

文章编号:1001-9081(2009)07-1894-03

## 动态调整学习速率的 BP 改进算法

王玲芝, 王忠民

(西安邮电学院 计算机系, 西安 710061)

(wlz@xyou.edu.cn)

**摘要:**在基本反向传播(BP)算法中,学习速率往往固定不变,限制了网络的收敛速度和稳定性。因此,提出一种动态调整 BP 网络学习速率的算法,以 BP 网络输出层节点的实际输出值与期望输出值的平均绝对值误差及其变化率为自变量,找出学习速率与两个自变量之间的函数关系。根据网络的实际学习情况,对学习速率进行动态调整。实例仿真结果表明,改进的 BP 算法在保持网络稳定性的同时,具有更快的收敛速度。而且,该算法只需恰当地选取几个参数,不受条件限制,因此具有普遍的适用性。

**关键词:**反向传播算法;学习速率;动态调整;平均绝对值误差;变化率

**中图分类号:** TP183 **文献标志码:** A

## Improved BP algorithm based on dynamically adjusting learning rate

WANG Ling-zhi, WANG Zhong-min

(Department of Computer Science, Xi'an University of Posts and Telecommunications, Xi'an Shaanxi 710061, China)

**Abstract:** The learning rate is usually invariable in basic Back Propagation (BP) algorithm, thus the constringency rate and stabilization of network are constrained. Therefore, a BP algorithm based on dynamically adjusting learning rate was proposed. The average absolute error between actual output value of the output layer node and the expected output value and its change ratio were regarded as independent variables, then the function relation of learning rate and two independent variables was found. According to the actual learning circumstance of network, the learning rate was adjusted dynamically. Through the instance simulation, the improved BP algorithm is of more fast constringency rate while keeping the good stabilization than basic BP algorithm. Further more, the algorithm can select the appropriate number of parameters without any condition, and it is therefore of general applicability.

**Key words:** Back Propagation (BP) algorithm; learning rate; dynamic adjustment; average absolute error; change ratio

### 0 引言

反向传播(Back Propagation, BP)算法是一种用于前向多层神经网络的反传学习算法。目前, BP 算法已成为应用最多且最重要的一种前向神经网络学习算法,也是前向网络得以广泛应用的基础<sup>[1]</sup>。

BP 算法是一种很有效的学习算法已成为神经网络的重要模型之一<sup>[2]</sup>。由于它具有理论依据坚实、推导过程严谨、物理概念清晰及通用性好等优点,使它至今仍然是前向网络学习的主要算法。但 BP 算法也存在着一些不足之处,其中一个重要的缺点就是该学习算法的收敛速度非常慢,常常需要成千上万次迭代,而且随着训练样例维数的增加网络性能会变差。而通常在 BP 算法的具体应用中,网络的学习速率往往是固定不变的,这样就使得算法的学习速度和网络稳定性之间产生很大的矛盾。因此,提出一种动态调整学习速率的 BP 算法,使算法具有更快的收敛速度和更好的网络性能。而且该算法简单,易于实现。

### 1 动态调整学习速率的 BP 改进算法

#### 1.1 BP 网络

设 BP 网络的结构如图 1 所示,有  $M$  个输入节点,输入层

节点的输出等于其输入。输出层有  $L$  个输出节点,网络的隐含层有  $q$  个节点,  $w_{ij}$  是输入层和隐含层节点之间的连接权值。  $w_{jk}$  是隐含层和输出层节点之间的连接权值,隐含层和输出层节点的输入是前一层节点输出的加权和,每个节点的激励程度由它的激发函数决定。

