一种基于遗传算法的优化分类器的方法*

季文赟, 周傲英, 张 亮, 金 文

(复旦大学 计算机科学与工程系,上海 200433)
E-mail: {wyji,ayzhou,zhangl}@fudan.edu.cn
http://www.cs.fudan.edu.cn

摘要: 提出了一种通过遗传算法(GA)对单个分类器进行优化以及对多个分类器进行组合优化的方法.该方法使用叠加(stacking)的策略.经典的叠加策略分为两步,该方法将遗传算法作为叠加策略的第 2 步.实验结果表明,遗传算法可以较好地完成优化任务,同单个分类器比较,它可以提高分类的精度.在对分类器进行组合优化方面,它得到比单个分类器更高的精度以及使分类结果具有更好的可理解性.

关 键 词: 分类;遗传算法;优化;机器学习;数据挖掘;分类规则
中图法分类号: TP18 文献标识码: A

数据分类在统计学、机器学习、神经网络以及专家系统中已经被广泛研究.近来,它又成为数据挖掘中的一个重要研究方面[1].解决数据分类问题已经有很多方法,包括机器学习方法,统计学方法,神经网络方法.对于分类问题,本文采用的方法如下:一种情况是选择一种分类方法,然后用遗传算法对分类结果进行优化.另外一种情况是先并行地使用几种不同的分类方法,然后用遗传算法作为综合方法对分类结果集合进行组合优化.组合优化的依据在于:在一套训练集上使用一种方法就定义了一个唯一的模型,不同的方法也许产生不同的模型,一些方法在某些预测任务上性能很好而在另外一些则较差,它们的预测错误很有可能是分散的,因此可以用遗传算法把这些方法综合起来以提高精度.遗传算法[2]是在很多类型的问题中都很适用的一种优化技术.它可以搜索空间的全局最优解而不必考虑局部解,除了目标函数以外不必具有任何特定的知识,并且有很强的容错性和易于应用.因此,它很容易同其它技术杂交.优化某个算法或者对几个算法进行组合优化具有相同的框架结构,它们的区别在于:在单个方法的优化情况下,当遗传算法在确定其初始种群时,每个个体具有相同的值,而在组合优化时,每个个体具有相应于各个方法得到的值.

已有一些组合的方法,它们得到的结果比使用单个方法更为精确.朴素的方法是多数投票(plurality voting,简称 PV),这种方法可应用于错误是不相关的情况.SCANN(stacking,correspondence analysis,nearest neighbor)[3]可以应用于错误是相关的情况,与以前的组合法相比,它的优点在于:对于性能差的学习模型更不敏感以及具有更高的精度,但是组合后的结果在可理解性上较差.叠加方法(stacking)[4]是一种组合多种模型的综合框架,它可以用来发现并且纠正所使用的学习算法中的系统偏差.本文用叠加方法作为组合的基本框架,在这点上同 SCANN 类似,区别在于本文的第一层归纳(level-1 induction)用遗传算法,其优点是在精度相等的情况下,遗传算法的输入,输出均为规则集,易于理解,在第 1 节中将详细讨论这个框架.

* 收稿日期: 2000-02-15; 修改日期: 2000-07-12
基金项目: 国家自然科学基金资助项目(69933010);国家重点基础研究发展规划 973 资助项目(G1998030414)
作者简介: 季文赟(1971－),男,上海人,博士,主要研究领域为 Web 信息管理,数据挖掘;周傲英(1965－),男,安徽郎溪人,博士,教授,博士生导师,主要研究领域为数据库,数据挖掘,Web 信息管理;张亮(1963－),男,湖北武汉人,博士,副教授,主要研究领域为多媒体技术,支持多媒体应用的数据库技术;金文(1966－),安徽芜湖人,博士生,主要研究领域为数据库,数据挖掘.

1 基于遗传算法的组合方法

1.1 组合方法的基本步骤

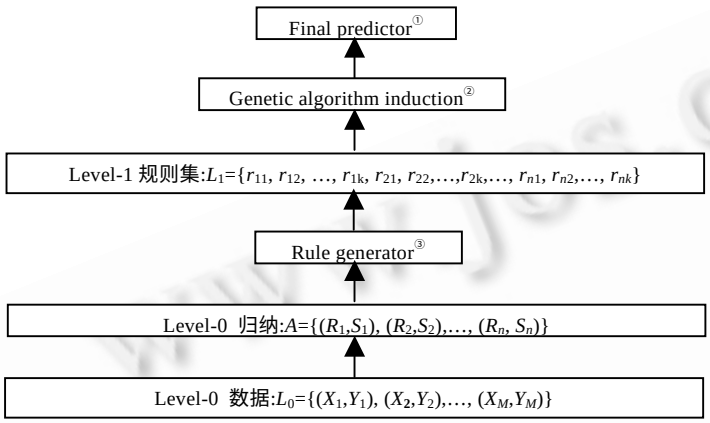
(1) 一个用以学习数据的集合,$L=\{(x_m,y_m),m=1,\ldots,M\}$,其中 x_i 是输入属性,y_i 是输出属性,M 是例子的数目.

