

文章编号:1001-9081(2005)07-1661-02

基于免疫遗传算法的多层前向神经网络设计

罗 菲,何明一

(西北工业大学 电子信息学院,陕西 西安 710072)

(luo_fei@163.com)

摘 要:利用一种基于免疫功能的遗传算法,设计多层前向神经网络,用于实现多层前向神经网络结构的确定和权值空间的搜索。仿真实验结果显示该算法具有比遗传算法和动量 BP 算法更好的全局收敛性和快速学习网络权值的能力。

关键词:多层前向神经网络;免疫遗传算法;抗体;浓度

中图分类号:TP183 **文献标识码:**A

Multilayer feed-forward neural networks based on immune genetic algorithm

LUO Fei, HE Ming-yi

(Electronic and Information School, Northwestern Polytechnical University, Xi'an Shaanxi 710072, China)

Abstract: An optimizing method based on immune genetic algorithm was presented for designing multilayer feed-forward neural network. This algorithm could decide the structure of the multilayer feed-forward neural networks and search the proper weight of this network. The simulation experiments show that this algorithm has the better ability of convergent on whole solution space and the capacity of fast learning than genetic algorithm and momentum BP algorithm.

Key words: multilayer feed-forward neural networks; immune genetic algorithm; antibody; density

0 引言

以 MLP(多层感知机)网络为代表的多层前馈神经网络^[1],因其优良的非线性逼近性能,其应用已受到越来越多领域的关注。它通过监督学习的方式实现模式识别和函数逼近的功能。以一个简单的两层前向神经网络为例,第一层为输入层;第二层是输出层。网络的输入向量为 $X = (x_1, x_2, \dots, x_m)$, W^1 为输入向量与第一层之间的连接权值矩阵, W^2 为第一层与第二层之间的连接权值矩阵, $f(\cdot)$ 是输入层神经元的活动函数, $g(\cdot)$ 是输出层神经元的活动函数,则输出向量 $Y = (y_1, y_2, \dots, y_n)$ 为:

$$Y = g(W^2 f(W^1 X)) \quad (1)$$

假设网络的期望输出为向量 $T = (t_1, t_2, \dots, t_n)$, 则定义网络输出端的误差代价函数为:

$$E = \frac{1}{2} \sum_i (t_i - y_i)^2 \quad (2)$$

网络学习的目的是通过调节网络权重 W^1 和 W^2 ,使 E 最小。

多层前向网络设计过程涉及较多方面,例如网络的层数、每层的神经元节点数、初始值的选取、学习速率的确定、还有训练算法的选定或改进等。其中,网络的结构与参数的设计是多层前向神经网络设计的关键。其设计方法主要有两种:一是采用试探训练法,这种方法采用增减式的探测方法获取网络的结构和权值,其缺点是不能避免反向传播算法的不足,网络结构难以确定,网络训练速度慢并且容易陷入局部收敛。二是用遗传算法来进行多层前向网络的设计。但在遗传算法中,初始群体是随机产生的,如果分布不均匀时,易出现未成熟收敛,陷入局部小点。

近年来,随着基础医学研究和生物学研究的发展,关于

免疫系统的运行机制也逐渐为人们所了解,生物免疫系统作为人体信息处理系统的一个主要组成部分,它所具有的许多信息处理机制和功能特点,如自我非自我的抗原识别与抑制、学习与记忆、自适应环境的能力以及能与体内其他成分进行协调的特点等,对开发设计新的智能计算模型,具有重要的现实意义和发展前景。

针对上述两种方法存在的不足,国内外学者利用免疫遗传机理来设计神经网络^[2-4]。本文吸取了他们的经验,借鉴生物免疫机制,提出了一种基于免疫遗传算法的多层前向神经网络设计方法来实现网络结构和权值的确定。免疫遗传算法是遗传算法和生物免疫思想的结合,既能够保留遗传算法的全局搜索特性,又能避免未成熟的收敛。

1 免疫遗传算法

1.1 免疫遗传算法的特点

免疫遗传算法可以看作一种新型融合算法,也是一种改进的遗传算法,一种新型计算智能方法,是具有免疫功能的遗传算法。与免疫算法一样,它把待求解的问题对应为抗原,问题的解对应于抗体。免疫遗传算法与免疫算法在形式上看,就是在免疫算子中加入了遗传算子。与遗传算法相比,增加了抗原识别、记忆功能和调节功能,没有附加复杂的操作,没有降低遗传算法的鲁棒性,算法兼顾了搜索速度、全局搜索能力和局部搜索能力。免疫遗传算法一般形式只是要表明免疫机制对遗传算法的作用,根据具体问题设计算法。