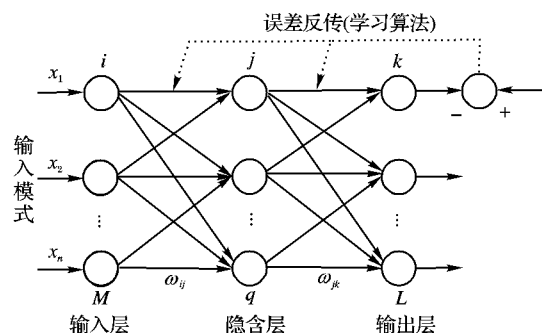


图 1 BP 网络

在训练网络的学习阶段,设有  $N$  个训练样本,先假定用其中的某一固定样本中的输入输出模式  $X_p$  和  $\{d_{pk}\}$  对网络进行训练,则输入层的第  $i$  个节点的输入和输出分别为:  $net_i = x_i, O_i = net_i$ ; 隐含层的第  $j$  个节点的输入和输出分别为:  $net_j = \sum_{i=1}^M \omega_{ij} O_i, O_j = f(net_j)$ ; 输出层的第  $k$  个节点的输

收稿日期:2009-01-06;修回日期:2009-02-26。

基金项目:陕西省自然科学基金研究计划项目(SJ08-ZT14);西安市科技计划项目(CXY08017(1))。

作者简介:王玲芝(1981-),女,河南平顶山人,助教,主要研究方向:智能控制、计算机控制;王忠民(1967-),男,陕西蒲城人,教授,博士,主要研究方向:嵌入式系统、智能控制、机器人技术、智能信息处理。

入和输出分别为:  $net_k = \sum_{j=1}^q \omega_{jk} O_j$ ,  $O_k = f(net_k)$ 。以上各式中,  $i = 1, 2, \dots, M$ ,  $j = 1, 2, \dots, q$ ,  $k = 1, 2, \dots, L$ ,  $f$  为具有偏置和调节的激发函数。

若网络输出与期望输出值  $d_k$  不一致, 则将其误差信号从输出端反向传播 (见图 1), 并在传播过程中对加权系数不断修正, 使在输出层节点上得到的输出结果尽可能接近期望输出值  $d_k$ 。对样本  $p$  ( $p = 1, 2, \dots, P$ ) 完成网络加权系数的调整后, 再送入另一样本模式对, 进行类似学习, 直到完成  $P$  个样本的训练学习为止。

设每一样本  $p$  的输入输出模式的二次型误差函数定义为:

$$E_p = \frac{1}{2} \sum_{k=1}^L (d_{pk} - O_{pk})^2 \quad (1)$$

系统的平均误差代价函数为:

$$E = \frac{1}{2} \sum_{p=1}^P \sum_{k=1}^L (d_{pk} - O_{pk})^2 = \sum_{p=1}^P E_p \quad (2)$$

式(1)、(2)中,  $P$  为样本模式数,  $L$  为网络输出节点数。BP 算法的最终问题就是如何调整链接系数以使代价函数  $E$  最小。通常 BP 算法中网络权值调整规则是基于式(1)的一阶梯度法的优化方法, 即最速下降法。

将式(1)简写为:

$$E = \frac{1}{2} \sum_{k=1}^L (d_k - O_k)^2 \quad (3)$$

权系数应按  $E$  函数梯度变化的方向进行调整, 使网络的输出接近期望的输出。

定义反传误差信号  $\delta_j = -\frac{\partial E}{\partial net_j}$ ,  $\delta_k = -\frac{\partial E}{\partial net_k}$ 。

输出层权值的调整公式为:

$$\Delta \omega_{jk} = -\eta \frac{\partial E}{\partial \omega_{jk}} = \eta \delta_k O_j \quad (4)$$

隐含层节点权值的调整公式为:

$$\Delta \omega_{ij} = -\eta \frac{\partial E}{\partial \omega_{ij}} = \eta \delta_j O_i \quad (5)$$

式中,  $\eta$  为学习速率,  $\eta > 0$ 。

BP 算法的核心是连接权值的调整, 而由权值调整式(4)、(5)可知, 每一层权值的调整都与 BP 网络的学习速率有关, 因此学习速率  $\eta$  也是影响 BP 算法性能好坏的一个重要参数。

