

遗传算法进化神经网络权值的新技术

张颖鹏

华南理工大学 广州 510640

e-mail: tczyp@yahoo.com.cn

摘要: 本文将介绍一种在不提高运算代价的前提下,大大提高遗传算法进化人工神经网络权值的效率和速度的原创技术:“伪隐藏层”技术。并详细讨论该技术的本质和实现过程。该技术将有效地帮助遗传算法跳出局部最优解,并增加收敛到全局最优解或者高效的次优解的可能性。

关键词: 遗传算法 神经网络 伪隐藏层

The New Technology of evolving Artificial Neural Networks Weights by Genetic Algorithms

ZHANG Ying-Peng

South China University Of Technology GuangZhou 510640

e-mail: tczyp@yahoo.com.cn

Abstract: In this text A kind of authorship technology is presented, which could improve the efficiency and velocity of evolving Artificial Neural Networks weights by Genetic Algorithm(GA) without augment the operation cost. It is “false hidden layer” technology. The essence and implementing process of this technique are discussed in detail. It could help GA to escape from local optimization value and make the probability of arriving at global optimization or local second-optimization value increased.

Key Words: Genetic Algorithm, Artificial Neural Network, false hidden layer.

1 引言

人工神经网络(ANN)是对人类大脑的一种模拟,它将人脑的一些基本特征和功能抽象出来用于解决各种现实问题。自80年代以来,ANN无论是在理论模型、学习算法还是实际应用等方面都取得了很大的进展。K. M. Hornik等人已证明,只要用一个具有单隐层的前馈网络就可以实现对任意复杂函数的任意逼近,然而如何对网络进行合理配置却是一个NP问题^[1];另一方面,传统的BP训练方法,具有收敛缓慢且容易陷入局部最优解等缺点,这些都极大地限制了ANN的应用与发展。

遗传算法(GA)是一种模拟自然选择和进化的随机搜索算法。GA适用于大规模的并行计算并能在复杂、多峰值、不可微的空间中以较大的概率找到全局最优解,因此能有效地克服传统ANN训练方法的缺点。近年来,越来越多的研究人员正从事GA与ANN相结合的研究工作^[2]。而本文将提及的“伪隐藏层”技术将进一步提高GA在复杂,多峰值的多维曲面空间中搜索全局最优解的能力,使GA成为一种更加高效的神经网络训练方法,GA的地位将大大提高。

2. “伪隐藏层”技术

2.1 被忽视的线性激活函数

传统神经网络的层与层之间的激活函数一般采用非线性激活函数，这个结论来自于下面这个定理：使用线性激活函数的多级网的功能不会超过单级网。下面给出该定理的证明。（更详细的证明过程请查阅参考文献[3]）

设有一 n 层网络， X 是其输入向量， $W^1W^2W^3\dots W^n$ 是各级连接矩阵。

$NET_1NET_2NET_3\dots NET_n$ 分别是各级的网络输入向量， $F_1F_2F_3\dots F_n$ 是各级神经元的激活函数，因为它们都是线性的，所以可以表示为：

$$F_i(NET_i) = K_iNET_i + A_i \quad 1 \leq i \leq n$$

则网络的输出向量为：

$$\begin{aligned} O &= F_n(\dots F_3(F_2(F_1(NET_1)))\dots) \\ &= F_n(\dots F_3(F_2(K_1XW^1 + A_1))\dots) \\ &= F_n(\dots F_3(K_2(K_1XW^1 + A_1) + W^2 + A_2)\dots) \\ &\dots\dots \\ &= K_n \dots K_3K_2K_1XW^1W^2W^3 \dots W^n \\ &\quad + K_n \dots K_3K_2A_1W^2W^3 \dots W^n \\ &\quad + K_n \dots K_3A_2W^3 \dots W^n \\ &\quad + K_n \dots K_{i+1}A_iW^{i+1} \dots W^n \\ &\quad + A_n \\ &= KXW + A \end{aligned}$$

$$\begin{aligned} K &= K_n \dots K_3K_2K_1 \\ W &= W^1W^2W^3 \dots W^n \\ A &= K_n \dots K_3K_2A_1W^2W^3 \dots W^n \\ &\quad + K_n \dots K_3A_2W^3 \dots W^n \\ &\dots\dots \\ &\quad + K_n \dots K_{i+1}A_iW^{i+1} \dots W^n \\ &\dots\dots \\ &\quad + K_nA_{n-1}W^n \\ &\quad + A_n \end{aligned}$$

从上述推导可知，这个多级网相当于一个激活函数为 $F(NET) = KXW + A$ ，连接矩阵为 W 的简单单级网络。显然，如果网络使用的是非线性激活函数，则不会出现上述问题。因此说，非线性激活函数是多级网络的功能超过单级网络的保证。

