

# 平均 1-依赖决策树集成算法

周传华<sup>1,2</sup>, 王 清<sup>1,3</sup>, 吴科主<sup>1</sup>, 赵保华<sup>2</sup>

(1. 安徽工业大学管理科学与工程学院, 安徽马鞍山 243002;  
2. 中国科学技术大学计算机科学与技术学院, 安徽合肥 230027;  
3. 复旦大学计算机科学技术学院, 上海 200433)

**摘要:** 基于平均 1-依赖贝叶斯分类器(AODE)算法的思想,提出了平均 1-依赖决策树集成算法(AODT),该算法通过使用每个输入属性和类别属性共同建立集成学习中的个体决策树分类器.同时,我们从多任务学习的角度探讨了 AODE 和 AODT 算法的工作原理.通过在 Weka 平台上使用 40 个 UCI 数据集的实验结果表明,该算法可以显著提高决策树学习算法的分类性能,并且具有很好的抗噪声性能.

**关键词:** 集成学习; 多任务学习; 决策树学习算法; 平均 1-依赖贝叶斯分类器

**中图分类号:** TP391.4 **文献标识码:** A **文章编号:** 0372-2112 (2010) 02-0434-05

## Averaged One-Dependence Decision Trees Ensemble Algorithm

ZHOU Chuan-hua<sup>1,2</sup>, WANG Qing<sup>1,3</sup>, WU Ke-zhu<sup>1</sup>, ZHAO Bao-hua<sup>2</sup>

(1. School of Management Science and Engineering, Anhui University of Technology, Ma' anshan, Anhui 243002, China;  
2. Department of Computer Science and Technology, University of Science and Technology of China, Hefei, Anhui 230027, China;  
3. School of Computer Science, Fudan University, Shanghai 200433, China)

**Abstract:** Averaged One-Dependence Estimators (AODE) ensemble naive Bayes classifiers by aggregating the predictions of a set of one-dependence estimators built for each attribute. Inspired by this, in this paper we propose a new method, namely Averaged One-Dependence Trees (AODT), to ensemble decision tree learning algorithms which enumerate each input attribute together with the class attribute to create different component one-dependence decision tree classifiers in the ensemble. We then give a multi-task view of AODE and AODT to explain how they work. We conduct all the experiments on the Weka platform and use the 40 widely used UCI data sets. The experimental results verify the method's effectiveness, efficiency and robustness.

**Key words:** ensemble learning; multi-task learning; decision tree; averaged one-dependence estimators

## 1 引言

泛化能力一直是机器学习所关注的一个重要问题,即学习算法在训练集上学习得到的模型可以很好地预测未知测试数据.集成学习是指利用训练数据建立若干个有差异的分类器,然后将它们各自的预测结果进行集成以作为最终预测结果.由于集成学习可以有效地提高学习系统泛化能力,因此从上世纪 90 年代开始,对集成学习理论和算法的研究逐渐成为了机器学习领域的一个热点<sup>[3,4,7,8]</sup>.为了构造一个强的集成学习系统,现在一般认为,集成中各个体分类器应该具有较高的精度同时个体之间具有较高的差异度<sup>[3]</sup>.

平均 1-依赖贝叶斯分类器 (Averaged One-Dependence Estimators, 简称 AODE)<sup>[1]</sup>是最近提出的一种有效

的朴素贝叶斯分类器的集成算法,它通过为每个输入属性分别建立一个超 1-依赖贝叶斯分类器,然后将它们对新的样例的预测值进行平均作为集成的最终结果.具有分类精确度高,学习效率高,支持增量学习等优点.在本文中,我们基于 AODE 算法的思想提出了平均 1-依赖决策树集成算法 (AODT),该算法通过使用每个输入属性和类别属性共同建立集成学习中的每个个体决策树分类器.实验结果表明其具有以下几个特性:

—在没有噪声的环境下,其分类性能可以与 Boosting 和 Random Forest 集成算法相媲美,同时其分类性能要显著地优于 Bagging 集成算法;

