

# 快速选定神经网络优化结构的方法

杨钟瑾 史忠科

(西北工业大学 西安710072)

**摘要** 本文引荐了一种选择神经网络优化结构的改进方法。常规交叉校验算法从大量候选模型中,通过训练和比较来搜寻网络优化结构。本文提出一种与规则化方法相关的快速算法,这种算法仅在少许适当的候选模型中选定网络优化结构。实验证明,这种快速算法与常规交叉校验算法相较而言,具有更快捷和更高效的优点。

**关键词** 交叉校验,规则化,结构优化,自底向上,自顶向下,遗传算法,前馈神经网络,归纳,反向传播算法

## Fast Approach for Optimal Network Architecture Selection

YANG Zhong-Jin SHI Zhong-Ke

(Northwestern Polytechnical University, Xi'an 710072)

**Abstract** In this paper a new method for selecting optimal network architecture is proposed. The formal Cross Validation (CV) performs a search of optimal architecture through large-scale models. We present a fast approach, which corresponds to Regularization, accomplishes selection of optimal network architecture by appropriate scale models. Comparative studies with formal CV, experimental results show that the fast selecting algorithm provides improvement of learning capability and evidences efficient.

**Keywords** Cross validation, Regularization, Architecture optimization, Bottom-up, Top-down, Genetic algorithm, Feedforward neural networks, Generalization, Backpropagation

## 1 引言

神经网络具有通过学习获得解决问题智能的特性,从而吸引众多的研究者与应用者。当古典参数技术对许多的自然与社会现象的建模束手无策时,神经网络技术却提供了捷径。神经网络具有模式识别、函数逼近、优化和联想记忆等性能。

前馈神经网络是解决非线性问题的极好模型<sup>[1]</sup>,它们的训练通过梯度下降法优化路径来实现<sup>[2]</sup>。在历史上,人们经常使用反向传播算法(BP算法)来训练前馈神经网络<sup>[3,4]</sup>。BP算法是被Rumelhart, Hinton和Williams发现和提出的<sup>[5]</sup>。BP算法有致命的缺点:容易陷入局部最小值,对初始条件过于敏感和收敛速度缓慢。但是,如果BP算法具备适合的初始条件和适当的网络结构,那么它就能充分发挥优势而且出色完成各项事务。随便给出初始条件和随意猜测网络结构的情形,都会导致迟缓的、繁杂的训练过程。理论研究<sup>[6]</sup>和仿真试验<sup>[7]</sup>都充分证明了以下两种事实:太大的网络结构容易造成归纳性差;太小的网络结构则造成学习能力低。因此,一种能快速且准确地指定神经网络结构的方法值得探究。

通过优化神经网络结构的方法,从而使所得网络的结构更简单、学习速度更快和归纳性能更好。前馈神经网络的结构优化包括:1)选择适当的输入神经元、隐含神经元和输出神经元;2)明确各神经元的非线性组合方式;3)确定各神经元之间的连接权数量。众所周知,单一的神经网络结构不可能适合所有问题<sup>[8]</sup>。在通常的神经网络结构选择过程中,除了具备预先了解问题复杂性的前提外,还要涉及许多的选择试验和错误。近年来,相继有一些结构优化方法被提出:1)规则化,由模型查找优化结构,主要通过训练和比较不同网络结构的途径来实现;备受推崇的方法有交叉校验<sup>[9]</sup>(CV)。2)自底向上方法,从相对小的结构开始,然后逐渐增加隐含神经元数量,从而增

强网络性能;著名的构造算法有瀑流关联,它是由Fahlman<sup>[10]</sup>提出的。3)自顶向下方法,与自底向上方法相反,它从臆断为足够复杂的结构开始,通过训练而尽可能裁减冗余的输入神经元、隐含神经元和输出神经元;修剪<sup>[11]</sup>就是其中一种有代表性的方法;而在修剪技术中,由Hassibi和Stork<sup>[12]</sup>提出的神经中枢手术优化算法(OBS)为佼佼者。4)遗传算法<sup>[13]</sup>,运用群体搜索技术,通过对当前神经网络种群实施选择、交叉和变异等一系列遗传操作,从而产生新子代群体,然后从中筛选具有最好性能的神经网络个体。

