决策树中避免过度拟合的方法

王黎明,刘 华

(武汉理工大学 计算机科学与技术学院,湖北 武汉 430070)

摘 要: 通过学习训练数据集来构造分类树的策略可能无法达到最好的泛化性能。随机噪声和某些决策仅基于少量训练数据的情况都会导致决策树的分类精度下降,并且过度拟合训练数据集。避免过度拟合主要是通过对树的剪枝来实现,包括预剪枝和后剪枝。后剪枝方法有很多种,主要从计算复杂性、误差估计和算法理论基础角度分析其中的REP、MEP和规则后剪枝算法。

关键词: 噪声;过度拟合;误差;后剪枝;降低误差剪枝;最小误差剪枝;规则后剪枝

中图分类号:TP18 文献标识码:A 文章编号:1672-7800(2006)10-0080-03

0 前言

决策树学习是逼近离散值函数方法的归纳推理算法,是数据挖掘中最为重要的分类方法之一,被广泛地运用于模式识别和机器学习。Breiman 的 CART(Classification And Regression Trees)系统和 Quinlan 的 ID3 算法是决策树学习的先驱,之后人们不断加以改进,提出了各种决策树优化算法。Quinlan 的改进决策树归纳包 C4.5 目前被普遍采用。由于决策树对噪声数据有很好的健壮性,并且能够学习析取表达式,学习是目前应用最为广泛的归纳推理算法之一。

决策树通过对一组训练数据进行学习,将得到的函数表示成一棵决策树。但通过学习训练数据集来构造分类树无法达到最好的泛化性能,特别是当训练数据中有噪声,或训练样例的数量太少以至于不能产生目标函数有代表性的采样时,该策略便会遇到困难。而 ID3 算法产生的决策树能过度拟合训练样例。

1 噪声与过度拟合(overfitting)

给定一个假设 H,如果在假设空间上存在另一个假设 H',使在训练集上 H 的错误率差比 H'小,而在测试集上 H'的错误率比 H 小,那么称假设 H 对于训练数据是过度拟合的。

图 1 显示了 ID3 算法学习中的过度拟合的影响。

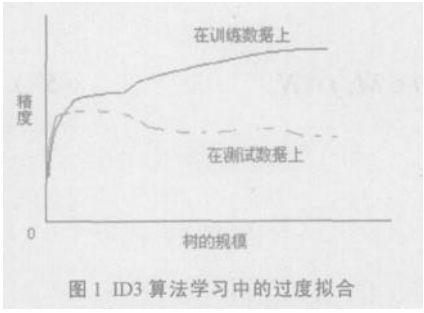

图 1 ID3 算法学习中的过度拟合

横轴表示决策树创建过程中树的结点总数,纵轴表示决策树的预测精度。实线表示在训练集上的精度,虚线表示在测试集上的精度。可以看出,随着树的增长,训练样集的精度单调上升的,而测试集的精度先上升后下降。

导致 H 比 H'在前期更好地拟合训练样例而后来的测试样例表现较差,这种情况发生的一种可能原因是训练样例中含有噪声。噪声包含分类噪声和属性噪声,这两种噪声都可能导致过度拟合。

例如,对于正确的训练样本,ID3 生成图 2 (a)表示的决策树 T_1。如果在训练样本中增加一条如下的训练"-"例:

<medium, green, circle, Class ="+">,

该例子却将错误地标记为正例。此时 ID3 将会建立一个更为复杂的决策树 T_2,如图 2 (b)所示。

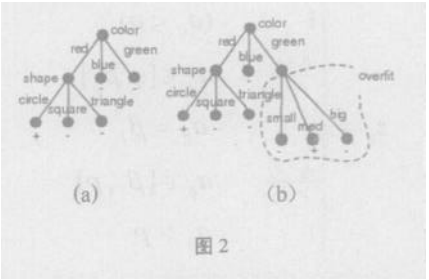

图 2

由于新的样例被标记为"+"例,所以 ID3 会在决策树中第 2 层最右边的叶结点上增加新的分枝,并且在这个结点的子树中作进一步搜索。只要新的噪声样例与原来结点所包含的样例有任何差异,ID3 就会找到一个新的决策属性把该噪声样例从这些样例中分离出来。结果是 ID3 会得到另外一棵决策树 T_3,它仍然比 T_1 复杂。但新的决策树能完美地拟合训练样例,而较简单的树 T_1 则无法做到。然后新的决策树 T_3 也仅仅是拟合训练样例中噪声数据的结果,可以断定在取自同一实例分布的测试集,T_1 会优于 T_3。

