基于随机化属性选择和邻域覆盖约简的集成学习

朱鹏飞,胡清华,于达仁
(哈尔滨工业大学能源学院,黑龙江哈尔滨 150001)

摘 要: 提高分类模型的分类精度和可靠性是分类建模追求的目标.针对目前规则学习方法应用于分类时稳定性差以及分类精度低的问题,本文通过随机化邻域属性约简,搜索一组分类精度较高的属性子集,在不同的属性子集上采用邻域覆盖约简方法学习分类规则,得到多个规则集.最后通过简单投票融合不同规则集上的分类结果获得对象的类别.实验表明,基于随机化邻域约简的集成学习方法分类性能优于或与其它相关的分类器相当,并且在噪声扰动下具有更强的鲁棒性.

关键词: 邻域;随机约简;集成学习;规则学习;分类器
中图分类号: TP391.4 **文献标识码:** A **文章编号:** 0372-2112 (2012) 02-0273-07
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2012.02.011

Ensemble Learning Based on Randomized Attribute Selection and Neighborhood Covering Reduction

ZHU Peng-fei, HU Qing-hua, YU Da-ren
(*Department of Energy Science and Technology, Harbin Institute of Technology, Harbin, Heilongjiang* 150001, *China*)

Abstract: Improving accuracy, robustness and understandability is the objective of classification modeling. Regarding instability and performance limitation of existing rule learning techniques, we introduce an ensemble classifier based on randomized neighborhood reduction and neighborhood covering reduction. A set of reducts are obtained with randomized attribute reduction. A collection of rule sets are derived from the reducts based on neighborhood covering reduction. And then the classification result is output by combining the classification decision of different rule sets. The experiment result shows that the proposed technique is better than or equal to other classifiers, and is more stable when deals with noisy data.

Key words: neighborhood; randomized reduction; ensemble learning; rule extraction; classifier

1 引言

规则学习建立的分类模型具有很好的可理解性,基于规则的分类系统已在生物工程[1]、股票分析[2]、故障诊断[3]等领域得到广泛应用.目前的规则学习方法在提取规则时,由于样本少难以学习到一组精确的分类规则,而且采用贪心搜索时陷入局部极值[2],易受噪声影响导致分类性能不稳定,分类精度较低.

集成学习把若干子分类器集成起来,通过一定方式融合多个子分类器的输出得到最终的分类[5~7],从而取得比单个分类器更好的性能.在规则学习中融合多个规则集的输出能降低分类错误率、提高分类的稳定性[8].

对于集成学习,如何获得有差异的子分类器是影响集成学习效果的关键[8].Bagging[9]和 Boosting[10]等方法通过对训练样本进行处理获得不同的基分类器.对于具有多输入特征的实例,可以通过抽取不同的输入特征子集分别进行训练,获得不同的分类器[7].Hu 在文献[12]中应用集成粗糙子空间的方法,通过粗糙集生成多个特征子空间,在不同子空间中学习分类器,取得了良好的效果.

针对规则学习在分类中存在的鲁棒性差以及分类精度不高的问题,本文首先通过随机化邻域约简,得到一系列分类性能较强的属性子集,并基于邻域覆盖约简算法在不同的属性子集上进行规则学习,得到多个规则集.对于一个新的样本,通过融合不同规则集上的分类结果得到最终的输出.实验结果表明,基于随机化邻域约简的集成规则集方法分类性能优于或与其它相关的分类器相当,并且在噪声扰动下具有更强的稳定性.

收稿日期:2011-01-01;修回日期:2011-06-26
基金项目:国家自然科学基金(No.60703013,No.10978011)

2 基本概念

2.1 邻域粗糙集

由于 Pawlak 提出的粗糙集理论不能处理数值型数据,Hu 构建了基于邻域粒化的粗糙集模型[13].

给定决策表 $<U,A,D>$,$U=\{x_1,\cdots,x_n\}$ 是全部样本构成的集合,$A=\{a_1,\cdots,a_N\}$ 是描述样本的属性集合,D 是分类决策属性.

定义 1 $<U,\Delta>$ 是非空度量空间,$x\in U$,$\delta\geqslant 0$,称点集

$$\delta(x)=\{y\mid\Delta(x,y)\leqslant\delta,y\in U\}$$

为 x 的 δ 邻域,其中 Δ 为距离函数.

若 Δ 为欧氏距离,则样本 x 的 δ 邻域为以 x 为中心,δ 为半径的超球体.

定义 2 给定 $<U,A,D>$,如果 A 生成一组邻域关系,则称 $<U,A,D>$ 为邻域决策系统.