交叉校验是一种规则化方法<sup>[14]</sup>,针对不同的网络结构大小来调整规则水准,并通过比较和评估候选模型,从而决定最佳模型。与其它方法相较而言,交叉校验有不俗的特点:1)在海量数据处理和解决复杂问题时,交叉校验方法更胜一筹;2)拥有全局数据处理的优点;3)运用交叉校验方法选择网络优化结构直截了当。然而,为了网络优化结构的获得,常规交叉校验算法通常在大范围的候选模型中搜寻,这就造成昂贵的计算花销。为了使CV算法具有更直接和更高效的特性,本文提出了一种从少许适当的候选模型中选定网络优化结构的改进算法。

本文第2节详细叙述改进的交叉校验算法;第3节介绍改进算法的实验仿真结果;最后给出相关结论。

## 2 快速CV算法

我们希望神经网络经由学习来提高归纳性能。鉴于此点,可将网络优化结构的选择视为:依据规则标准,网络通过学习,从众多的候选模型中,选取最优结构。交叉校验算法在学习过程中,由比较与评估候选模型来选择最优结构。现将网络优化结构的选择归类为组合优化问题:

$$\begin{aligned} & \min E(A_L) \\ & s. t. A_1 \subset A_2 \subset \dots \subset A_L \end{aligned}$$

$$A_l \in \{F(x, w) : w \in W_l\}, l=1, 2, \dots, L$$

其中  $E$  为目标函数, 目标欲使所选定的网络结构的归纳性误差最小; 约束条件为候选模型的有限数目共是  $L$  个, 各个候选模型的网络结构之间有如下关系:  $A_1 \subset A_2 \subset \dots \subset A_L$ ; 决策变量  $A_l$  为第  $l$  个候选模型的网络结构, 表示具有相似结构和权向量  $w$  的网络集合, 其中  $w \in W_l, W_l$  是多维权向量空间, 并且  $A_l$  中的每一个元素满足输入-输出映射函数  $F(x, w)$ , 其中  $x$  代表输入向量。

首先用 BP 算法对候选模型进行训练评估, 尔后将已训练的神经网络进行校验, 目标是选取具有最小归纳性误差的网络优化结构。为了实现该目标, 训练样本数据集被随机地分割为两个相互独立子集: 评估子集和校验子集。在此, 用  $D$  代表整个训练样本数据集,  $T^m$  代表第  $m$  轮训练时的评估子集,  $V^m$  代表第  $m$  轮训练时的校验子集。假设训练样本数据集包含  $N$  种样本,  $r$  是决定训练样本数据集  $D$  分割的参数 ( $0 < r < 1$ )。这样,  $|T^m| = [(1-r)N], |V^m| = [rN]$ 。它们满足:

$$(\forall m) (1 \leq m \leq t \rightarrow T^m \subset D \wedge V^m \subset D)$$

$$D = T^m \cup V^m \quad T^m \cap V^m = \phi$$

在这里,  $t$  代表整个训练次数。每一个评估子集  $T^m$  进一步分割成 5 个独立子集  $T_i^m$ , 关系如下:

$$(\forall i) (1 \leq i \leq 5 \rightarrow T_i^m \subset T^m) \quad T^m = \bigcup_{i=1}^5 T_i^m$$

$$(\forall i, j) (1 \leq i, j \leq 5 \wedge i \neq j \rightarrow T_i^m \subset T^m \wedge T_j^m \subset T^m \wedge T_i^m \cap T_j^m = \phi)$$