—在有一定的噪声的环境下,其分类性能要显著地优于 Boosting 和 Random Forest 集成算法,同时其分类性能要稍优于 Bagging 集成算法;

—AODT 算法的实现只需要在既有决策树学习算法的基础上加少量修改,并且支持并行化。

## 2 平均 1-依赖决策树集成算法

### 2.1 平均 1-依赖贝叶斯分类器

朴素贝叶斯(Naïve Bayes)分类器是一种简单高效的学习算法,它假设各个属性之间在给定类别属性的情况下是相互独立的.假定训练样例可以描述为  $n$  维属性向量  $\langle A_1, A_2, \dots, A_n \rangle$ ,朴素贝叶斯算法是在给定样例  $a$  的属性值向量  $\langle a_1, a_2, \dots, a_n \rangle$  时,使用以下公式选择具有最大后验概率(MAP)的类别标记  $c$  作为对样例  $a$  的预测结果:

$$\arg \max_{c \in C} p(c) \prod_{i=1}^n P(a_i | c) \quad (1)$$

图 1(a)给出了朴素贝叶斯算法的图形表示,在朴素贝叶斯中每个属性节点都仅以类别属性为自己的父节点.尽管朴素贝叶斯算法简单并且在很多真实的数据集上效果很好,但是其属性之间条件独立性假设在现实世界中并不总是成立的,这在一定程度上影响了其分类性能。

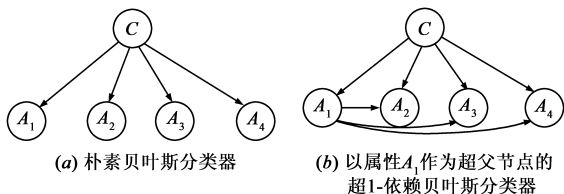


图 1 朴素贝叶斯分类器和超 1-依赖贝叶斯分类器的图形表示

平均 1-依赖贝叶斯分类器算法<sup>[1]</sup>是一种有效的朴素贝叶斯分类器的集成算法,它通过为每个输入属性分别建立一个超 1-依赖贝叶斯分类器,然后将它们对新的样例的预测值进行平均作为集成的最终结果.即在给定样例  $a$  的属性值向量  $\langle a_1, a_2, \dots, a_n \rangle$  时,使用以下公式选择具有最大后验概率(MAP)的类别标记  $c$  作为对样例  $a$  的预测结果:

$$\arg \max_{c \in C} \sum_{i: 1 \leq i \leq n \wedge F(a_i) \geq \lambda} P(c, a_i) \prod_{j=1 \wedge j \neq i}^n P(a_j | c, a_i) \quad (2)$$

其中,  $F(a_i)$  代表属性值  $a_i$  在训练样例中出现的次数,  $\lambda$  则是事先确定的门限值,在最初的研究中<sup>[1]</sup>设定  $\lambda = 30$  以避免不精确的概率估计,但随后的研究表明<sup>[2]</sup> 这样反而会降低算法的分类精确度,所以后续研究均设定  $\lambda = 1$ 。

AODE 算法可以被看作是首先以每个输入属性作为超父节点建立一个超 1-依赖贝叶斯分类器,然后在预测实例  $a$  的类别标记时,通过将 这些超 1-依赖贝叶

斯分类器对每个类别标记  $c$  的预测概率进行平均,并选取具有最大平均后验概率值的类别标记  $c$  作为集成学习的输出.图 1(b)给出了超 1-依赖贝叶斯分类器的图形表示,在超 1-依赖贝叶斯分类器中每个属性节点除了以类别属性为自己的父节点外,还都以给定的超父节点属性为父节点。

