doi:10.11835/j.issn.1000-582X.2018.04.007

基于非平衡数据的随机森林分类算法改进

魏正韬,杨有龙,白　婧
(西安电子科技大学 数学与统计学院,西安 710126)

摘　要:随机森林算法作为一种组合分类器有较好的分类性能,适合多样的分类环境。算法同样也存在一些缺陷,例如算法处理非平衡数据时不能很好地区分正类和负类。针对这一问题,通过对抽样结果增加约束条件来改进 Bootstrap 重抽样方法,减少抽样对非平衡性的影响,同时尽量保证算法的随机性。之后利用生成数据的非平衡系数给每棵决策树进行加权处理,提升对非平衡数据敏感的决策树在投票环节的话语权,从而提升整体算法对非平衡数据的分类性能。通过上述两种改进可以明显提高随机森林在决策树数量不足情况下的分类精度。

关键词:非平衡数据;随机森林算法;有条件的 Bootstrap 重抽样;加权的决策树
中图分类号:TP391.4　　**文献标志码**:A　　**文章编号**:1000-582X(2018)04-054-09

An improved random forest algorithm based on unbalanced data

WEI Zhengtao, *YANG Youlong*, *BAI Jing*
(School of Mathematics and Statistics, Xidian University, Xi'an 710126, P.R.China)

Abstract: Random forest algorithm has better classification performance as a combination of classification and is suitable for a variety of classification environments, but it also has some flaws. For example, it can not distinguish positive and negative class when dealing with unbalanced data. By setting conditions on sampling results, we improve the Bootstrap sampling method, reduce the influence of sampling on non-equilibrium and ensure the randomness of this algorithm. Then, we weight every decision tree according to the non-equilibrium coefficient of the generated data to enhance the discourse right of the decision tree which is sensitive to the non-equilibrium data and improve the classification performance of the whole algorithm dealing with unbalanced data. With these two above improvements, the new algorithm can significantly improve classification performance when the number of decision tree is insufficient.

Keywords: unbalanced data set; random forest; conditional Bootstrap resampling; weighted decision tree

大数据时代的本质是在数据处理的基础上实现硬件智能化,其目标是真正实现人与物的智能交互,甚至物与物的智能交互,建立一个全新的生产模式及产业结构。现在全球数据量呈现井喷式的增长,最近几年数据量的增长速度都是保持在 50%以上,而当前数据总量的 80% 都是最近两年产生。如此庞大的数据不仅仅对现有计算机的计算能力提出了挑战,也让传统的数据处理办法显得捉襟见肘,分类相关算法在这样的背景下应运而生。

收稿日期:2017-10-20
基金项目:国家自然科学基金资助项目(61573266)。
Supported by National Natural Science Foundation of China (61573266).
作者简介:魏正韬(1990—),男,主要从事概率图模型与数据分析的研究,(E-mail)xdweizhengtao@163.com;
杨有龙(联系人),男,教授,博士生导师,主要从事概率图模型与数据分析的研究,(E-mail)youlongy @126.com。

分类算法是解决分类问题的方法，是数据挖掘、机器学习和模式识别中一个重要的研究领域。常见的分类算法有决策树分类、KNN 算法、支持向量机、朴素贝叶斯、随机森林算法等。随机森林算法是 Breiman L[1] 于 2001 年提出的一个可处理高维度和非线性样本的分类器组合模型，并在众多领域得到了广泛应用。

在随机森林算法发展的过程中，算法对非平衡数据分类时性能不足缺陷逐渐地显露出来。黄衍等人[2]对比非平衡分类问题上随机森林和支持向量机的性能，随机森林算法在非平衡分类数据的处理显著逊色于支持向量机的结论。

现有国内外对随机森林算法改进方法大致可以分为 3 类：①将新的理论引入随机森林算法，随机森林算法本身就是 Breriman 将自己的 Bagging 算法[3]与 Tin Kam Ho 的 Random Subaspace 算法[4]结合得到的算法，很多学者致力于将其他算法与随机森林算法结合来提升性能，例如 Gall[5]，Ishwaran 等人[6]。②数据预处理融入随机森林算法、针对随机森林算法处理非平衡数据时性能不足的问题，将数据预处理后再利用随机森林算法分类，可有效提升随机森林对非平衡数据的敏感度。例如，吴琼[7]、杜军[8]等人。③对随机森林算法构建过程进行优化。在构建随机森林算法的过程中，增加决策树的强度可以有效提升随机森林算法的性能，例如雍凯等人[9]为属性进行权重评估，在生成决策树时优先选择权重大的属性借此提升单个决策树的分类性能。

