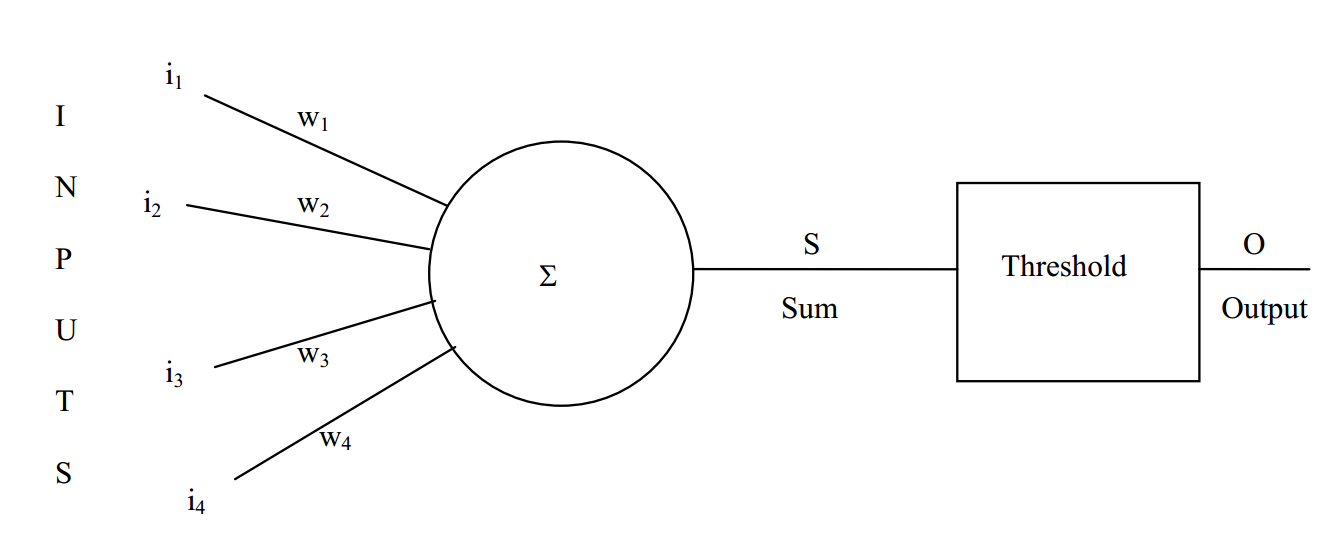
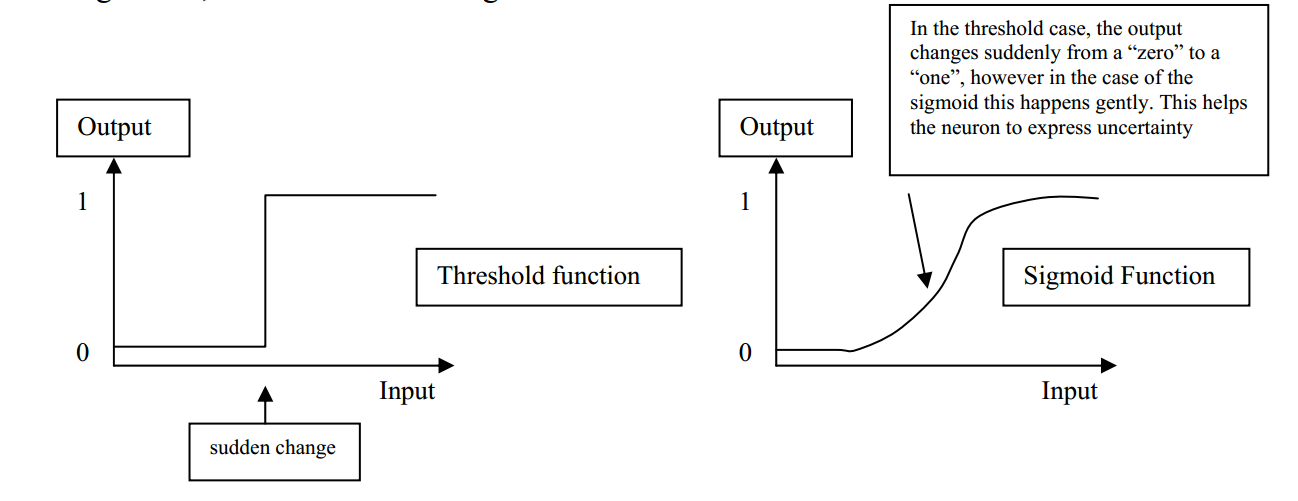
In 1949 Donald Hebb2 postulated one way for the network to learn. If a synapse is  
used more, it gets strengthened – releases more Neurotransmitter. This causes that  
particular path through the network to get stronger, while others, not used, get weaker.  
You might say that each connection has a weight associated with it – larger weights  
produce more stimulation and smaller weights produce less. These were the first steps  
to understanding the learning mechanism of the network. There are still gaps in our  
understanding of learning in the Biological Network, but this need not delay us from  
turning to Artificial Networks and how they are trained

神经网络的灵感来自真正的神经元的观察，上面这段话说明，不同的突触synapse是有不同的比重的，用的多比重就大，用的少比重就小，所以人们根据这种比重的关系，搞出了第一个神经元模型。