### 2.2 平均 1-依赖决策树集成算法

信息增益(Information Gain)是目前使用最为广泛的构造决策树的一种划分标准.传统的决策树学习算法在构造决策树时仅使用类别属性的信息增益来选择划分属性,也就是说构造的决策树的结构仅仅依赖于类别属性.类似在超 1-依赖贝叶斯分类器中,超父节点属性同类别属性一样被设置为所有其它属性的父节点,即其它属性都同时依赖于超父节点属性和类别属性.受此启发,我们将超父节点属性的信息加入到信息增益中,从而使得学习算法在构造决策树时,同时依赖超父节点属性和类别属性的信息增益来决定最佳的划分属性,我们相应称之为超 1-依赖决策树模型.为了表达这种依赖关系,在本文中,我们将类别属性和超父节点属性信息增益的通过各自熵的归一化之和作为新的划分标准.采用熵归一化处理的目的是为了在类别属性取值的个数和超父节点属性取值的个数差距比较大时,防止算法偏向于多数的一方,以至于无法反应少数一方的信息.具体如式(3)所示:

$$\text{ODTIG}(T, A, \text{SPA}, C) = \text{IG}(T, A, C) / \text{Entropy}(T, C) + \text{IG}(T, A, \text{SPA}) / \text{Entropy}(T, \text{SPA}) \quad (3)$$

其中  $T$  表示训练集,  $A$  表示划分属性,  $\text{SPA}$  表示超父节点属性,  $C$  表示类别属性.  $\text{ODTIG}(T, A, \text{SPA}, C)$  表示使用属性  $\text{SPA}$  作为超父节点属性的超 1-依赖决策树模型在给定划分属性  $A$  上的信息增益,  $\text{IG}(T, A, C)$  表示给定划分属性  $A$  相对于属性  $C$  的信息增益,  $\text{Entropy}(T, C)$  表示属性  $C$  的相对于训练集  $T$  的熵,其余符号定义类似。

通过分别使用每个输入属性作为超父节点属性,我们得到一个有差异的超 1-依赖决策树集合.同时,为了进一步提高每个超 1-依赖决策树之间的差异度,我们取消生成决策树的剪枝操作.当给定一个新的样例  $a$  时,通过使用集成中每个个体超 1-依赖决策树对样例  $a$  的预测概率的算术平均值作为集成的预测结果。

值得注意的是, AODT 算法通过分别使用每个输入属性作为超父节点属性和类别属性共同建立集成中的每个超 1-依赖决策树,因此 AODT 的集成规模等于给定的训练样例的输入属性的个数,这点同 AODE 类似.同时,每个超 1-依赖决策树的建立过程并不依赖其它超 1-依赖决策树,即它们是相互独立的,因此算法支持并

行化.

### 2.3 从多任务学习角度解释 AODE 和 AODT 的工作原理

多任务学习 (Multi-Task Learning) 能够通过使用共享表示同时学习多个相关的任务, 相对于分别独立学习这些任务来说, 其可以显著地提高这些任务的泛化能力. 近年以来, 多任务学习以及和其相关的迁移学习逐渐成为机器学习领域的研究热点之一, 并被理论<sup>[6]</sup>和实践<sup>[5]</sup>证明是有效的. 在多任务学习中, 辅助任务相对于主任务来说提供了额外的偏置 (Bias), 这些额外的偏置使得学习算法在搜索假设空间时, 倾向于选择能够同时解释主任务和辅助任务的假设<sup>[5]</sup>. 在实际的应用中, 我们可以通过选择若干同主任务相关但具有不同偏置的辅助任务, 然后通过分别使用这些不同的辅助任务和主任务学习得到若干不同的分类器. 由于这些辅助任务与主任务相关联, 因此每个分类器具有较高的分类精确度, 同时这些不同辅助任务具有不同的偏置, 因此构建的不同的分类器之间具有差异性. 通过将这些分类器进行集成, 可以产生良好的分类结果<sup>[12]</sup>.

