

基于子集类蚁群模型的属性相对约简算法^{*})

梁云川¹ 李德玉²

(忻州师范学院计算机系 忻州 034000)¹ (山西大学计算机与信息技术学院 太原 030006)²

摘要 粗糙集属性约简是一个典型的 NP-hard 问题。提出了一种基于子集类蚁群模型的属性相对约简算法,该算法采用转移概率对每个属性随机搜索,直到获得一个分类能力与决策属性分类能力一致的属性子集。提出的基于信息素变异的蚁群算法,不仅提高了解的质量,而且有效避免了早熟收敛。106 组病例数据的实验结果表明,该算法能够发现较好的决策表相对约简与决策规则。

关键词 粗糙集,子集类蚁群算法,属性约简,信息素变异

Algorithm for Relative Reduction Based on Subset Ant Colony Optimization Model

LIANG Yun-chuan¹ LI De-yu²

(Computer Department of Xinzhou Teacher University, Xinzhou 034000, China)¹

(School of Computer and Information Technology, Shanxi University, Taiyuan 030006, China)²

Abstract Reduction in rough set theory is a typical NP-hard problem. A new algorithm for relative reduction which is based on Subset ant colony algorithm was proposed. By using the strategy of diversion probability, every attribute is randomly searched by ants until an attribute subset is obtained that has the same discerning capability with the decision attribute. The presented ACO algorithm based on pheromone mutation not only improves the solution quality but also avoids stagnation. The experiment with 106 illness cases shows that the algorithm can discover better relative reduction and decision rules.

Keywords Rough set, Subset ant colony algorithm, Reduction, Pheromone mutation

1 引言

Z. Pawlak 于 1982 年提出的粗糙集理论在人工智能和数据分析的许多领域得到了广泛应用。属性约简是粗糙集理论中非常重要的一个研究课题,苗夺谦等提出了一个基于信息熵的启发式约简算法^[1],梁吉业等提出了粗糙集数据分析中的度量和约简算法^[2,3],邵明文、张文修、吴伟志提出了信息系统知识约简便算法^[5],李德玉等提出了信息系统中约简的极大相容块方法^[6],王国胤等提出了基于条件信息熵的决策表约简算法^[4,7]。然而,Wong, S. K. M 和 Ziarko, W 证明^[8],求属性约简的复杂性是随着决策表的规模增长呈指数增长的,是一个典型的 NP-Hard 问题,而上述研究成果是针对属性个数较少的信息表或决策表。蚁群模型是根据蚁群觅食活动的规律而建立的一个利用群体智能进行优化搜索的模型,相对其他的各类启发式搜索算法,蚁群算法具有明显的优越性,在一系列 NP-完全组合优化问题的求解中取得了卓有成效的结果。本文提出了一种基于子集类蚁群算法的属性约简算法,实验表明该算法能够快速求出大规模信息表的全局最优约简。

2 粗糙集属性相对约简问题描述

下面给出属性约简问题的数学描述:

给定一个含 n 个条件属性的决策表,定义一个 n 阶完全无向图 $G(V, E)$,其中 V 是顶点集,即 $\{1, 2, \dots, n\}$,用来表示

决策表的属性列, E 是图的边集。这样,决策表的相对约简问题就是在无向图 G 中找到具有最少顶点数的子集 R ,使该顶点子集的分类能力与给定数据集决策属性的分类能力一致。

