

改进的 BP 神经网络在交通流量预测中应用

郇洪江, 宫宁生, 胡 斌

(南京工业大学 信息科学与工程学院, 江苏 南京 210009)

摘 要: 针对传统 BP 学习算法收敛速度慢, 对步长依赖明显等缺点, 提出一种利用搜索较优步长的 BP 算法. 在网络训练中, 能够在每次迭代中搜索出一个相对合理的步长, 从而使步长的选择对学习速度的影响大大降低. 对交通流量预测仿真结果表明, 新算法对步长选择的依赖性小于传统 BP 算法.

关键词: 神经网络; BP 算法; 搜索较优步长; 交通流量预测

中图分类号: TP183

文献标识码: A

文章编号: 1000-7180(2010)01-0106-03

Application of Modified BP Neural Network in Traffic Flow Forecasts

HUAN Hong-jiang, GONG Ning-sheng, HU Bin

(School of Information Science and Engineering, Nanjing University of Technology, Nanjing 210009, China)

Abstract: Throughing the study the neural network can give a output which match the expectation output, if use it in traffic flow forecast may achieve the good effect. The traditional BP algorithm has some limitations such as the learning convergence speed is very slow and the learning process is dependent on the learning rate. In view of the limitations, this paper presents a BP neural network based on searching for more adaptive learning rate algorithm. The traffic flow forecasts simulation results show that the new algorithm is less dependent on the learning rate than the traditional BP algorithm.

Key words: neural network; BP algorithm; searching for more adaptive learning rate; traffic flow forecasts

1 引言

人工神经网络是借鉴人脑的结构和特点, 通过大量简单处理单元(神经元或节点)互连组成的大规模并行分布式信息处理和非线性动力系统. 利用神经网络可以通过学习达到其输出与期望输出相符的结果, 具有很强的自学习、自适应、鲁棒性、容错性及存储记忆的能力, 用它来进行交通流量预测可以达到良好的效果. BP 算法^[1]是应用最广泛的一种人工神经网络模型. 但标准的 BP 学习算法存在着收敛速度慢, 步长选择对收敛速度影响大, 而且容易陷入局部极小等缺点. 因此, 文中提出一种利用搜索较优步长的 BP 算法, 仿真结果表明, 新算法对步长选择的依赖性小于传统 BP 算法.

2 常规 BP 算法

BP 模型是一种适用于非线性的模型识别和分类预测问题的人工神经网络^[2-3], 最基本的 BP 网络是由输入层、隐层和输出层组成 3 层前馈网络, 每层有若干个互不连接的神经元节点, 相邻两层节点通过权值连接.

BP 学习过程分为两个阶段: (1) 工作信号正向传播: 当给定网络一个输入模式时, 它由输入层单元传到隐含层单元, 经过隐含层单元处理后再送到输出层单元, 由输出层单元处理后产生一个实际输出模式; (2) 误差信号反向传播: 如果实际输出模式与期望输出模式有误差, 且不满足允许误差要求, 则将误差值沿连接通路逐层反馈并修正各层连接权值.

3 搜索较优步长 BP 算法

3.1 搜索较优步长算法

文中提出的搜索较优步长算法^[4-5],可以在每一步的迭代后得出一个比较合理的步长值,算法的主要思想是:设定一个当前步长 $\eta(n)$,一个步长搜索向量 ηL , ηL 中共有 L 个元素,选择 1 和在 1 的左右相对靠近的 L 个数,例如:

$$\eta L = \{1.1, 1.05, 1.01, 1.0, 0.99, 0.95, 0.9, 0.8, 0.7, 0.6\}.$$

对于一个样本 k ,先前向计算 BP 网络每层神经元的输入信号和输出信号,由期望输出和实际输出计算样本误差 $E_k(n)$,若不满足允许误差要求,则反向计算每层神经元的局部梯度 δ . 对于 $l = 1, 2, \dots, L$,利用如下公式计算网络每层神经元的输入信号和输出信号,以及样本误差 $E_{kl}(n)$:

$$\text{隐层: } Net_{il}(n) = \sum_{m=1}^M (w_{mi}(n) + \eta(n) \cdot \eta L_l \cdot \delta_i(n) \cdot x_{km}(n)) x_{km}(n)$$

$$v_{il}(n) = \frac{1}{1 + e^{-Net_{il}}} \quad (i = 1, 2, \dots, I)$$

$$\text{输出层: } Net_{pl}(n) = \sum_{i=1}^I (w_{ip}(n) + \eta(n) \cdot \eta L_l \cdot \delta_p(n) \cdot v_i(n)) v_{il}(n)$$

$$y_{pl}(n) = v_{pl}(n) = \frac{1}{1 + e^{-Net_{pl}}} \quad (p = 1, 2, \dots, P)$$

$$E_{kl}(n) = \frac{1}{2} \sum_{p=1}^P (d_p - y_{pl})^2$$

选取使样本误差 $E_{kl}(n)$ 最小的 ηL_l ,

$$\eta(n+1) = \eta(n) \cdot \eta L_l$$

按下式计算权值修正量:

$$w_{ip}(n+1) = w_{ip}(n) + \eta(n+1) \delta_p(n) v_i(n)$$

$$w_{mi}(n+1) = w_{mi}(n) + \eta(n+1) \cdot$$

$$\delta_i(n) x_{km}(n)$$

按此方法,不断循环,直到样本误差 $E_k(n)$ 小于允许误差为止,完成样本 k 的学习,进行下一个样本的学习.