从图 1(b) 中我们可以看出在超 1-依赖贝叶斯分类器中, 超父节点属性同类别属性一样被设置为所有其它属性的父节点, 即每个属性之间在给定类别属性和超父节点属性值时属性才是相互独立的. 同样在超 1-依赖决策树的学习过程中, 使用超父节点属性和类别属性的信息增益来共同决定最佳的划分属性. 因此, 超父节点属性在一定程度上扮演的是和分类主任务属性的类似的角色. 在一定程度上, 我们可以认为, 超 1-依赖贝叶斯分类器和超 1-依赖决策树分别可以被看作是使用超父节点属性作为辅助任务的朴素贝叶斯和决策树算法的多任务版本. AODE 和 AODT 通过使用不同的输入属性作为辅助任务和分类主任务共同建立集成个体学习器, 只要不同的输入属性和分类主任务是相关的, 那么产生的多任务学习器就具有较好的性能, 同时如果不同的输入属性代表不同的偏置, 那么产生的不同个体学习器之间就具有一定的差异性, 通过将它们进行集成可以产生较好的性能.

## 3 实验结果与分析

### 3.1 实验数据和对比算法

实验基于 Weka 机器学习平台<sup>[11]</sup>, 并采用 40 个广泛使用的 UCI 数据集<sup>[10]</sup>. 由于我们提出的算法不能处理缺失数据情况, 同时需要每个辅助任务属性必须为离散属性, 所以在实验中我们采取以下三个步骤对每个数据集进行预处理: 采用 Weka 平台中无监督的属性

过滤器 ReplaceMissingValues 处理数据集中的缺失数据; 采用 Weka 平台中的有监督的属性过滤器 Discretize (即使用 MDL 离散方法) 对数据集中的连续值属性进行离散化处理; 采用 Weka 平台中的无监督的属性过滤器 Remove 过滤掉数据集中的冗余属性.

在实验中, 我们对比了本文提出的 AODT 算法同主要的集成学习算法 Bagging、Boosting 和 RandomForest 的分类性能. 由于 Weka 中已经实现了 C4.5、Bagging、Boosting 和 RandomForest 算法, 所以我们仅在 Weka 平台的框架下实现了我们提出的 AODT 算法. Bagging 和 Boosting 的基学习器均使用 C4.5 算法, 而 RandomForest 则使用自有的基学习器. 同时 Bagging、Boosting 和 RandomForest 的集成规模设定为 50. 所有其它的参数均使用 WEKA 的默认值.

### 3.2 实验结果及分析

通过使用 10 次 10 折交叉验证来计算算法在数据集上的分类准确度, 并且每个算法都使用相同的 10 个用于划分 10 折的随机种子. 表 1 给出了上述算法在 40 个数据集上的分类精确度和方差、平均分类准确度和总体 rank 值<sup>[9]</sup>. 表 2 给出了我们使用 95% 置信度的双侧  $t$ -test 比较不同算法的分类性能的对比结果, 其中每个条目  $w/t/l$  表示该行对应的算法同该列对应的算法相比, 在  $w$  个数据集上胜出, 在  $t$  个数据集上平手, 在  $l$  个数据集上落败. 从表 2 中, 我们可以看出, 首先 AODT 算法可以显著地提高决策树学习算法的分类性能 (在 17 个数据集上胜出, 4 个数据集上落败), 其次 AODT 算法的性能要显著优于 Bagging 算法 (在 8 个数据集上胜出, 4 个数据集上落败), 最后, AODT 的性能同目前公认的最好的决策树的集成算法 Random Forest 和 Boosting 相比, 其性能基本相当. 值得指出的是, 在本文的实验设置中, AODT 算法的集成规模在大多数的数据集上都要显著低于 Boosting 和 Random Forest.

