

57-60

一种模糊计算的多层前向神经网络——FCBP*

Multiple Layer BP Neural Network for Fuzzy Calculating

周小佳
Zou Xiaojia

张安邦 ✓
Zang Anbang

周家启
Zou Jiaqi

TP18

(重庆大学电气工程系, 重庆, 630044, 第一作者 27 岁, 男, 博士生)

A 摘要 提出的 FCBP(Fuzzy Calculating BP)算法克服了 BP 算法对样本的敏感性, 减少了输入点数, 减轻了输入层的负担, 用于模糊推理及模式识别中就更显其优越性。

关键词 神经网络; 模糊计算 / 模糊计算 BP 算法; BP 算法
中国图书资料分类法分类号 TN711

FCBP

ABSTRACT The FCBP(Fuzzy Calculating BP) algorithm which is proposed by this paper has overcome the sensitivity for samples, reduced the number of input layer's samples, lightened the burden of input layer, it is suitable to fuzzy inference and pattern recognition.

KEYWORDS neural networks; fuzzy calculation / fuzzy calculating BP; BP algorithms

0 引 言

BP 模型是一个多层映射有监督的学习网络, 采用的是最小均方差的学习方式。它在专家系统和自适应控制中有广泛的应用。但 BP 算法仍存在着许多问题, 如收敛速度慢; 易陷入局部极小点, 形成振荡; 对输入样本正交性的依赖; 对学习参数(学习速度、动态项等)的敏感性等等。不少学者对 BP 算法作了改进, 如为解决局部极小值问题的模拟退火法, 牛顿法; 为解决收敛速度慢问题的快速 BP、FBP 算法等。这些算法在一定程度上解决了局部极小值和收敛速度等问题, 但对于算法对输入样本的依赖性问题以及输入向量维数增加时输入层负担过重等问题仍没能解决。笔者利用模糊集理论, 提出了 FCBP 算法, 该算法适于对输入向量维数较多的模糊问题进行学习训练。

1 FCBP 模型及算法工作原理

在专家系统知识表达之中

$$\text{IF } X_1(A_1) \text{ OR } X_2(A_2) \cdots \text{OR } X_n(A_n) \text{ THEN } Y(B) \quad (1)$$

$$\text{IF } X(A) \text{ THEN } Y_1(B_1) \text{ AND } Y_2(B_2) \cdots \text{AND } Y_m(B_m) \quad (2)$$

* 收文日期 1995-04-24

其中, X, X_1, X_2, \dots, X_n 是规则的前件项; Y, Y_1, Y_2, \dots, Y_m 是规则的后件项; A, A_1, A_2, \dots, A_n 是规则前件项的可信度; B, B_1, B_2, \dots, B_m 是规则后件项的可信度。

对于式(1)和(2)的规则形式,可转换成几个单一前件项形式的规则。于是,每一个前件项就是我们具有一定可信度的考察对象的一个侧面。对于前件是由多个前件项的“与”,如:

$$\text{IF } P_1 \text{ AND } P_2 \text{ AND } \dots \text{ AND } P_n \text{ THEN } Y \quad (3)$$

其中, P_1, P_2, \dots, P_n 是规则的前件项; Y 是规则的后件项。

公式(3)的规则形式,前件的可信度是其前件项可信度的“与”运算的结果。其模糊计算规则形式可由图1所示。其中, X_1, X_2, \dots, X_n 是 N 条规则组成的模糊计算层的输出结果; B_1, \dots, B_n 是各条规则的前件项,“○”是模糊计算神经元,利用模糊运算法则进行模糊计算。如果对每一个前提的前件项赋给模糊化向量(由端口测定或由用户给定的),由这些前件项经模糊运算就可计算出每条规则前件的可信度,如图1中的 X_1, X_2, \dots, X_n 。如果再经过一个BP网络进行前件激活,就构成了FCBP模型,也就可以进行规则前提的模糊量化处理,从而最终得到可用于推理的模糊向量。

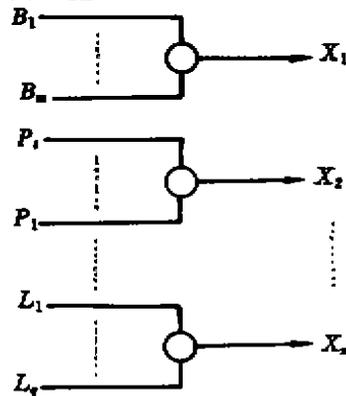


图1 模糊计算规则

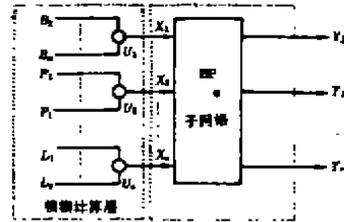


图2 FCBP模型

FCBP模型如图2所示,输入层是模糊计算层,采用图1的模糊计算规则。其中每个模糊计算神经元 u_1, u_2, \dots, u_n 都是按照如下的模糊运算法则进行模糊运算

