第一节 介绍

近二十年来，越来越多的研究人员和从业者应用神经网络（NNs）来解决各种问题，如模式分类和函数近似。对于分类问题，输出以及输入也是离散的。另一方面，函数近似或回归问题具有连续的输入和输出，函数或回归可能是非线性的。在许多应用中，需要从训练有素的神经网络中提取知识，以便用户更好地了解网络如何解决问题。大多数现有的研究工作都集中在提取用于解决分类问题的符号规则[1]。已经设计了很少的方法来从经过训练的神经网络中提取规则用于回归[2]。

从神经网络生成的规则应该足够简单，供人们使用。函数近似规则通常采取以下形式：if（条件满足），然后预测\ y=f（x） ，哪里 \ F（X）  是一个常数或线性函数 \ X ，数据的属性。这种规则是可接受的，因为它们与传统的参数回归统计方法相似。

单一规则不能很好地逼近由网络执行的非线性映射。一个可能的解决方案是将数据的输入空间划分为子区域。同一子区域内所有样本的预测将通过单个线性方程来执行，其系数由网络连接的权重确定。通过更精细地分配输入空间，产生更多的规则，并且每个规则可以更准确地近似网络输出。然而，一般来说，大量的规则（每个规则仅适用于少量样本）不能向用户提供有意义或有用的知识。因此，在规则准确性和规则简单性之间必须达到平衡。

本文描述了一种称为规则提取的方法，用于从逼近神经网络（REFANN）中提取规则，用于从训练的神经网络中提取用于非线性函数近似或回归的规则。这表明，REFANN产生的规则几乎与提取规则的原始网络一样准确。对于一些问题，有足够少的规则可以获得有关问题领域的有用知识。REFANN在具有单个隐藏层和一个线性输出单元的网络上工作。

为了减少规则数量，冗余隐藏单元和不相关的输入属性首先通过在应用REFANN之前通过称为NN修剪的修剪方法（N2PFA）来消除。然后隐藏单元的连续激活功能通过三件式或五件式线性函数近似。分段线性函数的域的各种组合将输入空间划分成子区域，使得可以将相同子区域中的所有输入的函数值计算为输入的线性函数。

本文组织如下：第二部分介绍函数近似，修剪和规则提取的相关工作。第三节 -V描述我们的算法，即N2PFA，非线性激活函数的近似和REFANN。第六部分介绍了两个说明性的例子，说明算法的工作原理。已经进行了广泛的实验，其显示了所提出的函数近似方法的有效性。这些实验的结果见第七节。最后，第八部分总结了论文，讨论了提取的规则的可解释性和对我们贡献的总结。

1. 相关作品和动机

现有的NN函数近似方法通常采用径向基函数（RBF）网络[3]或RBF网络的组合等方法[4]。RBF网络的缺点是它们通常分配一个单元来覆盖输入空间的一部分。因此，许多单元需要充分覆盖整个输入空间，特别是对于具有复杂分布模式的高维输入空间。具有太多隐藏单位的网络不适合规则提取，因为需要许多规则来表达其输出。相比之下，我们的方法采用具有一个隐层的多层神经网络，已被证明是一个通用函数逼近器。

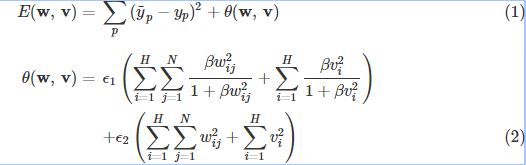
从这样的NN提取的规则数量随着网络中隐藏单元数量的增加而增加。为了平衡规则准确性和规则简洁性（如第一部分所述），必须确定适当数量的隐藏单位，并在文献中提出两种一般方法。建构性算法从几个隐藏单元开始，并根据需要添加更多单位以提高网络精度[5] [6] [7]。另一方面，破坏性算法从大量隐藏单元开始，并删除那些被认为是冗余的[8]。有用输入单元的数量对应于数据的相关输入属性的数量。典型的算法通常从每个属性分配一个输入单元开始，对所有输入属性进行训练，然后去除与不相关数据属性对应的网络输入单元[9]，[10]。已经提出了输入属性对网络预测精度的贡献的各种措施[11] - [12] [13] [14] [15]。我们选择了破坏性方法，除了生产隐藏单位数量最少的网络之外，我们还希望尽可能多地删除冗余和不相关的输入单元。

大多数现有的已发表的报告都侧重于提取用于解决分类问题的符号规则。例如，MofN算法[16]和GDS算法[17]提取MofN规则; BRAINNE [18]，Bio-RE，Partial-RE，Full-RE [19]，RX [20]，NeuroRule [21]和GLARE [22]产生分离正常形式（DNF）规则; 和FERNN [23]根据哪种规则更合适地提取MofN或DNF规则。

另一方面，已经设计了很少的方法来从经训练的NN中提取规则用于回归。ANN-DT [24]是一种能够从功能近似网络中提取规则的算法。该算法将NN称为“黑盒子”，它根据网络的输入和相应的输出产生决策树，而不分析隐藏单元的激活值和网络的连接权重。相比之下，我们的REFANN方法从最小尺寸网络中获得规则。它从原始网络中修剪单元，并通过分段线性函数近似隐藏单元激活函数来提取线性规则。

第三节 网络训练和修剪算法

在本节中，我们描述了N2PFA训练和修剪算法。可用的数据样本(Ip,yp)， p=1,2,…,K 哪里输入 一世p∈ [Rñ 和目标 yp∈R，首先被随机分为三个子集：训练，交叉验证和测试集。使用训练数据集，一个网络H 隐藏的单位被训练，以便最小化平方误差的总和 E(w,v) 增加了罚款期限 θ(w,v)



（待续）