定义 3 给定 $<U,A,D>$,D 将 U 划分为 N 个等价类:$X_1,X_2,\cdots,X_N$,$B\subseteq A$ 生成 U 上的邻域关系 N_B,那么决策 D 关于 B 的邻域下近似和上近似分别为:

$$\underline{N_B}D=\{\underline{N_B}X_1,\underline{N_B}X_2,\cdots,\underline{N_B}X_N\},$$

$$\overline{N_B}D=\{\overline{N_B}X_1,\overline{N_B}X_2,\cdots,\overline{N_B}X_N\},$$

其中,$\underline{N_B}X_i=\{x\mid\delta(x)\subseteq X_i\}$,$\overline{N_B}X_1=\{x\mid\delta(x)\cap X_i\neq\phi\}$.

定义 4 给定 $<U,A,D>$,决策属性 D 对条件属性 $B\subseteq A$ 的依赖度为

$$\gamma_B(D)=\frac{Card(\underline{N_B}D)}{Card(U)}.$$

给定邻域决策系统 $<U,A,D>$,$B\subseteq A$,$a\in B$,如果

(1) $\gamma_B(D)=\gamma_A(D)$

(2) $\forall\ a\in B:\gamma_{(B-a)}(D)<\gamma_B(D)$

则称 B 是一个属性约简.

本质上约简是一组保持原始数据近似能力的特征子集,一般存在多个可以保持原始数据近似能力的约简.在不同的约简中分类信息不同,从而可形成互补,通过集成多个约简的信息可提高系统的泛化能力[12].

2.2 基于覆盖约简的规则学习

当前关于覆盖约简的研究主要分为两类.第一类的目的是约简覆盖中的冗余元素[18~20],称为覆盖元约简,可用于规则学习.另一种是约简多个覆盖族中不影响决策上下近似计算的冗余或无关的覆盖[21,22],称为覆盖集约简,用于属性约简.此处中主要讨论第一类覆盖约简[23].

定义 5 给定论域 $U=\{x_1,\cdots,x_n\}$,$C=\{F_1,F_2,\cdots,F_k\}$ 为 U 的一族非空子集,并且 $\bigcup_{i=1}^{k}F_i=U$,称 C 为 U 的一个覆盖,F_i 为覆盖元.

如果计算每个样本的邻域,那么邻域族就形成了论域的一个覆盖,每个样本的邻域是一个覆盖元,称为邻域覆盖元.在文献[13]中,每个样本的邻域大小是固定的.本文中不同样本的邻域大小将根据其在特征空间中的位置进行计算,δ 的大小设置为样本的分类间隔[4].

定义 6 给定一个样本集 $<U,A,D>$,$x\in U$. $NH(x)$ 是样本 x 的最近同类样本,$NM(x)$ 是样本 x 的最近异类样本,则样本 x 的分类间隔定义为:

$$M(x)=\Delta(x,NM(x))-\Delta(x,NH(x)).$$

如果 $M(x)$ 小于零,此时样本按照最近邻规则将被错分.实验中,设置此类样本的 $M(x)=0$.按照以上规则设置邻域大小,如果不存在属性值相同而类别不同的样本,那么每个样本的邻域将一致的属于同一决策类.

邻域族 $C=\{\delta(x_1),\delta(x_2),\cdots,\delta(x_n)\}$ 形成了论域的逐点覆盖,称 $<U,C>$ 为一个邻域覆盖空间,$<U,C,D>$ 为一个邻域覆盖决策系统.

定义 7[23] $<U,C,D>$ 是一个邻域覆盖决策系统,X_i 是某一决策类.如果 $\exists\ \delta(x)\in C$,使得 $\delta(x')\subseteq\delta(x)\subseteq X_i$,则称 $\delta(x')$ 相对于 X_i 是相对一致可约的;否则,称 $\delta(x')$ 是相对一致不可约的.

定义 8[23] 给定 $<U,C,D>$,如果对于任意的决策类 X_i,都不存在 $\delta(x')\in C$,使得 $\delta(x')\subseteq\delta(x)\subseteq X_i$,则称 $<U,C,D>$ 是相对可约的;否则,称 $<U,C,D>$ 是相对不可约的.

定义 9[23] $<U,C,D>$ 是一个邻域覆盖决策系统,$C'\subseteq C$ 是从 C 中去除冗余覆盖元所得到的一个覆盖 $<U,C',D>$ 是相对不可约的,称 C' 是 C 的一个 D 相对约简,表示为 $reduct_D(C)$.

性质 1 $<U,C,D>$ 是一个邻域覆盖决策系统,$reduct_D(C)$ 是 C 的 D 相对约简,那么 $<U,reduct_D(C),D>$ 也是一个领域覆盖决策系统,$\forall\ \delta(x)\in C$,$\exists\ \delta(x')\in reduct_D(C)$,使得 $\delta(x)\subseteq\delta(x')$.

在覆盖元约简后,覆盖决策系统中没有冗余的覆盖元.所有被选中的覆盖元在近似决策类的时候都是有用的.给定一个覆盖决策系统的约简,产生覆盖规则的形式如下:

如果 $x'\in\delta(x)$,则样本 x' 和邻域覆盖元 $\delta(x)$ 的决策一致

邻域覆盖约简的理论框架为从训练样本中学习规则提供了一种理论机制,通过邻域覆盖约简可以得到一组规则集.