训练样例中的随机噪声会导致过度拟合,同时,这些噪声也有可能直接导致样本冲突,比如对某两个样例,相同的属性描述却有着截然相反的分类。另外,当属性的描述不完备,例如重要属性值缺省,或属性值不足以判别分类时,也会导

作者简介:王黎明,武汉理工大学硕士研究生,研究方向为机器学习;刘 华,武汉理工大学硕士研究生,研究方向为人工智能。

致样本冲突。样本冲突必然会导致学习所得到的决策树对训练样本过度拟合。

事实上，当训练样例没有噪声干扰时，过度拟合也有可能发生，特别是当某些决策仅基于少量数据，而这些数据并不能客观反映训练集的大体趋势时。例如，少量的样例被关联到叶子结点时，很可能出现一些巧合性，使得一些属性恰好可以很好地分割样例，但却与实际的目标函数没有任何关系。

过度拟合在机器学习中是一个重要的实践难题。Mingers1989 年在相关文献中指出，对于涉及 5 种带有噪声的和缺省属性数据的不同研究，过度拟合使决策树的精度降低了 10%～25%。

2 避免过度拟合

由于实际问题中存在太多不确定因素，用决策树算法对训练集分类时，所得到的决策树规模太大。研究证明，大而复杂的决策树并不意味着可以得到更加准确的规则集。另外，寻找最小决策树被证明是 NP 问题，所以在现实中找不到绝对的最小决策树。为了避免过度拟合，只能通过分析造成过度拟合的原因，来寻找一些简化技术以修剪决策树。

避免决策树学习中过度拟合的途径有很多，主要分为两大类：预剪枝(pre-pruning)和后剪枝(post-pruning)。

预剪枝是指在决策树算法完全正确分类训练数据之前，如果某个结点的支持度不够，则及早停止该结点子树的增长。尽管这一方法看起来很直接，如何精确估计何时停止树的增长是相当困难的。由于该方法不必生成整棵树，且算法相对简单，效率也高，适合解决大规模问题，所以现仍得到广泛的应用。对过度拟合的树进行后修剪的方法被证明在实践中更为成功。后剪枝首先构造完整的决策树，允许树过度拟合训练数据，然后对那些置信度不够的结点的子树用叶结点来替代，该叶结点所属的类用子树中占统治地位的实例所属的类来替代。

无论用那种方法得到正确的决策树，一个关键的问题是用什么样的准则来判定哪些子树要被修剪。

交叉验证(Cross-validation) 使用与训练样例截然不同的独立测试集来评价通过后修剪从树上修剪结点的效用。

统计测试 通过对训练集的统计测试，来估计扩展一个特定的结点是否能改善在训练集外的整个实例的性能。例如，Quinlan 使用的卡方测试(chi-square)。

最小长度描述（MDL）方法基于一种启发式规则，使用一个明确的标准（编码的长度）来衡量训练样例和决策树的复杂度，判别该结点的复杂度是否比记忆例外情况的复杂度更高。

其中第一种方法是最普通的，很多后修剪方法都是它的变种。当然，用于验证的测试集应该足够大，以便提供具有统计意义的实例样本。下面介绍常用的几种后剪枝方法。

3 后剪枝方法

3.1 Reduced Error Pruning（REP）

REP 方法由 Quinlan 在文献[6]中首先提出，它是一种最简单的剪枝方法，其过程如下所述：

自底向上，对于树 T 的每个非叶子子树 S，用一个叶结点来代替生成一棵新树，其中该叶结点标记为 S 中占统治地位的类。如果新树得到一个较小或相等的对测试集分类错误个数，且子树 S 不包含具有相同性质的子树，则 S 被删除，由叶结点替代。