由于线性激活函数不能使多级网络的功能超过单层网络，所以一直不受重视，也因此隐藏层一般都不会使用线性激活函数。

2.2 看似多余的“伪隐藏层”

而“伪隐藏层”恰恰是激活函数为线性激活函数的隐藏层。由上述定理可知这种隐藏层的加入不会增加整个神经网络的表达能力，所以命名为“伪隐藏层”。这个（或几个）看似

多余的伪隐藏层虽然不能增加神经网络的表达能力但是对于遗传算法来说(至少对遗传算法来说)它能起到增生搜索空间,分割搜索空间的作用。使最优解从一个变为无数个。(当然,因为搜索空间的增大,最优解的总体概率持平。)

下面解释最优解为什么有无数个。由前面怎么的定理可知对于单层神经网络每个输出对应一个固定的 K, W, A 。假设当 $K = K', W = W', A = A'$ 时输出是最优解。那么对于添加了“伪隐藏层”的神经网络来说只要使下面的式子成立就得到最优解。

$$\begin{aligned}
 K' &= K_n \dots K_3 K_2 K_1 \\
 W' &= W^1 W^2 W^3 \dots W^n \\
 A' &= K_n \dots K_3 K_2 A_1 W^2 W^3 \dots W^n \\
 &\quad + K_n \dots K_3 A_2 W^3 \dots W^n \\
 &\quad \dots \dots \\
 &\quad + K_n \dots K_{i+1} A_i W^{i+1} \dots W^n \\
 &\quad \dots \dots \\
 &\quad + K_n A_{n-1} W^n \\
 &\quad + A_n
 \end{aligned}$$

由此可知,有无数种组合可以得到最优解。

2.3 “伪隐藏层”技术的本质

希望通过以下这个淘金比喻来形象地解释一下“伪隐藏层”的本质。

设想有两种情况供你选择。

第一种情况:在一个广袤的砂砾池里面,埋藏着一块完整的金块。现在让你找出藏匿在这个砂砾池里面的那块金块。

第二种情况:把金块削成一块块金属,散放到砂砾池里面,现在只要你找到其中的一份金属。(金子或者金属都会发光。)

我想你肯定喜欢后一种情况吧,并且希望把那块金块被削得越细越好。最好随便捉一把沙子,其中就有你要的金属。

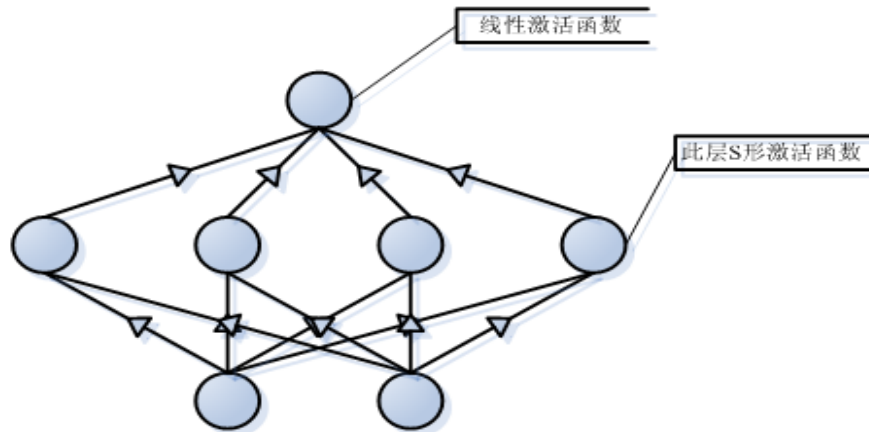
那么对于遗传算法来说,金块或者金属就是最优解,而金子的光泽覆盖的地方就是一些次优解。而把金块削开的就是“伪隐藏层”。

“伪隐藏层”本质上是对输入作了预处理,从高维空间向低维空间作一个映射。高维空间的无数多个点对应于低维空间的某一个点。那么遗传算法就能在一个更加高维的曲面上去寻找最优解,当算法陷入局部最优解的时候,她可以朝更多的方向去寻找走出局部最优解的路线(具体就是她可以通过对多个权值的修改达到粗调和微调的作用),而不是被困于局部最优解所设的“山峰”。