## 1.2 学习速率的动态调整设计

网络学习的过程中, 在开始阶段, 要求学习速率  $\eta$  选取较大的值以加快学习速度; 学习接近优化区时,  $\eta$  值必须相当小, 否则权值将产生振荡而不收敛, 进而影响网络的稳定性。所以有必要设计一个动态变化的学习速率, 以达到加快学习响应速度、增强网络稳定性的目的。

考虑两个简单函数  $f_1$  和  $f_2$ :

$$f_1(x_1) = k_1 \cdot \left[ 1 - \exp\left(-\frac{x_1^2}{b_1^2}\right) \right] \quad (6)$$

$$f_2(x_2) = k_2 \cdot \exp\left(-\frac{x_2^2}{b_2^2}\right) \quad (7)$$

$k_1$ 、 $k_2$  称为比例因子, 调节曲线的上下幅度, 从宏观上调整  $f_1$  和  $f_2$  的比例;  $b_1$ 、 $b_2$  称为权重因子, 调节曲线的左右伸展, 从微量上修正  $f_1$  和  $f_2$  的权重。 $f_1$  和  $f_2$  函数的大体曲线形状如图 2 所示。

构造另一个函数  $f_3$ ,  $f_3 = f_1 + f_2$ 。现假定  $x_1$  在某一时间段

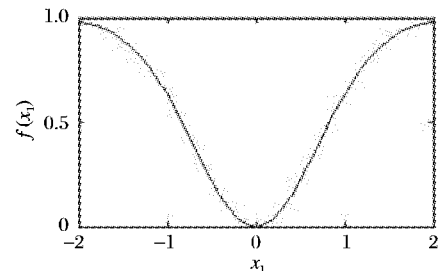
内取值范围为  $0 \sim 0.5$ ,  $x_2$  为  $x_1$  在前后两个时刻内的变化量, 同时, 对  $k_1$ 、 $k_2$  和  $b_1$ 、 $b_2$  在实数范围内赋值并进行调节, 则  $f_1$ 、 $f_2$ 、 $f_3$  的曲线形状可以为如图 3 所示。

观察  $f_3$  的曲线,  $f_3$  是一个减函数。由于  $x_1$  和  $x_2$  都是时间的函数, 则  $f_3$  的值随着时间的增大是逐渐减小的。根据 BP 网络学习速率的选取规则, 随着训练次数的增加, 学习速率  $\eta$  具备  $f_3$  的增减性, 那么用  $\eta$  代替  $f_3$  即可得到一个动态变化的学习速率。现选取 BP 网络输出层节点的实际输出和期望输出的平均绝对值误差  $e$  及其变化率  $ec$  两个变量:

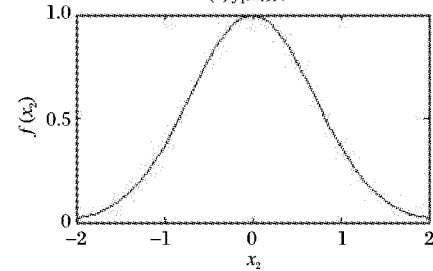
$$e = \frac{1}{L} \sum_{k=1}^L |d_k - O_k| \quad (8)$$

