

对于模糊 ARTMAP 的研究及改进

刘进涛,王万森

(首都师范大学 信息工程学院,北京 100037)

摘要:在模糊 artmap 网络中, F_2 层每增加一个的神经元就表示增加了一个聚类,这是该网络的优点同时也增加了网络的负荷。文中对模糊 artmap 算法进行了简要介绍并分析了算法的收敛复杂度。在此基础上,提出数据集划分法以提高网络的收敛速度。最后通过试验比较算法改进前后的网络收敛速度的差异,证明改进后网络的收敛速度明显提高。

关键词:模糊 artmap; 预警系数; 数据划分

中图分类号: TP183

文献标识码: A

文章编号: 1673-629X(2006)11-0016-03

Analysis and Improvement of Fuzzy ARTMAP

LIU Jin-tao, WANG Wan-sen

(Information Engineering College of Capital Normal University, Beijing 100037, China)

Abstract: One of the properties of fuzzy artmap, which can be both an asset and a liability, is its capacity to produce new neurons (templates) on demand to represent classification categories. This property allows fuzzy artmap to automatically adapt to the database without having to arbitrarily specify network structure. Provide one method for speeding up the fuzzy artmap algorithm, the method referred to as the data partitioning partitions the data into subsets for independent processing. Provide experimental results on a Beowulf cluster of workstations for the approach that confirms the merit of the modifications.

Key words: fuzzy artmap; vigilance parameter; data partitioning

0 引言

模糊 artmap 网络是由 Carpenter 等于 1991 年提出的,它是一种有导师聚类算法,具有收敛速度快、能够实现实时学习的特点^[1,2]。模糊 artmap 网络中 F_2 层的每个神经元代表一个的聚类,这使得网络不必事先确定具体的网络结构;但是在输入数据量很大时,算法的收敛速度会变得很慢。文中提出了数据划分的方法,以解决在海量输入数据情况下网络收敛慢的问题。

1 模糊 artmap 网络

1.1 网络结构

模糊 artmap 网络是一个四层网络^[2](见图 1)。其中比较重要的是输入层 F_1 、聚类层 F_2 、输出层 F_2' 。输入层的输入向量 I 的维数为 $2M_a$:

$$I = (a, a^c) = (a_1, a_2, \dots, a_{M_a}, a_1^c, a_2^c, \dots, a_{M_a}^c) \quad (1)$$

其中 $a_i^c = 1 - a_i$; $\forall i \in \{1, 2, \dots, M_a\}$, 且向量 a 的每个元素都属于区间 $[0, 1]$ 。

模糊 artmap 网络中有两个重要的权重矩阵。第一个是从 F_2 到 F_1 的权重矩阵 W_{ij}^a , 其中 $1 \leq j \leq N_a, 1 \leq i \leq M_a$ 。从 F_1 层的所有结点到 F_2 层的结点 j 的联接权重表示成向量为: $W_j^a = (w_{j1}^a, w_{j2}^a, \dots, w_{j,2M_a}^a)$, 该向量就表示一个聚类类别。第二个值得注意的矩阵是从 F_2 层的结点 j 发散到 F_2' 层的结点 k 的权重矩阵 W_{jk}^{ab} 。从 F_2 层的结点 j 发散到 F_2' 的所有结点的权重向量可表示为: $W_j^{ab} = (W_{j1}^{ab}, W_{j2}^{ab}, \dots, W_{j,N_a}^{ab})$ 。

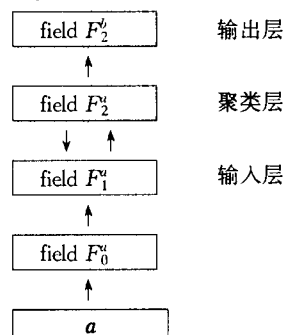


图1 artmap 网络结构

模糊 artmap 网络包含两个重要的参数——选择参数 β_a 和基线预警参数 $\bar{\rho}_a$ 。 β_a 的取值范围是 $(0, \infty)$, $\bar{\rho}_a$ 的取值范围是 $[0, 1]$ 。这两个参数值的选取直接影响着 F_2 层中结点的个数。当两个参数值都比较大时, F_2 层将产生更多的结点。