1　随机森林算法处理非平衡数据

数据集多而复杂造成了实际中同一种算法处理不同数据集时的性能波动。为了高效处理数据集，客观衡量算法的性能。根据数据集自身的某一个特点或者某一类特点对数据进行分类成为一种行之有效的方法。非平衡数据就是基于这一背景下诞生的产物。

当一个数据集呈现出不等分布的特性时这些数据集叫做非平衡数据集。非平衡数据集是指数据集中某一类的样本数量明显少于其他类样本的数目，其中占数量最多的一类样本被称为多数类（正类），而占数量最少的一类则称为少数类（负类）。如果数据集中某一类的数量远远大于另一类的数量时，这类数据集叫做类间不平衡。现实生活中非平衡数据并不少见，例如疾病监测，生病人数往往远远少于健康的人数，例如网络攻击监测，攻击行为数远远少于正常访问，因此，非平衡数据研究具有重要的应用价值和理论意义。

为了方便表述有如下定义：

①假设样本集 S 中样本数为 $|S|=m$。

②$S=\{(x_i,y_i)\}$，$i=1,\cdots,m$，这里 x_i 满足 $x_i\in X$，而 X 是维度为 n 的空间，$X=\{f_1,f_2,\cdots,f_n\}$，并且 $y_i\in Y=\{1,\cdots,C\}$ 是样本的特征值。

③$S_{\min}$ 是少数类样本，$S_{\max}$ 是多数类样本，满足 $S_{\min}\cap S_{\max}=\{\varnothing\}$ 并且 $S_{\min}\cap S_{\max}=\{S\}$。

④R 为数据集不平衡度，$R=\left|\dfrac{S_{\max}}{S_{\min}}\right|$。

1.1　非平衡数据分类器性能评价标准

由于非平衡数据本身的特殊性，所以非平衡数据并不可平衡数据，主要依靠训练精度作为标准来评价分类性能。针对上述问题，建立非平衡数据分类精度混淆矩阵：

其中 Pos 为正类样本，Neg 为负类样本；

$N=Pos+Neg$ 是全体学习样本。

TP（true positive）和 TN（true negative）分别表示被分类正确的多数类样本和被分类正确的少数类样本。

FP（false positive）和 FN（false negative）分别表示被分类错误的多数类样本和被分类错误的少数类样本。

表 1　混淆矩阵

Table 1　Confusion matrix

系数	Predicted Positive	Predicted Negative	
正类样本	TP	FN	Pos
负类样本	FP	TN	Nrg
	$PPOS$	$PNEG$	N

各个指标的具体计算公式如下：

$$TP\ rate = TP/Pos = TP/(TP + FN), \tag{1}$$

$$FP\ rate = FP/Neg = FP/(TN + FP), \tag{2}$$

精度为

$$\text{accuracy} = (TP + TN)/\text{N}, \tag{3}$$

查准率为

$$\text{precision} = TP/(TP + FP), \tag{4}$$

查全率为

$$\text{recall} = TP\ rate。 \tag{5}$$

根据以上定义考虑非平衡数据分类器分类性能时要考虑精度(accuracy)，查准率(precision)以及查全率(recall)。对于一个非平衡数据分类器而言这些指标越高，分类器性能越好。

1.2 真正类趋势图

为了直观地比较算法之间性能的差异还可以通过 TP/FP 散点对比图，也就是算法把正类数据分类正确的程度。TP/FP 散点对比图中以 FP rate 为横轴，以 TP rate 为纵轴。上图中每一个黑点都代表了一个分类器。

