

人工神经网络及其 matlab 实现

人工神经网络基本理论

1、人工神经网络模型拓扑结构

人工神经网络是由大量简单的基本元件（神经元）相互连接，通过模拟人的大脑处理问题的方式，进行信息并行处理和非线性转换的复杂网络系统。

优点：多输入、多输出实现了数据的并行处理以及自学能力。

神经网络的拓扑结构包括网络层数、各层神经元数量以及各神经元之间的相互连接方式，三者都根据实际情况再具体确定取值。

2、常用激励函数

a) 阈值型（一般只存在于 MP 简单分类的情形中）

$$f(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases}$$

b) 线性型（一般只用在输入神经元和输出神经元）

$$f(x) = x$$

c) S 型（常用于隐含神经元）

$$f(x) = \frac{1}{1 + e^{-x}} \text{ 或 } f(x) = \frac{1 - e^{-x}}{1 + e^{-x}}$$

3、常见神经网络理论

a) BP 网络基本数学原理

b) RBF 网络基本数学原理

BP 神经网络的结构设计

1、鲨鱼闻血腥味道与 BP 神经网络训练

BP 神经网络的训练过程是依靠“负梯度下降”的训练算法，即误差调整的方向总是沿着下降速度最快的方向进行。

2、透视神经网络的学习步骤

假设由三层 BP 网络，第一层为输入层，第二层为中间层（隐含层），第三层为输出层，每一层由神经元构成，每层神经元之间由权值相互连接。其中，**输入层神经元数量由输入样本的维数决定，输出层神经元数量由输出样本的维数决定**，隐含层神经元合理选择。

a) 准备训练网络的样本

由冶炼技术可知：冶炼的“初始温度”会受到出钢时间、钢水净重量、吹止温度、高碳锰铁、低碳锰铁、硅锰铁、硅铁、铝块、增碳剂、中碳锰铁、包龄、运输时间、等待时间这十三个因素影响。这 13 个因子便是输入样本，“开始温度”便是网络的输出样本。（由于开始温度的拿捏在冶炼过程中十分重要）。

b) 确定网络的初始参数

需要说明的是，下表所列的参数都可以根据实际训练情况调整。（另外，数据采用

批量输入，这是因为 matlab 对矩阵的操作是非常快且方便的，而对循环的操作较慢）

例：

参数	数值
最大训练次数	5000
隐含层（中间层）神经元数量	12
网络学习速率	0.035
训练的目标误差	0.65×10^{-3}
是否添加动量因子	否

c) 初始化网络权值与阈值

在此做以下约定：

- ①训练步骤是按照数据批量形式进行的；
- ②阈值写成与权值相似的形式，也就是说，阈值看作样本输入为 1 的随机数；
- ③数据输入是按照每一行输入，不是每一列，因此要把原始数据进行转置。

初始化网络：给网络的权值和阈值赋予随机数矩阵。因为是十三个输入因子，十二个隐含层（中间层）神经元，则第一层与第二层之间的权值 $w_{ij}(t)$ 为 12×13 的随机数矩阵。（传递过来的信息不是直接全部交给中间层，而是进行加权后传递给中间层）。我们可以把权值矩阵的大小看为：这一层神经元数量 \times 上一层神经元数量。

神经网络的一个阈值指用来激活神经元而设置的一个临界值。（有多少个神经元就会有多个阈值），则第二层神经元的阈值为 12×1 的矩阵。

同理，第二层与第三层神经元的权值为 1×12 的一个矩阵，第三层神经元的阈值为一个 1×1 的矩阵。

d) 计算第一层神经元的输入与输出

为了简化理解：假设 X 为输入样本，其数据规模为 13×30 的矩阵。假设第一层神经元中放置的为线性函数（也可以是别的激励函数），所以网络第一层输入和输出都等于实际样本的值， $O_1 = X$ ，为一个 13×30 的矩阵。

e) 计算第二层神经元的输入

f)

对于第二层，神经元的输入 I_2 一定来自第一层所有神经元的值与阈值的和。即： $I_2 = w_{ij} \times x + B_{ij} \times \text{ones}$ （都为 1 的矩阵）。这是因为我们把阈值看为输入为 1 的样本。（样本要进行归一化处理）

g) 计算第二层神经元的输出

假设隐含层（中间层）神经元激励函数为单 S 型函数，即为 $f(x) = \frac{1}{1+e^{-x}}$

所以，第二层神经元的输出为 $O_2 = \frac{1}{1+e^{-I_2}}$ 。由此可知，第二层输出 O_2 是一个 12×30 的数据矩阵。

h) 计算第三层的输入和输出

第三层和第二层的输入道理类似， $I_3 = w_{jk} \times O_2 + B_{jk} \times \text{ones}$ ，是一个 1×30 的矩

阵。

第三层的输出：通常也将第三层神经元设置为线性函数。即 $O_3=I_3$ 。

i) 计算能量函数 E

计算能量函数的目的是达到一定误差就可以停止训练网络。假设实际输出样本为 Y，则易得

$$E = \sum (Y - O_3)^2$$

j) 计算第二层和第三层之间权值和阈值调整量（要注意，这里要反复看）

这里是 BP 网络的核心内容，用到鲨鱼追寻血腥味的原理，同样是一个链式偏微分。

$$\Delta w_{jk} = -\eta \frac{\partial E}{\partial w_{jk}} = -\eta \times (Y - O_3) \times O_2$$

