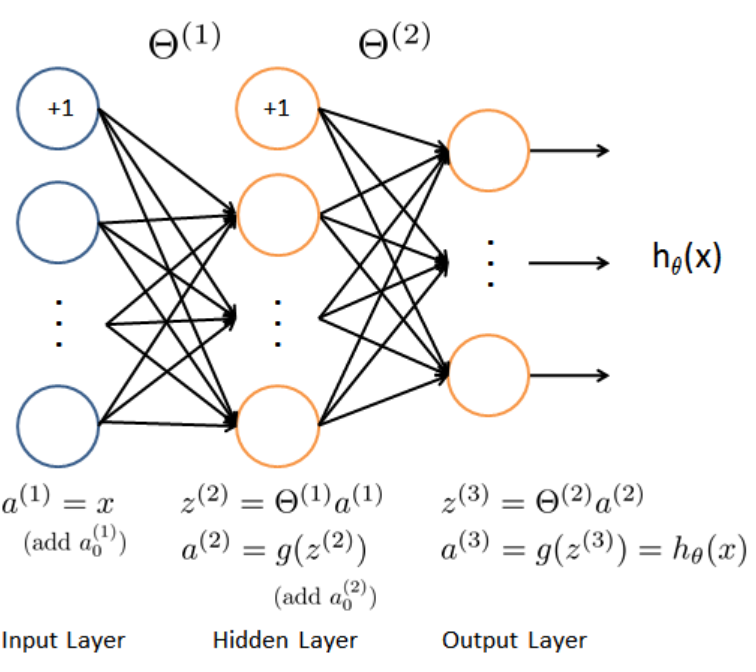
BP说明文档

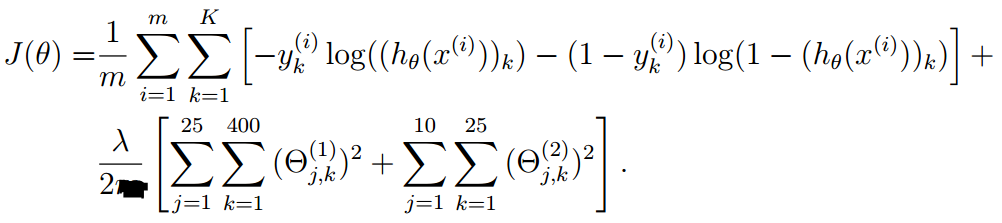
##### 网络结构



1. 从左到右依次是输入层，隐藏层，输出层共三层的网络。向前传递的计算公式如图中所示。其中Theta表示层层之间的参数矩阵（包含了bias），a是每一层向下一层输入向量，z表示每一层从上一层得到的输出向量，g是激活函数。hθ(x)为最总的输出向量。
2. 图中标有+1的节点表示bias节点，该节点值恒为1，这样参数Theta中由该节点延伸出的线上的权重就是bias，只是使得计算公式更统一一点。
3. 每层激活函数g选取为sigmoid函数。