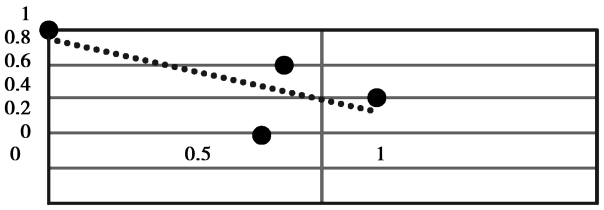

图 1 *TP/FP* 散点对比图

Fig.1 Comparison of *TP/FP*

点越靠近左上角分类器性能越好，也就是代表分类器的点与(0,1)距离越小分类性能越好，在实际应用中 $FP\ rate$ 越小，$TP\ rate$ 越大分类性能越好。

随机森林算法中决策树的数量与分类性能成正相关，因为决策树的数量越多相当于投票的专家越多，分类结果越好，相应的时间复杂度和空间复杂度就会增加，在到达一定的数量时分类性能达到最佳。刘敏等[10]利用 UCI 数据集对随机森林算法中决策树的数量与数据集的关系进行了实验分析，得出当决策树的数量为 100 时大部分的分类性能最好。当决策树数目不同得到的 ROC 点也不同，将这些点连起来，如果一个分类器的 ROC 散点图总在另外一个分类器左上方，可以说在 ROC 散点图左上方的那条分类性能较好。

接下来会用两种评价来对比改进后的随机森林算法与原始随机森林算法的区别。

2 处理非平衡数据分类的方法

2.1 数据预处理，过抽样与欠抽样

非平衡数据的处理方法可以总结为过抽样和欠抽样，也就是通过重抽样，以及去噪声和冗余数据等方法来降低数据不平衡度 $R\left(R = \left|\frac{S_{\max}}{S_{\min}}\right|\right)$，从而提升分类器性能，其中过抽样是提升少数类样本的学习样本，相反欠抽样是降低多数类的学习样本。

常见的过抽样方法代表是由 Chawla 等[11]提出的 SMOTE 技术，其核心思想是通过插值在少数类样本插入人造样本。Han 等[12]在 SMOTE 方式的基础上提出了 Borderline-SMOTE 技术，通过在适当的地方插值，从而保证增加的样本是有意义的。

而欠抽样技术常见的有 Tomek link、一致子集(consistent subset)、编辑技术(Wilson's editing)editing)以及单边选择(one-sided selection)等[13-14]。Barandela 等[15]整理对比了上述算法的优缺点。另外

Dehmeshki 等[16]提出了数据过滤技术，根据构造的规则将发现数据中的安全区，将安全区中的样本点舍弃不用，这种方法本质上也是一种欠抽样技术。

所有的欠抽样技术或者过抽样技术都取得了很好的效果，因此，在处理一个非平衡数据问题时人们常常两种技术同时使用。

2.2　算法层面的改进

改进现有算法对非平衡数据的敏感性[10]和根据非平衡数据的特点设计新的算法都是处理非平衡数据分类的方法。

例如：考虑误分代价[17-19]的 Cost-sensitive 学习。

Step1　建立代价矩阵 C，其中 $C(i,j)$表示，将类标号为 j 的样本误分为类 i 的代价，一般假设 $C(i,i)=0$。

Step2　最小化条件风险[17]。

$$R(i \mid x) \sum_j P(j \mid x)C(i,j)。 \tag{6}$$

这里的 $R(i|x)$表示将 x 分为 i 类的平均代价。对 x 来说应该选择 $k=\arg minR(i|x)$作为其类别号。

Step3　假设 j 是少数类时，一般情况下选择 $C(i,j)>C(j,i)$的情况。一方面这样降低少数类的误分代价，在实际情况下少数类被错分的情况造成的后果往往比多数类被错分造成的结果严重得多，另一方面误分代价将分类边界向多数类偏移，可以提升少数类的分类精度。