3 序列类问题和子集类问题

在序列类问题中,假设节点集合 $S = \langle a_1, a_2, \dots, a_j, a_{j+1}, \dots, a_n \rangle$,序列 $S' = \langle a_1, a_2, \dots, a_j \rangle$ 表示问题的一个部分解,集合 $R = S - S' = \langle a_{j+1}, a_{j+2}, \dots, a_n \rangle$ 表示剩下的节点,这些节点必须被逐步选择加入到部分解 S' 中,从而构成问题的一个可行解。从集合 R 中选择下一个节点 a_{j+1} 的概率 $P_{a_j, a_{j+1}}^t(t)$ 与部分解序列 S' 中最后一个元素 a_j 密切相关,即蚂蚁根据边 (a_j, a_{j+1}) 上的信息素 $\tau_{a_j, a_{j+1}}$ 和局部启发信息 $\eta_{a_j, a_{j+1}}$ 的共同指引,从集合 R 中选择一个节点 a_{j+1} 加入部分解序列 S' 中,直至集合 R 中所有节点都被加入到 S' 中为止,序列 S' 中各节点的序列是问题答案的关键。

在子集类问题中并不关心部分解中节点的顺序,部分解表示为集合 $S' = \langle a_1, a_2, \dots, a_j \rangle$,集合 $R = \langle a_{j+1}, a_{j+2}, \dots, a_m \rangle$ 表示剩下的可以被逐步选择加入到部分解的节点,部分解 S' 是顶点集合而并不是序列,集合 R 中的元素并不是都将被选择, R 是在满足相应问题的约束条件下,可以被加入部分解集合。最近加入到部分解集合 S' 的元素 a_j 不影响选择下一个元素的过程。从集合 R 中选择下一个节点 a_{j+1} 的概率 $P_{a_j, a_{j+1}}^t(t)$ 由其自身的信息素量 $\tau_{a_j, a_{j+1}}$ 和局部启发信息 $\eta_{a_j, a_{j+1}}$ 决定。顺序类问题的解长度固定,只是求已知元素的一个排

^{*}) 基金项目:国家自然科学基金项目(No. 60573074),忻州师范学院基金资助项目。梁云川 硕士,讲师,研究方向为智能优化计算与数据挖掘;李德玉 博导,教授,研究方向为计算智能与数据挖掘。

列顺序,而子集类问题的解长度不固定,求的是其元素的一个满足特定条件的子集。

4 适应度函数

约简的好坏评判标准包括两个方面,一是约简的分类质量,分类质量越高该约简越优;二是约简的长度,即约简中的属性个数,长度越短,约简越优。据此,本文提出如下的约简适应度函数:

$$Best(R_i) = e^{|\underline{R}(R_i)|/|U|} / L_{R_i} \quad (1)$$

其中, R_i 是一个约简, $|\underline{R}(R_i)|/|U|$ 是约简 R_i 的分类质量, L_{R_i} 是约简 R_i 的长度。从公式(1)可以看出约简的适应度与其分类质量成正比,与约简的长度成反比。

5 基于子集类蚁群算法的粗糙集属性约简算法框架

传统蚁群算法更适用于解决序列表问题,它将信息素留在边上,由边 (a_j, a_{j+1}) 上的信息素和局部启发信息确定选择从顶点 a_j , 通过路径 (a_j, a_{j+1}) 转移到顶点 a_{j+1} 的概率^[9]。而子集类问题的蚁群算法将信息素留在顶点上,由每个顶点的信息素和局部启发信息确定选择顶点加入子集(部分解)的概率。

定义顶点 a_j 在时刻 $t+1$ 的信息素更新策略公式为:

$$\tau_{a_j}(t+1) = (1-\rho)\tau_{a_j}(t) + \Delta\tau_{a_j}(t, t+1) \quad (2)$$

其中,

$$\Delta\tau_{a_j}(t, t+1) = \sum_{k=1}^m \Delta\tau_{a_j}^k(t, t+1) \quad (3)$$

即,信息素增加量 $\Delta\tau_{a_j}(t, t+1)$ 为从 t 到 $t+1$ 时刻所有蚂蚁在顶点 a_j 留下的信息素之和;其中 m 表示 t 到 $t+1$ 时刻经过顶点 a_j 的蚂蚁数量。

第 k 只蚂蚁在顶点 a_j 留下的信息素 $\Delta\tau_{a_j}^k(t, t+1)$ 可以定义如下:

$$\Delta\tau_{a_j}^k(t, t+1) = \begin{cases} F(L_k), & \text{第 } k \text{ 只蚂蚁选择 } a_j \\ 0, & \text{否则} \end{cases} \quad (4)$$