由于在很多的实际应用中, 给定的数据集中往往包含一定的噪音数据, 所以健壮性也是评价学习算法的重要特性之一. 表 3 给出了在引入 5% 随机类别噪音的情况下, 使用 95% 置信度的双侧  $t$ -test 比较不同算法的分类性能的对比结果. 从表 3 中, 我们可以看出, 在引入一定噪音的情况下, 首先 AODT 算法在噪音环境下依然可以显著地提高决策树学习算法的分类性能 (在 17 个数据集上胜出, 5 个数据集上落败), 其次 AODT 算法的性能要显著优于 Boosting 算法 (在 21 个数据集上胜出, 3 个数据集上落败) 和 Random Forest 算法 (在 12 个数据集上胜出, 5 个数据集上落败), 并且, AODT 算法同 Bagging 相比, 其性能基本相当.

表 1 算法在 40 个 UCI 数据集上的分类准确度和方差

Datasets	C4.5	Bagging	Boosting	Random Forest	AODT
adult	86.74 ± 0.39	86.94 ± 0.36	85.49 ± 0.40	85.59 ± 0.40	86.56 ± 0.32
anneal	98.78 ± 0.91	98.79 ± 0.87	99.61 ± 0.62	99.42 ± 0.84	96.37 ± 2.32
audiology	77.22 ± 7.68	80.97 ± 7.50	84.82 ± 7.13	79.78 ± 6.87	83.07 ± 6.46
autos	76.39 ± 9.55	83.89 ± 8.51	86.38 ± 6.94	87.06 ± 6.86	84.98 ± 7.89
balance-scale	69.32 ± 3.89	69.26 ± 3.80	69.31 ± 3.90	69.00 ± 3.84	70.31 ± 4.30
breast-cancer	75.26 ± 5.04	73.76 ± 5.85	66.04 ± 8.21	70.37 ± 7.34	69.97 ± 7.04
breast-w	94.65 ± 2.51	95.49 ± 2.57	94.94 ± 2.74	95.55 ± 2.61	95.37 ± 2.62
car	92.22 ± 2.01	93.59 ± 1.79	96.72 ± 1.50	94.42 ± 1.54	95.17 ± 1.42
colic	84.72 ± 5.94	85.10 ± 5.68	79.48 ± 6.36	82.91 ± 5.62	84.24 ± 5.56
colicORIG	76.09 ± 5.31	76.09 ± 5.31	76.09 ± 5.31	73.29 ± 5.52	75.59 ± 5.60
credit-a	86.58 ± 3.53	86.17 ± 3.56	82.72 ± 4.36	84.36 ± 4.01	85.25 ± 3.94
credit-g	72.17 ± 3.49	73.66 ± 3.69	71.69 ± 3.99	73.76 ± 3.63	74.08 ± 3.12
diabetes	77.34 ± 4.91	77.33 ± 4.72	77.16 ± 4.38	76.59 ± 4.78	78.35 ± 4.33
glass	75.23 ± 9.46	77.10 ± 9.13	75.28 ± 9.40	76.94 ± 8.07	75.79 ± 8.70
heart-c	77.32 ± 6.20	80.40 ± 6.38	79.57 ± 6.56	80.78 ± 5.83	80.61 ± 6.13
heart-h	80.96 ± 6.90	80.04 ± 7.24	82.34 ± 6.32	81.84 ± 6.35	80.08 ± 6.97
heart-statlog	82.26 ± 7.32	84.04 ± 6.46	83.22 ± 6.81	83.41 ± 6.85	81.63 ± 7.08
hepatitis	81.32 ± 9.48	83.26 ± 10.18	83.61 ± 8.93	85.52 ± 8.03	84.35 ± 8.87