2.3　组合方法

组合方法的主要思想在于将多个分类器组合成一个分类器，以提高分类性能。其中，提升是被广泛使用的技术。通过提升，多个弱分类器可以组合成一个强分类器。AdaBoost 是采用提升技术算法的代表。在该算法中，最终得到的分类器是多个弱分类器的线性组合形式：

$$H(x)=sign\left(\sum a_t H_t(x)\right)。 \tag{7}$$

3　改进的随机森林算法

3.1　改进的 Bootstrap 重抽样

3.1.1　BBootstrap 重抽样对非平衡数据分类的影响

抽样方法的最终目的都是期望通过抽样得到的样本尽可能靠近总体。随机森林算法使用的 Bootstrap 算法是随机森林分类算法随机性的保证。Bootstrap 重抽样方法每次抽取总体的 2/3 作为一个训练样本，不断地重复这一个抽取动作，以期望用一系列大小为原训练样本 2/3 的训练样本搭建出一个空间，通过这个空间达到无限接近总体的目的。

这样做显然具有很高的合理性，因为当抽样的数目越多，搭建出来空间的特点越接近整体的特点。当抽样的次数达到一定的程度后，是可以无限趋近于原样本的。但这样做具有很强的随机性，具体体现在以下 3 个方面：

1）每一次抽样的样本彼此不同，相互独立。

2）每一组抽样结果都是唯一的，每再抽一次样都会得到一组全新的抽样结果。

3）训练样本不同，不同的训练样本训练出来的决策树显然也不会相同，因此，同一组数据，会训练出很多不同的随机森林。

但是 Bootstrap 的随机性会加剧抽样子集的非平衡性问题。非平衡数据之所以难处理，从本质上来说是因为数据分配不均匀造成，从非平衡系数 R 的角度可以把 Bootstrap 重抽样的结果分成 3 种情况：

1）训练子集中没有少数类，训练子集中非平衡系数 R'无法计算。

2）训练子集中少数类很少，训练子集中非平衡系数 R'大于原非平衡系数 R。

3）训练子集中少数类样本充足，训练子集中非平衡系数 R'小于等于原非平衡系数 R。

由于 Bootstrap 抽取每一个子集时是简单随机抽样，所以上述 3 种情况出现的概率是相同的。但是第 1

种和第2种情况抽样得到的子集对于处理非平衡问题没有帮助的，可以称之为无效子集。无效子集的存在会导致构建出来可以代替整体的子集空间所需要的子集数量增加。这样一方面增加了抽样成本，另一方面在用这些无效子集训练出来的决策树效率不仅低下，而且会干扰最终的投票结果。

3.1.2 添加约束条件的 Bootstrap 重抽样

提出了一种增加约束条件的 Bootstrap 重抽样算法，对抽样结果进行筛选，剔除抽样结果中的无效子集，从而保证得到的训练子集中非平衡系数 R' 小于等于原非平衡系数。

改进后的随机森林算法的 Bootstrap 重抽样的工作步骤如下：

Step1 从数据 S 中抽取 2/3 的数据样本，记录观测值再放回。

Step2 从数据 S 中抽取数据时设定代价，保证抽样得到的数据集非平衡度不超过原数据集：

$$\left|\frac{S'_{\max}}{S'_{\min}}\right|\leqslant R, \tag{8}$$

$S'_{\max}$ 为抽样后的多数类数据集，$S'_{\min}$ 为抽样后的少数类数据集，其中

$$|S'_{\max}\cup S'_{\min}|=\frac{2}{3}|S|。 \tag{9}$$

Step3 如果抽样得到的数据集满足 step2 中的条件则重复 step1，总计重复 c 次，c 为拟构造决策数的数量。

实验证明上述改进的抽样算法是可以有效地提升原算法对非平衡数据的敏感性，但是增加限制条件毫无疑问地增加了训练成本。即使训练的复杂度并不会影响分类器的复杂度，但毕竟每次都要计算抽样出的训练集的非平衡系数 R'，这确实增加了训练成本。

对于一个非平衡系数为 R 的数据集，在去除单位后，可以令

$$|S_{\min}|=1\&|S_{\max}|=R, \tag{10}$$

这样

$$|S|=R+1。 \tag{11}$$

那么相应抽样后的少数类样本数量 $|S'_{\min}|\in[0,1]$，根据公式(8)，有以下的推导：

$$\left|\frac{S'_{\max}}{S'_{\min}}\right|=R'\leqslant R\Rightarrow\frac{\frac{2}{3}(R+1)-S'_{\min}}{S'_{\min}}\leqslant R\Rightarrow S'_{\min}R\geqslant\frac{2}{3}R+\frac{2}{3}-S'_{\min}\Rightarrow\left(S'_{\min}-\frac{2}{3}\right)\mathrm{R}\geqslant\left(\frac{2}{3}-S'_{\min}\right),$$

