一种新的组合分类器学习方法

郭华平[1] 袁俊红[1] 张 帆[1] 邬长安[1] 范 明[2]
(信阳师范学院 信阳 464000)[1] (郑州大学 郑州 450052)[2]

摘 要 提出了一种新的基于决策树的组合分类器学习方法FL(Forest Learning)。与bagging和adaboost等传统的组合分类器学习方法不同,FL不采用抽样或加权抽样,而是直接在训练集上学习一个森林作为组合分类器。与传统组合学习方法独立地学习每个基分类器,然后把它们组合在一起的做法不同,FL学习每个基分类器时都尽可能地考虑对组合分类器的影响。首先,FL使用传统的方法构建森林的第一棵决策树;然后,逐一构建新的决策树并将其添加到森林中。在构建新的决策树时,结点的每次划分都考虑对组合分类器的影响。实验结果表明,与传统的组合分类器学习方法相比,FL在大部分数据集上都能构建出性能更好的组合分类器。

关键词 森林学习,边界理论,贡献增益,特征变换

中图法分类号 TP181 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2014.07.059

New Ensemble Learning Approach

GUO Hua-ping[1] YUAN Jun-hong[1] ZHANG Fan[1] Wu Chang-an[1] FAN Ming[2]
(Xinyang Normal University, Xinyang 464000, China)[1] (Zhengzhou University, Zhengzhou 450052, China)[2]

Abstract This paper proposed a new decision tree-based ensemble learning method called FL(Forest Learning). Unlike traditional ensemble learning approaches, such as bagging and boosting, FL directly learns a forest on all training examples as an ensemble rather than on examples obtained by sampling from training set. Unlike the approach of learning ensemble by independently training each classifier and combining them for prediction, FL learns each classifier considering its influence on ensemble performance. FL first employs traditional algorithm to train the first decision tree, and then iteratively constructs new decision trees and add them to forest. When constructing current decision tree, FL considers the influence of each partition on ensemble performance. Experimental results indicate that, compared to traditional ensemble learning methods, FL induces ensemble with much better performance.

Keywords Forest learning, Margin-based theory, Contribution gain, Feature transformation

1 引言

组合学习是机器学习、模式识别和数据挖掘研究中非常活跃的研究领域[1-3]。与一般学习方法仅仅从训练实例集学习单个模型的做法不同,组合学习方法试图学习一个基分类器库,然后组合库中每个成员的预测以期获得更好的分类性能。该思想源于如下观察:一般情况下,委员会做出正确决策比单个委员做出正确决策的可能性更高。在机器学习中,组合分类方法使用给定的训练实例集 D 创建 M 个分类器,即 $H=\{h_j \mid j=1,2,\cdots,M\}$。预测时,它通过对每个基分类器 h_j 的预测(加权)投票来进行分类。已经发表的大量的研究成果在理论和实践上都表明,给定相同的训练信息,组合分类器往往表现出比单个分类器更好的泛化能力[4-6]。

按照一种普遍承认的观点,组合分类器成功的关键是构建有差异且具备高准确率的基分类器[7]。目前,存在很多构建有差异而又准确的基分类器方法,如:通过随机(加权)抽样为每个分类器构建不同的训练实例集,进而构建有差异而又准确的基分类器(如bagging[8]和adaboost[6]);通过特征变换将训练实例集映射到不同的特征空间为基分类器构建不同的实例集(random forest[9]、rotation forest[10]和COPEN[11]),进而构造一组有差异而又准确的基分类器;通过操纵算法的参数或结构构建准确且有差异的组合分类器[12,13]。

与以上方法不同,本文提出了一种新的基于决策树的组合分类器学习方法——森林学习FL(Forest Learning)。FL构建组合分类器时尽可能地避免产生那些无助于提高组合分类器泛化性能的决策树和决策树的分枝。与bagging和adaboost等传统的组合分类器学习算法不同,FL不采用抽样或加权抽样,而是直接在训练集上学习一个森林作为组合分类器。与传统的独立地学习每个基分类器,然后把它们组合在一起的做法不同,FL学习每个基分类器时都尽可能地考虑对

到稿日期:2013-09-21 返修日期:2013-12-09 本文受863项目:大规模汉语词义知识相关特征提取与构建工程(2012AA011101),河南科技厅重点项目:基于自适应蚁群算法的传感器网络节能覆盖研究(12A520035)资助。

郭华平(1982-),男,博士,讲师,CCF会员,主要研究方向为数据挖掘、机器学习,E-mail:hpguo.gm@gmail.com;袁俊红(1961-),女,高级实验师,主要研究方向为数据挖掘;张 帆(1982-),女,硕士,讲师,主要研究方向为信息安全;邬长安(1959-),男,教授,主要研究方向为模式识别、数字图像处理;范 明(1948-),男,教授,博士生导师,CCF高级会员,主要研究方向为数据挖掘、机器学习、数据库。

组合分类器的影响。首先，FL 使用传统的决策树构建方法构建森林的第一棵树；然后，逐一构建新的决策树添加到森林中。在构建新的决策树时，结点的每次划分都考虑对组合分类器的影响，选择那些有利于提高组合分类器性能的“最优”划分。

在 24 个实例集上的实验结果显示：FL 可以构造显著优于单个决策树的组合分类器；较之于 bagging、adaboost 和 random forest，FL 在大部分实例集上更好地提高了分类器的准确率；与 rotation forest 相比，FL 也表现出明显优势。这些结果表明：(1)构建每个分类器过程中，充分考虑结点划分对组合分类器性能的影响将有助于提升它的分类准确率；(2)我们提出的指标能很好地反映结点划分对森林性能的影响。

本文第 2 节介绍问题描述及算法的基本思想；第 3 节介绍构造的度量指标和单棵决策树学习算法的具体细节；第 4 节详细描述森林学习算法；第 5 节给出实验结果及相关讨论；最后总结全文。

2 问题描述及算法基本思想

2.1 问题描述

设 $D=\{\boldsymbol{x}_i \mid i=1,2,\cdots,N\}$ 是训练实例集，其中 $\boldsymbol{x}_i$ 给出第 i 个实例的诸属性值，$y_i \in \{1,\cdots,L\}$ 是与实例 $\boldsymbol{x}_i$ 相关联的真实类标记。假定我们已经在训练实例集 D 上学习了一个组合分类器 $F=\{T_1,\cdots,T_M\}$，其中每个 T_j 都是一棵决策树。F 也可以看作一个森林。本文把组合分类器和森林视为同义词，因为我们只讨论基于决策树的组合分类器。