$$ec = e(t) - e(t-1) \quad (9)$$

其中,  $e(t)$  和  $e(t-1)$  表示第  $t$  和  $t-1$  时的平均绝对值误差。

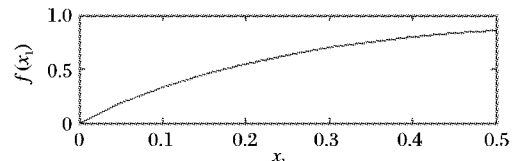


(a)  $f_1$  函数

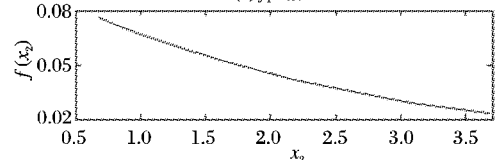


(b)  $f_2$  函数

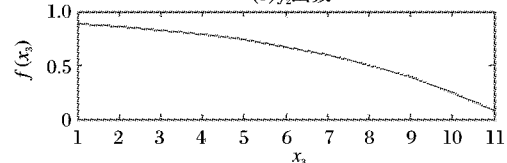
图2  $f_1$  和  $f_2$  函数的曲线形状



(a)  $f_1$  函数



(b)  $f_2$  函数



(c)  $f_3$  函数

图3 参数取适当值时  $f_1$ 、 $f_2$ 、 $f_3$  的曲线形状

$e$  能够反映出网络实际输出和期望输出之间存在偏差的整体水平, 而  $ec$  能够反映这个偏差的变化方向。如果  $e = 0$ , 则说明实际输出与期望输出一致, 网络已经不需要再进行学习; 如果  $e \neq 0$ , 则要观察  $ec$  的值来判断  $e$  的变化方向, 调整网络的学习情况。这样, 同时依据  $e$  和  $ec$  的实时变化, 根据网络

学习需要调整学习速率,以期使网络的实际输出尽可能地接近期望输出。因此,用  $e$  和  $ec$  分别代替  $x_1$  和  $x_2$ ,则学习速率  $\eta$  是以  $e$  和  $ec$  为自变量的函数,而  $e$  和  $ec$  都是与训练次数相关的量。取  $\eta$  的表达式形式为:

$$\eta = f(e, ec) = f_1(e) + f_2(ec) \quad (10)$$

在网络学习过程中,训练次数不断增大,  $\eta$  的值随着  $e$  和  $ec$  而改变。根据实际情况和调节规则可方便地设计  $f_1(e)$  和  $f_2(ec)$  参数的大小。 $f_1(e)$  和  $f_2(ec)$  的函数形式如下:

$$f_1(e) = k_1 \cdot \left[ 1 - \exp\left(-\left(\frac{e}{b_1}\right)^2\right) \right] \quad (11)$$

$$f_2(ec) = k_2 \cdot \exp\left(-\left(\frac{ec}{b_2}\right)^2\right) \quad (12)$$

$k_1$ 、 $k_2$  和  $b_1$ 、 $b_2$  的值根据实际需要和多次反复实验确定。

## 2 实例仿真及结果分析

例 1 在 Matlab 环境下,分别利用基本 BP 算法和改进 BP 算法对函数  $f(x) = \sin x + \cos(2x) + \sin(3x)$  进行学习仿真。取训练集为  $X = [-1:0.1:1]$ ,设网络的隐含层和输出层均为 8 个节点。隐含层的输入输出变换函数为 S 型函数,输出层变换函数为比例函数。

基本 BP 算法中,学习速率经过反复实验确定最优值为 0.45。改进 BP 算法中,在第一次训练时给学习速率赋初值为 1.1,参数  $k_1 = 1$ ,  $k_2 = 0.1$ ,  $b_1 = 0.5$ ,  $b_2 = 5$ 。图 4 为基本 BP 算法中函数拟合曲线及误差曲线,图 5 为改进 BP 算法中函数拟合曲线、误差曲线及网络学习过程中学习速率  $\eta$  的变化曲线。

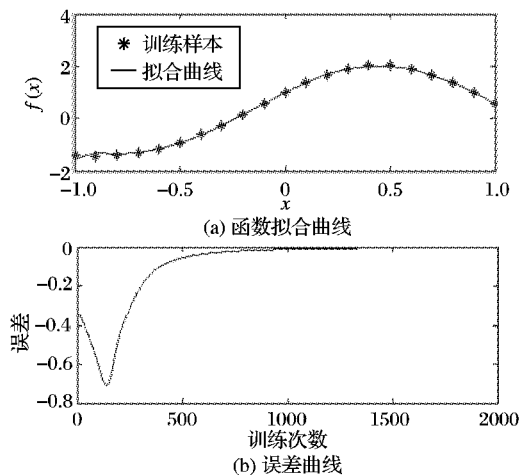


图 4 基本 BP 算法的仿真结果

从仿真结果图 4、5 可以看到,基于动态调整学习速率的 BP 算法与基本 BP 算法相比,前者的拟合效果要优于后者。同时,基本 BP 算法需要训练 2000 次,而改进后的 BP 算法只需训练 250 次。这样在保持网络性能稳定的同时,也大大加快了学习速度。