##### 参数调整算法

1. **Loss function**

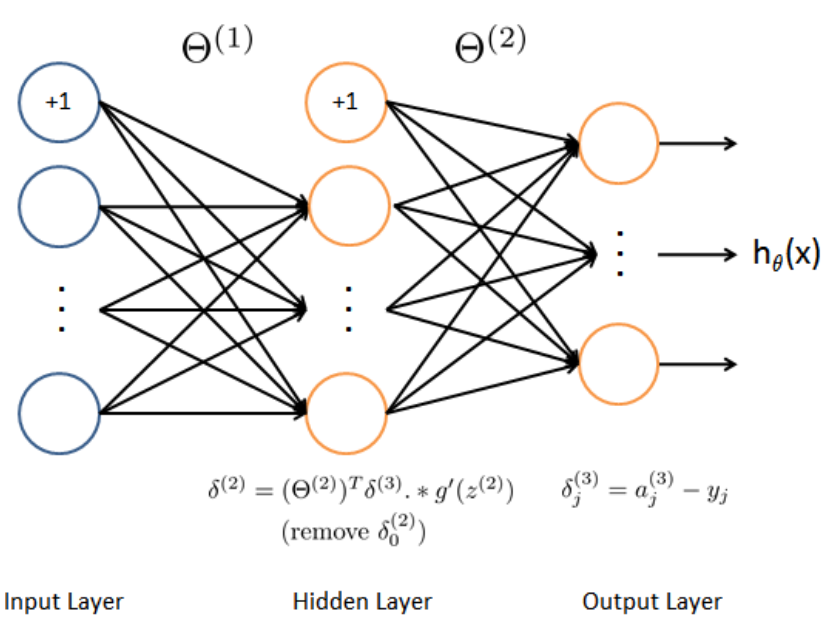


Loss function的前一部分是误差项，后一部分是正则化项。

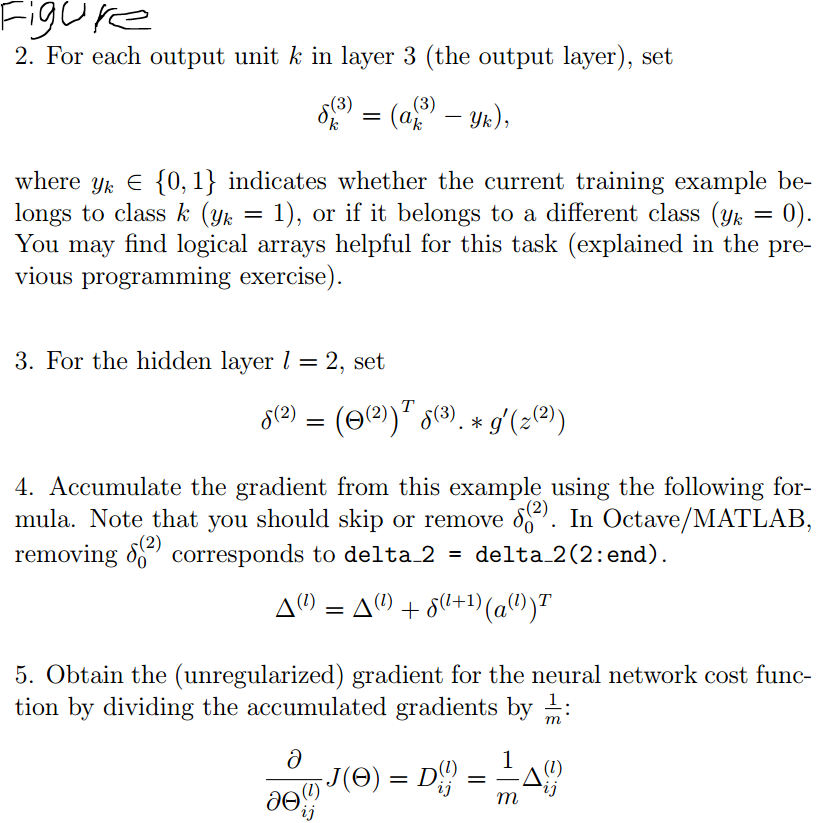
误差项使用的是cross entropy。然后这是一个由二分类问题扩展而来的loss function。即对于一个多分类问题，当成是每一个节点的二分类问题。每一个节点只需考虑自己，如果正确答案是自己，就向着使得自己变大的方向调整，反之向着使得自己变小的方向调整。这与softmax相比，少考虑了输出节点之间的相互影响，但是其由二分类问题自然扩展而来，要简单许多。

添加正则化项是为了防止过拟合。

1. **利用back propagation更新参数。**



1. loss function中误差项和正则化项是分开的，这极大方便了我们的计算。这里先计算误差项的梯度。
2. 图中δj（l）表示第l层，第j各unit的误差。
3. 按层更新参数的方法网上有很多，下图是来自coursera上一门机器学习课程提供的公式。需要注意的是，我们前面应用了一些方法使得bias和权重参数Theta混在一起了。所以在更新的时候需要特别小心的注意到它不能影响到第一层与第二层之间的参数。



1. **Mini-batch方法**

对于大的样本，如果每次都将所有样本的误差都累积