设 $T_j \in F$ 为任意决策树，v 为 T_j 的任意结点。令 $E(v) \in D$ 为从根结点 $root(T_j)$ 沿着一条路径到达结点 v 的训练实例的集合。假设每棵决策树 T_j 的每个结点 v 都包含一个向量 $(p_1^v,\cdots,p_L^v)$，其中 p_l^v 是 $E(v)$(到达结点 v 的实例集合)中实例属于类 l 的比例。如果 v 是 T_j 的树叶结点，并且实例 $\boldsymbol{x}_i \in E(v)$，则我们把 p_k^v 记作 $p_k^{(j)}$，并称对于实例 $\boldsymbol{x}_i$，T_j 返回向量 $(p_1^{(j)},\cdots,p_L^{(j)})$，指明 $\boldsymbol{x}_i$ 属于类 l 的概率为 $p_l^{(j)}$。

上述假定是合理的，决策树分类算法如果未提供这些概率，稍加调整都可以提供这些信息。

对于每个待分类实例 $\boldsymbol{x}_i$，组合分类器 F 也返回一个向量 $(p_{i1},\cdots,p_{iL})$，指明 $\boldsymbol{x}_i$ 属于类 l 的概率为 p_{il}，其中

$$p_{il}=\frac{1}{M}\sum_{j=1}^{M} p_l^{(j)},\ l=1,\cdots,L \tag{1}$$

组合分类器 F 预测 $\boldsymbol{x}_i$ 属于类 $F(\boldsymbol{x}_i)$，其中 $F(\boldsymbol{x}_i)=\arg\max_l(p_{il})$。

我们的问题是：如何构建森林 $F=\{T_1,\cdots,T_M\}$ 中的每棵决策树，使得 F 具有更好的泛化能力。

2.2 森林学习基本思想

理论上，给定一个属性集，可以构建指数级数量的决策树。尽管其中一些比其它更优越，但是试图通过穷举法找到最优决策树在计算上是不可行的。进而试图构建最优的包含 M 棵决策树的组合分类器是一件更加困难的任务。

本文采用贪心的方法学习一个森林：逐一构建每棵决策树，在构建新的决策树时，结点的每次划分都考虑对组合分类器的影响。该方法的核心是构建合理的度量指标评估划分结点对组合分类器的影响。论文构造了一个称作贡献增益(ConGain)的度量指标来确定最优的划分，具体细节在第 3 节讨论。

3 决策树学习方法

本文提出的森林学习方法的核心是如何保证学习每棵决策时充分考虑划分结点对森林的影响。针对该问题，本节提出了一种称作贡献增益的度量指标，用于监督决策树生长过程。为了避免过早陷入细节，3.1 节首先给出了单棵决策树的基本学习思想及具体算法细节，然后 3.2 节给出了指标的相关细节，最后 3.3 节给出了快速搜索划分条件的方法。

3.1 单棵决策树学习思想

学习高准确率的决策树的核心是构建合理的度量以确定划分实例的最佳方法。很多度量可以用来确定最佳划分，如 C4.5[14] 和 CART[15] 使用的度量。然而这些度量指标并没有考虑到基分类器对组合分类器的影响，不能用于指导论文提出的森林学习方法。

算法 1 CreateTree——创建决策树的算法

输入：D——训练实例集；A——属性集合
输出：以结点 v 为根的决策树
开始：
(1)v=ceateNode(D,A)//创建结点
(2)v. distribution=distribution(D)//获得结点上数据分布
(3)if stop(v) then
(4)　v. leaf=true
(5)　return v
(6)end if
(7)v. test_cond=bestSplit(D,A) //根据式(3)获得最优的测试条件
(8)令 O={o_i|o_i 是 v 的一个可能的输出}
(9)for 每个输出 $o_i \in$ O do
(10)　$D_i=\{\mathbf{x}_i \mid$ v. test_cond($\mathbf{x}_i$)=o_i 并且 $\mathbf{x}_i \in D\}$
(11)　v_i=CreateTree(D_i,A)
(12)　将 v_i 作为结点 v 的派生结点添加到树中
(13)end for
(14)return v

为了避免过早陷入细节，假定已经定义了 $Con(v,F,\boldsymbol{x}_i)$，它是组合分类器 F 对 $\boldsymbol{x}_i$ 分类时结点 v 的贡献。如果 $\boldsymbol{x}_i \notin E(v)$($E(v)$为训练集 D 中到达结点 v 的实例集)，则 $Con(v,F,\boldsymbol{x}_i)=0$。如果 $\boldsymbol{x}_i \in E(v)$，则 $Con(v,F,\boldsymbol{x}_i)$ 的定义在 3.2 节详细讨论。

根据以上定义，我们定义结点 v 对组合分类器 F 的贡献(简称结点 v 的贡献，记作 $Con(v,F)$)，定义为

$$Con(v,F)=\sum_{\boldsymbol{x}_i \in D} Con(v,F,\boldsymbol{x}_i)=\sum_{\boldsymbol{x}_i \in E(v)} Con(v,F,\boldsymbol{x}_i) \tag{2}$$

$Con(v,F)$ 体现了结点 v 对 F 的分类准确率的影响。

为了确定划分条件的优劣，我们需要比较子女结点(划分后)的贡献和父结点(划分前)的贡献，它们之间的差越大，测试条件的效果越好。我们使用这种差异来确定划分效果，形式化，有：

$$ConGain(A_t,F)=\sum_{v_i \in children(v)} Con(v_i,F)-Con(v,F) \tag{3}$$

其中，A_t 表示属性划分条件，$children(v)$ 表示 v 的子女结点。

森林学习(FL)中，学习单棵决策树方法非常简单：与传统决策树(如 CART 和 C4.5)一样，FL 以递归方式建立每棵决策树。对于当前结点 v，FL 使用式(3)确定最优划分。根据划分条件，分布 v 中的实例到相应的子女结点。然后，对每个子女节点重复该过程。算法 1 给出了 FL 学习每个决策树的归纳算法。

算法 1 首先创建一个结点 v，并关联 v 与实例集 D 的分

布(行 1—2),然后判断结点是否停止划分。若满足停止划分条件,则设置 v 为叶子结点并返回结点 v(行 3—6);否则,划分结点 v 中的实例集并生长决策树(行 7—14),其中,第 7 行使用式(3)搜索最优的测试条件。9—13 行根据划分的测试条件分布 v 中的实例到每个子女结点,进而生长每个子女结点。