因此，有且只有

$$S'_{\min}\geqslant\frac{2}{3}S_{\min}。 \tag{12}$$

式(8)才成立。从而上述改进的 Bootstrap 重抽样算法的约束条件也可以改为保证每次抽到的少数类样本至少要占总体少数类样本的 2/3。

3.2 加权的随机森林算法

3.2.1 决策树分类性能差异分析

传统随机森林算法使用的基分类器是 CART 决策树，该决策树使用的是随机选取属性进行分裂。CART 算法采取的是最小基尼系数分割。具体分割方法如下：

$$Gini_{\text{split}}\mathrm{S}=\left|\frac{S_1}{S}\right|Gini(S_1)+\left|\frac{S_2}{S}\right|Gini(S_2),$$

其中：

$$Gini(S)=1-\sum_{i=1}^{m}P_i^2。$$

在一个非平衡系数为 $R'R'$ 二类非平衡数据问题中

$$\sum_{i=1}^{m}P_i^2=\left(\frac{1}{R'+1}\right)^2+\left(\frac{R'}{R'+1}\right)^2 Gini(S)=1-\sum_{i=1}^{m}P_i^2=1-\left\{\left(\frac{1}{R'+1}\right)^2+\left(\frac{R'}{R'+1}\right)^2\right\},$$

$$Gini_{\text{split}}S=\left|\frac{S_1}{S}\right|\left\{1-\left(\left(\frac{1}{R'_1+1}\right)^2+\left(\frac{R'_1}{R'_1+1}\right)^2\right)\right\}+\left|\frac{S_2}{S}\right|\left\{1-\left(\left(\frac{1}{R'_2+1}\right)^2+\left(\frac{R'_2}{R'_2+1}\right)^2\right)\right\},$$

在理想的基础上 $S_1/S_2=R$，这意味着多数类和少数类被完全分开。

$$Gini_{\text{split}}S=\left|\frac{R}{R+1}\right|\left\{1-\left(\left(\frac{1}{R'_1+1}\right)^2+\left(\frac{R'_1}{R'_1+1}\right)^2\right)\right\}+\left|\frac{1}{R+1}\right|\left\{1-\left(\left(\frac{1}{R'_2+1}\right)^2+\left(\frac{R'_2}{R'_2+1}\right)^2\right)\right\},$$

$$Gini_{\text{split}}S=\left|\frac{R}{R+1}\right|\left\{1-\left(\frac{1+R'^2_1}{R'^2_1+2R'_1+1}\right)\right\}+\left|\frac{1}{R+1}\right|\left\{1-\left(\frac{1+R'^2_2}{R'^2_2+2R'_2+1}\right)\right\}。$$

这样在对于非平衡数据求最小基尼系数时就可以转换为一个关于非平衡系数的问题。目标节点 S_1 是多数类样本，S_2 是少数类样本。根据非平衡数据节点尽量纯的原则，R'_1 一定满足 $1<R<R'_1$。R'_2 一定满足 $0<R'_2<1$。那么如何使 $Gini_{\text{split}}S$ 最小，就是 R'_1 趋近于 R，R'_2 趋近于 0，因此，构造决策树的数据集的非平衡系数 R 越靠近 1 越能纯净地把正负类分开。

3.2.2　加权的随机森林分类算法 Bootstrap 重抽样

由 Bootstrap 重抽样可知，训练 CART 决策树的数据集是随机产生的，因此，这些数据集的非平衡系数是不同的，这就导致由这些数据集训练出来的决策树的能力是有差异的。但是在随机森林算法中，这些决策树并不会区别对待，分类性能差的决策树和分类性能好的决策树拥有相同的投票权，这显然是不合理的。如果能对这些决策树进行区别对待，提升性能好的决策树的话语权无疑可以提升随机森林算法的投票效率。在这一基础上提出了一种基于非平衡系数的加权随机森林算法。具体实现步骤如下：

