# 神经网络公式推导总结

## 1.示例网络及其参数

以一个三层的神经网络为例，如下图所示：

其中输入层向量：



Input Layer

Hidden Layer

Output Layer

1st

2nd



隐藏层： 权值矩阵：





偏置向量：





每层未经激励函数激活的结果向量：





每层经激励函数激活的结果向量：



输出层向量：

激励函数选用sigmoid函数，定义公式为：，其导数可以用自身表示，即：

根据网络，对于每一层的矩阵和矩阵元素之间我们都可以得到一些关系式：

1st hidden layer:

 （1）

 （2）

 （3）

2nd hidden layer：

 （4）

 （5）

 （6）

Output layer:

 （7）

 （8）

 （9）

## 神经网络的函数模型

假设训练样本个数为100，则对第i个样本建立函数模型，形式为：

 （10）

其中是第i个样本的样本真值，由训练数据可知，为已知量。是通过神经网络计算出来的预测值，为计算量，满足关系式：

 （11）

为误差，包括函数模型未学习到的部分和随机噪声，随着训练的进行，神经网络学习程度的提升会进一步“压榨”误差，最后误差可近似视为随机噪声，并满足均值为0的高斯分布，即：

 （12）

故其分布律为：

 （13）

结合（10），有：

 （14）

所以可以说服从均值为，方差与误差服从的分布的方差相同的高斯分布，即：

 （15）

为权值和偏置组成的参数矩阵，神经网络训练的目标就是通过训练出最佳的，使得预测出来的尽可能接近样本真值，根据极大似然估计，有：

 （16）

在这里做个极强的假设，假设这100个样本独立且同分布，则有：

 （17）

展开整理（17）式：

 （18）

做对数变换，得到：

 （19）

进一步整理：

 （20）

将括号中的项设为误差函数，将问题转变为“找到使误差函数最小的θ”：

 （21）

## 权值和偏置的更新

为了便于说明，在这里讨论的是一个样本的情况，于是将误差函数可以设为：

 （22）

为了最小化误差函数，我们采用梯度下降的方法，从后向前进行权值和偏置的更新，下面详细阐述权值的更新过程，偏置更新过程类似，只列举结果：

### Output layer

 （23）

要求误差函数对输出层权值矩阵的偏导数，我们可以先讨论其对矩阵中每一个元素的偏导数，根据偏导数链式法则我们有：

 （24）

将预测值与真实值差值设为：

 （25）

结合（7）（9）（22）（25）式可将（24）式分部解算：

 （26）

 （27）

 （28）

由此可以得到：

 （29）

 （30）

将（30）式用矩阵形式表示：



 （31）

 （32）

T为误差传递中重复出现的传递变量，所以这层的权值梯度最终可表示为：

 （33）

### 2nd hidden layer

类似（23）（24）式，我们可以得到：

 （34）

 （35）

对（35）式分部解算：



（36）

 （37）

 （38）

根据（36）（37）式解算此层的传递变量T：

 （39）

将其矩阵形式展开可以得到：

 (40)

所以结合以上几式，进而得到：

 （41）

### 1st hidden layer

第一层对权值更新的过程与前面类似，这里就不再赘述，此层的传递变量和权值梯度分别计算如下：

 （42）

 （43）

以上是各层权值的更新过程，偏置的更新过程与之类似，不再详细阐述。

## 总结

根据上述对于一个样本的参数更新过程，我们可以总结出对于整个100个样本的参数更新公式如下：

 （44）

 （45）

 （46）

 （47）

 （48）

 （49）

用E-M算法思路来解决神经网络的问题：

1. step:given model parameters:



Forward propagation estimate missing observations:



M-step:given:



Back propagation update:



Input:



Output:



初始化随机值：



For m=1:10000

随机找十个样本



代入公式（44）-(49),根据E-M步骤更新权值和偏置，调节ɑ的值

End