$$CF(\text{Certainty Factor})(X_j) = \bigotimes_{i=1}^n CF(P_i)$$

其中,“ \bigotimes ”表示模糊运算法则,即:“ \wedge (取小)”、“ \vee (取大)”、“ Σ (求和)”的运算; X_j 表示第 j 个规则的可信度; P_i 表示第 j 个规则的第 i 个前件项的可信度。

并且对于不同的规则形式要采用不同的模糊运算法则。对于式(3)所示的规则要采用“ \wedge (取小)”运算,对于式(1)和(2)所示的规则需采用“ \vee (取大)”运算。在该模糊计算层后是传统的BP子网络。其输入是模糊计算层的输出结果,网络层数视问题的复杂程度而定。BP子网络的输出也是FCBP模型的输出,即 Y_1, Y_2, \dots, Y_n 是可用于推理的模糊向量。输出节点数是模糊向量维数。由于FCBP网络增加了模糊计算层,使进入传统BP子网络的输入值正交性较好并大大减少了传统BP子网络的节点数。由于该网络的模糊计算层是非权重连接的计算神经元,因此使网络性能更加稳定并易于实现。

因为FCBP网络是采用误差反向传播的学习算法,所以它要求各个神经元的激活函数是连续且可导。但由于其模糊计算层中的模糊计算神经元激活函数是采用的模糊运算法则,所以有不可导区间。于是我们用下式对“ \wedge (取小)”运算近似表达

$$\min(a, b) \approx \frac{1}{2} [a + b - \sqrt{(a - b)^2 + \varepsilon^2} + \varepsilon] \quad (4)$$

其中 ε 为一小数, 决定了近似的精确程度。上式对自变量 a, b 的一阶导数(用 $\min'(1)(a, b)$ 表示) 可求得:

$$\frac{\partial \min(a, b)}{\partial a} = \min'(1)(a, b) = \frac{1}{2} [1 - (a - b) / \sqrt{(a - b)^2 + \varepsilon^2}] \quad (5)$$

由于 $\min(a, b)$ 与 a, b 出现的顺序无关, 有

$$\frac{\partial \min(a, b)}{\partial b} = \min'(1)(b, a) = -\frac{1}{2} [1 - (a - b) / \sqrt{(a - b)^2 + \varepsilon^2}] \quad (6)$$

所以, 近似表达式(4) 对于 a, b 均为连续可导的。

当自变量数目 $N > 2$ 时, $\min(a_i; i = 1, 2, \dots, N)$ 对其中任一变量 k 的一阶导数为:

$$\min'(1)(a_i; i = 1, 2, \dots, N) = \min'(1)(a_i, \min'(a_j; j = 1, 2, \dots, k - 1)) \quad (7)$$

由上所述, 对于“ \vee (取大)”运算的近似表达式及其对自变量各阶导数可同样得出。

将上述导数公式运用于误差反向传播算法中, 网络联结权值修正量可写为:

$$\sigma_{nj} = \frac{\partial E_p}{\partial O_{nj}} \cdot \min'(1)(Net_{nj}; j = 1, 2, \dots, n) \quad (8)$$

通过有教师指导的训练, 可实现对规则前件的激活。

2 FCBP 算法过程

在 FCBP 网络中, 笔者对于 BP 子网络采用的是 Sigmoid 函数形式。假设用户给定的输入是几个前件项“与”的形式, 如式(3) 所示。由此式及各阶导数可得出 FCBP 的算法步骤如下:

1) 初始化 FCBP 网络中的权值和阈值。

2) 给出 FCBP 网络结构——模糊计算层中各规则的前件数 m, l 和 q ; 训练样本数, 即规则数目 N ; BP 子网络各层节点数以及采样样本数目。

3) 提供训练用学习资料——输入向量 $\{S(m), P(l), L(q)\}$ 和目标输出 Y , 训练次数以及控制误差 E 。

4) 通过模糊计算层, 计算各训练样本前件的可信度。

5) 用模糊计算层的输出送入 BP 子网络计算该子网络各层节点的实际输出及隐层各节点状态。

$$O_{nj} = f(net_{nj}) = f(\sum W_{\mu} X_i + \theta_j)$$

6) 利用误差反传算法计算训练误差。

$$\delta_{nj} = O_{nj} \cdot (1 - O_{nj}) \cdot (t_{nj} - O_{nj}) \quad (\text{输出层}) \quad \delta_{nj} = O_{nj} \cdot (1 - O_{nj}) \cdot \sum \delta_i \cdot W_{\mu} \quad (\text{隐层})$$