hypothyroid	99.28 ± 0.42	99.31 ± 0.41	99.50 ± 0.37	99.27 ± 0.40	99.38 ± 0.38
ionosphere	89.49 ± 5.12	91.03 ± 5.33	93.11 ± 4.08	92.99 ± 3.88	91.88 ± 4.11
iris	93.87 ± 4.89	94.47 ± 5.02	94.60 ± 5.42	94.19 ± 5.81	94.33 ± 5.14
kr-vs-kp	99.44 ± 0.37	99.46 ± 0.37	99.60 ± 0.31	99.23 ± 0.46	98.72 ± 0.64
labor	87.13 ± 15.32	91.07 ± 11.36	91.13 ± 12.03	92.27 ± 11.30	90.03 ± 11.64
letter	78.75 ± 0.77	81.87 ± 0.95	89.94 ± 0.75	92.01 ± 0.61	91.47 ± 0.63
lymph	76.51 ± 10.14	79.46 ± 10.20	84.19 ± 8.33	82.69 ± 7.77	78.76 ± 9.81
mushroom	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0
nursery	97.18 ± 0.46	97.42 ± 0.43	99.75 ± 0.16	99.06 ± 0.33	98.89 ± 0.32
primary-tumor	41.01 ± 6.59	45.19 ± 6.16	42.63 ± 6.61	41.27 ± 6.09	46.11 ± 6.19
segment	95.23 ± 1.37	95.68 ± 1.22	96.41 ± 1.22	96.91 ± 1.05	96.33 ± 1.14
sick	97.82 ± 0.75	97.84 ± 0.76	97.63 ± 0.78	97.70 ± 0.73	97.75 ± 0.72
sonar	80.60 ± 8.85	83.07 ± 8.84	82.74 ± 7.72	82.64 ± 7.96	84.48 ± 8.72
soybean	92.63 ± 2.72	94.05 ± 2.61	94.04 ± 2.61	93.73 ± 2.69	94.28 ± 2.62
tic-tac-toe	85.57 ± 3.21	94.73 ± 2.04	98.92 ± 1.03	96.29 ± 1.93	96.71 ± 1.69
vehicle	70.77 ± 3.86	70.84 ± 4.00	72.53 ± 3.95	73.25 ± 3.72	73.24 ± 3.93
vote	96.27 ± 2.79	96.27 ± 2.67	94.80 ± 3.05	96.13 ± 2.96	96.34 ± 2.66
vowel	79.54 ± 4.01	82.41 ± 3.72	86.48 ± 3.25	90.04 ± 3.02	88.12 ± 3.16
waveform-5000	76.36 ± 1.77	78.78 ± 1.77	82.19 ± 1.59	84.47 ± 1.54	78.01 ± 1.84
wine	92.58 ± 5.32	95.55 ± 4.99	97.37 ± 3.92	98.65 ± 2.54	96.01 ± 4.86
yeast	59.46 ± 3.40	59.76 ± 3.46	59.46 ± 3.40	59.50 ± 3.42	60.68 ± 3.63
zoo	92.61 ± 7.33	93.10 ± 7.48	96.07 ± 5.89	92.85 ± 7.07	93.68 ± 6.95
Mean	83.67 ± 4.60	85.18 ± 4.43	85.69 ± 4.26	85.89 ± 4.11	85.67 ± 4.34
Overall rank	4.025	2.8375	2.7375	2.75	2.65
Overall rank	--	2.675	2.475	2.4625	2.3875