假设所有决策树原始投票权重为 1，计算每棵树的权重方式为每棵树自身数据集的 R_n 比上所有抽样后数据集的平均数，权重函数如下：

$$C(n)=\frac{R_n}{\sum_1^N R_n/N},\tag{12}$$

把构造好的决策树每一个决策树的投票权重设定为 1，这对随机森林算法决定一个样本类别是没有影响的。除以平均数是为了降低某些非平衡度高的数据集对结果有过分的影响力。

例如：对于一颗决策树来说，该决策树的非平衡度 R_n 为 9 而所有抽样数据集非平衡度的平均数为 10，该决策树的投票权重为 $9/10\times1=0.9$。

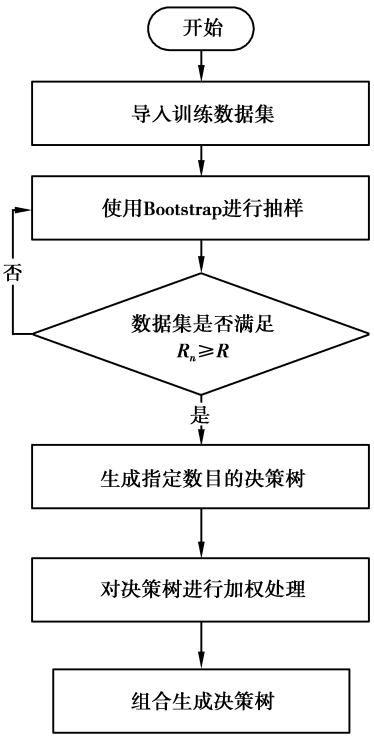

图 2　改进后的算法流程图

Fig.2　The improved random forests algorithm flowchart

4　算法合理性分析

4.1　实验设计

笔者提出了 2 种改进，一是改进的 Bootstrap 重抽样，二是在第 1 种改进的基础上利用非平衡系数对决策树进行加权后投票，为了可以清晰地证明算法改进的有效性。

每种算法分别比较决策树的数量 N 在 10，20，…，90，100 时的分类精度以及 ROC 曲线图。由于随机森林算法处理非平衡数据时的不稳定性，所以为了保证结果的稳定性，每次实验都重复 100 次取平均数。

实验数据采用来自 KEEL [20]，数据集仓库中 9 个非平衡二类分类数据，这些数据集的不平衡系数 (imbalance ratios，R) [21]分布在 1.5～9 范围内。表 2 描述了 *Dataset* 为数据集名，*total* 为数据集总样本数，*dim* 为维数，*R* 为不平衡系数。

表2 实验数据集

Table 2 Experimental data sets

Datasets	*total*	*dim*	*R*
heart	270	14	1.25
Wdbc	569	31	1.68
yeast1	1 484	8	2.46
haberman	306	3	2.70
vehicle2	846	18	2.88
vehicle3	846	18	2.99
new-thyroid1	215	5	5.14

4.2 实验结果分析

从图3可以清晰地看出改进后的算法相比于原算法对herbeman的分类精度有明显提升,其中在改进的Bootstrap重抽样算法基础上进行的加权随机森林算法可以进一步提升精度。