函数 F 依据具体问题定义,表示第 k 只蚂蚁在顶点 a_j 留下的信息素的量,其中 $F(L_k) = Q/L_k$, 第 k 只蚂蚁构造的解的质量即 L_k 为目标函数的值, Q 为常数,可根据经验指定值,这里, L_k 为第 k 只蚂蚁构造的约简的长度,即约简中的属性个数。

假设节点集合 $S = \langle a_1, a_2, \dots, a_j, a_{j+1}, \dots, a_n \rangle$, 集合 $S' = \langle a_1, a_2, \dots, a_{j-1} \rangle$ 表示问题在 t 时刻蚂蚁 k 构造的一个部分解,则 t 时刻蚂蚁 k 选择节点 a_j 加入部分解的概率 $p_{a_j}^k(t)$ 的定义如下:

$$p_{a_j}^k(t) = \begin{cases} \frac{[\tau_{a_j}(t)]^\alpha [\eta_{a_j}]^\beta}{\sum_{i \in allowed_k(t)} [\tau_i(t)]^\alpha [\eta_i]^\beta}, & a_j \in allowed_k(t) \\ 0, & \text{否则} \end{cases} \quad (5)$$

其中, $allowed_k(t)$ 为候选顶点集合。因此, $\tau_{a_j}(t)$ 和 η_{a_j} 越大, 顶点 a_j 被选择加入部分解 S' 的概率越大。这里 $\eta_{a_j} = 1/(1 - |POS_{a_j}(D)|/|POS_D(D)|)$, 即 η_{a_j} 的值与结论属性 D 对属性 a_j 的依赖度成正比, 结论属性 D 对属性 a_j 的依赖度越大, 表明属性 a_j 的分类能力与结论属性 D 的分类能力越接近, 其局部启发信息 η_{a_j} 就越大。初始时, 每个属性均无蚂蚁走过, 其上的信息素浓度 $\tau_{a_j}(t)$ 均相等, 所以最初蚂蚁搜索 η_{a_j} 大的属性的概率也越大, 随着搜索次数的增加, 蚂蚁对能够构成约简的属性的信息素不断增加, 局部启发信息 η_{a_j} 值对蚂蚁的转移方向的影响变得越来越不明显。

粗糙集属性约简与基本子集类问题稍有不同, 即子集类问题当一只蚂蚁选择了顶点 a_j 将其加入部分解集后则可肯定该顶点 a_j 在当前蚂蚁求得的解集中必定包含该顶点, 而粗糙集属性约简问题当顶点 a_j 被加入部分解集后也不一定就是最终要求的解集, 还需对求得的解集进行剪枝, 所以只有剪枝后的解集才是该蚂蚁最终求得的解集, 若该解集为当前最优, 则应在该解集各顶点增加信息素, 若该解集不是当前最优解, 则不应增加该解集各顶点的信息素量。所以, 信息素增加量 $\Delta\tau_{a_j}(t, t+1)$ 可以定义如下:

$$\Delta\tau_{a_j}^k(t, t+1) = \begin{cases} Q/L_k & \text{第 } k \text{ 只蚂蚁选择了顶点 } i, \\ & \text{且为当前最优约简} \\ 0, & \text{否则} \end{cases} \quad (6)$$

其中, $\Delta\tau_{a_j}(t, t+1)$ 表示第 k 只蚂蚁在顶点 a_j 留下的信息素的量。第 k 只蚂蚁构造的解的质量即 L_k 为当前最优约简的长度, 即最优约简的属性个数。只有当找到比当前最优解更好的解时, 找到的新解各顶点的信息素才会增加, 否则 $\Delta\tau_{a_j}(t, t+1) = 0$, 这时各顶点信息素根据公式(1)进行更新, 使信息素进行挥发, 这样可以进一步加快收敛, 靠近核属性。