算法 1 的核心是根据式(3)搜索最优划分,该式需要累计结点在每个实例 $\boldsymbol{x}_i$ 上对森林的贡献,具体细节见 3.2 节。

3.2 计算 $Con(v,F,\boldsymbol{x}_i)$

相关的实践结果表明,AdaBoost[6] 具有很高的泛化能力。为了解释这种现象,Schapir 等[17] 提出了一种叫做实例边界的概念。设 $H=\{h_1,h_2,\cdots,h_L\}$ 是一个包含 L 个基分类器的组合分类器,并且每个分类器 h_t 将实例 $\boldsymbol{x}_i$ 映射(分类)到一个类标号,即 $h_t(\boldsymbol{x}_i)=y\in\{-1,1\}$。又设 $y_i\in\{-1,1\}$ 是实例 $\boldsymbol{x}_i$ 的真实类标号。训练实例 $\boldsymbol{x}_i$ 的边界定义为

$$margin(\boldsymbol{x}_i)=\frac{y_i\sum_t\alpha_t h_t(\boldsymbol{x}_i)}{\sum_t\alpha_t} \tag{4}$$

其中,α_t 是分类器 h_t 的权重。不失一般性,规范化 α_t,$t=1,2,\cdots,L$,使得 $\sum_t\alpha_t=1$ 并且 $0\leqslant\alpha_t\leqslant 1$。则式(4)可改写为

$$margin(\boldsymbol{x}_i)=y_i\sum_t\alpha_t h_t(\boldsymbol{x}_i) \tag{5}$$

不难看出,实例的边界实质上是取值范围为[−1,1]的一个数;当且仅当 $margin(\boldsymbol{x}_i)=0$ 时,实例 $\boldsymbol{x}_i$ 在正确分类与错误分类的边界线上;边界的绝对值表示分类器对判定结果的置信度;边界的符号表示分类器是否正确分类 $\boldsymbol{x}_i$。它们证明,对于任意的正数 θ,组合分类器泛化错误的上界是

$$\hat{\Pr}[margin(\boldsymbol{x},y)\leqslant\theta]+\tilde{O}\left(\sqrt{\frac{d}{m\theta^2}}\right) \tag{6}$$

这表明:给定 θ,具有较大边界分布的组合分类器泛化能力较强。

为了将边界概念直接应用到森林学习中,我们定义训练集 D 的边界(训练边界)为 D 中实例边界的平均值,形式化地,有:

$$margin(D)=\frac{1}{N}\sum_{\boldsymbol{x}_i\in D}margin(\boldsymbol{x}_i) \tag{7}$$

不难看出 $margin(D)$ 的取值范围为[−1,1]。根据式(6)和式(7),我们给出以下两个条件:1)对于具有相同训练边界的组合分类器,训练误差较小的组合分类器泛化能力较好;2)对于具有相同训练误差的组合分类器,训练边界较大的组合分类器泛化能力较好。

假定 v 是决策树 T_j 的结点,$\boldsymbol{x}_i\in E(v)$。令 $f_m=\arg\max_l(p_{i1},p_{i2},\cdots,p_{iL})$,$f_s=\arg\max_l(\{p_{i1},p_{i2},\cdots,p_{iL}\}-p_{if_m})$,即 f_m 和 f_s 分别是 $\{p_{i1},\cdots,p_{iL}\}$ 中最大元素和次大元素的下标。(注:p_{il} 是森林预测实例 $\boldsymbol{x}_i$ 属于类 l 的后验概率)。显然,f_m 是组合分类器 F 对 $\boldsymbol{x}_i$ 的类预测。令 $e_t=\{\boldsymbol{x}_i\mid F(\boldsymbol{x}_i)\neq y_i\wedge\boldsymbol{x}_i\in D\}$,$e_f=\{\boldsymbol{x}_i\mid F(\boldsymbol{x}_i)=y_i\wedge\boldsymbol{x}_i\in D\}$,即 $e_t(e_f)$ 为组合分类器 F 正确(错误)分类的实例集合。

基于以上两个边界条件,定义 $Con(v,F,\boldsymbol{x}_i)$ 为

$$\begin{aligned}&Con(v,F,\boldsymbol{x}_i)\\&=\frac{I(\boldsymbol{x}_i\in e_t)(p^v_{f_m}-p^v_{f_s})+I(\boldsymbol{x}_i\in e_f)(p^v_{y_i}-p^v_{f_m})}{M(|margin(\boldsymbol{x}_i)|+\frac{1}{M})}\\&=\frac{numerator}{M(|margin(\boldsymbol{x}_i)|+\frac{1}{M})}=\frac{numerator}{M|margin(\boldsymbol{x}_i)|+1}\end{aligned} \tag{8}$$

在式(8)中,常数项 $1/M$ 用于避免分母为 0 或过小。(注:决策树对实例 $\boldsymbol{x}_i$ 的预测实际上是 $\boldsymbol{x}_i$ 到达的叶结点的预测)。如果 $\boldsymbol{x}_i\in e_t$,即组合分类器正确预测 $\boldsymbol{x}_i$,则 f_s 是获得最大后验概率的错误类标号。如果 $\boldsymbol{x}_i\in e_f$,即组合分类器错误预测 $\boldsymbol{x}_i$,f_m 是获得最大后验概率的错误类标号。所以,式(8)中项 $numerator/M$ 是结点 v(以及相应的宿主树)对实例 $\boldsymbol{x}_i$ 边界的贡献。设 $\boldsymbol{x}_i\in E(v)$,即 $\boldsymbol{x}_i$ 到达结点 v。式(8)合理性解释如下:

- 当 $\boldsymbol{x}_i\in e_t$ 时,$(p^v_{f_m}-p^v_{y_i})/M$ 是结点 v 对实例 $\boldsymbol{x}_i$ 的边界的真实贡献(见第 2.1 节)。当 $(p^v_{f_m}-p^v_{y_i})/M>0$ 时,结点的预测有助于组合分类器正确预测实例 $\boldsymbol{x}_i$,即结点 v 对 F 正确预测 $\boldsymbol{x}_i$ 的贡献为正。当 $(p^v_{f_m}-p^v_{y_i})/M<0$ 时,结点预测不利于组合分类器正确预测该实例,即结点 v 对 F 正确预测 $\boldsymbol{x}_i$ 的贡献为负。当 $\boldsymbol{x}_i\in e_f$ 时,也有类似的解释。
- 由于 $|margin(\boldsymbol{x}_i)|$ 反映了组合分类器 F 正确(或错误)分类 $\boldsymbol{x}_i$ 的置信度。如果式(8)中 $|margin(\boldsymbol{x}_i)|$ 很大,即 F 正确(或错误)分类 $\boldsymbol{x}_i$ 的置信度很高,那么划分结点 v 很难影响 F 对 $\boldsymbol{x}_i$ 的分类结果,因此,v 对 F 影响很小,即权重 $1/|margin(\boldsymbol{x}_i)|$ 很小。如果 $|margin(\boldsymbol{x}_i)|$ 很小(例如 $margin(\boldsymbol{x}_i)=0$),分裂结点 v 可能改变 F 对 $\boldsymbol{x}_i$ 的分类结果,因此,v 对 S 影响很大,即权重 $1/|margin(\boldsymbol{x}_i)|$ 很大。

通过以上方法,$Con(v,F,\boldsymbol{x}_i)$ 合理地利用了实例 $\boldsymbol{x}_i$ 的边界,进而式(2)和式(3)同时兼顾了前面的两个条件:1)对于具有相同训练边界的组合分类器,训练误差较小的组合分类器泛化能力较好;2)对于具有相同训练误差的组合分类器,训练边界较大的组合分类器泛化能力较好。

3.3 属性的划分方法

假设当前决策树的生长不影响实例的边界。对于离散属性,可以使用二元划分或多路划分方法分派到达决策树某个结点的实例[18]。为了方便操作,本文采用多路方法划分实例集合。例如,假设属性 A_t 包含 3 个可能的取值 a_1、a_2 和 a_3,按照 A_t 的 3 个不同取值把实例集合 D 划分为 3 个子集,其中 $D_k=\{\boldsymbol{x}_i\mid\boldsymbol{x}_i\in D\wedge A_t(\boldsymbol{x}_i)=a_k\}$。使用 D 和每个 D_k,按照式(3)计算结点 v 划分前后对森林 F 的贡献增益 $ConGain(v,F)$。$ConGain(v,F)>0$,表明根据属性 A_t 划分实例 D 有助于提升训练实例集边界分布,进而有助于提升组合分类器的泛化能力,此时,结点 v 生长为 $subtree(v)$。若 $ConGain(v,F)<0$,则不应该使用属性 A_t 划分 D。

对于数值属性,一般假设数值属性值服从某种分布或离散化。这里使用二值化方法离散化数值属性[18]。其核心思想是,按给定的属性排序实例集,然后逐一计算每个划分点(两个连续的实例在给定属性上的平均值)的指标值,对应于最优指标值的划分为最优划分。由于式(2)可重写为

$$\begin{aligned}Con(v,F)&=\sum_{\boldsymbol{x}_i\in D}Con(v,F,\boldsymbol{x}_i)\\&=\sum_{\boldsymbol{x}_i\in D}\frac{p^v_{f_m}-p^v_{y_i}}{M|margin(\boldsymbol{x}_i)|+1}\\&=(p^v_{f_m}-p^v_{y_i})\sum_{\boldsymbol{x}_i\in D}\frac{1}{M|margin(\boldsymbol{x}_i)|+1}\end{aligned} \tag{9}$$

令 $D'=D\cup\{\boldsymbol{x}_i\}$,进而更新结点 v 到 v'。结点 v' 对森林的贡献为

$$\begin{aligned}Con(v',F)&=\sum_{\boldsymbol{x}_i\in D'}Con(v,F,\boldsymbol{x}_i)\\&=\frac{\hat{p}^v_{f_m}-\hat{p}^v_{y_i}}{M|margin(\boldsymbol{x}_i)|+1}+\frac{\hat{p}^v_{f_m}-\hat{p}^v_{y_i}}{p^v_{f_m}-p^v_{y_i}}Con(v,F)\end{aligned} \tag{10}$$

根据式(10),我们可以将二值化方法离散化数值属性的时间复杂度优化到 $O(|D|)$,其中 $|D|$ 是数据集 D 的大小。

4 森林学习方法

本节展示基于决策树的组合分类器学习方法 FL(Forest Learning)的具体细节。与 bagging 和 adaboost 等传统的组合分类器学习算法不同,FL 不采用抽样或加权抽样,而是直接在训练集上学习一个森林作为组合分类器。首先,使用传统的方法构建森林的第一棵树;然后,逐一构建新的决策树并将其添加到森林中。在构建新的决策树时,使用式(3)定义的贡献增益考察结点的每次划分对组合分类器的影响。

分类器之间的差异性是成功构造组合分类器的关键问题之一。为了增加决策树之间的差异性,FL 首先使用特征变换方法把实例集映射到不同空间,进而在不同的空间中根据度量指标(见式(3))构建不同的决策树。文献[19]的研究表明,相比于其它特征变换方法,基于 PCA 的特征变换方法构建的旋转森林(Rotation Forest)具有最好的泛化能力,所以,本文也使用 PCA 作为默认的特征变换方法。算法具体过程如算法 2 所示。算法 2 迭代学习每棵决策树。对于每次迭代 j,算法使用如下 3 步学习一棵决策树 T_j。

• 随机地将属性集合划分为 K 个子集,这些子集可以相交也可以不相交(行 2)。FL 同时支持两种实现方式,为了最大化分类器的差异性,FL 默认使用不相交的子集。

• 对于每个属性子集 F_{jk},随机抽样非空类标号子集并获得相应的实例集合 D'。随机抽样实例集 D' 获得 D_{jk}(行 3—7)。将 PCA 方法应用到 D_{jk} 和 F_{jk} 得到学习变换矩阵 $\boldsymbol{P}_{jk}$。重组 $\boldsymbol{P}_{jk}$ 获得稀疏变换矩阵 $\boldsymbol{P}_j$(行 8)。

• 将 $\boldsymbol{P}_j$ 应用到 D 以获得训练实例集 D_j。以 D_j 为参数调用算法 1 学习决策树 T_j(行 13—15)。

算法 2 森林构造方法

训练阶段

输入:D—训练实例集合;M—森林中决策树总个数;Q—前 Q 个决策树直接使用 C4.5 构建;K—属性子集大小

输出:森林 F

(1)for j=1 to M do

(2) 把属性集 A 随机地划分成 K 个子集 F_{jk}($1\leqslant k\leqslant K$)