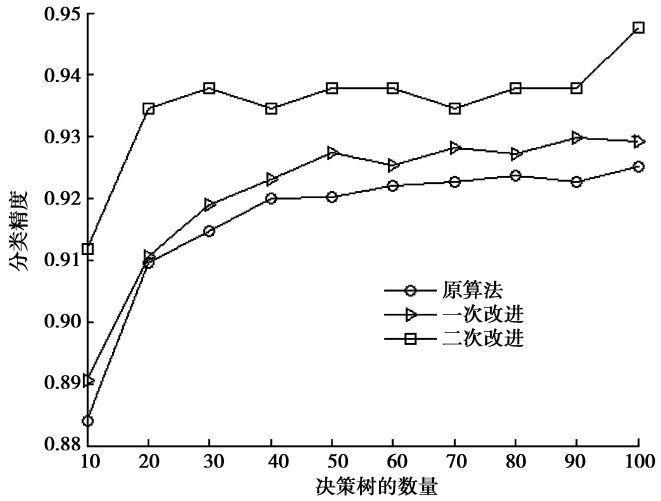

图3 精度对比图

Fig.3 Comparison of accuracy

从图4可以发现改进的算法不仅可以提升分类精度,更多的正类被正确地分到了目标类当中。

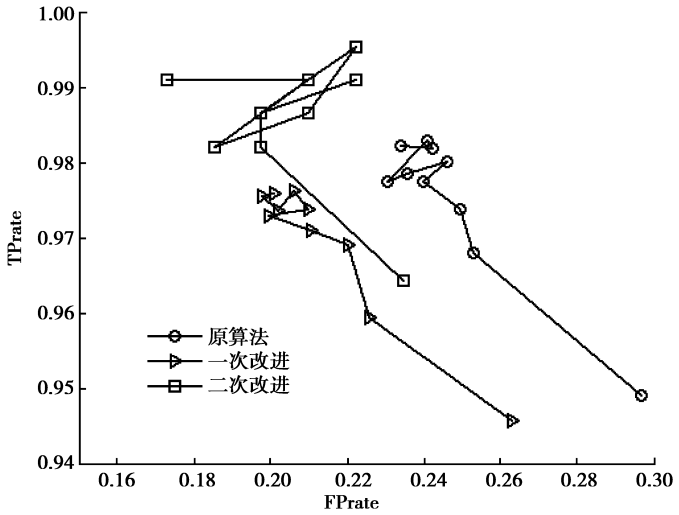

图4 TP/FP散点对比图

Fig.4 Comparison of TP/FP

实验表明，改进后的算法不仅对 herbman 分类有所提升，也可以对其他实验数据的分类性能也有显著的提升。

表 3　部分实验结果

Table 3　Part of the experimental results

决策树的数目		20	40	60	80
heberman	原算法	0.909 608	0.920 065	0.921 961	0.923 791
	一次改进	0.910 523	0.923 072	0.925 294	0.927 19
	二次改进	0.934 641	0.934 641	0.937 908	0.937 908
heart	原算法	0.943 63	0.954 074	0.957 259	0.958 444
	一次改进	0.947 481	0.953 111	0.959 111	0.960 667
	二次改进	0.951 852	0.959 259	0.962 963	0.966 667
vechicle2	原算法	0.993 617	0.994 385	0.995 331	0.995 567
	一次改进	0.993 853	0.994 444	0.994 917	0.995 39
	二次改进	0.996 454	0.995 272	0.995 272	0.995 272
vechicle3	原算法	0.940 898	0.958 629	0.971 158	0.975 177
	一次改进	0.941 371	0.959 338	0.973 995	0.977 541
	二次改进	0.945 626	0.964 066	0.976 359	0.979 314
wdbc	原算法	0.990 51	0.992 267	0.992 97	0.994 2
	一次改进	0.993 322	0.992 794	0.994 2	0.994 552
	二次改进	0.989 455	0.992 97	0.992 97	0.994 728
yeast1	原算法	0.940 903	0.952 965	0.955 863	0.954 043
	一次改进	0.946 294	0.954 852	0.958 288	0.960 108
	二次改进	0.944 07	0.957 547	0.957 547	0.962 938
yeast2vs4	原算法	0.989 572	0.992 802	0.992 84	0.994 086
	一次改进	0.990 623	0.993 307	0.995 019	0.995 486
	二次改进	0.992 218	0.994 163	0.996 109	0.996 109

由表 3 不难看出，在进行了算法改进后，随机森林算法拥有相同的决策树数目时，第一次改进和第二次改进都可以提升算法的分类精度，其中在一次改进基础上进行的二次改进分类精度也是优于仅仅进行一次改进的分类精度。

5　结　语

针对随机森林处理非平衡数据性能不足的问题，在原算法的基础上，提出了 2 种改进。改进的 Bootstrap 重抽样和加权的随机森林算法。实验表明 2 种改进均可以有效提升随机森林算法处理非平衡数据的性能，这一提升在决策树数量不足的情况下效果尤其明显。

参考文献：

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

[2] 黄衍，查伟雄. 随机森林与支持向量机分类性能比较[J]. 软件，2012(6):107-110.
HUANG Yan, ZHA Weixiong. Comparison on classification performance between random forests and support vector machine[J]. Software,2012(6):107-110.(in Chinese)

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

[4] Ho T K. Random Decision forests[C]// International Conference on Document Analysis and Recognition. Washington,

DC: IEEE Computer Society, 1995.