同样, 每一个校验子集  $V^m$  进一步分割为 2 个独立子集  $V_q^m$ , 关系如下:

$$(\forall q) (1 \leq q \leq 2 \rightarrow V_q^m \subset V^m) \quad V^m = \bigcup_{q=1}^2 V_q^m$$

$$(\forall q, s) (1 \leq q, s \leq 2 \wedge q \neq s \rightarrow V_q^m \subset V^m \wedge V_s^m \subset V^m \wedge V_q^m \cap V_s^m = \phi)$$

衡量  $A_l$  网络结构性能好坏的标准是依据其平均归纳性误差的大小, 平均归纳性误差  $E_{avg}$  有以下等式:

$$E_{avg}(A_l^m) = \frac{1}{10} \sum_{q=1}^2 \sum_{i=1}^5 E_{q_i}^m$$

其中,  $E_{q_i}^m$  表示  $A_l$  网络结构在第  $m$  轮训练时, 用评估子集  $T_i^m$  训练后, 再经校验子集  $V_q^m$  进行校验所得的均方误差。

常规 CV 算法的主要问题是候选模型范围过大。由于资源和时间的因素, 另外还有梯度递减算法固有缺陷的限制, 因此, 我们不得不考虑候选模型的范围。为了使 CV 算法更直接和更高效, 我们提出了定义候选模型适当范围的方法。虽然神经网络结构由许多因素决定, 但是隐含神经元却占有举足轻重的地位。隐含神经元在前馈神经网络中扮演重要角色—特征探测器, 在神经网络学习过程中, 隐含神经元不断地使训练样本数据特征化, 并将此特征记忆。在此, 我们通过改变隐含神经元的数量来改变网络结构。相关证明<sup>[15]</sup>得出: 具有 S 形函数的非线性三层神经网络结构, 可以逼近任意的非线性函数; 但是, 在这种只有单层隐含层的神经网络里, 各神经元相互影响。面对复杂情况, 这种相互影响成为改进函数逼近性的障碍; 可是, 拥有两层隐含层的神经网络更易于改进函数逼近性。Blum 和 Li<sup>[16]</sup>实验证明: 拥有两层隐含层的神经网络, 具有全局函数逼近性。由此, 我们在实验仿真过程中, 只针对四层神经网络结构进行研究, 并特定所有的隐含神经元使用 S 形函数。快速 CV 算法仅在小范围的候选模型中选定网络优化结构, 该范围为: 拥有  $INN$  个输入神经元的输入层, 其中  $INN$  视实际情况而定; 拥有  $H1N$  个隐含神经元的第二隐含层, 其中  $H1N \in [[2^{(m_0-1)}/m_0+1], [2^{m_0}/m_0+1]]$ ,  $m_0$  代表

网络中输入神经元数目; 拥有  $H2N$  个隐含神经元的第二隐含层, 其中  $H2N \in [[2^{(m_0-2)}/(m_0+2)+1], [[2^{(m_0-2)}/(m_0+2)+6]]$ ; 输出层有  $OUN$  个输出神经元, 其中  $OUN$  由实际情况定值。

以下是快速 CV 算法的概要:

- 步骤1 确定初始神经网络与明确候选模型的范围; 将初始神经网络作为当前候选模型。
- 步骤2 用评估子集  $T^m$  训练当前候选模型, 尔后经校验子集  $V_q^m$  进行校验并计算平均归纳性误差  $E_{avg}$ 。
- 步骤3 用评估子集  $T^m$  训练新候选模型, 尔后经校验子集  $V_q^m$  进行校验并计算平均归纳性误差  $E_{avg}$ 。
- 步骤4 比较当前候选模型与新候选模型的平均归纳性误差  $E_{avg}$  的大小, 将  $E_{avg}$  为小的候选模型保存并作为当前候选模型。
- 步骤5 在确定的范围内是否仍有新的候选模型? 若有, 则返回执行步骤3; 否则终止, 所得的当前候选模型就是要选定的优化网络结构。