3 规则集成

3.1 基本思想

本文提出的规则集成的结构如图 1 所示.给定一个决策表 $<U,A,D>$,条件属性集为 $\{a_1,a_2,\cdots,a_n\}$,根据随机化邻域约简算法可以得到一组属性约简集合

$\{AR_1, AR_2 \cdots AR_m\}$,在每个约简上利用覆盖约简算法可以得到一组规则集$\{R_1, R_2 \cdots R_m\}$.给定一个新的样本,在各规则集上给出不同的类别,通过加权投票的方式,最终可以得到这个样本的类标号.

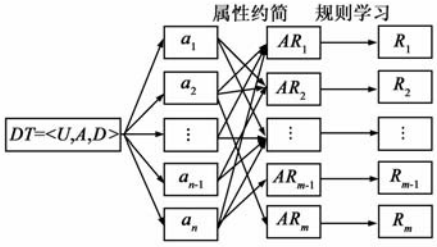

图1　基于邻域随机约简的规则学习过程

3.2　邻域随机约简

文献[13,15]提出了基于邻域粗糙集模型的贪心属性约简算法.从空集开始,每一步增加一个使得区分能力增长最大的特征,直到增加任何属性,区分能力都不再增长为止.通过这一步计算,将生成一个嵌套的特征子集序列:$B_1 \subset B_2 \subset \cdots \subset B_k$.这一步采用属性重要度来计算特征的区分能力.属性的重要度计算为

$$SIG(a, B, D) = \gamma_{B \cup a}(D) - \gamma_B(D)$$

由于每次都是选择区分能力最大的特征,这样就只能得到一个约简,称为在这种搜索方法下的贪心约简.

实际数据中存在多个可以保持原始数据近似能力的属性子集.为了尽可能利用这些不同子空间的分类能力,需要提出一种寻找多个约简的方法.Wu 等在文献[16]中提出了 WADF 方法来求得多个约简,通过随机删除非核属性中的一个属性,在删除后的属性集上寻找新的约简,即可得到多个属性约简.

给定 $< U, A, D >$, $B_1 \subset B_2 \subset \cdots \subset B_m \subseteq A$ 为一嵌套的条件属性子集序列,则条件属性子集相对于决策 D 的正域满足 $\gamma_{B_1}(D) \leqslant \gamma_{B_2}(D) \leqslant \cdots \leqslant \gamma_{B_m}(D) = \gamma_B(D)$.对于前向贪心搜索生成的嵌套序列,满足 $\gamma_{B_1}(D) < \gamma_{B_2}(D) < \cdots < \gamma_{B_m}(D) = \gamma_B(D)$.对于嵌套序列的生成,主要取决于每次新增加属性的选取,选取的方式不同,则得到的约简也可能不同,这就为寻找多个约简提供了一种方式.

本文为了得到多个约简,放宽了贪心算法每一步都选最佳属性的要求,而采用随机选择区分能力最大的前 F 个特征中的一个作为选中属性.这样通过多次运行该算法即可得到多个具有区分能力的属性子集.

选定随机数 F 后,每运行一次程序即可得到一个随机化的约简.这个算法的计算代价为$(2n - k)(k + 1) \times (k + 1) \times m\log m/2$,其中 n 和 m 为样本和特征的数目,k 为约简中属性的个数.

算法 1　基于邻域粗糙集的随机属性约简

输入:$< U, A, D >$,参数 delta 和随机数 N
输出:约简 red.
(1)　$\varnothing \rightarrow red$;
(2)　For $a_i \in A - red$
(3)　计算正域 $\gamma_{red \cup a_i}(D) = \dfrac{|POS_{red \cup a_i}(D)|}{|U|}$
(4)　计算属性 a_i 重要度

$$SIG(a_i, red, D) = \gamma_{red \cup a_i}(D) - \gamma_{red}(D)$$

(5)　end
(6)　选择 a_k, a_k 为属性集$\{A - red\}$中属性重要度 $SIG(a_k, red, D)$ 前 F 个最大中的一个
(7)　If $SIG(a_k, red, D) > 0$,
(8)　　$red \cup a_k \rightarrow red$
(9)　　回到第二步
(10)　else
(11)　　返回约简 red
(12)　end if

3.3　规则学习及规则集集成

得到约简后,需要基于约简学习规则集.本文采用邻域覆盖约简算法[23].

基于邻域覆盖约简的规则学习算法从一个空集开始,采用前向搜索策略,每次添加一条规则.每一步中,覆盖样本数目最多的邻域覆盖元会被选中并产生一条规则,而被覆盖的样本产生的邻域覆盖元会被删除.依次迭代,直到覆盖为空.