(3) for k=1 to K do

(4) 随机抽样非空类标号子集并获得相应的实例集合 D′

(5) 随机抽样 D′得到实例集 D_{jk}

(6) 将 PCA 应用于 D_{jk} 和 F_{jk} 得到其变换矩阵 $\mathbf{P}_{jk}$

(7) end for

(8) 组合 $\mathbf{P}_{j1},\mathbf{P}_{j2},\cdots,\mathbf{P}_{jK}$ 得到稀疏变化矩阵 $\mathbf{P}_j$

$$\mathbf{P}_j=\begin{bmatrix} a_{j,1}^{(1)},a_{j,1}^{(2)},\cdots,a_{j,1}^{(M_1)} & [0] & \cdots & [0] \\ [0] & a_{j,2}^{(1)},a_{j,2}^{(2)},\cdots,a_{j,2}^{(M_2)} & \cdots & [0] \\ \vdots & \vdots & \ddots & \vdots \\ [0] & [0] & \cdots & a_{j,K}^{(1)},a_{j,K}^{(2)},\cdots,a_{j,K}^{(M_K)} \end{bmatrix}$$

(9) if j>Q then

(10) 使用 $\mathbf{P}_j$ 将 D 映射到新空间以获得训练数据 D_j

(11) 在 D_j 上,使用 C4.5 构建决策树

(12) else

(13) 使用前 j−1 个分类器计算每个实例的边界并关联到相应的相应实例

(14) 使用 $\mathbf{P}_j$ 将 D 映射到新空间以获得训练数据 D_j

(15) T_j=CreateTree(D_j)//调用算法 1

(16) end if

(17)end for

预测阶段

(18)for j=1 to M do:

(19) $\mathbf{x}_j'=\mathbf{P}_j^T\mathbf{x}_i$

(20) 使用 T_j 评估每个类概率 $p_{il}^{(j)}$,$1\leqslant l\leqslant L$

(21)end for

(22)$p_{il}=\sum_{j=1}^{M}p_{il}^{(j)}/M,l=1,\cdots,L$

(23)return arg $\max_l\{p_{il}|l=1,\cdots,L\}$

值得注意的是,在初始阶段,使用 C4.5 直接学习决策树(行 10、11)。在预测阶段,FL 使用每个 $\boldsymbol{P}_j$ 把待预测实例 $\boldsymbol{x}_i$ 映射到相应的属性空间,然后使用 $\boldsymbol{T}_j$ 预测实例 $\boldsymbol{x}_i$,最后使用式(1)聚集组合分类器在每个类上的概率预测(行 18—22)。

5 实验

5.1 实验设置

本节设计了两个实验来评估森林学习方法(FL,Forest Learning):M(基分类器个数)对组合分类器性能的影响以及 FL 分类性能。使用 24 个 UCI 实例集(见表 1)测试算法性能。对于每个实例集,使用 5×2 折交叉验证[20]分析 FL 的性能。为了评估 FL 算法(本文提出的算法),选择旋转森林(rotation forest)[9]、adaboost[6]、bagging[8]、随机森林(random forest)[10]以及 C4.5[14]作为参考算法,其中 rotation fo-rest、ababoost 和 bagging 使用 C4.5 作为基分类器。在算法 2 中,设置 Q=1,即使用 C4.5 建立第 1 棵决策树(注:rotation forest 也是用 PCA 方法把训练实例集合映射到新的空间,进而在不同的空间里学习不同的基分类器)。实验使用数据挖掘工具洛阳铲(LySpoon)[21]完成。

表 1 实验用到的 24 个实例集的具体信息描述

实例集	大小	属性数	类个数
anneal	898	38	6
audiology	226	70	24
autos	205	25	6
balance-scale	625	4	3
breast-w	699	9	2
car	1728	4	6
cars	406	8	3
credit-g	1000	20	2
ecoli	336	8	8
flag	194	27	6
glass	214	9	7
heart-statlog	270	13	2
hypothyroid	3772	39	4
kr-vs-kp	3195	36	2
lymphography	148	19	4
machine	209	7	8
mofn-3-7-10	300	11	2
page-block	5473	11	5
promoters	106	27	2
segment	2310	19	7
sick	3772	30	2
vehicle	846	19	4
vowel	990	12	11
zoo	108	18	7

5.2 实验结果

实验1用于评估基分类器个数 M 对组合分类器性能的影响。3个实例集(breast-cancer、credit-g 和 lymphography)作为代表评估 M 对组分类器的影响。相关结果如图1所示，其中横轴表示组合分类器大小，纵轴表示分类器准确率。为了清晰起见，图中省去了它们的标准差。

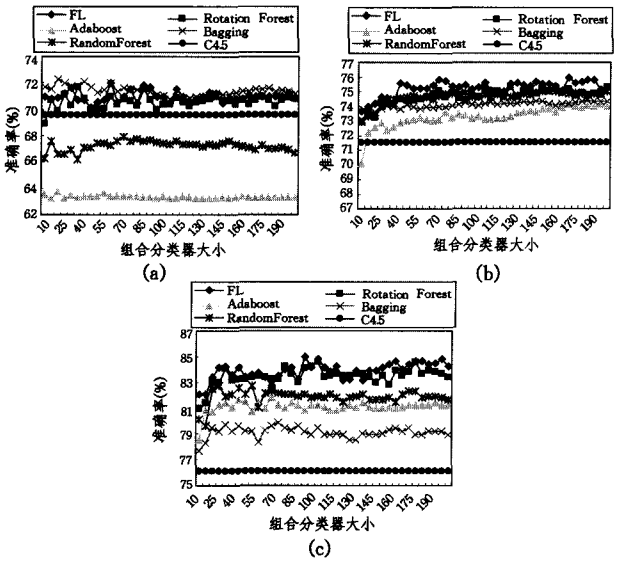

图1 组合分类器准确率在数据集 breast-cancer(a)、credit-g(b)和 lymphography(c)上的变化趋势

图1显示，总体来说，FL 分类准确率是其它算法的上限，只在实例集 breast-cancer 上略输于 bagging。具体地，在这3个实例集上，所有组合分类器的性能随着 M 的增加而呈上升趋势，当 $M\geqslant 30$ 时，组合分类器的性能基本稳定。故后面的实验设置 $M=30$。在 credit-g 和 lymphography 上，FL 较其它所有算法，都表现出优势。在 breast-cancer 上，当组合分类器规模较小时，FL 输 bagging 较多，但比其它组合分类算法更优越。值得注意的是，adaboost 和 random forest 在实例集合 breast-cancer 上的性能明显弱于单棵决策树(使用 C4.5 构建)。该结果的一个合理解释是，breast-cancer 上存在离群点，这导致 ababoost 过分加权那些难分的离群点，使得后续的分类器过分注重分类这些离群点而忽略了数据的整体分布。random forest 通过随机抽样的方法缓解了该问题。