### 3 仿真实验

在解决实际问题的初始, 网络的优化结构对于我们来讲是个未知数, 我们的目标是获得尽可能简单的结构和保持良好的性能。在这种情况下, 我们用归纳性误差作为衡量所选择网络性能好坏的标准。为了考察快速 CV 算法基于前馈神经网络的性能, 我们通过解决实际问题来进行。数据资料来源于十大公司在深市交易的原始数据, 数据记录时间为: 从 1991 年 1 月 2 日至今。实验对象是股票涨或跌的问题, 目标是分清股票的走势。在仿真实验中, 用“1”表示涨势, 用“0”表示跌势。实验对象有七个真值输入神经元和一个输出神经元:

输入神经元:

- OPT: 当日开盘价较昨日开盘价之走势;
- HPT: 当日最高价较昨日最高价之走势;
- LPT: 当日最低价较昨日最低价之走势;
- CPT: 当日收盘价较昨日收盘价之走势;
- TMT: 当日成交总金额较昨日成交总金额之走势;
- TVT: 当日总成交量较昨日总成交量之走势;
- CIT: 当日综合指数较昨日综合指数之走势。

输出神经元:

- SCT: 明日走势。

选用从 1996 年 7 月至 2001 年 6 月的原始数据作为实验数据, 针对每个公司实验所用的训练样本集包含 1200 种样本,  $r$  的取值范围为:  $0.15 \leq r \leq 0.3$ 。在这里, 评估子集  $T^m$  包含 1000 种样本, 校验子集  $V^m$  包含 200 种样本。根据实际情况, 我们定义初始神经网络结构(输入神经元数目-第一层隐含神经元数目-第二层隐含神经元数目-输出神经元数目)如下:  $7-10-4-1$ ; 实验所选定的候选模型范围为:  $INN=7, H1N \in [10, 19], H2N \in [4, 9], OUN=1$ 。

针对 BP 算法易陷入局部最小值和收敛速度缓慢的缺点, 我们采用了以下改善方法: 1. 随机读入样本数据; 2. 选择一些小的随机值作为初始权值和偏置值; 3. 改变学习速度。

表1 性能参数对照表

问题	方法	选定网络优化结构	$E_{avg}$	$T_{avg}$
股票 涨或跌	常规 CV 算法	7-16-7-1	0.1223	32.2 (M)
	快速 CV 算法	7-16-7-1	0.1248	1.1 (M)