在覆盖约简后,被选中的邻域覆盖元产生的规则集中有一些规则仅仅覆盖了少数的几个样本,需要对规则集进行了进一步的剪枝.首先,移除只包含一个样本的邻域覆盖元.其次,对规则按照各自覆盖样本数的大小进行排序,使得在训练集或者测试集上取得最高分类精度的前 L 条规则留下,其余的规则被删除.

计算邻域的距离公式不同,则邻域的形状也不同,最后得到的规则形式也不同.在本文中我们分别采用欧氏距离和无穷范数计算样本的邻域.

由于采用贪心搜索,这种规则学习方法非常高效.NCR 的时间复杂度是 $n\log n$.不足的是由于采用贪心的算法寻找规则集,往往得到的是次优解.

从每个约简都可抽取一个规则集,从而可以得到一系列的规则集,对于一个新来的样本在不同的规则集上会有相应的分类输出,经过简单投票的方法确定样本的最终类标号.

基于邻域随机约简的规则集成相对于其他分类器以及集成学习的方法主要有以下优势.首先,在规则学习前进行了属性约简,去除冗余和不相关属性的同时又保持了原始空间的区分能力,这样学习到的规则将具有更好的泛化能力.其次,基于随机约简的规则学习,能在不同的子空间中得到不同的规则集.通过不同规则集的集成可以使得整个规则学习过程更为稳定.

对于约简数目很少的数据集,这种方法无法体现出其优势.此时可通过其他子空间生成的方法[17]得到多个特征子空间,再和邻域覆盖约简相结合设计集成学习方法.

算法 2 基于邻域覆盖约简的规则学习

输入:训练集 $U\_train=\{(x_i,d_i)\},i=1,2,\cdots,n$;
测试集 $U\_test=\{(x_i',d_i')\},i=1,2,\cdots,n$.
输出:规则集 $R=\{r_1,\cdots,r_i,\cdots,r_h\}$,其中规则形式为$(x_i,m(x_i),d_i)$.
(1) 计算样本 x_i 的分类间隔 $m(x_i)\ i=1,2,\cdots,n$.如果 $m(x_i)<0$,设置 $m(x_i)=0$.
(2) 计算样本 x_i 的邻域覆盖元 $\delta(x_i)\ i=1,2,\cdots,n$,论域的覆盖表示为 C,移除只包含一个样本的邻域覆盖元.
(3) $R\leftarrow\varnothing$
(4) While($C\neq\varnothing$)
(5) 选择覆盖样本数最多的覆盖元 $\delta(x)$
(6) 添加一条规则$(x,m(x),y)$到规则集 R,其中 $m(x)$是样本 x 的分类间隔,y 是 x 的类标号.
(7) 移除被邻域覆盖元 $\delta(x)$覆盖的样本.
(8) end
(9) 按照规则覆盖样本数目由大到小对规则进行排序
(10) 计算使得在训练集或者测试集上分类精度最高的规则集 R

4 实验分析

实验分析包含四个部分,首先通过人造数据展示了基于邻域覆盖约简的规则学习的过程.同时在 4 个数据上,对比了随机约简上的分类精度以及贪心约简[24]上的分类精度.然后对比了基于随机约简的规则集、基于贪心约简的规则集及其他相关分类器的分类性能.最后,为检验提出方法的稳定性,在噪声数据上检验了基于随机约简的集成规则集分类器的抗噪能力.

首先构造了一个两类分类问题,共 40 个样本,各类都满足高斯分布,如图 2(a)所示.计算每个样本的分类间隔,可以得到各个样本的邻域大小,从而形成 40 个邻域覆盖元,如图 2(b)所示.经过覆盖约简后,最终得到了 2 条规则,每类样本仅生成一条规则,如图 2(c)所示.

从 UCI 数据库[11]中下载了 4 个数据集,分别是 wine、wdbc、iono 以及 wpbc,数据描述如表 1 所示.为了验证集成学习的必要性,我们在四个数据集上对比了基于随机约简学习到的规则集以及基于贪心约简学习到的规则集的分类性能.在实验中我们计算了 100 个随机约简产生规则集的分类性能,并按照约简中属性的个数画出了规则集分类精度的箱线图,如图 3 所示.图的最左侧用实线标出了基于贪心约简的分类精度.可以看出,贪心约简产生的规则的分类性能不一定总优于随机约简,因此这就说明只利用在贪心约简上学习到的规则集进行分类并不一定合理,分类性能还有提高的空间.