3. “伪隐藏层”技术的实现和验证

3.1 实现“伪隐藏层”的步骤。

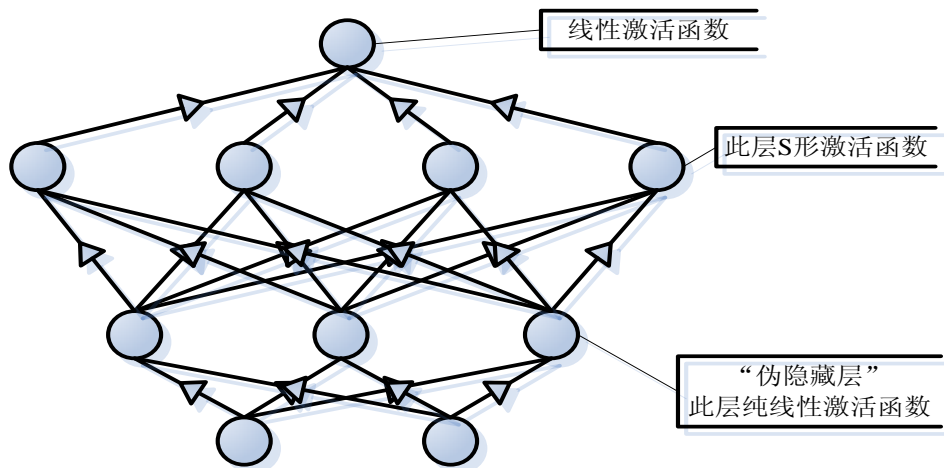
使用“伪隐藏层”技术的过程主要包括添加,训练(进化),整合等三个步骤。下面用一个简单的例子作为说明。下图是一个忽略偏置的两输入单输出,含有一个隐藏层的神经网络模型。



我们希望对这个神经网络模型使用“伪隐藏层技术”：

1. 添加“伪隐藏层”。

先添加一个激活函数为线性函数（一般直接使用纯线性函数）的隐藏层，神经元数目一般较下一层的神经元数目少。这里就取 3 个，那么所得到新的神经网络模型如下图所示：



2. 训练神经网络（进化权值）

这一步骤只要把以往对权值进化的处理方法用在新的神经网络模型上面就可以了。（但是有一个问题值得注意，就是变异算子的变异范围要减小几个数量级。）

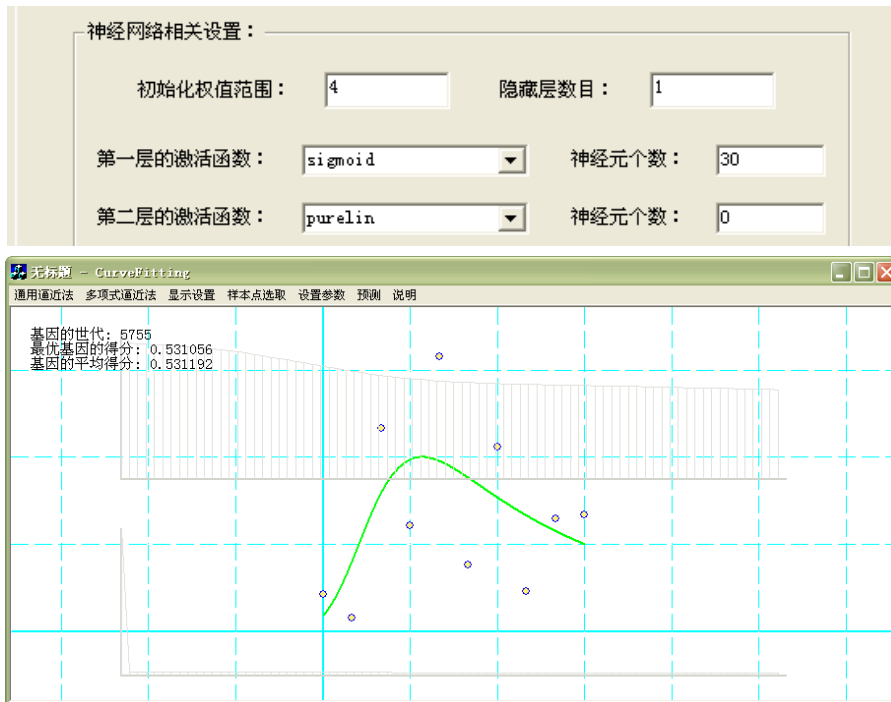
3. 整合神经网络

这一步骤要把添加“伪隐藏层”后的神经网络模型整合成添加“伪隐藏层”前的神经网络模型。整合规则由得到。由此也说明训练完成后的神经网络的运算代价并不会因为训练过程添加了“伪隐藏层”而有所增加。