实验2测试了 FL 的分类准确率，相关结果如表2和表3所列，其中，组合分类器的基分类器分别为未剪枝 C4.5 和剪枝 C4.5(除 FL 和 random forest)。在表中，黑体表示相应的算法(列)在相应的实例集(行)上准确率最高或准确率相同而标准差最小。所有的组合分类器大小都为 $M=30$(注：random forest 没有对应的剪枝策略，所以表3中没有给出 random forest 算法的实验结果)。

FL 在学习每棵决策树时，充分考虑了划分每个结点对组合分类器性能的影响，进而更好地保证了分类器间的互补性，因此，与其它算法相比较，FL 表现出明显优势。从表2中不难看出，在绝大多数实例集上(14)，第一种方法(FL，Forest Learning)都获得了最高的准确率。后面依次是 rotation forest(4)、adaboost(3)、bagging(2)、random forest(1)和 C4.5(0)。表3也表现出类似的结果。FL 在绝大多数实例集上(14)获得了最高的准确率。后面依次是 adaboost(7)、rotation forest(2)、bagging(1)和 C4.5(0)。

为了检验 FL 对其它算法的优势是否具有统计意义，选用显著度为95%的配对 t 测试，把 FL 分别与 rotation forest、adaboost、bagging、random forest 和 C4.5 进行了比较。结果见表2和表3。其中，结果旁边的黑点表示在相应的实例集(行)上，FL 显著优于相应的方法(列)；圆圈表示在相应的实例集上，FL 显著劣于相应的方法。

表2 森林学习算法及 C4.5 的准确率和标准差，其中，组合分类器(除 FL 和 random forest 外)基分类器使用未剪枝 C4.5 建立

实例集	FL	rotation forest	adaboost	bagging	random forest	C4.5
anneal	98.95±0.43	98.75±0.47	98.95±0.64	98.60±0.43●	**99.04±0.36**	98.33±0.53●
audiology	79.20±2.10	77.96±2.52	**80.97±2.78**	79.20±2.29	73.54±2.52●	74.96±2.13●
autos	74.54±3.98	74.45±4.49	74.84±5.77	**76.10±3.56**	74.35±5.20	69.77±5.70●
breast-cancer	**70.58±2.68**	70.35±3.90	62.80±4.23●	67.90±4.08	66.92±3.47●	66.57±5.08
balance-scale	**91.02±0.76**	90.82±1.15	79.36±1.89●	83.65±1.48●	81.34±1.97●	79.36±2.39●
breast-w	**97.39±0.66**	97.00±0.55	95.97±0.35●	95.96±0.78●	96.14±0.56●	94.11±1.20●
car	96.62±0.55	**96.70±0.68**	94.04±1.01●	91.39±1.08●	91.86±1.14●	90.17±1.07●
cars	79.36±2.85	79.01±2.90	**82.12±1.77**○	81.72±2.85○	82.36±3.15○	78.57±2.93
credit-g	**74.52±2.08**	73.92±2.53	72.74±1.73●	73.06±2.67	74.22±2.10	69.30±2.08●
ecoli	85.89±1.29	**85.95±1.69**	81.19±2.37●	82.44±2.56●	83.27±2.01●	79.76±3.02●
flag	**68.66±5.01**	67.11±4.25	62.89±3.40●	61.34±4.10●	64.85±3.49●	57.53±5.49●
glass	**72.47±2.69**	71.03±2.37	71.50±5.33	68.60±4.89	76.17±3.00○	62.52±6.36●
heart-statlog	**83.41±2.21**	82.81±2.20	80.52±3.48●	80.89±2.31●	80.30±2.86●	76.74±4.16●
hypothyroid	99.30±0.26	99.36±0.20	99.40±0.20	**99.41±0.18**	99.06±0.19●	99.35±0.23
kr-vs-kp	99.19±0.28	99.08±0.55	**99.29±0.18**	99.12±0.34	98.54±0.25●	98.95±0.35●
lymphography	**84.24±3.20**	83.78±3.19	80.68±4.64	78.78±2.12●	81.89±3.99	75.81±4.15●
machine	**92.82±3.06**	92.53±2.80	87.27±2.63●	85.84±2.86●	87.27±1.62●	85.45±3.12●
page-blocks	97.17±0.25	**97.22±0.30**	96.90±0.23	97.15±0.30	97.01±0.26●	96.69±0.24
promoters	**87.55±4.72**	86.98±4.40	86.42±5.32	83.02±5.41●	84.15±4.46	74.53±5.13●
segment	**97.97±0.37**	97.71±0.30●	97.58±0.42●	96.44±0.42●	97.46±0.31●	95.04±0.61●
sick	98.60±0.35	**98.62±0.28**	98.61±0.33	98.60±0.39	98.08±0.25●	98.37±0.56
vehicle	**77.54±1.84**	77.38±2.17	75.93±1.25●	73.29±0.93●	73.26±1.52●	69.65±2.26●
vowel	**94.15±1.45**	93.45±1.20	89.54±1.63●	85.05±2.47●	90.06±2.29●	72.12±3.06●
zoo	**92.29±3.24**	91.90±2.65	91.70±3.69	92.08±3.36	90.30±3.02	91.89±3.41