收稿日期: 2005-12-27

基金项目: 北京市自然科学基金资助项目(4032009)

作者简介: 刘进涛(1977-), 男, 河北人, 硕士研究生, 主要研究方向为人工智能、数据挖掘; 王万森, 教授, 博士研究生, 主要研究方向为人工智能、数据挖掘、不确定推理。

1.2 学习算法

首先对两个权重矩阵进行初始化,初始化矩阵 W_{ij}^a 的每个元素为 1, 矩阵 W_{jk}^b 的每个元素为 0。为了叙述方便, 介绍一下模糊运算的定义^[3]:

模糊“与”($\cdot \wedge \cdot$): $x_i \wedge y_i \equiv \min(x_i, y_i)$, x_i, y_i 分别是向量 x, y 的相对应的元素。

模糊“或”($\cdot \vee \cdot$): $x_i \vee y_i \equiv \max(x_i, y_i)$, x_i, y_i 分别是向量 x, y 的相对应的元素。

模糊“模” $|\cdot|: |p| = \sum_{i=1}^m (p^i)$, m 是向量 p 的维数。

学习算法步骤如下:

1) 通过公式(2)计算输入向量 I 和 F_2 层中结点 j 的选择函数值 T_j^a ($j=1, 2, \dots, N_a$)。系统设定的获胜类别条件为 $T_j = \max |T_j: j=1, 2, \dots, N_a|$, 其中 J 表示获胜神经元序号, 如果有多个 T_j 符合条件, 则选择序号最小的作为获胜神经元。

$$T_j^a = \frac{|I \wedge W_j^a|}{|W_j^a| + \beta_a} \quad (2)$$

2) 检测获胜的节点 j_{\max} 是否符合预警标准, 即公式(3)。如果符合则继续步骤 3), 否则就选取选择函数值序列中下一个结点, 然后重复步骤 2), 直到找到符合预警标准的结点 j_{ρ} 为止。

$$\frac{|I \wedge W_{j_{\max}}^a|}{|I|} \geq \rho_a \quad (3)$$

3) 检测 j_{ρ} 是否能在 F_2^b 层找到理想的输出向量 O' 。如果找到了理想的输出向量, 则对 $W_{j_{\max}}^a$ 和 $W_{j_{\max}}^b$ 进行调整, 如公式(4); 若没有找到理想的输出向量, 则对 ρ_a 进行调整, 如公式(5), 然后选取选择函数值序列中下一个结点, 重复步骤 2)。

$$W_{j_{\max}}^a \leftarrow W_{j_{\max}}^a \wedge I, W_{j_{\max}}^b \leftarrow O' \quad (4)$$

$$\rho_a \leftarrow \frac{|I \wedge W_{j_{\max}}^a|}{|I|} + \epsilon, \text{ 其中 } \epsilon \text{ 是非常小的正数} \quad (5)$$

2 复杂度分析

2.1 线性复杂度分析

图 2 给出了模糊 artmap 算法的伪码形式^[4]。训练集中的输入模式至少要被网络使用一次, 每个输入模式被网络使用的平均次数为 Γ , 则每个输入模式的运算次数为:

$$\text{Time}(I) = O(\Gamma \times |\text{templates}|) \quad (6)$$

在 F_2^b 层中结点数目不改变的情况下, 算法的复杂度很容易表示为:

$$\text{Time}(\text{FAM}) = O(\Gamma \times P \times |\text{templates}|) \quad (7)$$

如果训练集中的数据是固定的, 则 F_2^b 层中结点数目与训练集中数据的数目有一定的比例关系, 这里把这个比例系数称为压缩率 k , 用公式可以表示为:

$$|\text{templates}| = kP \quad (8)$$

最后算法的时间复杂度可以表示为:

$$O(\text{FAM}) = O(\Gamma P k P) = O(k \Gamma P^2) \quad (9)$$

FAM-LEARNING-PHASE(PATTERNS, $\bar{\rho}_a, \beta_a, \text{maxEpochs}$)

```

1 templates ← {}
2 iter ← 0
3 repeat
4     modified ← false
5     for each I in Patterns
6         do  $\rho_a \leftarrow \bar{\rho}_a$ 
7         LEARN-PATTERN(I, templates,  $\rho_a, \beta_a$ )
8         iterations ← iterations + 1
9
10 until (iter = maxEpochs) or (modified = false)
11 return templates