$E_{avg}$ : 平均归纳性误差  $T_{avg}$ : CPU 平均运行时间

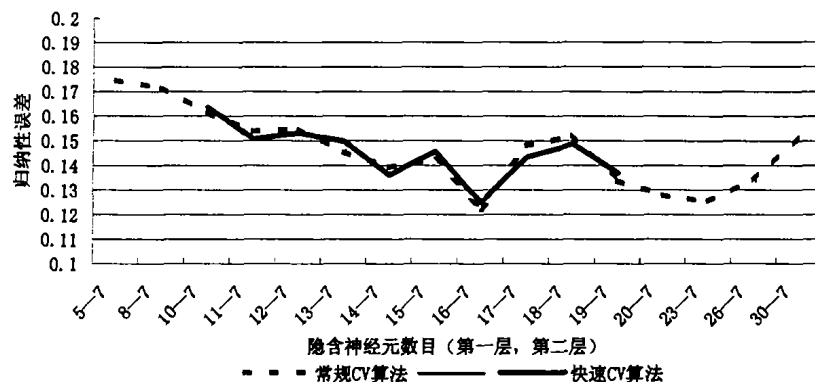


图1 归纳性误差对照图

通过仿真实验,我们获得快速 CV 算法的相关实验数据,并将这些数据与常规 CV 算法比较。表1给出了快速 CV 算法与常规 CV 算法在实验中所获得的性能参数对照结果。表1的第一列说明所要解决的问题;第二列指出实验所采用的算法;第三列给出实验最后所选定的网络优化结构;“ $E_{avg}$ ”列分别标明快速 CV 算法与常规 CV 算法在采用选定的网络优化结构时,运用各组实验数据进行验证后的平均归纳性误差;“ $T_{avg}$ ”表示 CPU 平均运行时间。表1的第三列显示快速 CV 算法与常规 CV 算法最后所选定的网络优化结构一样,虽然“ $E_{avg}$ ”项表明快速 CV 算法的平均归纳性误差比常规 CV 算法的差 0.0025,但是在“ $T_{avg}$ ”项却充分体现了快速 CV 算法的优势:CPU 平均运行时间比常规 CV 算法快 31.1 分钟。图1显示了快速 CV 算法与常规 CV 算法的平均归纳性误差参照结果。在图1中,我们仅仅给出部分网络结构的平均归纳性误差,但具有代表性。图1直接明了地表达网络结构为:7-16-7-1 的神经网络的平均归纳性误差最小。

由仿真实验结果表明,与常规 CV 算法相比,快速 CV 算法有如下优良性能:CPU 运行时间更短,网络收敛速度更快。

**结论** 本文对 CV 算法进行研究,提出和阐述常规 CV 算法的改进方法。实验结果显示,合理选择候选模型的适当范围,令常规 CV 算法扬长避短,从而使快速 CV 算法在处理复杂问题时,更实用和更有效。未来研究重点是将其改进方法推广应用到不同类型的神经网络。

### 参考文献

- Haykin S. Neural Networks, A Comprehensive Foundation. New York: Macmillan, 1994
- Fletcher R. Practical Methods of Optimization. New York: Wiley, 1990
- 史忠科. 神经网络控制理论. 西安:西北工业大学出版社, 1997
- (美)哈根(Hagan, M. T.),等著,戴葵,等译. 神经网络设计. 北京:机械工业出版社, 2002
- Rumelhart D E, Hinton G E, Williams R J. Learning Internal Representations by Error Propagation. In: Rumelhart D E, McClelland J L, eds. Parallel Distributed Processing: Explanations in the Microstructure of Cognition. Cambridge, MA: MIT Press, 1986, 1:318~362
- Kung S, Fallside F, Sorenson J A, Kamm C. Neural Networks for Signal Processing. In: Proc. of the 1992 IEEE Workshop, 1992, 1:255~266
- Guyon I, Wang P S P. Special Issue on Neural Networks and Pattern Recognition. Pattern Recognition Artificial Intelligence, 1993, 7(4):849~872
- Friedman J H. An Overview of Predictive Learning and Function Approximation. In: Friedman J H, Wechsler H, eds. From Statistics to Neural Networks: Theory and Pattern Recognition Applications. Proc. of the ASI Conf. Subseries F. New York: Springer-Verlag, 1994
- Moody J. Prediction Risk and Architecture Selection for Neural Networks. In: Cherkassky V, Friedman J, Wechsler H, eds. From Statistics to Neural Networks: Theory and Pattern Recognition Applications, NATO ASI Series F. New York: Springer-Verlag, 1994, 136:147~165
- Fahlman S E, Lebiere C. The Cascade-Correlation Learning Architecture. In: Touretzky D S, ed. Advances in Neural Information Processing Systems. San Mateo, CA: Morgan Kaufmann, 1990, 2:524~532
- Reed R. Pruning Algorithms - A Survey. IEEE Trans. on Neural Networks, 1993, 4(5):740~747
- Hassibi B, Stork D, Wolff G. Optimal Brain Surgeon and General Network Pruning. IEEE International Conference on Neural Networks, 1993, 1:293~299
- Holland J H. Adaptation in Natural and Artificial Systems. Ann Arbor, MI: University of Michigan Press, 1975
- Tikhonov A N, Arsenin V Y. Solution of Ill-Posed Problems. Washington, DC: W. H. Winston, 1977
- Hetch-Nielsen R. Theory of the Backpropagation Neural Network. In: Proc. of the Intl. Joint Conference on Neural Networks, New York: IEEE Press, 1989, 1:593~611
- Blum E K, Li L K. Approximation theory and feedforward networks. Neural Net, 1991, 4:511~515