(2) 学习算法 A 的集合,设学习算法有 N 个,$A_1,A_2,\ldots,A_N$.算法 A 由它表示的空间 R 和它对于空间的搜索 S 而定义,$A=(R,S)$.

(3) 用遗传算法将各种算法学习的结果综合起来,得到一个综合的解释.

1.2 基于遗传算法的组合方法的框架

本文采用叠加方法来作为组合的基本框架[4],与文献[4]的区别在于使用遗传算法作为第一层的归纳方法.相应的变化是,第一层的输入是规则,而不是数据.方法流程图如图 1 所示.经过第 0 层归纳(level-0 induction)以后,其中每个算法经过训练集的训练,都会生成一套对分类问题的表示,有的以决策树的形式来表达,有的以规则的形式来表达.因为对规则进行编码十分容易,所以本文使用规则形式.方法流程图中的规则产生器就是将各种形式的对于分类问题的表达转化为规则的形式,以此作为 Level-1 层的输入.Level-1 层经过遗传算法的综合,生成最后的分类器,它综合了各个单个分类器的优点,其精确度高于各单个分类器.结果以规则集的形式而存在.



①最后的预测器,②遗传算法归纳,③规则产生器.

Fig. 1 Flowchart of combining method based on genetic algorithm

图1 基于遗传算法的组合方法流程图

1.3 遗传算法优化规则集

(1) 规则编码

根据 Goldberg 的标准 GA,每一个个体代表问题的一组解,因此每一个个体应含有表达全部解的一组规则集.每一条规则相应于一条染色体,规则的多少即是染色体数目的多少.规则由规则的条件部分加上结论部分组成,它有如下的形式:"if($V_{1L}\ \theta_L A_1\ \theta_R V_{1R}$)and($V_{2L}\ \theta_L A_2\ \theta_R V_{2R}$)and …and($V_{nL}\ \theta_L A_n\ \theta_R V_{nR}$) then C_j",其中 A_i 代表某一属性,V_{iL} ,V_{iR} 分别代表左右边界值, θ_L, θ_R 代表左、右关系符,C_j 代表某个类.

(2) 适应函数的确定

本文采用如下信任分配算法:适应函数由两个参数确定,匹配值和不匹配值,当规则条件和结论均匹配,则匹配值增加 1;条件与结论不匹配,则不匹配值增加 1;条件不匹配,则均不变.它们的值由个体经过测试文件测试后决定.

(3) 匹配规则

遗传算法用来产生新的规则,利用一种限制交配策略,只允许同类的规则进行交叉,而且对于同一结论的规则,只允许其条件部分进化,假如条件和规则同时进化,则有可能会导致不收敛.本方法中同类规则的意义指对于选定需要演化的属性,其区间应有重合部分,匹配规则可以由机器根据规则集中属性出现的频度以及它们所处的区间来确定;也可以由人根据经验来确定两套规则集中各规则间的对应关系.

(4) 遗传算法的几个算子

选择算子:使用转盘式选择[2].

交叉算子:个体与个体之间的同类规则的相同基因位进行交叉,即一条规则中的某个属性或某几个属性的

对应的边界值与另外一条规则相应部位进行交叉.

变异算子:对于属性边界上的值进行突变,突变的范围需根据问题的实际情况而确定.

2 实验结果

2.1 数据集

为了与已有的结果对照,我们采用两个数据集:(1) Iris 数据库-数据集来源于美国加州大学 Irvine 分校的机器学习数据库(http://www.sgi.com/Technology/mlc/db/);(2) 来源于 Agrawal 等人[5]的人工生成数据集.它是数据挖掘中的基准数据.

2.2 对于单个算法的优化

2.2.1 Iris 数据集

我们用 C5.0(http://www.cse.unsw.edu.au/~quinlan/)来产生规则集.规则见表 1.

Table 1 C5.0-Based rule set of Iris

表 **1** 基于 C5.0 对于 Iris 的规则集

Rule 1:(覆盖 35 个例子) Petal-Length <= 1.9→class Iris-Setosa
Rule 2:(覆盖 32 个例子) Petal-Length > 1.9∧Petal-Length <= 5∧Petal-Width <= 1.6→class Iris-Versicolor
Rule 3:(覆盖 29 个例子) Petal-Width > 1.6→class Iris-Virginica
Rule 4:(覆盖 28 个例子) Petal-Length > 5→class Iris-Virginica
Default class:Iris-Setosa

优化以后,规则具有同 C5.0 相同的形式,区别在于:对于规则 2,属性 Petal-Length 右边界的值改变为 4.91,对于规则 4,属性左边界的值改变为 4.70,值的改变导致了精度的提高.表 2 是 C5.0 和经过遗传算法优化后的规则在训练集和测试集上的精度.