```

Where the procedure LEARN-PATTERN is:

LEARN-PATTERN(I, templates, ρ_a, β_a)

```

1 repeat
2     status ← FoundNone
3     jmax ← GET-MAX-ARG(I, templates,  $\rho_a, \beta_a$ )
4     if status = Foundone
5         then if class(I) = class( $w_{j_{\max}}^b$ )
6             then
7                 status ← ThisIsIT
8             else
9                 status ← Matchtracking
10             $\rho \leftarrow \rho(I, w_{j_{\max}}^b) + \epsilon$ 
11 until status ≠ Matchtracking

```

图 2 模糊 artmap 算法

2.2 非线性复杂度分析

在没有任何假设的前提下, 模糊 artmap 算法的时间复杂度可以表示为:

$$O \left(\underbrace{\sum_{i=1}^{\text{epochs}} k_i \Gamma P^2}_{\chi} \right) \quad (10)$$

假设集合 A 符合下面 3 个条件:

- 1) 训练次数与训练集中数据的多少直接相关;
- 2) Γ_i 不随着迭代的次数变化, 但它是 k_i 的线性函数;
- 3) $k_i = k^i$, 此公式的含义是聚类数目随着迭代次数的增加而减少。

在这 3 个条件的基础上, 公式化简为:

$$A = \sum_{i=1}^{\text{epochs}} k^i \Gamma P^2 = \Gamma P^2 \sum_{i=1}^{\text{epochs}} k^i = \Gamma P^2 k \left(\frac{1 - k^{\text{epochs}(P)}}{1 - k} \right) \quad (11)$$

从公式(11) 可是看出, 模糊 artmap 算法的收敛速度是训练集中输入模式数 P 的二次方程。这就说明了通过减少输入模式中输入模式的数目就可以减少算法的复杂度。文中接下来的部分, 就是通过划分训练集减小参数 P 。

3 模糊 artmap 网络的改进

通过采用划分数据空间的方法, 减小网络的输入模式数和聚类数目。数据划分就是将原来的数据集分成多个小的数据集, 并用这些小子数据集去训练不同的 artmap 网络。这种方法的优点是它降低了原算法的时间复杂度, 同时取得了很好的并行性。

在前面提到的 3 个前提下, 模糊 artmap 网络的聚类数是和输入模式数目成比例的。所以模糊 artmap 网络的收敛的时间复杂度可描述成 $O(n^2)$, 其中 n 表示训练集中输入模式数。通过划分训练集加快了算法的收敛速度。例如, 把训练集平均分成 p 等分, 每一份是 n/p , 通过数据的划分在理论上可以产生一个 p^2 的收敛加速度。

输入数据被设定在 M_a 维至 \bar{M}_a 的数据之间, 其中 $\bar{M}_a \ll M_a$ 。在这里对数据空间的数据进行区域划分。例如: 在一个三维空间中, 把每一维分成 10 份, 就相当于把三维空间的数据装入到 1000 个盒子里。如果用每个盒子中的数据训练不同的 artmap 网络, 则可以肯定保证每个网络的聚类类数不是很大。但可能存在这样的情况, 数据划分后训练的不同网络的聚类的总和大于没有进行数据划分训练的网络的聚类数。

每个模糊 artmap 网络中的聚类数是由每个数据区域中的数据决定的。同时每个数据区域的大小与预警参数 $\bar{\rho}_a$ 和输入模式的维数 M_a 有关, 尤其是在: $\text{size}(w_j^i) \leq M_a (1 - \bar{\rho}_a)$ 。在预警参数值比较大时, 就可以很好地解决上面所提到的问题。

4 应用实例

文中采用由 Jock A. Blackard 创建的 Coverttype 数据库作为测试的数据源^[5]。测试集是从该数据库中选取了 20000 个数据对作为输入模式对。在训练过程中预警系数 ρ_a 被设定为 0.96。下面两图是算法改进前后的网络的错误率(见图 3)和收敛时间(见图 4)^[6]的比较。