说明： O_3 是 w_{jk} 的函数，这里复合函数求导加矩阵求导同理可得：

$$\Delta B_{jk} = \frac{\partial E}{\partial B_{jk}} = -\eta \times (Y - O_3) \times \text{ones} = -16.8983。$$

k) 计算第一层与第二层之间权值和阈值调整量

先计算第二层与第三层，再计算第一层与第二层，体现了误差反向传播的思想。对传递函数求导会有一个特征： $f'(x) = f(x)(1 - f(x))$

$$\Delta w_{ij} = -\eta \frac{\partial E}{\partial w_{ij}} = -\eta \times w_{jk} \times (Y - O_3) \times O_2 \times (1 - O_2) \times X$$

l) 计算调整之后的权值和阈值

在这里进行简单的加法即可。

$$w_{jk}(t+1) = -\eta \frac{\partial E}{\partial w_{jk}} + w_{jk}(t) = \Delta w_{jk} + w_{jk}(t)$$

$$B_{jk}(t+1) = -\eta \frac{\partial E}{\partial B_{jk}} + B_{jk}(t) = \Delta B_{jk} + B_{jk}(t)$$

$$w_{ij}(t+1) = -\eta \frac{\partial E}{\partial w_{ij}} + w_{ij}(t) = \Delta w_{ij} + w_{ij}(t)$$

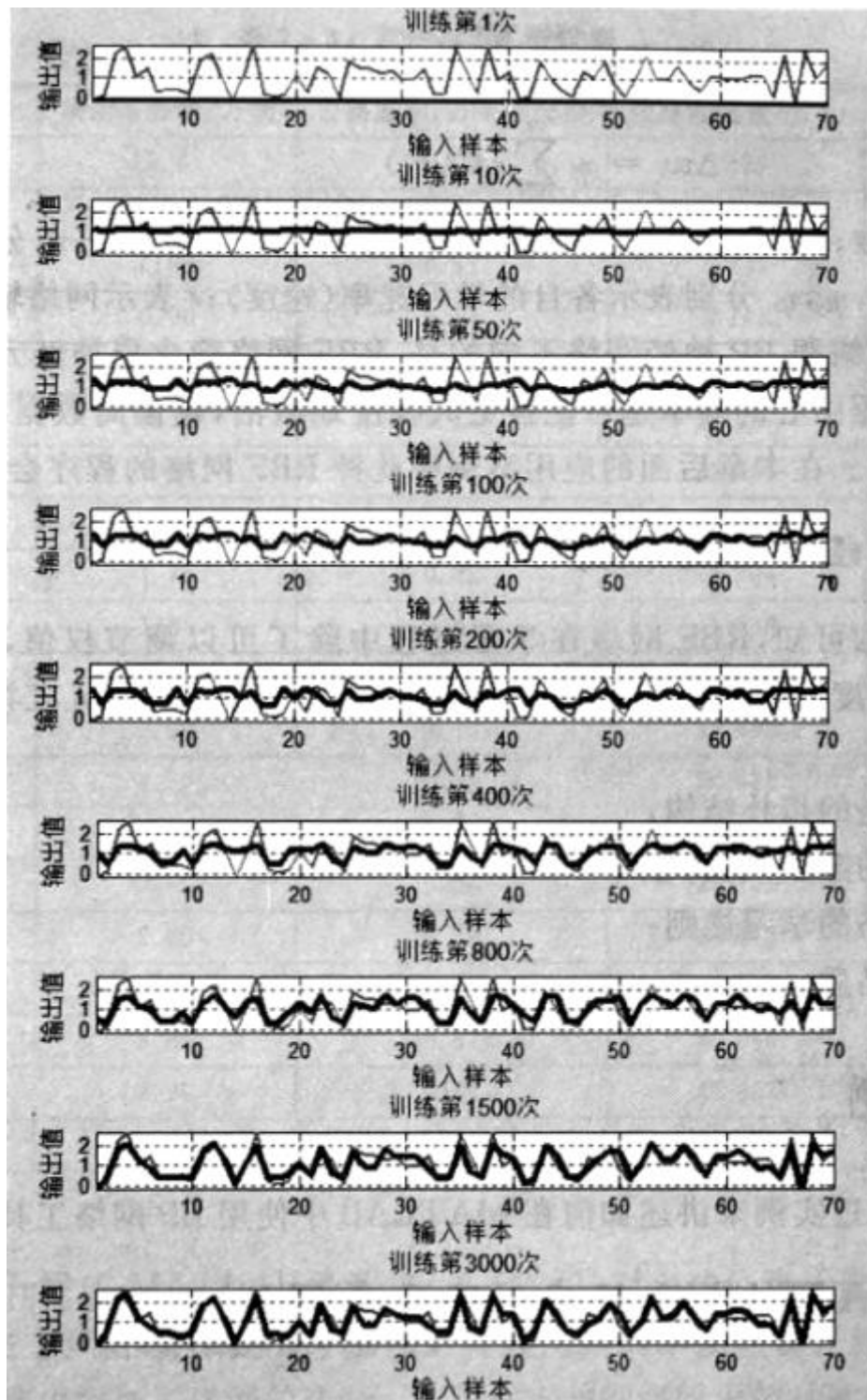
$$B_{ij}(t+1) = -\eta \frac{\partial E}{\partial B_{ij}} + B_{ij}(t) = \Delta B_{ij} + B_{ij}(t)$$

m) 网络输出的值需要还原

由于样本提前进行过归一化，那么现在将 O_3 还原为原始数据的量级。

3、BP 神经网络的动态拟合过程

神经网络的本质是拟合的过程。可以由图得知。



RBF 神经网络的结构设计