Table 2 Accuracy of C5.0 rules and GA-optimizing rules on Iris

表 **2** 在 Iris 数据集上 C5.0 和 GA 优化后规则的精度

	C5.0 rule set①		GA rule after optimization②	
	Training set③	Testing set④	Training set	Testing set
Setosa	25/25	25/25	25/25	25/25
Versicolor	24/25	23/25	24/25	23/25
Virginica	25/25	24/25	25/25	25/25
总体	74/75 98.67%	72/75 96%	74/75 98.67%	73/75 97.33%

①规则集,②优化后规则集,③训练集,④测试集.

表 2 的结果表明,遗传算法优化后的精度好于单个算法的精度.需要注意的是,在测试数据中,Iris-Versicolor 类的两个例子被误分为 Iris-Virginia 类,其原因在于这两个例子中的属性值极为相似,因此,用基于单属性来生成规则集(决策树算法把每次把一个属性作为分枝的标准)的算法是无法区分的.只有通过属性值的组合才可以把它们进行区分[6].但是采用属性值组合的算法,其产生的结果可理解性差.在数据挖掘应用中,由于输入数据量大,一般希望规则集可以容易地转化为 SQL 查询形式以利于查询和容易理解.而由属性值组合而形成的规则集难以进行这种转化,所以属性值组合的算法不是最佳算法,最佳算法应是那些同时兼顾精度和可理解性的算法.

2.2.2 人工合成数据集

本文用分类函数 2.函数定义见文献[5].

本文用 C4.5(ID3 的一种变体)来产生规则集,它产生 18 个规则.其中,8 条规则定义了 Group A 的条件,其余都是 Group B.规则集见文献[7].

根据属性的频率直方图,我们选择 age 和 salary 属性用遗传算法进行优化.对于文献[7]中的规则 6,我们可以得到新的规则形式:(25100<salary<74800)∧(age>60.8),这个结果是用遗传算法经过 20 世纪 70 年代后得到的结果(注意:在规则 6 中,属性 salary 偏离它的正确值很远).对于规则 20,我们得到新形式的规则:

(74800<salary<=124000)∧(39<age<=58.7),对于规则 10,我们得到(50100<salary<100200)∧(age<39.8).对照函数 2,优化后的边界值比 C4.5 得出的规则集的边界值更接近真实值,其它规则经过优化也有类似结果,因此精度得到很大提高.由此说明即便规则的边界值偏离了其正确的值(甚至是很多时),遗传算法也可以通过演化得到其正确值.

2.3 多个算法的组合优化

我们选择神经网络(一种连接主义者的方法)和 C5.0(一种符号主义者的方法)作为两种不同的算法.数据集是 Iris.根据文献[8],Iris 数据集基于神经网络(NN)规则集如下:

rule 1:Petal-Length<=1.9→Iris-Setosa

rule 2:if Petal-Length<=4.9∧Petal-Width<=1.6 → Iris-Versicolor

Default class:Iris-virginica

基于 C5.0 的规则集见表 1.

优化后,我们得到两套规则集.一套基于神经网络,它的形式没有改变,另外一套基于 C5.0,它列于表 3,由表 3 可以看出,在边界上其值发生改变,新的规则集其精度比原先提高,精度见表 4.在优化前,NN 规则集有较高的精度,但是可理解性较差,因为从 NN 规则,我们不知道 iris-virginica 的特性,只是知道它是缺省的类别;而 C5.0 形成的规则集可理解性好,但是精度差一些.优化后,我们得到了具有高精度和高可理解度的规则集.

Table 3 C5.0-Based rule set after optimization

表 **3** 优化后基于 C5.0 的规则集

Rule 1:(cover 35) Petal-Length <= 1.9→class Iris-setosa
Rule 2:(cover 32) Petal-Length>1.9∧Petal-Length<=4.95∧Petal-Width<=1.6→class Iris- Versicolor
Rule 3:(cover 29) ∧ Petal-Width>1.6→class Iris-Virginica
Rule 4:(cover 28) ∧ Petal-Length>4.95→class Iris-Virginica
Default class:Iris-Setosa

Table 4 Accuracy of multiple algorithms after optimization

表 **4** 多个算法优化后的精度

	Neural network rule set①		C5.0 rule set②		rule set after optimization③	
	Training set④	Testing set⑤	Training set	Testing set	Training set	Testing set
Setosa	25/25	25/25	25/25	25/25	25/25	25/25
Versicolor	24/25	23/25	24/25	23/25	24/25	23/25
Virginica	25/25	25/25	25/25	24/25	25/25	25/25
总体	74/75	73/75	74/75	72/75	74/75	73/75
	98.67%	97.33%	98.67%	96%	98.67%	97.33%

①NN 规则集,②C5.0 规则集,③GA 优化后规则集,④训练集,⑤测试集.