7) 利用式(8) 修正权值和阈值。

$$W_{\mu}(l+1) = W_{\mu}(l) + \eta \delta_j O_{\mu} \quad \theta_j(l+1) = \theta_j(l) + \eta \delta_j$$

8) 判断样本是否学习完成或训练是否满足精度要求, 即 $E \leq \sigma$ (σ 是一个很小的正数) 是否成立。若样本已学习完或精度满足则转到步骤 9), 否则转到步骤 3)。

9) 学习完成, 退出。

权值和阈值的修正公式中的学习速率 η , 是按梯度搜索的步长。 η 越大, 权值的改变越剧

烈,收敛速度越快,但易产生振荡; η 越小,权值的改变小,收敛速度越慢。为了使训练学习时收敛的速度快,应选取合适的 η 值。通过实际运用的研究, η 的选取原则为:在不导致振荡的前提下取尽可能大的 η 值。实验结果表明,在 $\eta = 0.83 \sim 0.98$ 时可获得较满意的收敛速度。

3 FCBP 和 BP 模型的比较

对于异或(XOR)问题,FCBP网络和BP网络仿真的结果如表中所示。为了说明两种算法对输入样本集的依赖性,本文对异或问题的输入值采用了四种不同的输入顺序。两种算法采用同样的训练参数(学习速度 $R = 0.88$,训练次数 $N = 1000$)。BP网络采用2-2(两节点、两层)结构,FCBP网络的基本BP算法采用2-3结构,模糊

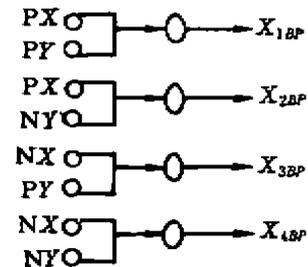


图3 FCBP模糊计算层结构

计算采用8-4网络结构。在FCBP网络的模糊计算层,输入是四个概念: X 正大(PX), Y 正大(PY), X 负大(NX), Y 负大(NY)的组合(如图3所示)。X,Y相当于样本空间中两个元素 $\{X, Y\}$,四个概念相当于样本空间元素的四个状态。X,Y对于四个状态的隶属度可由下式得出:

$$\text{当 } X = a \text{ 时, } X \text{ 对于 PX 的隶属度定义为: } \quad \text{对于 NX 的隶属度定义为:}$$

$$PX(x) = \begin{cases} 0 & X = a < 0 \\ 1/(1 + e^{-x}) & X = a \geq 0 \end{cases} \quad NX(x) = \begin{cases} 0 & X = a > 0 \\ 1/(1 + e^{-x}) & X = a \leq 0 \end{cases}$$

相应地, $Y = b$ 时对于PY和NY的隶属度与PX和NX的定义相仿。四个状态的输入经FCBP模糊计算层后送入基本BP子网络的是 X_{1BP} 和 X_{2BP} , X_{3BP} 和 X_{4BP} 等四个样本。在仿真计算中,模糊计算采用的是“V”(即:取大)运算,计算结果见下表:

FCBP 和 BP 网络的仿真结果表

网络模型	N	S	T/s	σ
BP 算法	00,01,10,11	0,1,1,0	93	0.003 1
	00,01,11,10	0,1,0,1	130	0.003 1
	00,11,01,10	0,0,1,1	143	0.011 07
	11,00,01,10	0,0,1,1	75	0.102 7
FCBP 算法	00,01,10,11	0,1,1,0	27	1.5
	00,01,11,10	0,1,0,1	29	4.5
	00,11,01,10	0,0,1,1	29	60
	11,00,01,10	0,0,1,1	28	1.0

通过仿真结果,可清晰地看出FCBP算法相对于BP算法的良好性能。FCBP网络对于输入值的敏感性小,不依赖于输入样本;随着样本输入顺序的变化,BP算法训练时的训练时间和学习误差都有较大的变化,FCBP算法训练时基本无变化;FCBP算法收敛速度比BP网络明显要快得多。将FCBP算法运用于模糊推理系统,可以使系统对不同问题的适应能力有很大的提高。

参 考 文 献

- 1 焦李成. 神经网络系统理论. 西安:西安电子科技大学出版社,1992. 19
- 2 余少波,胡守仁. 神经网络的能力. 海军工程学院学报,1989,(3):21
- 3 罗承忠. 模糊集引论. 北京:北京师范大学出版社,1989. 49~65