

60-61

神经网络

BP算法

多层神经网络

多层神经网络BP算法的初步研究

陆金桂 韦柳涛 王石刚 余俊 周旌 TP18

(华中理工大学 430074武汉)

摘要

In BP algorithm, the error function, which can not express the error of the sample-learning effectively, reduces the effectiveness of the algorithm. This paper presents a new error function, which is in the form of the relative error. The computer simulation results show that the new error function can help to reduce the relative error of the sample-learning.

目前,在数十种神经网络模型中以BP算法作为学习方式的多层神经网络是使用得最为广泛的一种。该网络是由Rumelhart和McClelland领导的研究小组在1985年实现的,已有效地解决了XOR、T-C匹配等感知器不能解决的问题。现已广泛应用于语音识别、模式分类、过程监控等领域。尽管多层神经网络的BP算法较有效,但由于它用于高度非线性映射的样本学习时存在一些问题而影响了BP算法的深入应用。基于此,本文对该算法进行了初步的研究。

1. 多层神经网络BP算法的改进

多层神经网络是一个在输入和输出层之间具有一层或多层的神经元前馈网络。用于多层神经网络的BP算法包含正向传播和反向传播过程。在正向传播过程中,样本输入值从输入层经隐单元层传递到输出层,然后进行误差反向传播过程,即将误差逐层传递到输入层,通过改变各神经元间的权值使得样本点的实际输出值和期望输出值间误差渐渐减少。

利用BP算法进行一些高度非线性映射的样本学习时存在一些问题,例如在一些样

本点学习效果较好,而在另一些样本点学习效果则较差。究其原因,BP算法中的误差函数 $E = \sum (Y_k - d_k)^2 / 2$ 是误差的绝对量,不能有效地表征样本学习的误差程度。因为对不同的样本点,即使其绝对误差量相同,但样本点的实际输出和期望输出之间误差程度也不等。例如有两个样本点,其期望输出分别为 $d_1 = 0.9$ 、 $d_2 = 0.001$,学习后样本点的实际输出分别为 $Y_1 = 0.81$ 、 $Y_2 = 0.091$,两个样本点的实际输出和期望输出间绝对误差 $E_1 = E_2 = 0.00405$ 。但第一个样本点的实际输出和期望输出间误差为10%,而第二个样本点误差为期望输出的90倍。再如两个样本点,其期望输出分别为 $d_1 = 0.9$ 、 $d_2 = 0.001$,实际输出分别为 $Y_1 = 0.81$ 、 $Y_2 = 0.011$ 。虽然第二个样本点的实际输出与期望输出间绝对误差比第一个样本点的小得多,但第一个样本点的实际输出与期望输出间误差为10%,而在第二个样本点误差为期望输出的10倍。由于在BP算法的学习过程中,学习效果的提高即样本点实际输出和期望输出间误差的减小是通过反向传播过程进行的,因此不能有效表征样本点的学习误差

就不能有效地提高学习效果。故本文在BP算法中用相对量形式的误差函数 $E = \sum (1 - Y_k/d_k)^2/2$ 代替原先的绝对量形式的误差函数。

在下面的算法推导过程中,选择sigmoid函数作为神经元作用函数 $f(x)$ 。在上述和下面有关的公式中, d_k 为第 k 个样本点的期望输出, Y_k 为第 k 个样本点的实际输出。神经元 i 在第 k 个样本点下的实际输出为 Y_{ik} , 神经元 j 在第 k 个样本点下的输入为 net_{jk} , 神经元 i 与 j 的权值为 W_{ij} 。为简单起见认为网络只有一个输出。

$$\text{定义 } E_k = (1 - Y_k/d_k)^2/2$$

$$\text{因 } net_{jk} = \sum W_{ij} Y_{ik}, Y_{jk} = f(net_{jk}),$$

$$\text{则 } \partial E_k / \partial W_{ij} = \partial E_k / \partial net_{jk} \cdot \partial net_{jk} / \partial W_{ij} \\ = \partial E_k / \partial net_{jk} \cdot Y_{ik}$$

$$\text{定义 } \delta_{jk} = \partial E_k / \partial net_{jk}$$

$$\text{则 } \partial E_k / \partial W_{ij} = \delta_{jk} Y_{ik}$$

当神经元 j 为输出单元时, $Y_{jk} = Y_k$, 则

$$\delta_{jk} = \partial E_k / \partial net_{jk} = \partial E_k / \partial Y_{jk} \cdot \partial Y_{jk} / \partial net_{jk} \\ = \partial E_k / \partial Y_{jk} \cdot \partial Y_{jk} / \partial net_{jk} \\ = (1 - Y_k/d_k) \cdot (-1/d_k) f'(net_{jk})$$

$$\text{因 } f'(net_{jk}) = Y'_{jk} = Y_{jk}(1 - Y_{jk}),$$

$$\text{故 } \delta_{jk} = (Y_k/d_k^2 - 1/d_k) Y_{jk}(1 - Y_{jk}).$$

当神经元 j 为隐含层单元时, 计算 δ_{jk} 的公式仍为采用绝对量形式误差公式时的有关公式。

2. 计算机模拟

我们根据上述思想编制了程序在 VAX-11/750 小型机上进行了计算机模拟。我们利用三层神经网络(输入层有三个神经元, 隐藏层有三个神经元, 输出层有二个神经元)对九个样本进行了非线性映射的学习。样本点的输入和期望输出值见下表。

样本号	1	2	3	4	5	6	7	8	9
输	2.0	1.0	0.1	0.1	1.5	1.5	0.5	3.5	9.5
	2.0	0.1	0.1	0.1	1.5	0.2	0.2	5.2	0.1
入	2.0	1.0	1.0	0.1	1.5	1.3	0.3	4.3	9.9
期 望	0.05	0.1	0.64	0.98	0.065	0.065	0.25	0.025	0.016
输 出	0.027	0.0875	0.56	0.50	0.03	0.055	0.15	0.01	0.012

为了考核利用改进的学习算法进行高度非线性映射的样本学习效果, 我们对同样迭代次数时利用两种算法进行学习得到的样本点的实际输出进行了比较, 证实了改进的算法能降低样本点实际输出和期望输出间的误差程度。但是利用改进的算法进行样本学习得到的样本点的绝对误差量有时比原来算法得到的略高。

主要参考文献

- [1] Lippmann R.P., An Introduction to Computing with Neural Networks, IEEE ASSP Magazine 4, 1981
- [2] Philip D. Wasserman, Neural Computing Theory and Practice, New York, 1989
- [3] 焦李成, “神经网络系统理论”, 西安电子科技大学出版社, 1990年