算法相比,本文算法的准确度在总体上有所提升.

由于 Letter 数据集的样本比较大,将 Letter 数据集平均分割为 5 部分,则每个数据集包含 4 000 条样本数据. 用本文算法对分割后的 5 个数据集进行增量学习以改进分类模型,得到图 4 的结果.

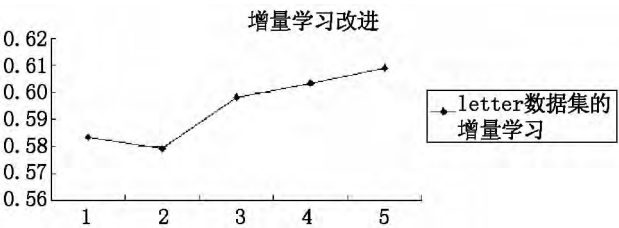


图 4 经 Letter 数据集增量学习后的对比图

由图 4 可知,随着增量学习次数增加,准确度随之上升. 另外,利用加权投票,可以弱化分类器中分类错误率,从而通过增量学习修正分类模型,以提高分类准确度.

5 结论

本文对数据挖掘方法进行了探讨,先将数据模糊化处理,然后通过后验熵和互信息来构建模糊决策树,并用增量学习方法改进模糊决策模型. 最后用实验验证了本文方法的有效性.

6 参考文献

[1] Jiawei Han ,Micheline Kamber ,Jian Pei. 数据挖掘概念与技术 [M]. 北京: 机械工业出版社 2012.

[2] 赵静娴 ,倪春鹏等. 一种高效的连续属性离散化算法 [J]. 系统工程与电子技术 2009 31( 1) :195-200.

[3] 王熙熙 ,谢凯. 基于聚类的数据预处理对模糊决策树产生的影响 [J]. 计算机工程与应用 2006 20( 1) :156-

- 160.
- [4] 梁路, 滕少华. 多属性模糊度量在可用性评估中的研究 [J]. 江西师范大学学报: 自然科学版, 2010, 34(4): 358-361.
- [5] 蒲元芳, 张巍, 滕少华, 等. 基于决策树的协同网络入侵检测 [J]. 江西师范大学学报: 自然科学版, 2010, 34(3): 302-307.
- [6] 谢竞博, 王熙照. 基于属性间交互信息的 ID3 算法 [J]. 计算机工程与应用, 2004, 30(3): 93-95.
- [7] 唐亮, 段建国, 徐洪波等. 基于互信息最大化的特征选择算法及应用 [J]. 计算机工程与应用, 2008, 44(13): 130-133.
- [8] 谢振平, 王士同, 王小明. 一种基于软边界球分的分裂式层次聚类算法 [J]. 模式识别与人工智能, 2008, 21(4): 559-570.
- [9] 汪中, 刘贵全, 陈恩红. 一种优化初始中心点的 K-Means 算法 [J]. 模式识别与人工智能, 2009, 22(2): 299-305.
- [10] 刘波, 梁活民. 基于增量决策树的快速 IDS 研究与实现 [J]. 计算机工程与应用, 2008, 44(7): 141-145.
- [11] Hu Rong, Xu Weihong, Kuang Fangjun. An improved incremental singular value decomposition [J]. International Journal of Advancements in Computing Technology, 2012, 4(2): 95-102.
- [12] 林明龙. 基于神经网络集成的增量式学习 [D]. 合肥: 中国科学技术大学计算机学院, 2012. 6.
- [13] Vitaly Levashenko, Elena Zaitseva, Seppo Puuronen. Fuzzy classifier based on fuzzy decision tree [C]. The International Conference on Computer as a Tool, Warsaw, September 2007: 823-827.
- [14] 丁世飞, 朱红, 徐新政等. 基于熵的模糊信息测度研究 [J]. 计算机学报, 2012, 35(4): 796-812.
- [15] 张燕平, 杜玲, 赵姝. 构造性覆盖方法的增量学习算法 [J]. 南京大学学报, 2009, 45(5): 699-705.
- [16] 张燕平, 曹振田, 赵姝等. 一种新的决策树选择性集成学习方法 [J]. 计算机工程与应用, 2010, 46(17): 41-45.
- [17] 张春霞, 张讲社. 选择性集成学习算法综述 [J]. 计算机学报, 2011, 34(8): 1399-1411.

## Fuzzy Decision Tree Based on Mutual Information and Incremental Learning

ZHANG Wei, NIE Jin, TENG Shao-hua

(School of Computer Science and Technology, Guangdong University of Technology, Guangzhou Guangdong 510006, China)

**Abstract:** The fuzzy decision tree used to determine a candidate tree classification attributes based on mutual information has been introduced, and then construct fuzzy decision trees. It corrects classification model to achieve better classification results through incremental learning. Finally, experimentally validated.

**Key words:** fuzzy decision tree; conditional information entropy; mutual information; ensemble learning; incremental learning

(责任编辑: 冉小晓)