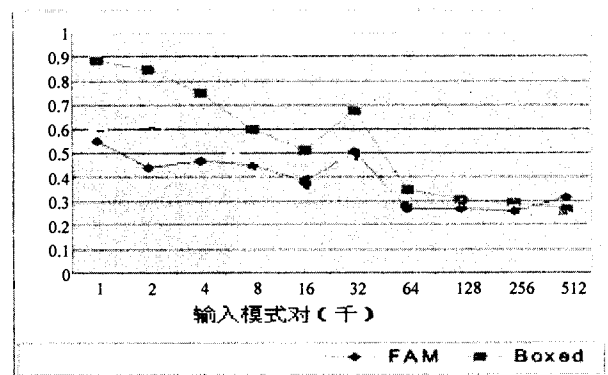


图 3 错误率比较

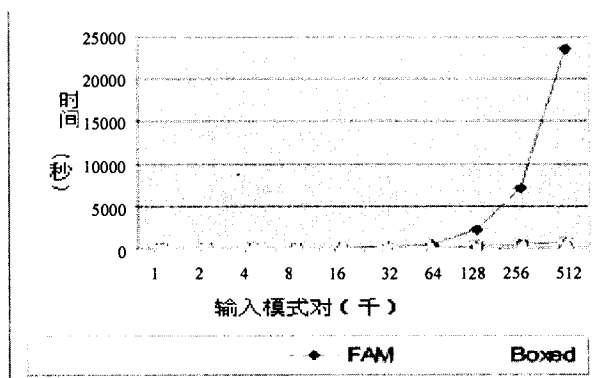


图 4 收敛时间比较

5 结论

上述事例表明, 改进后的模糊 artmap 网络在训练集中数据量较大的情况下, 网络的收敛时间明显缩短, 同时错误率明显降低。

参考文献:

- [1] Carpenter G A, Grossberg S, Reynolds J H. ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network[J]. Neural Networks, 1991(4):565-588.
- [2] Carpenter G A, Grossberg S, Markuzon N, et al. Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps[J]. IEEE Transactions on Neural Networks, 1992(3):698-713.
- [3] 秦安文. 模糊 ARTMAP 神经网络在一维信息聚类分析中的应用[J]. 山东工业大学学报, 1999(5):56-60.
- [4] Carpenter G A, Grossberg S, Reynolds J H. Fuzzy ART: An adaptive resonance algorithm for rapid, stable classification of analog patterns[C]//In International Joint Conference on Neural Networks, IJCNN'91. Seattle, Washington: IEEE/INNS/ENNS, 1991:411-416.
- [5] University of California, Irvine. UCI machine learning repository[EB/OL]. 2003. <http://www.icf.ucf.edu/mllearn/ML-Repository.html>.
- [6] 飞思科技产品研发中心. 神经网络理论与 MATLAB7 实现[M]. 北京: 电子工业出版社, 2005.

(上接第 15 页)

供了更加灵活的手段和新的思路。用 XML 技术所实现的页面信息无痕嵌入的方法正是这样一个有现实意义的研究尝试。相信随着 XML 在各领域应用的进一步推广, 越来越多针对 XML 在网页设计方法和网站建设应用技术的研究将带来 Internet 上的一场革命。

参考文献:

- [1] W3C Recommendation. Extensible Markup Language (XML) 1.0 (Third Edition)[EB/OL]. 2004-02-04. <http://www.w3.org/TR/2004/REC-xm1>.

- [2] W3C Recommendation. Extensible Stylesheet Language(XSL) Version 1.0[EB/OL]. 2001-10-15. <http://www.w3.org/TR/xsl/>.
- [3] 许斗, 陈恩红. 用 XML 表示和发布图文混合信息[J]. 计算机应用研究, 2001, 18(11):29-31.
- [4] Holzner S. Real World XML[M]. 2nd Edition. Berkeley: Peachpit Press, 2003.
- [5] Aitken P G. 微软 XML 技术指南[M]. 谢君英译. 北京: 中国电力出版社, 2003.