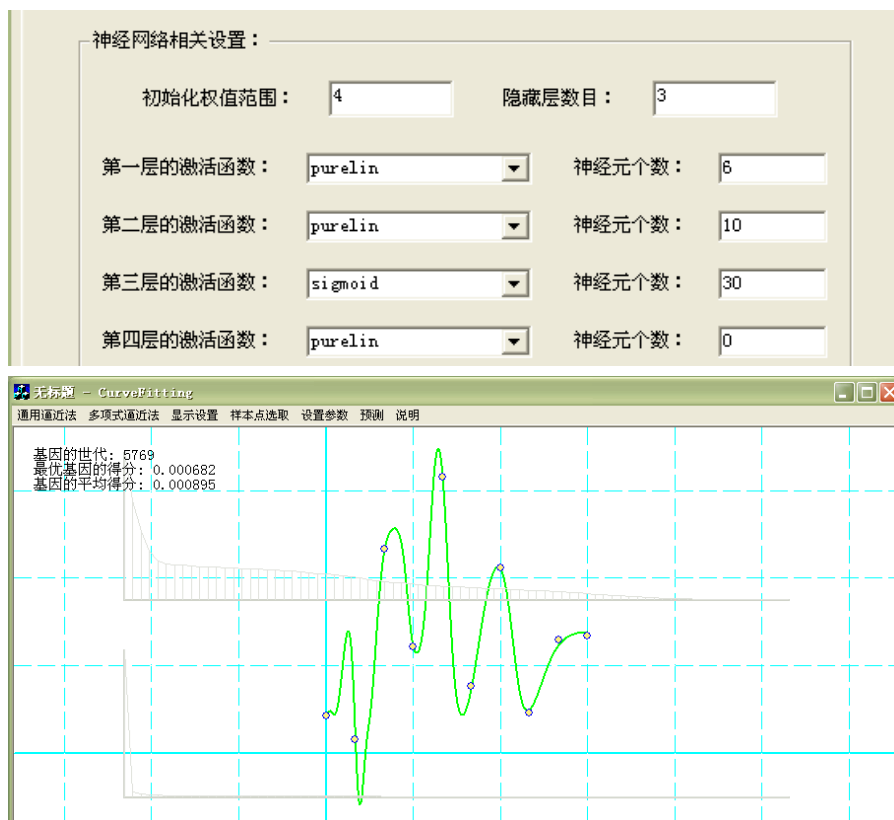
3.2 新技术的应用实例。

为了验证“伪隐藏层”技术的效果，专门写了一个二维曲线拟合的程序作为验证程序。以辽东湾多年的年极值冰厚构成一个非线性随机时间序列作为曲线拟合的样本数据。

试验一：创建一个单隐藏层，隐藏层含 30 个神经元的神经网络模型。训练 5000 多代后（用时 3 分钟）拟合结果不理想，解明显陷入了局部最优，如下图所示。



试验二：在试验一的神经网络模型的基础上，前后分别添加两个和一个伪隐藏层，详细数据请看下图。同样进化 5000 多代后（用时 3 分钟），函数拟合的结果非常理想，最后的均方差减小到 0.00682。从均方差的曲线记录可以看出，均方差是平稳递减的，说明算法几乎没有陷入过局部的最优解。这个神经网络的表达能力相当于一个单隐藏层的神经网络，却能经过短时间训练准确地拟合复杂的样本数据，是以往的算法所难以做到的。



4. 结论，与及算法的进一步完善

上面分别从理论和实践两方面证实了“伪隐藏层”技术的可行性和实用性。但是，值得注意的是，该技术在大大增加了 GA 的全局搜索能力的同时也同样无法避免 GA 一些固有的局限性。下面对算法的改进和完善作进一步的探讨。

1. 使用模拟退火技术或者自适应技术动态改变变异算子变异范围^[4]。这个技术可以有效缩短 GA 的进化周期。也可以进一步提高 GA 找到全局最优解的可能性。

2. 先用 GA 定位到全局最优解附近，然后利用 BP 算法进行局部寻优。实验证明这种混合方法比单独利用 GA 或 BP 算法的效率都要高^[5]。这是因为 GA 虽然具有较好的全局搜索能力，但其局部搜索能力仍比不上传统的 BP 算法。

值得注意的是“伪隐藏层”技术，是一种分割搜索空间，丰富化搜索路线的思想。利用 GA 去解决其它实际问题时（非训练神经网络的问题）也可以适当地构造一个“伪隐藏层”起到同样的作用。（做了一个多项式法拟合曲线的程序，把多项式的系数看作神经网络的权值，运用构造“伪隐藏层”法，也收到了很好结果，由于篇幅所限，这里不一一细述。）希望“伪隐藏层”技术得到更广泛的关注和重视，对它的深入研究将带动其它领域进一步的发展。

参考文献

- [1] Judd J S. Learning in networks is hard. Proceedings of the 1st IEEE International Conference on Neural Networks, Vol.2, pp.685~692, 1987
- [2] 万琼, 姚望舒, 王金根, 陈世福, 谢俊元. 进化算法在人工神经网络中的应用研究. 中国人工智能学会第 10 届全国学术年会论文集. 2003
- [3] 蒋宗礼. 人工神经网络导论. 高等教育出版社. 2001
- [4] 王小平, 曹立明. 遗传算法——理论、应用与软件实现 西安交通大学出版社 2002
- [5] Xiangping Meng, Huaguang Zhang, Wanyu.Tan. A hybrid method of GA and BP for short-term economic dispatch of hydrothermal power systems. Issue 3-4, pp. 341~348, January 2000