针对蚁群算法本身容易陷入局部寻优的缺陷, 提出一种信息素变异的蚁群算法改进策略。该策略更改了传统蚁群算法一成不变的信息素更新策略, 使信息素更新策略在算法运行过程中动态调整, 其调整后的信息素更新策略为:

$$\tau_{a_j}(t+1) = \begin{cases} (1-\rho)\tau_{a_j}(t) + \Delta\tau_{a_j}(t, t+1) & \text{找到新解} \\ (1-\rho)\tau_{a_j}(t) & \text{若顶点 } a_j \text{ 在剪枝时被删除} \\ \text{初始值} & \text{未找到新解且信息素很低} \end{cases} \quad (7)$$

通过该信息素更新策略的修改, 当蚁群较长时间不能发现新解时, 将信息素很低的属性节点的信息素恢复初值, 这样在迭代后期本来不可能被选择的属性节点的偏移概率增大, 使其有可能被选择, 这样蚁群的搜索过程可以脱离陷入局部寻优的困境, 使其找到全局最优解的可能性增加。

根据上述讨论, 基于蚁群算法的属性相对约简算法描述如下:

Procedure 基于蚁群算法的粗糙集属性约简

设置参数, 初始化信息素的分布和各个属性的依赖度

while 不满足结束条件 do

for $m=1$ 到 蚁群中的蚂蚁数 do

while 没有找到约简 do

根据转移概率公式(5)选择 1 个顶点;

加入部分解子集;

更新候选顶点集合;

end of while

对找到的约简 R_i 进行剪枝;

求找到的约简的适应度 $Best(R_i)$;

If $Best(R_i) > NowBest(R)$ Then

记录当前找到的约简 R_i ;

$NowBest(R) = Best(R_i)$;

End If

根据公式(7)对信息素进行全局更新;

end of for

end of while

输出找到的最优解

end of procedure

6 实验

在下面的分析中, 实验所用的实例数据为从山西省吕梁

市人民医院心脏外科获得风湿性二尖瓣狭窄疾病病例资料 106 份,建立风湿性二尖瓣狭窄疾病知识库,知识表达系统见表 1 所示,得到 15 种属性。该知识表达系统无重复实例,定义前 14 种(序号除外)为条件属性,最后一种(患病否)为决策属性。可转化为决策表,见表 2 所示。

经过多次对各个参数研究发现,实验参数取为:蚂蚁数量取 14 只,最大迭代次数为 84 次,最小约简质量 $\gamma_R(F) = 0.78, \alpha=2, \beta=3, \rho=0.2$ 效果为最佳^[10]。但计算过程中发现,运算 20 次,总有 4~5 次不能找到最优解,甚至距离最优

解比较远。分析其中一次未能找到最优解的运行结果图,如图 1 所示。在左图中发现,算法搜索初期,蚂蚁搜索的顶点较均匀,每个属性节点都有访问记录,但算法在后期访问的属性节点却非常集中,除了属性 1,2,4,6,7,9,11,12,14 外,剩余的属性根本没有访问一次;从右图可以看出,算法搜索到的当前最优解会长期停滞不动,到达最大迭代次数后,找到的最佳适应度为 0.2477,仍然没有找到全局最优解(适应度为 0.2549)。分析后,得出算法容易陷入局部寻优的结论,这也是蚁群算法本身最大的缺陷。

表 1 风湿性二尖瓣狭窄疾病知识库

序号	呼吸 困难	心悸	咳嗽	咯血	胸痛	二尖瓣 面容	压迫 症状	心尖区舒 张期杂音	心尖区第一 心音亢进	肺 A 区每二心 音亢强和分裂	X 线呈 梨形	心电图二 尖瓣型 P 波	右心功 能不全	风湿病史	患病否
1	端坐	有	无	后期	无	无	无	有	有	有	有	有	无	无	否
2	端坐	有	夜间	后期	无	有	无	有	无	无	有	有	无	无	是
3	端坐	有	无	后期	无	无	无	有	无	有	无	有	无	无	否
4	无	有	夜间	后期	无	有	无	有	有	无	有	无	无	无	是
5	无	无	无	有	无	无	无	无	有	有	无	有	无	有	是
...															