表 2 在无噪声情况下使用 95%置信度的双侧  $t$ -test 比较的结果

$w/t/l$	C4.5	Bagging	Boosting	Random Forest
Bagging	14/26/0			
Boosting	18/17/5	16/17/7		
Random Forest	17/18/5	12/22/6	9/23/8	
AODT	17/19/4	8/28/4	9/24/7	5/28/7

表 3 在噪声的情况下使用 95%置信度的双侧  $t$ -test 比较的结果

$w/t/l$	C4.5	Bagging	Boosting	Random Forest
Bagging	16/24/0			
Boosting	8/18/14	5/17/18		
Random Forest	12/20/8	6/21/13	15/21/4	
AODT	17/18/5	7/27/6	21/16/3	12/23/5

## 4 结论

本文从基于多任务集成学习的角度阐述了平均 1-依赖贝叶斯分类器算法的工作原理,并提出了平均 1-依赖决策树集成学习算法.实验结果表明,该算法可以显著提高决策树学习算法的分类性能,并且具有很好的抗噪声性能.平均 1-依赖集成算法的优点是,可以用于不稳定的基学习器(如决策树)以及稳定的基学习器(如 Naïve Bayes),具有很好的健壮性以及支持并行化等.

值得注意的是,在本文实验中使用的 40 个 UCI 数据集,虽然代表了许多不同的具体应用的特性,但是它们无论是属性个数还是样例的个数都是较小或中等规模的.因此,本文的实验结果和结论对于这些较小或中等规模的 UCI 数据集上是成立的.但是,对于属性和样例个数非常大的应用场合,如生物信息,时间序列和文本等领域的大规模高维数据.对于这些应用,一般都需要根据数据的特性对标准的学习算法做相应的调整以有效地处理这类数据,AODT 算法亦然.因此如何修改 AODT 算法以处理该应用下的数据还有待进一步的研究.

### 参考文献:

- [1] G I Webb, J Boughton Z Wang. Not so naïve bayes: aggregating one-dependence estimators[J]. Machine Learning, 2005, 58(1): 5-24.
- [2] Y Yang, G I Webb. To select or to weigh: A comparative study of linear combination schemes for superparent-one-dependence estimators[J]. IEEE Trans on Knowledge Discovery and Engineering, 2007, 19(12): 1652-1665.
- [3] T G Dietterich. Ensemble methods in machine learning[A]. 1st International Workshop on Multiple Classifier Systems[C]. Cagliari, Italy: Springer, 2000, 1587. 1-15.
- [4] R E Banfield, L O Hall, K W Bowyer, W P Kegelmeyer. A comparison of decision tree ensemble creation techniques[J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2007, 29(1): 173-180.
- [5] R Caruana. Multi-task learning[J]. Machine Learning, 1997, 28(1): 41-75.
- [6] J Baxter. A model for inductive bias learning[J]. Journal of Artificial Intelligence Research, 2000, 12(1): 149-198.
- [7] 王珏,周志华,周傲英.机器学习及其应用[M].北京:清华大学出版社,2006.

- [8] Z H Zhou, J X Wu, W Tang. Ensembling neural networks: many could be better than all[J]. Artificial Intelligence, 2002, 137(1): 239-263.
- [9] J Demsar. Statistical comparisons of classifiers over multiple data sets[J]. Journal of Machine Learning Research, 2006, 7(1): 1-30.
- [10] C Blake, C J Merz. UCI repository of machine learning databases[DB/OL]. <http://www.ics.uci.edu/mllearn/MLRepository.html>, Department of ICS, University of California, Irvine, 1998.
- [11] I H Witten, E Frank. Data Mining: Practical Machine Learning Tools and Technology with Java Implementation[M]. Morgan Kaufmann, 2000.
- [12] Q Wang, L Zhang, M M Chi, J K Guo. MTForest: ensemble decision trees based on multi-task learning[A]. Proceedings of the European Conference on Artificial Intelligence (ECAI 2008)[C]. Amsterdam, Netherlands: IOS Press, 2008. 122-126.

### 作者简介:



**周传华** 男,1966 年 12 月出生,安徽含山,博士研究生,教授,主要研究方向为 Web 智能、信息安全与系统集成.  
E-mail: chzhou863@ustc.edu



**王清** 男,1981 年 11 月出生,安徽巢湖,博士研究生,讲师,主要研究方向为机器学习、数据挖掘.  
E-mail: wangqing@fudan.edu.cn

**吴科主** 男,1985 年 12 月出生,硕士研究生,主要研究方向为机器学习、数据挖掘.

**赵保华** 男,1947 年 8 月出生,中国科技大学计算机科学与技术学院教授,博士生导师,中国通信学会通信软件技术委员会副主任委员,中国计算机学会软件工程专业委员会委员,程控交换技术与通信网国家重点实验室学术委员会委员,《计算机信息与技术》杂志编委,主要研究领域为协议理论与协议工程、软件工程.