运用 REP 方法得到是关于测试集原始树的最精确的子树，并且是有此精度的规模最小的树；另外它的计算复杂性是线性的，因为决策树中的每个非叶子结点只需要访问一次就可以评估其子树被剪枝的概率。但是该方法也存在不足之处。由于在测试集中一些不会出现的稀少训练数据应对原始树的部分在剪枝过程中要被忽略，一旦测试集比训练集小得多的时候，分类的精度将受到极大的限制。

尽管有这些缺点，REP 方法仍被作为一种基准来评价其他剪枝方法的性能，它了解决策树方法学习的优点和缺点提供了一个好的初始切入点。

3.2 Minimum Error Pruning（MEP）

MEP 方法由 Niblett 和 Bratko 首先提出。它从树的叶结点开始，向上搜索一棵单一的树以使分类误差的期望概率达到最小。但它并不需要测试集。

如果训练集类数为 k，对于树中的当前非叶子结点 t，假设所包含的样本数为 N(t)，属于主导类 i 的样本数目为 $N_i(t)$，不属于主导类的样本数为 E (t)，显然 E(t) = N (t) - $N_i(t)$。这样，结点 t 中某一样本实例属于类 i 的期望概率为：

$$P_i(t)=\frac{N_i(t)+P_{ai}\cdot m}{N(t)+m}$$

其中，P_{ai} 是第 i 类的先验概率，m 是设置的参数，用来确定先验概率在评估 $P_i(t)$ 中的权值。

如果将当前结点 t 标识为类 i 的叶结点，将引起分类误差。为了简单起见，假设所有的结果类是等概率的，即 $P_{ai} = 1/k$，并且结果类的概率是均匀分布的，即 m = k，i = 1,2,…k。则结点 t 分类误差的期望概率为：

$$EER(t)=min\{1-P_i(t)\}$$

$$=min\{\frac{N(t)-N_i(t)+m\cdot(1-P_{ai})}{N(t)+m}\}$$

在 MEP 方法中，对于决策树中的叶子结点，利用上式计算决策树中叶结点的误差概率；而对于非叶子结点 t，假设 t 的子结点为 $t_1, t_2, \cdots, t_m$，首先计算该结点 t 的误差，称之为静态误差 STE(t)，然后计算 m 个分枝的误差，并且加权相加，权值为每个分枝拥有的训练样本的比例，称之为动态（回溯）误差 DYE (t)。如果 STE (T)≤DYE (T)，则对结点 t 的子树进行剪枝。

后来 Cestnik 和 Bratko 利用贝叶斯方法做出了一些改进，他们将 P_i (t)称作“m-概率估计”。他们认为 m 的值可以根据问题域的不同进行调整。一般说来，m 的值越大，树的剪裁程度就越大。如果 m 趋于无穷大，则 $P_i(t) = P_{ai}$。由于 P_{ai} 是训练集中第 i 类的样本所占的百分比，这样整棵树被剪裁为一个叶子结点，此时拥有最小的误差概率。但事实上，较高的 m 值并不能自动生成规模较小的树，这种非单调性增加了 MEP 的计算复杂度：如果增加 m 的值，裁剪操作必须从原始的树开始进行。

m 值的选取是至关重要的。Cestnik 和 Bratko 建议采用专家系统，根据数据中不同的噪音程度甚至是通过选择生成树为 m 设置适当的值。但大多数情况下，这

样的专家是很难得到的。因此后来的改进版本采用独立的修剪数据集,首先在该修剪数据集上计算分类精度,以得到不同的 m 值;然后根据最低经验误差率选择规模最小的树。

3.3 Rule Post-pruning (规则后修剪)

后剪枝还包括一种改变数据结构的修剪方法,规则后修剪。该方法的一个变体用在 C4.5 中。它通过将决策树转化成规则集来简化决策树。规则后修剪的步骤可以概括为:

(1) 从决策树的根结点到叶子结点的每一条路径创建一条规则,将决策树转化为等价的规则集。

(2) 对规则集进行评价。对于每一条规则,如果省略它后剩余的规则在训练集上的分类错误不会增加,则将其修剪。

(3) 按照修剪过的规则的估计精度对它们进行排序,并按所排的顺序应用这些规则分类后来的实例。

评价规则精度除了使用与训练集不相交的测试集外,还可以仅仅使用训练集对其评价。不过这里使用保守估计 (pessimistic estimate) 来弥补训练数据有利于当前规则的估计偏置。该方法为 C4.5 采用。

将决策树转化为规则集的好处在于:规则集可以区分决策结点使用的不同上下文。对于不同的路径,属性测试的修剪可以不同。如果直接修剪树,要么完全删除决策结点,要么保留其初始状态。

3.4 其他后剪枝方法

还有一些后剪枝方法,如 Pessimistic Error Pruning (PEP)、Cost-Complexity Pruning(CCP)和 Error-Based Pruning(EBP)等。其中 PEP 方法是为克服 REP 方法需要独立测试集的缺点提出的,它是唯一使用自顶向下的策略,在实际应用中表现出了较高的精度。EBP 是 PEP 的改进,改进之处在于在剪枝时,除了删除子树之外,还将该子树嫁接到被删除子树根接点的父结点上。如果训练集较小且要求剪枝精度较高,可首选 PEP。

4 小结

文中分析了决策树产生过度拟合训练样例的原因,并介绍了几种避免过度拟合的剪枝方法。在实际运用中,可根据不同方法的特点具体选择哪种或哪几种策略。避免过度拟合主要涉及决策树的剪枝和精度两方面的问题,应该在这两者之中找到平衡点。在不影响分类正确率或有更高分类正确率的前提下,使优化后的决策树有尽可能小的规模,并能推导出尽可能短的分类规则。

参考文献:

[1] B.Cestnik and I.Bratko, On Estimating Probabilities in Tree Pruning, Machine learning: EWSL-91, Y.Kodra-toff, ed. Lecture Notes in Artificial Intelligence. Berlin: Springer-Verlag, 1991, 482: 138-150.

[2] Elomaa T, Kaariainen M. An analysis of reduced error pruning [J]. Journal of Artificial Intelligence research, 2001,15:163-187.

[3] Floriana Esposito, Donato Malerba, Giovaai Semeraro, A Comparative analysis of methods for Pruning decision Trees [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997,19(5): 476-490.

[4] J.R. Quinlan. Induction of Decision Trees [J]. Machine Learning, 1986 (1): 81-106.

[5] J.R. Quinlan, C4.5: Programs for Machine Learning. San Mateo, Calif.:Morgan Kaufman, 1993.

[6] J.R. Quinlan, Simplifying Decision Trees Int'1 J. Man-machine Studies, 1987, 27: 221 - 234.

[7] L.Breiman, J.Friedman, R. Olshen, and C. Stone, Classification and Regression Trees. Belmont, Calif.: Wadsworth Int'1, 1984.

[8] Mingers, J. An Empirical Comparison of Pruning Methods for Decision-tree Induction. Machine learning, 4(2): 227-243.

[9] Oates T, Jensen D. The Effects of Training Set on decision Tree [A]. Proc of the 14th Int'1 Conf on Machine Learning[C]. Nashville: Morgan Kaufman, 1997: 254-262.

[10] T.Niblett and I.Bratko, Learning Decision Rules in Noisy Domains, Proc. Expert Systems86, Cambridge: Cambridge University Press, 1986.

(责任编辑:刘双琴)

Methods to Avoid Overfitting in Decision Trees

Wang Liming ,Liuhua

(Department of Computer Science and Technology, Wuhan University of Technology, Wuhan 430070)

Abstract: Learning a decision tree through a training set may not lead to the tree with the best generalization performance. The noises in the training set can make the decision tree overfit the training set and reduce the accu-racy of classification. Moreover, the algorithm might be making some decisions toward the leaves based on very little data and may not reflect reliable trends in the training data. Generally, the authors exploit pruning methods to avoid overfitting. There are two methods for pruning, pre-pruning and post-pruning. The paper mainly emphasizes REP,MEP and Rules Post-pruning in term of computational complexity, error estimation and theoretical principle.

Keywords: noise; over-fitting; post-pruning