例 2 为了与文献[4]中提出的算法进行比较,现用改进的 BP 算法逼近  $[0, \pi]$  上的函数  $f(x) = 2\sin(2x)$ ,在  $[0, \pi]$  上均匀地取 30 个点,同样采用 15 个隐含层节点数的一个隐层的单输入单输出网络。逼近曲线及误差曲线如图 6 所示,训练次数为 200 次,误差为  $1 \times 10^{-5}$ 。文献[4]中同样的函数及网络结构,经过 1000 次迭代,误差为  $3.749231 \times 10^{-4}$ 。因此,该改进算法与文献[4]中提出的算法相比,算法相对简

单,训练次数少,精度更高。

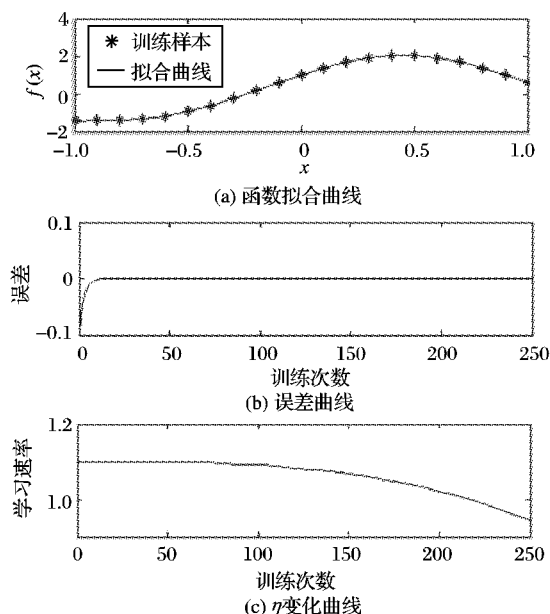


图 5 改进 BP 算法的仿真结果

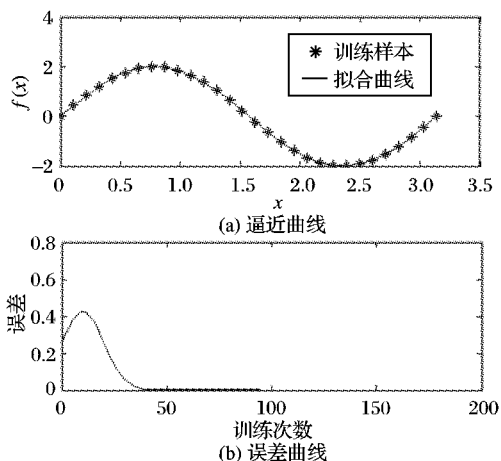


图 6 逼近曲线及误差曲线

## 3 结语

本文提出了一种基于正态函数的动态调整 BP 网络学习速率的算法,给出了学习速率与平均绝对值误差和其变化率之间的函数关系,根据平均绝对值误差和其变化率的实时变化对学习速率进行动态调整,以满足 BP 网络学习的实际需要。通过改进的动态学习速率 BP 算法和基本固定学习速率 BP 算法对相同非线性函数的拟合仿真比较,该算法具有更好的逼近非线性函数的能力,而且加快了学习速度。由于该算法只需恰当地选取几个参数,不受任何条件限制,因此具有普遍的适应性。

### 参考文献:

- [1] 易继锴,侯缘彬. 智能控制技术[M]. 北京: 北京工业大学出版社, 1999.
- [2] 蔡自兴,徐光祐. 人工智能及其应用[M]. 9 版. 北京: 清华大学出版社, 2005.
- [3] 王永庆. 人工智能原理与方法[M]. 西安: 西安交通大学出版社, 2003.
- [4] 王子才,施云惠,崔明根. 一种具有最优动态学习率的 BP 算法[J]. 系统仿真学报, 2001, 13(6): 775-776, 815.