表 1 数据描述

Data	Features	Class	Instances
Wine	13	3	178
Wdbc	30	2	569
Iono	34	2	351
wpbc	33	2	198

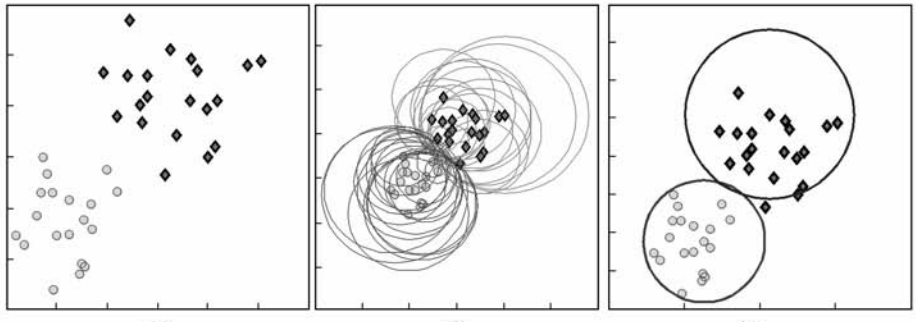

图2 人造数据上规则学习过程演示

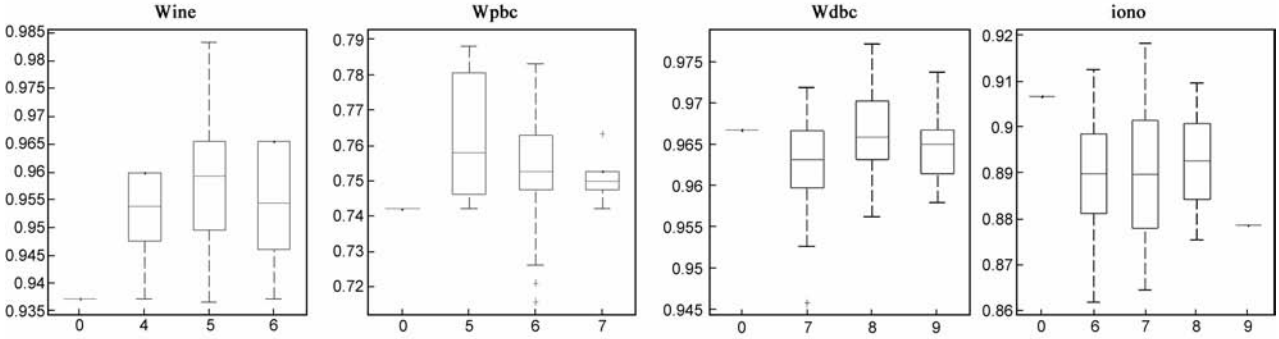

图3 按照约简中属性数目进行统计的分类精度的箱线图

表 2　常见分类器和规则学习方法实验效果

Data	1-NN	EROS[12]	NEC	LVQ	LSVM	CART		C4.5		Jripper	
	Accuracy	Accuracy	Accuracy	Accuracy	Accuracy	Accuracy	No.	Accuracy	No.	Accuracy	No.
Wine	94.9±5.0	98.3±2.7	96.6±2.9	96.0±2.8	98.9±2.3	88.2±6.2	10	91.6±7.3	5	93.8±8.1	3
wdbc	95.4±3.3	98.2±1.4	94.6±2.5	95.4±3.6	97.7±2.5	93.1±2.4	9	93.3±3.8	13	94.3±2.0	5
Iono	86.4±4.9	95.7±4.6	83.5±4.7	86.1±4.4	87.6±6.5	89.7±3.4	3	91.4±2.7	18	88.9±3.0	5
wpbc	70.7±6.7	81.7±6.5	78.3±7.3	75.3±6.8	77.4±7.7	76.3±6.2	1	73.2±6.8	16	70.7±6.5	1
average	**86.9**	**93.5**	**88.3**	**88.2**	**90.4**	**86.8**	**5.8**	**87.4**	**13.0**	**86.9**	**3.5**

表 3　基于最优约简规则集分类效果

Data	ERC_NR(R)Test		ERC_NR(R)Training		ERC_NR(S)Test		ERC_NR(S)Training	
	Accuracy	No.	Accuracy	No.	Accuracy	No.	Accuracy	No.
wine	98.3±2.8	5.1	93.7±7.0	5.9	98.3±2.8	7.4	94.3±4.0	11.5
wdbc	98.1±1.5	20.8	96.7±2.1	32.3	95.1±2.7	12	93.5±3.0	22.2
iono	91.8±4.4	19.1	90.7±4.9	27	91.5±5.0	12.8	90.6±5.0	15.1
wpbc	79.3±4.3	4.2	74.2±6.4	2	81.8±4.7	10.8	75.7±7.9	15.1
average	**91.9**	**12.3**	**88.8**	**16.8**	**91.7**	**10.8**	**88.8**	**16.0**