3 结 论

本文将分类技术与进化方法结合起来,把遗传算法作为一种优化技术应用于单个和多个分类器上.实验结果表明,这样的结合可以得到较好的分类效果,得到了比单个算法更精确或更易理解的结果.分类技术处理原始训练数据,得到初步的结果规则集.遗传算法作为不同规则的组合器,从一个更高层次对规则集进行优化,它是通过优化规则的条件部分的边界值来提高预测精度.由于它可以搜索解空间的全局最优解,因此当演化结束后,边界值就可以到达它的最佳值从而反映分类问题的实际解.它又是鲁棒的,尽管边界的值一开始可以偏离其正确位置很远,遗传算法也可以找到其正确位置.同时,基于遗传算法的组合方法又具有可扩展性,一旦有了新的分类算法,可以把算法得到的分类结果转化为规则集,把新的规则集作为遗传算法的输入,同已有的规则集一起进行演化,这样就有可能得到更好的结果,从而实现增量挖掘的功能.遗传算法的限制在于同时演化的参数不能太多.当参数太多时,它就不能演化到最佳值.解决的方法是可以逐步进行演化,每次演化选取几个参数.另外一个缺点是训练时间,训练时间主要消耗在扫描测试集.解决的方法是在算法中设置阈值,在扫描测试集数据时,

当群体中的个体与测试例子的不匹配数达到某一阈值时,便淘汰该个体,这样不适合环境的个体在扫描了几个例子后便被淘汰,不必扫描整个数据集,这对于大规模的数据集极为适合.

References:

[1] Fayyad, U.M, Piatetsky-Shapiro, G., Smyth, P., *et al*. Advances in Knowledge Discovery and Data Mining. Cambridge, MA: AAAI/MIT Press, 1996.

[2] Goldberg, D.E. Genetic Algorithms in Search, Optimization, and Machine Learning. New York: Addison-Wesley, 1989.

[3] Merz, C.J. Using correspondence analysis to combine classifiers. Machine Learning, 1999,36(1~2):33～58.

[4] Wolpert, D.H. Stacked generalization. Neural Networks, 1992,5(2):241～259.

[5] Agrawal, R., Imielinski, T., Swami, A. Database mining: a performance perspective. IEEE Transactions on Knowledge and Data Engineering, 1993,5(6):914～925.

[6] Setiono, R. Techniques for extracting rules from artificial neural networks. In: Plenary Lecture Presented at the 5th International Conference on Soft Computing and Information Systems. Iizuka, Japan, 1998. http://www.comp.nus.edu.sg/~rudys/ publications. html.

[7] Lu, H., Setiono, R., Liu, H. NeuroRule: a connectionist approach to data mining. In: Umeshwar, D., Peter, M. D., Shojiro, N., eds. Proceedings of the 21st VLDB Conference. Zürich, Switzerland: Morgan Kaufmann, 1995. 478～489.

[8] Setiono, R., Liu, H. Understanding neural networks via rule extraction. In: Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI 95). Montreal: Morgan Kaufmann, 1995. 480～485.http://www.informatik.uni-trier.de/~ley/db/conf/ijcai/ijcai95.html#Setion0L95

A Method to Optimize Classifiers by Using Genetic Algorithms *

JI Wen-yun, ZHOU Ao-ying, ZHANG Liang, JIN Wen

(*Department of Computer Science and Engineering, Fudan University, Shanghai* 200433, *China*)

E-mail: {wyji,ayzhou,zhangl}@fudan.edu.cn

http://www.cs.fudan.edu.cn

Abstract: This paper focuses on methods of optimizing a single classifier and combining multiple classifiers by genetic algorithms (GA). The method uses the strategies of stacking. There are two steps in classical strategies of stacking, and GA is used as the second step in the method. Experimental results show that it performs well on the task of optimization. Comparing with the single algorithm, it enhances the precision. In task of combining optimization, it can obtain more understandable result than constituent learners.

Key words: classification; genetic algorithm; optimization; machine learning; data mining; classification rules

* Received February 15, 2000; accepted July 12, 2000

Supported by the National Natural Science Foundation of China under Grant No.69933010; the National Grand Fundamental Research 973 Program of China under Grant No.G1998030414