1.2 免疫遗传算法的实现

一般的,免疫遗传算法(IGA)的实现采取以下步骤^[5],如图1所示:

1) 随机创建抗体和抗原的群体;

收稿日期:2005-01-07;修订日期:2005-03-14

作者简介:罗菲(1979-),男,四川成都人,硕士研究生,主要研究方向:人工免疫系统、人工神经网络、进化算法;何明一(1958-),男,四川盐亭人,博士生导师,主要研究方向:神经网络与智能信息处理、图像工程与可视化、对地观测图像处理、三维信息获取与处理。

- 2) 抗体与抗原匹配;
- 3) 根据抗体的亲和力对抗体作评价;
- 4) 用标准的遗传算法进化抗体。

为了保证在遗传算法中抗体群体的多样性,De Castro 等人引入适应度来进一步完善免疫特性^[5]:

- 1) 抗原被逐一处理;
- 2) 只选择抗体细胞中的一个子集来与抗原匹配;
- 3) 抗体中存在对抗原识别的竞争,这样只有与抗原具有最高亲和力的抗体才被选中进行再生;
- 4) 使用高频变异来进一步提高抗体的多样性。

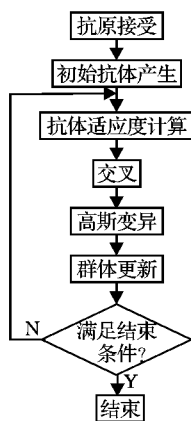


图1 免疫遗传算法

2 基于免疫遗传算法的多层前向神经网络设计

本文应用上述的免疫遗传算法的一般实现步骤,将其用于多层前向神经网络的设计。把待求解的问题对应为抗原,问题的解对应为抗体。在此问题中,抗体对应一种网络结构。IGA 首先接受一个抗原(对应特定问题),然后随机产生一组初始抗体(对应候选解);接着计算每一代抗体的适应度,对抗体进行变异;再通过基于抗体浓度和亲和力的群体更新策略生成下一代抗体群;直到满足终止条件,算法结束。需要说明的是,在设计多层前向神经网络时,假设其输入(包括输入节点数及输入值)和输出(包括输出节点数及输出值)是已知的,激活函数也预先确定,要设计的主要是隐层节点数和其相关权值及阈值。

2.1 抗体基因编码

多层前向神经网络的学习过程是一个复杂的连续参数优化问题,如果采用二进制编码,会造成编码串过长,在计算抗体时又要解码成实数,从而影响网络的学习精度和算法的运行时间。因此,本算法采用了实数编码方式。每个抗体的基因编码方式如下:(n (隐节点数目),与第1个隐节点相关的权,与第1个隐节点相关的阈值,与第2个隐节点相关的权,与第2个隐节点相关的阈值,……,与第 n 个隐节点相关的权,与第 n 个隐节点相关的阈值)。

2.2 抗体适应度函数的设计

设抗体群的数目为 n ,每个抗体对应一种网络,作为网络结构的组合。由式(2)计算出每个抗体对应的误差 E_i ,对应于免疫遗传算法中的抗原。则B细胞分泌的抗体与抗原的适应度函数可设计为 $F(i)$,其对应于免疫理论的亲和力概念:

$$F(i) = \frac{1 + \text{const}}{e^{E_i} + \text{const}}, \text{const 为大于0的常量} \quad (3)$$

这里对抗体适应度函数进行归一化处理,由式(3)知,当抗体最佳时适应度函数的值为1。

2.3 抗体浓度的定义

将抗体中要设计的参数(如 W)映射为免疫理论中的抗体 i 中的等位基因 j ,表示为 $G^j(i)$ 。 n 个抗体就构成了一个非空免疫系统集合 X ,可以度量抗体之间在集合 X 中距离,规定:

$$d_i = \sum_k \sqrt{\sum_j (G^k(i) - G^k(j))^2} \quad (4)$$

为抗体 i 和集合 X 中其他抗体的距离度量。因此,抗体 i 的浓度应该与抗体与其他抗体的距离成反比,则可以定义抗体浓度 A_i 为:

$$A_i = \frac{1}{1 + d_i} \quad (5)$$

由式(5),集合 X 中相同抗体越多,则抗体的浓度越大;反之,抗体的浓度越小。将抗体浓度归一化,当所有抗体相同时,其浓度最大为1。

2.4 抗体的抑制和促进

根据免疫系统抑制高浓度、低亲和力抗体的产生,促进低浓度、高亲和力抗体分泌的特性,由式(3)~(5)可推导出基于抗体浓度的概率选择公式:

$$P(i) = \alpha(1 - A_i)F(i) \quad (6)$$

其中 α 为0到1之间的可调参数。由式(6)知,集合 X 中的某个抗体浓度越大,亲和力越低,则该抗体被选中的概率就越小;反之,抗体的浓度越小,亲和力越高,其被选中的概率就越大。因此,基于抗体亲和力和浓度的选择策略在理论上保证了了解的多样性。