表 3　森林学习算法及 C4.5 的准确率标准差，其中，组合分类器(除 FL 外)基分类器使用剪枝 C4.5 建立

实例集	FL	rotation forest	adaboost	bagging	C4.5
anneal	**98.98±0.43**	98.73±0.48●	98.89±0.50	98.00±0.87●	97.86±0.88●
audiology	79.20±2.10	77.17±2.82	**81.06±1.52**	79.03±2.47	75.75±2.61●
autos	74.98±3.98	74.74±4.01	**75.91±4.90**	75.22±2.71	67.42±6.49
breast-cancer	71.58±2.68	70.28±3.51	63.50±3.65●	**71.82±1.95**	69.58±1.66
balance-scale	**91.02±0.76**	90.59±1.08●	80.10±1.76●	83.46±1.19●	79.29±1.80●
breast-w	96.88±0.66	**97.02±0.55**	96.11±0.69	95.99±0.80	94.39±1.01●
car	**96.62±0.55**	96.49±0.65	94.14±1.14●	90.00±0.99●	87.75±1.10●
cars	79.36±2.85	79.06±2.87	**81.63±2.05**○	81.72±2.61○	78.52±3.77
credit -g	**74.52±2.08**	74.18±2.23	72.36±1.68●	74.04±2.22●	71.52±1.75●
ecoli	**86.89±1.29**	86.13±1.59	82.14±1.94●	82.56±2.93●	80.48±2.26●
flag	**68.66±5.01**	67.42±4.19	60.93±3.31●	61.34±5.03●	55.67±4.45●
Glass	70.47±2.69	70.56±2.54	**71.68±4.66**	68.50±4.64	63.18±6.06●
heart-statlog	**83.41±2.21**	82.30±2.38	79.33±2.89●	81.33±2.54	78.67±3.16●
hypothyroid	99.30±0.26	99.36±0.22	**99.41±0.22**	99.40±0.19	99.36±0.23
kr-vs-kp	**99.19±0.28**	99.11±0.40	99.32±0.16	99.11±0.30	99.07±0.23
lymphography	**84.24±3.20**	84.05±3.04	81.35±2.53●	79.73±2.47●	76.08±3.49●
machine	**92.82±3.06**	92.34±2.76	87.37±2.97●	85.65±2.84	85.26±3.09●
page-blocks	97.17±0.25	**97.23±0.25**	96.83±0.27	97.18±0.24	96.83±0.20
promoters	87.55±4.72	86.98±4.40	**88.68±3.44**	81.70±6.42	73.96±4.34
segment	**97.97±0.37**	97.68±0.30●	97.71±0.62	96.42±0.48●	95.10±0.63●
sick	98.60±0.35	98.49±0.31	**98.68±0.37**	98.54±0.39	98.42±0.48●
vehicle	**77.54±1.84**	77.35±2.10	76.29±1.87	73.50±0.92●	69.72±2.24●
vowel	**93.73±1.45**	93.00±1.24●	88.67±1.80●	83.39±2.98●	69.94±2.66●
zoo	**92.29±3.24**	91.90±2.65	91.69±3.82	92.08±3.36	91.89±3.41

正如表 2 所列，相比于未剪枝 C4.5，相应的 FL 在 18 个实例集上表现出显著优势。除此之外，相比于 random forest 和基于未剪枝 C4.5 的 adaboost 和 bagging，FL 分别在 12、13 和 16 个实例集合上显示出明显优势。与 rotation forest 相比，FL 略显优势(在 2 个实例集上显著好)。另外，仔细比较 FL 和 rotation forest，虽然 FL 只在两个实例集上显著胜出，但是几乎在所有的实例集合上 FL 的准确率都高于 rotation forest。

表 3 显示的结果与表 2 类似。相比于 C4.5，FL 在 16 个实例集上表现出显著优势。相比于 adaboost 和 bagging，FL 都在 10 个实例集上显出明显优势。与 rotation forest 相比，FL 也略显优势(在 5 个实例集上显著好)。

结合以上结果，我们总结：(1)FL 可以构造显著优于单个决策树的组合分类器；(2)相较于其它高级森林学习方法，提出的森林学习方法能更好地提高分类器的准确率；(3)提出的指标能较好地考虑划分结点对森林性能的影响，进而很好地监督每棵决策树的学习过程。

结束语　本文提出了一种新的组合分类器学习方法——森林学习方法(FL)。与传统的森林学习方法不同的是，FL 在学习当前决策树时，通过本文设计的指标来评估决策树的生长对森林性能的影响。为了增加基分类器间的差异性，FL 使用特征变换方法把实例集映射到不同空间，进而在不同的空间中根据度量指标构建不同的决策树。

在 24 个实例集上的实验结果显示，相较于 bagging、adaboost 和 random forest，FL 在大部分实例集上更好地提高了分类器的准确率；与 rotation forest 相比，FL 也表现出明显优势。这些结果表明：(1)构建每棵分类器过程中，充分考虑结点划分对组合分类器性能的影响将有助于提升它的分类准确率；(2)我们提出的指标能很好地反映结点划分对森林性能的影响。

森林学习的核心是评估决策树分支对组合分类器性能的影响，因此寻找其它评估函数是未来的一个研究方向。另外，这种学习组合分类器的方法对基于规则的组合分类器也适用，所以，将森林学习思想推广到基于规则的组合分类器学习中也是一个值得研究的问题。

参 考 文 献

[1] Gayar N E, Kittler J, Roli F, et al. Multiple Classifier Systems [C] // Proceedings of 9th International Workshop on MCS. LNCS 5997, Springer 2010

[2] Sansone C, Kittler J, Roli F, et al. Multiple Classifier Systems [C] // Proceedings of 10th International Workshop on MCS. LNCS 6713, Springer 2011

[3] Zhou Z-H, Roli F, Kittler J, et al. Multiple Classifier Systems [C] // Proceedings of 11th International Workshop on MCS. LNCS 6713, Springer 2013

[4] Sun Y, Todorovic S, Li J, et al. Unifying the error-correcting and output-code AdaBoost within the margin framework[C] // Raedt L D, Wrobel S, eds. Proceedings of the 22nd International Conference on Machine Learning. 2005: 872-879

[5] Bartlett P L, Traskin M. AdaBoost is Consistent[J]. Journal of Machine Learning Research, 2007, 8: 2347-2368

[6] Freund Y, Schapire R F. A decision-theoretic generalization of on-line learning and an application to boosting[J]. Journal of Computer and System Sciences, 1997, 55(1): 119-139

[7] Dietterich T G. Ensemble methods in machine learning[C] // Kittler J, Roli F, eds. Proceedings of the 1st International Workshop on Multiple Classifier Systems. 2000: 1-15

[8] Breiman L. Bagging predictors[J]. Machine Learning, 1996, 24(2): 123-140

[9] Rodríguez J J, Kuncheva L I, Alonso C J. Rotation Forest: A New Classifier Ensemble Method[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006, 28(10): 1619-1630