表 4　基于随机约简的集成规则集分类效果

Data	ERC_NRR(R)Test		ERC_NRR(R)Training		ERC_NRR(S)Test		ERC_NRR(S)Training	
	Accuracy	No.	Accuracy	No.	Accuracy	No.	Accuracy	No.
wine	100.0±0.0	6.3	96.1±3.8	9.2	99.4±1.8	7.7	97.0±0.6	10.1
wdbc	98.3±1.4	21.5	97.2±2.3	28	97.7±1.4	18.7	95.6±2.5	21.3
iono	94.1±4.1	18.0	90.7±4.9	27	92.9±5.0	11.3	90.6±5.0	15.1
wpbc	83.3±2.6	4.5	78.7±5.4	7.6	84.4±4.9	10.8	77.2±5.7	11.8
average	**93.9**	**12.9**	**90.7**	**16.9**	**93.6**	**12.1**	**90.1**	**14.6**

表 5　噪声数据上常见分类器和规则学习方法实验效果

Noisy data	1-NN	EROS[12]	NEC	LSVM	CART		C4.5		Jripper	
	Accuracy	Accuracy	Accuracy	Accuracy	Accuracy	No.	Accuracy	No.	Accuracy	No.
wine(0.1)	93.6±7.4	96.5±4.9	92.6±3.9	95.4±4.6	83.7±7.1	7	86.5±6.9	9	85.9±6.4	5
wdbc(0.1)	90.5±4.7	94.1±2.9	94.0±3.3	95.6±1.5	89.8±4.3	8	87.4±5.1	18	91.7±3.2	6
iono(0.1)	81.6±6.2	92.6±5.6	79.6±7.8	85.9±6.0	89.1±5.4	3	86.6±4.7	14	89.2±5.3	8
wpbc(0.1)	64.1±1.1	81.7±6.6	74.2±10.9	75.3±4.1	77.3±5.2	4	74.7±4.7	14	77.7±6.4	3
average	**82.5**	**91.2**	**85.1**	**88.1**	**85.0**	**8**	**83.8**	**13.8**	**86.1**	**5.5**
wine(0.2)	78.7±7.2	84.9±6.3	78.9±11.2	85.4±6.6	75.3±8.7	7	77.5±7.7	15	74.2±6.3	6
wdbc(0.2)	83.8±3.1	89.6±3.4	88.9±5.4	90.9±4.3	82.9±5.3	8	81.7±4.8	28	81.2±4.5	6
iono(0.2)	75.6±7.1	90.7±5.6	76.2±6.5	76.8±5.8	86.9±6.8	6	81.8±6.7	22	84.3±5.9	5
wpbc(0.2)	55.5±11.4	76.7±8.6	63.2±6.4	84.4±5.0	76.3±5.8	4	73.2±6.5	11	72.2±7.0	2
average	**73.4**	**85.5**	**76.8**	**84.3**	**80.4**	**6.3**	**78.6**	**19**	**78.0**	**4.8**

表 6　噪声数据上基于最优约简规则集的分类效果

Noisy data	ERC_NRR(R)Test		ERC_NRR(R)Training		ERC_NRR(S)Test		ERC_NRR(S)Training	
	Accuracy	No.	Accuracy	No.	Accuracy	No.	Accuracy	No.
wine(0.1)	97.7±3.0	8.1	92.1±7.1	10.2	96.0±3.8	7.2	90.4±6.7	10.3
wdbc(0.1)	95.6±1.9	25.6	92.1±3.2	41.9	94.9±2.9	16.6	90.5±4.9	25.3
iono(0.1)	80.7±8.5	11.8	78.7±7.9	9.7	88.7±7.0	11.8	85.5±6.4	20.8
wpbc(0.1)	78.8±4.5	3.7	72.1±5.0	4.3	76.3±3.0	1	73.2±7.3	3.1
average	**88.2**	**12.3**	**83.8**	**16.3**	**89.0**	**9.2**	**84.9**	**14.9**
wine(0.2)	82.0±7.3	7.3	72.9±11.4	9.2	82.5±7.4	8.8	75.8±10.5	9.6
wdbc(0.2)	87.2±4.6	13.2	83.7±6.4	15	87.4±4.8	17.9	80.9±4.8	23.3
iono(0.2)	80.9±4.8	13.4	78.1±5.5	13.5	87.3±5.6	11.5	82.3±6.6	14
wpbc(0.2)	80.3±3.6	4.3	75.7±5.5	9	80.8±4.6	4.8	75.7±4.1	11.3
average	**82.6**	**9.6**	**77.6**	**11.7**	**84.5**	**10.8**	**78.7**	**14.6**