考虑到如果对于每次大循环中的所有样本的都使用搜索较优步长算法,则会大大增加每次循环的计算量,所以本算法每一次大循环只选择一组样本使用搜索较优步长算法进行学习,其他样本都使用定步长的算法进行学习.一次大循环中学习次序在使用搜索较优步长算法的样本之前的样本,使用上一次大循环结束时的步长,学习次序在使用搜索较

优步长算法的样本之后的样本,使用经过搜索较优步长算法调节后的步长.

由于 BP 算法以样本误差 $E_k(n)$ 作为指导,所以一次大循环将所有样本学习完毕总体误差 E 并不一定下降,为了使学习过程中总体误差 E 更快的下降,也为了收敛到全局最小点的概率增大,具体在选择时用哪一组样本使用搜索较优步长算法时,使用随机选择和以总体误差 E 下降为标准判断相结合的方法.

3.2 搜索较优步长 BP 算法

搜索较优步长 BP 算法具体步骤如下:

(1) 设置变量和参量:

$X_k = [X_{k1}, X_{k2}, \dots, X_{km}]$, ($k = 1, 2, \dots, N$) 为输入向量,或称训练样本, N 为训练样本的个数.

$W_{MI}(n)$ 为第 n 次迭代时输入层与隐层之间的权值向量.

W_{IP} 为第 n 次迭代时隐层与输出层之间的权值向量.

$Y_k(n) = [Y_{k1}(n), Y_{k2}(n), \dots, Y_{kp}(n)]$, ($k = 1, 2, \dots, N$) 为第 n 次迭代时网络的实际输出.

$d_k = [d_{k1}, d_{k2}, \dots, d_{kp}]$, ($k = 1, 2, \dots, N$) 为期望输出.

$\eta(n)$ 为学习速率; n 为迭代次数.

$\eta L = [\eta L_1, \eta L_2, \dots, \eta L_L]$, 为步长搜索向量.

c 为每次大循环选择的样本的序号 ($c = 1, 2, \dots, N$). Flag 为随机标志.

(2) 初始化,赋给 $W_{MI}(0)$, $W_{IP}(0)$ 各一个较小的随机非零值. $k = 0$. Flag = true.

(3) 若 Flag = true,则为 c 赋一个 $(1 - N)$ 的随机值;否则 c 不变.

(4) 输入样本 X_k , $n = 0$. ($k = 1, 2, \dots, N$).

(5) 对输入样本 X_k ,前向计算 BP 网络每层神经元的输入信号和输出信号 v .

$$\text{隐层: } Net_i = \sum_{m=1}^M w_{mi} x_{km};$$

$$v_i(n) = \frac{1}{1 + e^{-Net_i}} \quad (i = 1, 2, \dots, I)$$

$$\begin{aligned} \text{输出层: } Net_p &= \sum_{i=1}^I w_{ip} v_i(n); y_p(n) = v_p(n) \\ &= \frac{1}{1 + e^{-Net_p}} \quad (p = 1, 2, \dots, P) \end{aligned}$$

(6) 由期望输出 d_k 和上一步求得的实际输出 $Y_k(n)$ 计算误差 $E_k(n)$,判断其是否满足要求,若满足转至(9),不满足转至(7).

(7) 判断, $n + 1$ 是否大于最大迭代次数, 若大于转至(9), 若不大于, 对输入样本 X_k , 反向计算每层神经元的局部梯度 δ .

输出层: $\delta_p(n) = y_p(n)(1 - y_p(n))(d_p(n) - y_p(n))$ ($p = 1, 2, \dots, P$)

隐层: $\delta_i(n) = v_i(n)(1 - v_i(n)) \sum_{p=1}^P \delta_p(n) w_{ip}(n)$ ($i = 1, 2, \dots, I$)

若 $c = k$, 则使用搜索较优步长算法如下:

隐层: $Net_{il}(n) = \sum_{m=1}^M (w_{mi}(n) + \eta(n) \cdot \eta L_l \cdot \delta_i(n) \cdot x_{km}(n)) x_{km}(n)$

$$v_{il}(n) = \frac{1}{1 + e^{-Net_{il}}}$$

输出层: $Net_{pl}(n) = \sum_{i=1}^I (w_{ip}(n) + \eta(n) \cdot \eta L_l \cdot \delta_p(n) \cdot v_i(n)) v_{il}(n)$

$$y_{pl}(n) = v_{pl}(n) = \frac{1}{1 + e^{-Net_{pl}}}$$

$$E_{kl}(n) = \frac{1}{2} \sum_{p=1}^P (d_p - y_{pl})^2$$

选取使样本误差 $E_{kl}(n)$ 最小的 ηL_l ,

$$\eta(n+1) = \eta(n) \cdot \eta L_l.$$

(8) 按下式计算权值修正量, 并修正权值: $n = n + 1$, 转至(5). 若 $c < k$, 则:

$$w_{ip}(n+1) = w_{ip}(n) + \eta \delta_p(n) v_i(n)$$

$$w_{mi}(n+1) = w_{mi}(n) + \eta \delta_i(n) x_{km}(n)$$

若 $c = k$, 则:

$$w_{ip}(n+1) = w_{ip}(n) + \eta(n+1) \delta_p(n) v_i(n)$$

$$w_{mi}(n+1) = w_{mi}(n) + \eta(n+1) \cdot \delta_i(n) x_{km}(n)$$

(9) 若 $k = N$, 则判断是否所有的训练样本的误差都满足允许误差, 是则结束; 否则 $k = 1$, 若 E 与比前一次大循环结束时相比上升, 则 $\text{Flag} = \text{ture}$, 若 E 与比前一次大循环结束时相比下降或不变, 则 $\text{Flag} = \text{false}$, 转至(3). 若 $k \neq N$, 则 $k = k + 1$, 转至(4).

4 仿真结果及分析

为了考察搜索较优步长 BP 算法的改进效果, 使用 Visual C++ 编程实现搜索较优步长 BP 算法和传统 BP 算法, 并使用南京某路口一天中 8 点到 22 点每小时的车流量作为样本, 在相同的精度下, 让两个算法分别学习^[6-7].

传统 PB 算法, 在步长为 0.2 时, 迭代总次数达

到 13336708 次, 约是步长为 20.0 时迭代总次数的 37 倍. 在步长为 0.5 时, 迭代总次数达到 4460206 次, 约是步长为 20.0 时迭代总次数的 12 倍. 可以看出迭代总次数的差距十分明显, 反映了传统 BP 算法的不稳定. 而文中算法在初始步长为 0.2 时, 迭代总次数为 330999 次, 约是步长为 20.0 时迭代总次数的 1.3 倍. 在初始步长为 0.5 时, 迭代总次数达到 288567 次, 约是步长为 20.0 时迭代总次数的 1.1 倍. 可以看出不同初始步长条件下的迭代总次数之间没有明显差距, 反映了新算法是比较稳定的.

通过仿真结果的分析可以看出, 传统 BP 算法对步长的选择十分敏感, 步长选择不合理时, 迭代总次数会非常大. 搜索较优步长 BP 算法对于初始步长的选择依赖性不强, 经过一段时间的调整, 可以得到一个相对合理的步长. 从稳定性上看, 搜索较优步长 BP 算法对比传统 BP 算法还是有一定的优越性的.

5 结束语

文中针对传统 BP 学习算法收敛速度慢, 对步长依赖明显等缺点, 提出一种利用搜索较优步长的 BP 算法. 交通流量预测仿真结果显示新算法使学习步长具有自适应性, 经过一段时间的调整就可以得到一个相对合理的步长, 同时新算法的运行比传统 BP 算法更加稳定.

参考文献:

- [1] 杨居义. 基于 BP 网络的交通运输能力预测[J]. 微电子学与计算机, 2008, 25(9): 202-205.
- [2] 梁曼君, 石竹. 提高 BP 神经网络学习速度的算法研究[J]. 计算机应用与软件, 1995(4): 59-61.
- [3] 李歧强, 顾永如, 钱积新. 神经网络的具有自适应动量和步长的伪牛顿算法[J]. 信息与控制, 1998, 27(2): 146-151.
- [4] 叶东毅, 陈昭炯. 一种基于模型逼近度的变步长 BP 算法[J]. 福州大学学报: 自然科学版, 1995, 23(2): 26-29.
- [5] 李众立, 王成端. 神经网络自适应学习步长研究[J]. 电子科技大学学报, 1996, 25(6): 643-647.
- [6] 孙佰清, 潘启树, 冯英俊, 等. 提高 BP 网络训练速度的研究[J]. 哈尔滨工业大学学报, 2001, 33(8): 439-441.
- [7] 张伟. 基于神经网络的数字识别的研究[J]. 微电子学与计算机, 2006, 23(8): 150-152.

作者简介:

- 鄢洪江 男, (1983-), 硕士研究生. 研究方向为神经网络.
 宫宁生 男, 副教授. 研究方向为神经网络.
 胡 斌 男, (1982-), 硕士研究生. 研究方向为神经网络.