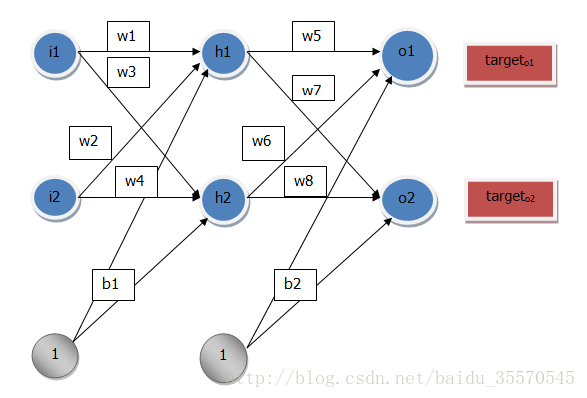
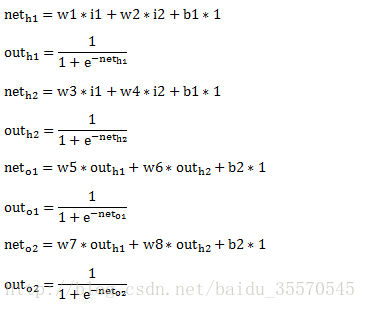
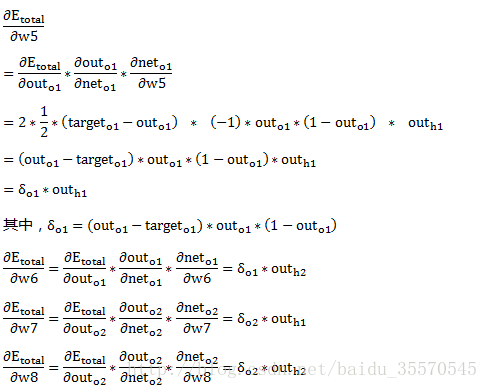
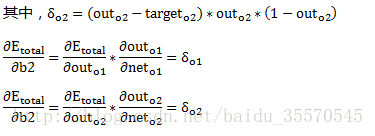
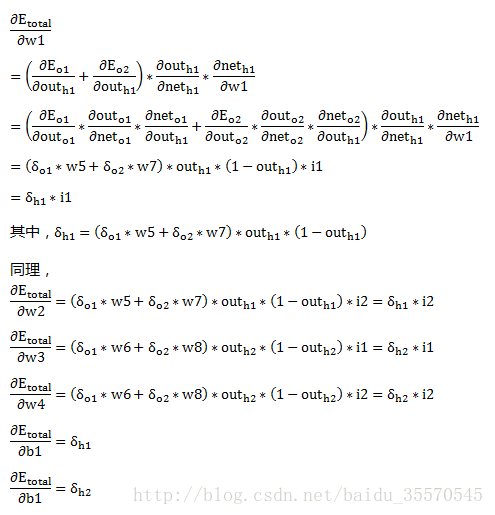
BP算法-让你看清误差传播的每一步

2017年03月14日 22:22:51

**摘要**：本文通过最简单的一步步计算试着搬动BP算法这块大石头，通过发现其中的规律来更好的理解BP算法。   
**预备知识**：链式求导法则。   
  
1、给定输入值，向前传播，直到输出神经元有实际输出为止。   
  
2、误差计算方法   
C:\Users\10488\Desktop\20170314221610881.png  
3、反向传播（最后一层）   
  
  
上述两式乍一看，似乎可以在δo1和δo2之间划等号，这也是最近困扰我好久的一个地方。可以说，δo1就是δo1，δo2就是δo2，互不相干，跟target值有关。那为什么会出现上述的尴尬局面呢？主要是因为图中把每一层对应的神经元的阈值都初始化成同一个数，计算neto1使用b2，计算neto2也是用b2，就误以为两个“b2”永远相等。其实不然，经过第一次反向传播后，两个“b2”就会各自回到自己的轨道上。所以δ\_o1一般不等于δ\_o2。对于下面的b1也是同样的道理。

4、反向传播（非最后一层）   


**归纳计算方法**：w的偏导数 =右边神经元的误差\*左边神经元的输出，b的导数=右边神经元的误差。

1. 对于最后一层的导数中的误差：（实际输出 - 目标值）\*激活函数的导数；
2. 对于非最后一层的导数中的误差：（与该神经元相连的右边的所有权值\*相对应前一层的误差的总和）\*激活函数的导数。