表 2 由表 1 转化来的决策表

	a	b	c	d	e	f	g	H	i	j	k	l	m	n	o
r1	1	1	2	4	2	1	2	1	2	1	1	1	2	2	0
r2	1	1	1	4	2	1	2	1	2	2	1	1	2	2	1
r3	1	1	2	4	2	2	2	1	2	1	2	1	2	2	0
r4	2	1	1	4	2	1	2	1	1	2	1	2	2	2	1
r5	2	2	2	1	2	2	2	2	1	1	2	1	2	1	1
...															

其中,属性:a—呼吸困难;b—心悸;c—咳嗽;... m—风湿病史;o—患病否;条件属性记录值:1—(a)端坐;(b)有;(c)夜间;2—无;3—初期;4—后期;决策属性记录值:1—患病;0—未患病。

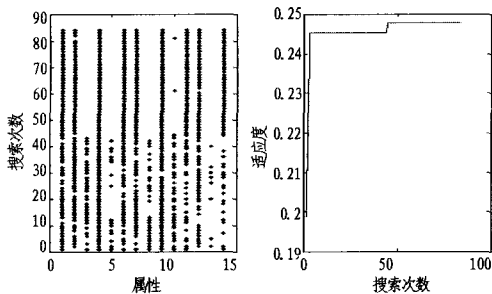


图 1 一次未找到全局最优解的结果图

图 2 为采用信息素变异更新策略的一次运行结果图,从左图可以看出,蚁群在迭代初期和后期,对各属性顶点均有访问,只是在最终找到的最优约简属性顶点 1,2,3,4,7,9,11,12,14 上搜索的密度要大,从右图可以看出在搜索过程中,适应度值以阶梯形状逐渐向最优解靠拢,在未找到最优解之前,不会很长时间停留在某个适应度值。

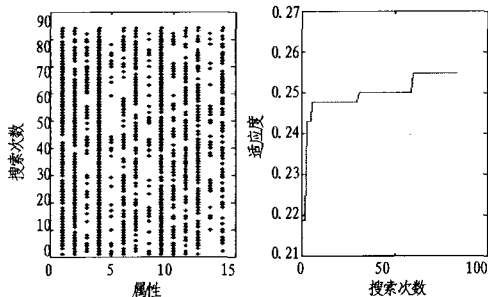


图 2 采用信息素变异更新策略的一次运行结果图

经实验统计,采用传统蚁群算法进行粗糙集属性约简,能找到最优约简的概率约为 80%,而采用信息素变异更新策略

的蚁群算法能找到最优约简的概率约为 95%以上。

7 实验结果

本算法在 Matlab7.0 环境下进行运行,输出属性约简集。对 106 份风湿性二尖瓣狭窄疾病病例资料进行分析,要求最小约简质量为 $\gamma_R(F) = 0.78$,约简的适应度函数定义为公式(1),将原来的 14 个诊断参数减少为 9 个,即 1,2,3,4,7,9,11,12,14,分别表示呼吸困难、心悸、咳嗽、咯血、压迫症状、心尖区第一心音亢进、X 线呈梨形、心电图二尖瓣型 P 波和风湿病史。与临床所得数据信息作比较,数据约简量为 35.71%,诊断正确率为 83.02%,其适应度为 0.2549。若要求最小约简质量为 $\gamma_R(F) = 0.88$,约简的适应度函数定义为公式(1),将原来的 14 个诊断参数减少为 11 个,即 1,2,3,4,6,7,9,10,11,12,14,分别表示呼吸困难、心悸、咳嗽、咯血、二尖瓣面容、压迫症状、心尖区第一心音亢进、肺 A 区每二心音亢强和分裂、X 线呈梨形、心电图二尖瓣型 P 波和风湿病史。与临床所得数据信息作比较,数据约简量为 21.43%,诊断正确率为 88.68%,其适应度为 0.2207。在不明显降低诊断正确率的条件下,提高了推理程序的运算速度,也为实际的推理解释打下了基础。