2.5 抗体群的遗传更新

变异只对每个抗体中的权值分量进行,因为权重基因型与网络神经元之间是多对一的映射关系,这使得杂交操作变得低效或无效。所以这里仅选择遗传算法中的变异操作实现抗体群的更新。将各抗体解码为相应的网络结构,按式(7)和(8)变异网络的所有权值,其中 β 是-1到1之间的一个常数, μ_i 为变异算子, $N(0,1)$ 为高斯算子。变异后再重新将隐节点分量和权值分量组成一个新的抗体。

$$\mu_i = \sqrt{1 - F(i)} \quad (7)$$

$$\Delta W = \beta \mu_i N(0, 1) \quad (8)$$

因为抗体适应度函数已做了归一化处理, $F(i)$ 的取值范围为 $[0,1]$,所以由式(7)知,抗体亲和力越大,则变异算子越小,对最佳抗体的变异为0。

3 仿真实验

表1 XOR问题3种算法的学习结果对比

误差精度	最大学习步数	动量 BP 算法			免疫遗传算法			遗传算法		
		学习步数	学习时间/s	收敛比 (%)	学习步数	学习时间/s	收敛比 (%)	学习步数	学习时间/s	收敛比 (%)
0.100	500	376	1.121	75	38	0.107	87	45	0.127	82
0.010	1000	754	2.313	81	77	0.243	90	184	0.439	76
0.001	2000	1462	3.145	73	191	0.523	83	321	0.748	69

3.1 XOR问题

用2-3-1多层前向网络结构,以同样的10组初始权值和阈值,分别利用本文算法(IGA)、遗传算法(GA)和动量BP算法(MOBP)进行学习,结果如表1所示。其中学习步数和学习时间都是在收敛的情况下有意义,为50次收敛训练的平

均学习步数和学习时间,而收敛比为50次重复学习中收敛次数所占的比例,学习时间的单位为秒。从表中可知,IGA比GA和MOBP有更快的收敛速度和更好的全局收敛性。

3.2 函数逼近

(下转第1665页)

```

* [j] = q →
[ ShortSysCall [i] 满足特征库中第 j 类“非我”某条规则
→ R[i][j] = 1
□ ShortSysCall [i] 不满足特征库中第 j 类“非我”的任
何一条规则 → R[i][j] = 0
j = j + 1;
]
]
count[j] = count[j] + R[i][j];
r = count[j]/N;
[ r < r0 → i = i + 1
□ r ≥ r0 →
[ 此“非我”程序属于第 j 类“非我”;
i = p;
]
]
]
]

```

4.4 实验结果

测试的“非我”程序 sm_10763.int, sm_280.int, fwd_loops_3.int 和 syslog_local_2.int, 分别记作 nonself_test1, nonself_test2, nonself_test3 和 nonself_test4. susendmailcp 类“非我”, decode 类“非我”, forwarding loops 类“非我”和 syslogd 类“非我”分别记作类 1, 类 2, 类 3 和类 4。

表 1 测试程序的实验结果

测试程序	不同类“非我”					总数	r	所属类
	类 1	类 2	类 3	类 4	未知			
nonself_test1	65	0	0	0	8	73	0.89	类 1
nonself_test2	0	3	0	0	8	3	1.00	类 2
nonself_test3	0	0	48	0	16	64	0.75	类 3
nonself_test4	0	0	0	195	77	272	0.72	类 4

实验中取相关系数阈值 $r_0 = 0.70$, 可正确地判别出测试“非我”程序所属的“非我”类, 实验结果如表 1 所示。其中:

- 1) “不同类‘非我’”列中的数字为满足该类“非我”的系统短调用序列个数;
- 2) “总数”列中的数字为测试“非我”程序的系统调用短

序列除掉在 SELF 库中出现的系统调用短序列后的个数;

- 3) “r”列中的数字表示测试的“非我”程序与一类“非我”的相关系数, 这类“非我”要求: 测试“非我”程序的系统调用短序列满足这类“非我”的个数最多。

实验结果分析:

- 1) 由表 1 可以看出, 将“非我”程序按入侵行为进行分类, 它们的系统调用短序列有明显的差别, 除了少数被判为未知类外, 绝大部分判出满足的“非我”类是正确的, 通过选择适当的 r_0 , 可以辨别出属于哪类“非我”。

- 2) 由于用于训练集的“非我”程序有限, 所以被判断为未知类的系统调用短序列的数目相对较多。随着训练集的扩充, 特征库中规则的增加, r_0 的取值可相应地增大, 从而实验结果更为准确。

参考文献:

- [1] HOFMEYER SA, FORREST S, SOMAYAJI A. Intrusion Detection using Sequences of System Calls[J]. Journal of Computer Security, 1998, 6(3): 151-180.
- [2] FORREST S, HOFMEYER SA, SOMAYAJI A, et al. A sense of self for UNIX processes[A]. Proceedings of the 1996 IEEE Symposium on Computer Security and Privacy[C]. IEEE Press, 1996.
- [3] LEE W, STOLFO SJ, CHAN PK. Learning Patterns from Unix Process Execution Traces for Intrusion Detection[A]. Proceedings of the AAAI97 Workshop on AI Approaches to Fraud Detection and Risk Management[C]. 1997.
- [4] LEE W, STOLFO SJ. Data mining approaches for intrusion detection[A]. Proceedings of the 7th USENIX Security Symposium[C], 1998.
- [5] 王凤先, 常胜, 刘振鹏, 等. 一种仿生物免疫的计算机安全系统模型[A]. 2002 年全国开放式分布与并行计算学术会议论文集[C]. 武汉: 华中科技大学出版社, 2002. 72-76.
- [6] 王凤先, 胡陈丽, 张岩. 基于系统调用的入侵检测[J]. 计算机工程与应用, 2004, 40(35): 79-81.
- [7] 戴英侠, 连一峰, 王航. 系统安全与入侵检测[M]. 北京: 清华大学出版社, 2002.
- [8] 徐漫江, 曹元大. 一种分析系统调用序列的入侵检测系统设计与实现[J]. 北京理工大学学报, 2004, 24(4): 342-344.

(上接第 1662 页)

多层前向神经网络的一个重要应用就是函数逼近, 本文取用 2-3-1 多层前向网络结构来逼近一个非线性函数 $Y = 5\cos(\frac{e^x + 1}{x})$ 的 $[0, 20]$ 段, 在 $[0, 20]$ 中均匀取 40 个点, 它们和被模拟的函数在这些点的取值构成了

训练样本。分别用上述三种方法训练网络, 取多层前向神经网络在 40 个样本处的学习误差的平方和作为目标函数。训练的结果见表 2, 从中可以看出, IGA 比其他两种方法脱离局部极小的能力更好, 收敛速度更快。

参考文献:

- [1] HAGAN MT, DEMUTH HB, BEALE MH. 神经网络设计[M]. 戴葵, 等译. 北京: 机械工业出版社, 2002.
- [2] 何永勇, 诸福磊, 钟秉林. 基于进化计算的神经网络设计与实

表 2 函数逼近 3 种算法学习结果对比

免疫遗传算法		动量 BP 算法		遗传算法	
运算时间/s	目标函数	运算时间/s	目标函数	运算时间/s	目标函数
0	0.759 509 800E+01	50	0.414 567 000E+01	10	0.632 004 900E+01
10	0.329 452 200E+01	100	0.136 049 800E+01	30	0.517 962 400E+01
50	0.147 475 000E+01	200	0.678 664 600E+00	50	0.385 235 200E+01
100	0.672 597 800E+00	350	0.299 284 900E+00	150	0.683 270 500E+00
150	0.317 680 100E+00	500	0.251 077 800E+00	300	0.281 917 900E+00
300	0.208 167 000E+00	750	0.246 354 700E+00	500	0.230 425 600E+00
500	0.189 614 900E+00	1 000	0.223 910 000E+00	1 000	0.209 606 400E+00

现[J]. 控制与决策, 2001, 16(3).

- [3] DE CASTRO LN. Immune and Neural Network Models: Theoretical and Empirical Comparisons[J]. International Journal of Computational Intelligence and Applications, 2001, 1(3).
- [4] WANG L. A Novel Neural Network Based on Immunity[A]. Proceedings of the International Conference on Artificial Intelligence[C], 2002.
- [5] DE CASTRO LN. Artificial Immune Systems: A Novel Computational Intelligence Approach[M]. Springer-Verlag, 2002.