表 7 噪声数据上基于随机约简的集成规则集的分类效果

Noisy data	ERC_NRR(R)Test		ERC_NRR(R)Training		ERC_NRR(S)Test		ERC_NRR(S)Training	
	Accuracy	No.	Accuracy	No.	Accuracy	No.	Accuracy	No.
wine(0.1)	98.9±2.3	6.3	96.0±3.8	7.6	98.8±2.5	8.7	94.9±5.1	13.4
wdbc(0.1)	97.9±2.0	17.2	95.1±2.3	25.0	97.0±2.2	15.9	94.4±3.2	25.3
wpbc(0.1)	89.5±5.6	15.9	87.0±5.2	17.0	91.2±5.3	17.3	89.2±5.0	18.0
iono(0.1)	81.3±4.7	3.7	76.8±3.9	2.4	82.3±6.4	5.9	76.8±3.2	9.3
average	**91.9**	**10.8**	**88.7**	**13**	**92.3**	**12.0**	**88.8**	**16.5**
wine(0.2)	91.0±4.6	7.9	85.9±7.2	7.6	90.4±5.6	8.1	87.6±7.5	11.4
wdbc(0.2)	94.4±3.1	15.0	90.5±4.1	16.7	92.6±4.5	20.5	87.9±4.3	24.6
iono(0.2)	88.7±4.7	11.3	83.9±7.2	12.4	88.9±4.5	15.7	84.4±4.9	18.0
wpbc(0.2)	80.3±5.5	4.2	76.3±3.0	3	83.3±3.3	5.4	77.3±3.3	11.4
average	**88.6**	**9.6**	**84.2**	**9.9**	**88.8**	**12.4**	**84.3**	**16.4**

此后,在 4 个数据集上分别计算了最近邻分类器、邻域分类器、LVQ 以及 SVM 的分类精度,同时也计算了相关决策树算法包括 CART、C4.5 以及 Jripper 等的分类精度和规则数,如表 2 所示.表 3 给出了贪心约简学习到的规则集的分类效果,表 4 则给出了基于随机约简的集成学习的分类效果,其中 ERC_NR 表示在贪心约简上进行规则学习,ERC_NRR 在随机约简上进行学习,(R)表示规则的形式为超球体,(S)表示学习到的规则形式为超立方体,Test 表示在测试集上对规则进行了剪枝,Training 表示在训练集上对规则进行了剪枝.邻域大小影响邻域属性约简的生成[13],若邻域接近于零,则属性约简中属性个数很少,这可能造成单个规则集的分类精度低于 50%,使得随着规则集数目的增加,集成分类精度下降[7].因此邻域大小的选取影响模型的性能,本文中邻域大小设置为 0.15.对于集成的规则集的个数,即随机约简的数目,本文中设置为 100.

对比表 2 和表 4 可以看出,ERC_NRR 的分类性能相对于决策树算法分类精度最高提高了 7%,平均提高了 5%.与 1-NN、NEC 以及 LVQ 等分类器相比性能也有了大幅度的提高.经对比可以发现 ERC_NRR 的分类性能要优于 SVM 或与其相当.同时我们也与文献[12]中的方法进行了对比,实验中选择 CART 作为基分类器,从实验结果可以看出二者分类性能相当.对比表 2~表 4 发现虽然基于贪心约简得到的规则的分类性能也有了一定的提高,但相对于基于随机约简集成的分类精度还是要低约 2%,这从实验验证了规则集成的优势.

为了考察学习算法的鲁棒性,在数据的属性中注入噪声.首先生成一个服从标准正态分布的 $n \times m$(n 为样本数,m 为属性数)的噪声数据,然后乘以系数 a 后加入到原始数据中.本文 a 值分别设定为 0.1 和 0.2,其中 wine(0.2)表示 a 取值为 0.2 时的被噪声污染数据.表 5~表 7 为噪声数据下相应的实验效果.对比诸如噪声前后的分类效果可以发现,ERC_NRR 在保持良好分类性能的同时也保持了相应的稳定性,分类精度变化最小.当 $a=0.1$ 时,噪声数据下基于贪心约简的规则集的分类性能变化平均下降了 4%,而 ERC_NRR 的平均分类性能则下降了 1.6%.当 $a=0.2$ 时,这种优势更加明显.这表明基于随机约简的集成规则集在稳定性能方面的优越性.

5 结论

目前的规则学习方法稳定性差,分类性能有待提高.本文利用基于邻域粗糙集的随机属性约简方法,得到保持原始空间近似能力的多个特征子集,通过基于邻域覆盖约简的规则学习算法,在不同的属性约简上得到多个规则集.分类时融合不同规则集输出的分类结果,产生最后的分类输出.实验效果表明这种集成学习方法在原始数据和噪声数据上分类性能要优于其他相关分类器.

参考文献

[1] J He, H Hu, B Chen, PC Tai, R Harrison. Rule Extraction from SVM for Protein Structure Prediction[M]. Tioga Publishing, 1983. 83 – 129.

[2] S K Kim, J Il Park. A structural equation modeling approach to generate explanations for induced rules[J]. Expert Systems with Applications, 1996, 10(3/4): 403 – 416.

[3] 刘金福,于达仁,胡清华,王伟.基于加权粗糙集的代价敏感故障诊断方法[J].中国电机工程学报,2007,27(23):93 – 99.
Liu Jin-fu, Yu Da-ren, Hu Qing-hua, Wang Wei. Cost-sensitive fault diagnosis based on weighted rough sets[J]. Proceedings of the CSEE, 2007, 27(23): 93 – 99. (in Chinese)

[4] R Gilad-Bachrach Ranb, A Navot, N Tishby. Margin based feature selection-theory and algorithms[A]. International Conference on Machine Learning[C]. ACM Press, 2004. 43 – 50.