[5] Gall J, Lempitsky V. Class-specific Hough forests for object detection[C]// Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2009.

[6] Ishwaran H, Kogalur U B, Blackstone E H, et al. Random survival forests[J]. Journal of Thoracic Oncology Official Publication of the International Association for the Study of Lung Cancer, 2011, 6(12):1974-1975.

[7] 吴琼, 李运田, 郑献卫. 面向非平衡训练集分类的随机森林算法优化[J]. 工业控制计算机, 2013, 26(7):89-90.
WU Qiong, LI Yuntian, Zheng Xianwei. Optimized random forests algorithm for imbalanced training sets[J]. Industrial Control Computer, 2013, 26(7):89-90.(in Chinese)

[8] 杜均. 代价敏感学习及其应用[D]. 武汉: 中国地质大学, 2009.
DU jun. Cost sensitive learning and its application[D]. China University of Geosciences,2009.(in Chinese)

[9] 雍凯.随机森林的特征选择和模型优化算法研究[D]. 哈尔滨: 哈尔滨工业大学, 2008.
YONG Kai. Research on feature selection and model optinization of random forest[D]. Harbin: Harbin Institute of Technology,2008.(in Chinese)

[10] 刘敏, 郎荣玲, 曹永斌. 随机森林中树的数量[J]. 计算机工程与应用, 2015, 51(5):126-131.
LIU Min, LANG Rongling, CAO Yongbin. Number of trees in random forest[J]. Computer Engineering and Applications, 2015, 51(5):126-131.(in Chinese)

[11] Chawla N V, Bowyer K W, Hall L O, et al. SMOTE: synthetic minority over-sampling technique[J]. Journal of Artificial Intelligence Research, 2011, 16(1):321-357.

[12] Han H, Wang W Y, Mao B H. Bordeline-SMOTE: A new over-sampling method in imbalanced data sets learning[J]. Lecture Notes in Computer Science, 2005, 3644(5):878-887.

[13] Batista G E A P A, Prati R C, Monard M C. A study of the behavior of several methods for balancing machine learning training data[J]. Acm Sigkdd Explorations Newsletter, 2004, 6(1):20-29.

[14] Kubat M, Matwin S. Addressing the curse of imbalanced training sets: One-sided selection[C]// International Conference on Machine Learning, 2000.

[15] Barandela R, Valdovinos R M, Sánchez J S, et al. The imbalanced training sample problem: Under or over sampling[J]. Lecture Notes in Computer Science, 2004.

[16] Dehmeshki J, KarakÖY M, Casique M V. A rule-based scheme for filtering examples from majority class in an imbalanced training set[C]// International Conference on Machine Learning and Data Mining in Pattern Recognition. Verlag: Springer, 2003.

[17] Elkan C. The foundations of cost-sensitive learning[C]// Proc of the 17th International Joint Conference on Artificial Intelligence(IJCAI01). Washington DC: University of California, San Diego, 2001.

[18] Domingos P. Metacost: A general method for making classifiers cost-sensitive[C]// ACM Sigkdd International Conference on Knowledge Discovery and Data Mining. ACM, 2002.

[19] Fan W, Stolfo S J, Zhang J, et al. AdaCost: Misclassification cost-sensitive boosing[C]// Sixteenth International Conference on Machine Learning.Morgan Kaufmann Publishers Inc. 1999.

[20] Alcalá-Fdez J, FernáNdez A, Luengo J, et al. KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework[J]. Jaurnal of Multiple-Valued Logic & Soft Computing, 2011, 17: 255-287.

[21] 彭立志. 基于机器学习的流量识别关键技术研究[D]. 哈尔滨: 哈尔滨工业大学, 2015.
PENG Lizhi. Research on key technologies of machine learning based traffic identification[D]. Harbin: Harbin Institute of Technology,2015.(in Chinese)

(编辑 王维朗)