[10] Breiman L. Random Forests [J]. Machine Learning, 2001, 45 (1):5-32

[11] Zhang D, Chen S, Zhou Z-H, et al. Constraint projections for ensemble learning[C]// Fox D, Gomes C P, eds. Proceedings of the 23rd AAAI Conference on Artificial Intelligence. Chicago, IL, 2008:758-763

[12] Hecherman D. Bayesian Networks for Data Mining [J]. Data Mining and Knowledge Discovery, 1997, 1(1):79-119

[13] Lin H-T, Li L. Support Vector Machinery for Infinite Ensemble Learning[J]. Journal of Machine Learning Research, 2008, 9: 285-312

[14] Quinlan J R. C4. 5: Programs for Machine Learning [M]. Morgan-kaufmann Publisher, San Mateo, CA, 1993

[15] Breiman L, Friedman J H, Olshen R, et al. Classification and Regression Trees[M]. London: Chapman and & Hall, 1993

[16] Ho T K. The Random Subspace Method for Constructing Decision Forests[J]. IEEE Trans. Pattern Analysis and Machine Intelligence, 1998, 20(8):832-844

[17] Schapire R E, Freund Y, Bartlett P, et al. Boosting the Margin: A New Explanation for the Effectiveness of Voting Methods [J]. Annals of Statistics, 1998, 26(5):1651-1686

[18] Tan P, Steinbach M, Kumar V. 数据挖掘导论[M]. 范明, 范宏建, 等译. 北京: 人民邮电出版社, 2008

[19] Rodríguez J J, Kuncheva L I, Alonso C J. An Experimental Study on Rotation Forest Ensembles[C]// MCS 2007. LNCS 4472, 2007:459-468

[20] Rudin C, Schapire R E, Daubechies I. Open Problem: Does AdaBoost Always Cycle?[J]. Journal of Machine Learning Research, 2012, 23:1-4

[21] http://nlp.zzu.edu.cn/LySpoon.asp

(上接第 245 页)

搜索长路径，由于两方搜索是交替进行的，如果一方搜索结束，另一方搜索则很可能已经找到最短路的从起点开始(或到终点)的子路。因此，BiBFS 对于长路径查询有高正确率。

3. 结果分析

实验结果表明，对于大规模图中的长路径查询，双向优先搜索有较高的运行效率和正确率。在路网中，一般来说，起始点和终止点的地理位置越远，其最短路径越可能是长路径。启动查询时，可先检查起止点的地理位置，若相距较远，则可采用双向广度优先搜索；反之，则采用单向广度优先搜索。

结束语 最短路径查询是图研究中的一个经典问题。目前大部分研究假设图中每条边只有一种权值。然而，有些应用需要考虑图中每条边有多种权值。多权值路网中，最短路的子路不一定也是最短路，使得单权值路网中的大部分求解最短路算法不适用。本文提出了一种双向广度优先搜索方法及剪枝策略，用以求解多权值路网中的最短路径近似解。实验表明，该算法适用于长路径搜索。与单向广度优先搜索相比，该算法有更快的运行效率；与基于 Dijkstra 算法的贪心算法相比，该算法有更高的准确率。在路网中查询最短路径时，若起始点和终止点的地理位置相距较远，则适合采用双向广度优先搜索求解。

参 考 文 献

[1] Geisberger R, Sanders P, Schultes D, et al. Contraction Hierarchies: Faster and Simpler Hierarchical Routing in Road Networks[C]// WEA. 2008:319-333

[2] Abraham I, Fiat A, Goldberg A V, et al. Highway dimension, shortest paths, and provably efficient algorithms[C]// SODA. 2010:782-793

[3] Bast H, Funke S, Matijecvic D. Transit: ultrafast shortest-path queries with linear time preprocessing[C]// Proc. of the 9th DIMACS Implementaion Challenge. 2006:175-192

[4] 林澜, 闫春钢, 蒋昌俊, 等. 动态网络最短路问题的复杂性与近似算法[J]. 计算机学报, 2007(4):608-604

[5] 王树西, 吴政学. 改进的 Dijkstra 最短路径算法及其应用研究[J]. 计算机科学, 2012, 39(5):223-228

[6] 黄贵玲, 高西全, 靳松杰, 等. 基于蚁群算法的最短路径问题的研究和应用[J]. 计算机工程与应用, 2007, 43(13):233-235

[7] 戴树贵, 孙强, 潘荫荣. 带限制条件的多权最短路径近似算法[J]. 计算机工程, 2003, 29(7):88-91

[8] Yang Ya-jun, J Xu-yu, Gao Hong, et al. Finding the optimalpath overmulti-cost graphs[C]// CIKM'12. ACM, New York, NY, USA, 2012:2124-2128

[9] Dijkstra E W. A note on two problems in connection with graphs [J]. Numerical Mathematics, 1959(1):269-271

[10] Pohl I. Bi-direcitonal search[J]. Machine Intelligence, 1971(6): 128-140

(上接第 249 页)

[6] 徐星, 吴昱. 基于扩散机制的杂交粒子群优化算法[J]. 计算机应用研究, 2011, 11:4156-4159

[7] Pires, Solteiro E J, et al. Particle swarm optimization with fractional-order velocity[J]. Nonlinear Dynamics, 2010, 61(1/2): 295-301

[8] Zhan Zhi-hui, et al. Adaptive particle swarm optimization [J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 2009, 39(6):1362-1381

[9] Pan, Indranil, Das S. Brief Introduction to Computational Intelligence Paradigms for Fractional Calculus Researchers[M]// Intelligent Fractional Order Systems and Control. Springer Berlin Heidelberg, 2013:63-85

[10] 杨伟超. Alpha 稳定分布噪声下通信信号调制识别研究[D]. 哈尔滨: 哈尔滨工程大学, 2012

[11] Weron A, Weron R. Computer simulation of Lévy α-stable variables and processes[M]. Springer Berlin Heidelberg, 1995: 379-392

[12] 吕晓蕊. Alpha 稳定分布随机变量的产生[J]. 计算机与数字工程, 2012, 40(3):32-34

[13] GEATbx: Example Functions (single and multi-objective functions)[EB/OL]. http://www.geatbx.com/docu/ fcnindex-01.html#P86_3059

[14] Xin Yao, Liu Yong, Lin Guang-ming. Evolutionary programming made faster[J]. IEEE Transactions on Evolutionary Computation, 1999, 3(2):82-102