需要说明的是,论文所用算法在处理大数据量、属性较多信息系统时有明显优势,但如果数据信息量非常小,还是传统方法比较简单。

参考文献

- [1] 苗夺谦,胡桂荣.知识约简的一个启发式算法.计算机研究与发展,1999,36(6):681-684
- [2] Liang J Y, Xu Z B. The algorithm on knowledge reduction in incomplete information systems. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 2002, 10 (1): 95-103
- [3] 梁吉业,曲开社,徐宗本.信息系统的属性约简.系统工程理论与实践,2001,21(12):76-80
- [4] 王国胤,于洪,杨大春.基于条件信息熵的决策表约简.计算机学报,2002,25(7):759-766
- [5] 邵明文,张文修,吴伟志.信息系统知识约简简便算法[J].计算机科学,2003,30(11):25-28
- [6] Leung Y, Li De-yu. Maximal consistent block technique for rule

[7] 胡峰,代劲,王国胤. 一种决策表增量属性约简算法[J]. 控制与决策, 2007, 32(3): 268-277

[8] Liu H, Yu L. Feature Selection for Data Mining [EB/OL]. http://www.public.asu.edu/~huanliu/feature_selection.html,

[9] Dorigo M L M. Ant colony system: a cooperative learning approach to the traveling salesman problem[J]. IEEE Transactions on Evolutionary Computing, 1997, 1(1): 53-56

[10] 吴春明, 陈治, 姜明. 蚁群算法中系统初始化及系统参数的研究[J]. 电子学报, 2006, 34(8): 1530-1533

(上接第 138 页)

同特征的样本,过程神经元对样本的特征抽取能力大大高于传统神经元,这一结果也从侧面验证了上述结论。

2.2 隐节点数

对于过程神经网络来说,过程神经元隐层节点数的选取无疑是十分重要的。过程神经元隐节点数选取过小,将导致对样本过程模式特征提取能力太弱,训练无法收敛;节点数选取过多,会导致两个问题:一是网络的计算量增大,由于过程神经网络自身特点,已产生较大的计算量,再增加网络计算量,导致的可能结果是,对整个过程神经网络性能的整体影响;二是由于抽取能力太强而产生的“过拟合”现象,而对于过程神经网络而言,“过拟合”现象的出现,会导致较传统神经网络而言泛化误差的迅速降低。其原因在于:过程神经网络输入时变函数的产生,是由多个几何点式的常量经函数拟合得到,而拟合的过程不可避免地存在误差,因此,输入时变函数,在包含优于传统神经网络常量输入的大量有用信息特征的同时,无疑也增加了噪声,而噪声的产生多少与具体的函数拟合方法和精度有关,但从噪声的量上看,过程神经网络输入函数所包含的噪声量无疑应大于传统神经网络常量输入的情况,因此,当过程神经网络一旦产生“过拟合”现象时,会出现随着训练误差的逐渐变小,测试误差迅速增大的情况,因为过程神经元隐层所具有的对过程模式和空间累积效应的双重特征提取功能,导致网络对输入函数中所包含的大量噪声的快速提取,出现了网络泛化性能的快速下降。

表 3 测试误差对比

网络序号	隐层数	隐层节点数	最小误差要求	测试误差
1#	2	4,4	0.10	0.0386
2#	2	6,6	0.10	0.0675
3#	2	6,6	0.05	0.3067

从表 3(各实验条件同表 1)中看到,当网络出现“过拟合”现象时,随着隐层节点数增加和最小误差要求的提高,网络泛化能力迅速下降。

3 前馈过程神经网络网络结构构造算法

基于上述讨论结论,给出前馈过程神经网络网络结构构造算法:

- 1)选择单过程神经元隐层;
- 2)选择足够少的过程神经元隐节点数;
- 3)训练网络;
- 4)训练误差是否满足要求,如果满足转 6),否则转 5);
- 5)增加隐节点数或增加一个一般神经元隐层,转 3);
- 6)挑选测试集样本;
- 7)测试误差是否满足要求,如果满足转 9),否则转 8);
- 8)增加样本训练集的数量,转 3);
- 9)结束。

其中步骤 5)可选择性地增加过程神经元隐节点数或增加一个一般神经元隐层。增加一个过程神经元隐节点出现“过拟合”现象时,可采用增加若干个一般神经元隐节点的方法避免“过拟合”现象。算法中,要求训练集与测试集的数据应能够很好地表示输入域的整体情况,测试集与训练集应相

互独立。

4 应用实例

在林业木材科学研究中,木材生长密度是进行木材合理利用决策的重要依据。选取某一生长条件下的特定树种作为样本,对木材生长密度进行等距采样,根据木材林地条件对数据进行合理的早晚材划分,得到木材生长密度的时间序列,通过动态调整网络过程神经元的互连权函数和一般神经元的权值,拟合时间序列数据内在的映射关系,实现对木材生长密度的预测。

选取具有代表性的、共有 30 个年轮的某地区木材生长密度数据,共组成 18 个样本。过程神经网络结构构造过程如下:

1)初始化时,网络层数为 1,即只有 1 个过程神经元隐层,过程神经元隐层节点数为 2,学习精度 $\epsilon < 0.03$,测试误差限用平方和误差 sse 表示,取值为 0.005。以其中 14 个样本作为训练集,另外 3 个样本作为测试集,训练 132 次收敛,利用 4 测试样本预测后的平方和误差 sse 为 $0.0077 > 0.005$,测试误差不满足要求。

2)合并测试集中的一个样本到样本集中,仍然使用上述网络结构训练,经 173 代进化后收敛,对 3 个测试样本进行生长密度预测结果见表 4,平均绝对误差为 0.0134,平均相对误差为 1.51%,平方和误差 sse 为 $0.00073 < 0.005$,测试误差满足要求,网络构造和训练结束。

表 4 测试样本预测结果

测试集样本序号	1	2	3
实测密度值	0.8800	0.8460	0.8960
预测密度值	0.8646	0.8488	0.8740
绝对误差	0.0154	0.0028	0.0220
相对误差(%)	1.7500	0.3310	2.4550

结束语 对过程神经网络的结构与泛化能力进行了分析,得出了如下结论:其过程神经元隐层起主要作用,一般神经元隐层并非是必需的,对于相同特征的样本,过程神经元对样本的特征抽取能力大大高于传统神经元;给出了一个基于提高泛化能力的前馈过程神经网络网络结构构造算法,并应用一个实例验证了其有效性。同时与传统神经网络类似,样本的数量、质量也不可避免地对过程神经网络的泛化能力产生影响,除此以外,过程神经网络中时变输入函数的拟合精度,基于正交基展开的过程神经网络中正交基的选取,对过程神经网络的泛化能力都有一定的影响,而对于影响过程神经网络泛化能力的上述因素的研究,还有待于进一步深入探讨。

参 考 文 献

[1] 何新贵,梁久祯. 过程神经网络的若干理论问题[J]. 中国工程科学, 2000, 12(2): 40-44

[2] 何新贵,梁久祯,许少华. 过程神经网络的训练及其应用[J]. 中国工程科学, 2001, 4(3): 31-35

[3] 焦李成. 神经网络系统理论[M]. 西安:西安电子科技大学出版社, 1990

[4] Hornik, Stinchcombe K M, White H. Multi-Layered Feed-Forward Neural Networks are Universal Approximations [J]. Neural networks, 1990, (2): 359-366