[5] T G Dietterich. An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization[J]. Mach Learn, 2000, 40: 139 – 157.

[6] 周传华,王清,吴科主,赵保华.平均 1-依赖决策树集成算法[J].电子学报,2010,38(2):434 – 438.
Zhou Chuan-hua, Wang Qing, Wu Ke-zhu, Zhao Bao-hua. Averaged one-dependence decision trees ensemble algorithm[J]. Acta Electronica Sinica, 2010, 38(2): 434 – 438. (in Chinese)
[7] G Valentini, F Masulli. Ensembles of learning machines[A]. Proc of Valentini02. Neural Nets WIRN Vietri-02, Series Lecture Notes in Computer Sciences[C]. Springer-Verlag, 2002. 3 – 19.
[8] Zhou Z-H, Yu Y. Ensembling local learners through multimodal perturbation[J]. IEEE Trans. SMC – Part B: Cybernetics. 2005, 35: 725 – 735.
[9] L Breiman. Bagging predictors[J]. Machine Learning. 1996, 24(2): 123 – 140.
[10] RE Schapire. The strength of weak learnability[J]. Machine Learning, 1990, 5(2): 197 – 227.
[11] C Blake, C J Merz. UCI repository of machine learning databases[DB/ OL]. http://www. ics. uci. edu/mlearn/MLRepository. html, Department of ICS, University of California, Irvine, 1998.
[12] Hu Q H, Yu D R, Xie Z X, Li X D. EROS: ensemble rough subspaces[J]. Pattern Recognition. 2007, 40: 3728 – 3739.
[13] Hu Q H, Yu D R, Xie Z. Neighborhood classifiers[J]. Expert Systems with Applications, 2008, 34: 866 – 876.
[14] 米爱中,郝红卫,郑雪峰,涂序彦,一种自整定权值的多分类器融合方法[J].电子学报,2009,37(11):2604 – 2609.
Mi Ai-zhong, Hao Hong-wei, Zheng Xue-feng, Tu Xu-yan. A method of multiple classifier fusion with self-adjusting weights [J]. Acta Electronica Sinica, 2009, 37(11): 2604 – 2609. (in Chinese)
[15] Hu Q H, Yu D R, Liu J F, Wu C X. Neighborhood rough set based heterogeneous feature subset selection[J]. Information Sciences, 2008, 178: 3577 – 3594.
[16] Wu Q, Bell D, Mcginnity M. Multiknowledge for decision making[J]. Knowledge Inform. Systems, 2005, 7: 246 – 266.
[17] Ho T K. The random subspace method for constructing decision forests[J]. IEEE Trans, PAMI, 1998, 20(8): 832 – 844.
[18] Zhu W, Wang F Y. Reduction and axiomization of coving generalized rough sets. Information Science. 2003, 152: 217 – 230.
[19] Zhu W, Wang F Y. On three types of covering-based Rough Sets[J]. IEEE transactions on knowledge and data engineering. 2007, 19(8): 1131 – 1144.
[20] Yang T, Li Q G. Reduction about approximation spaces of covering generalized rough sets[J]. International Journal of Approximate Reasoning. 2010, 51: 335 – 345.
[21] Chen D G, Wang C Z, Hu Q H. A new approach to attribute reduction of consistent and inconsistent covering decision systems with covering rough sets[J]. Inf Sci 2007, 177(17): 3500 – 3518.
[22] Wang C Z, Wu C X, Chen D G. A systematic study on attribute reduction with rough sets based on general binary relations[J]. Inf Sci 2008, 178(9): 2237 – 2261.
[23] Du Y, Hu Q H, Zhu P F. Rule learning for classification based on neighborhood covering reduction[J]. Inf Sci doi: 10.1016/j.ins.2011.07.038
[24] 杜卫锋,秦克云.不协调决策表几种约简标准及其关系分析[J].电子学报,2011,39 (6):1336 – 1340.
Du Wei-feng, Qin Ke-yun. Analysis of several reduction standards and their relationship for inconsistent decision tables. Acta Electronica Sinica[J]. 2011, 39(6): 1336 – 1340. (in Chinese)

作者简介

朱鹏飞　男,1986 年生于河南,硕士研究生,动力机械及工程专业,研究方向为智能控制与故障诊断.

胡清华　男,1976 年生于湖南,副教授,博士生导师.主要研究方向为机器学习、数据挖掘及其应用.
E-mail: huqinghua@hit.edu.cn

于达仁　男,1966 年生于山西,教授,博士生导师,长江学者特聘教授.主要研究方向为建模、仿真以及电力系统控制.