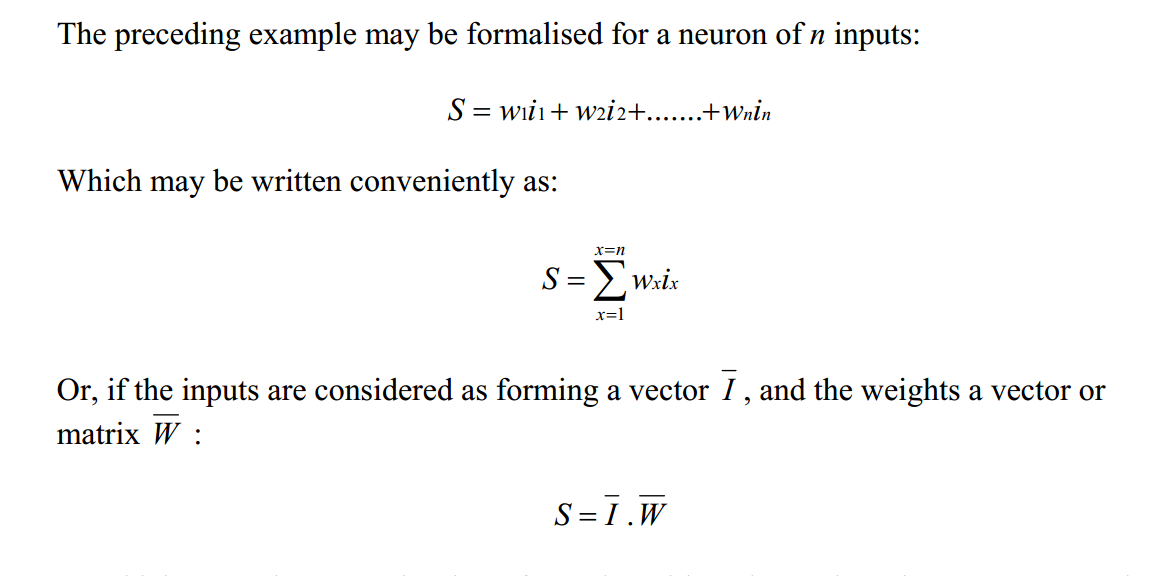
i像是一个个的突触，输入到神经处理单元，但是不同的突触起的作用不一样，所以给了一个w，w就代表每个突触的比重。

这样S求和之后的结果，经过一个阈值处理，输出的0,1这种间断的信号。人们觉着还是连续的好，所以s后面开始跟类似sigmod函数这样的。



阈值是一个突然跳跃，太吓人了。sigmod是一个缓慢上升而且连续的，确实好些？

下面是把计算转换成矩阵运算

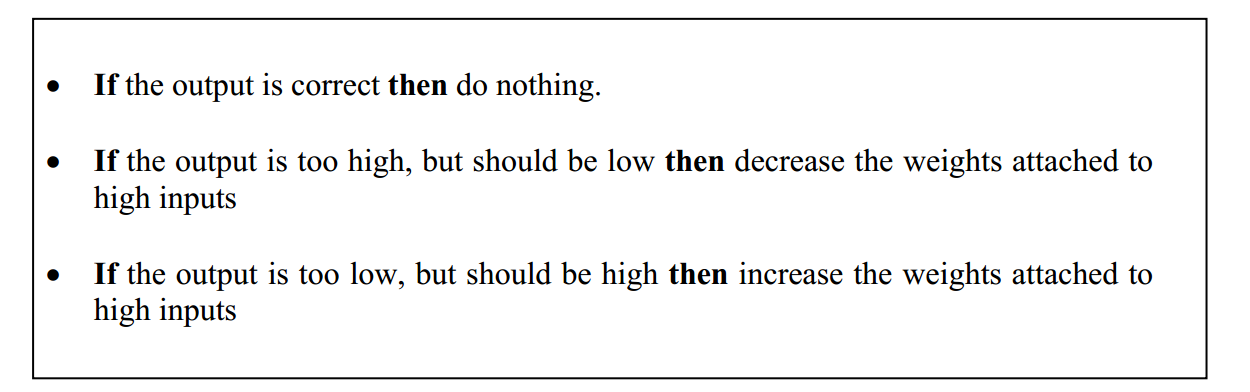


矩阵运算的优点呢？

Even if there were some “interference” to the basic picture (some of the white pixels  
weren’t quite 0 and some of the shaded ones weren’t quite 1) the neuron would still  
recognise it, as long as the total sum was greater than 1.5. So the neuron is “Noise  
Tolerant” or able to “Generalise” which means it will still successfully recognise the  
picture, even if it isn’t perfect - this is one of the most important attributes of Neural  
Networks

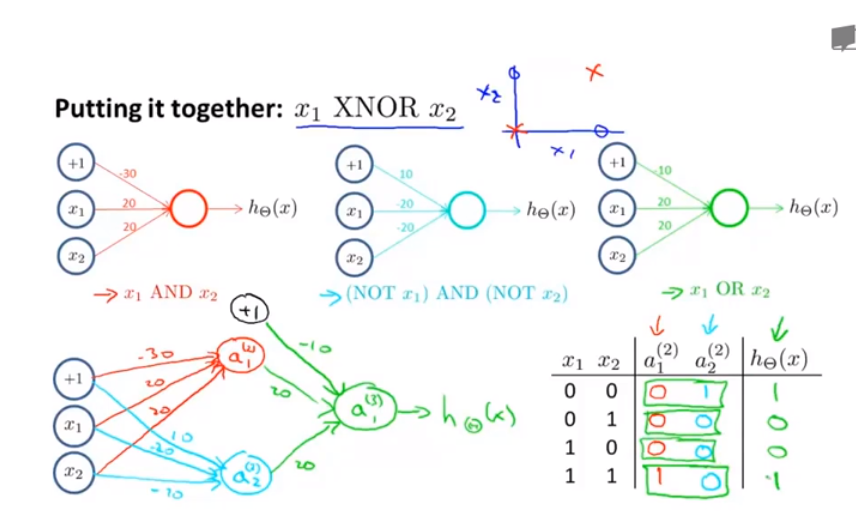
神经网络一个最大的优点是，抗干扰性。

算法学习的过程

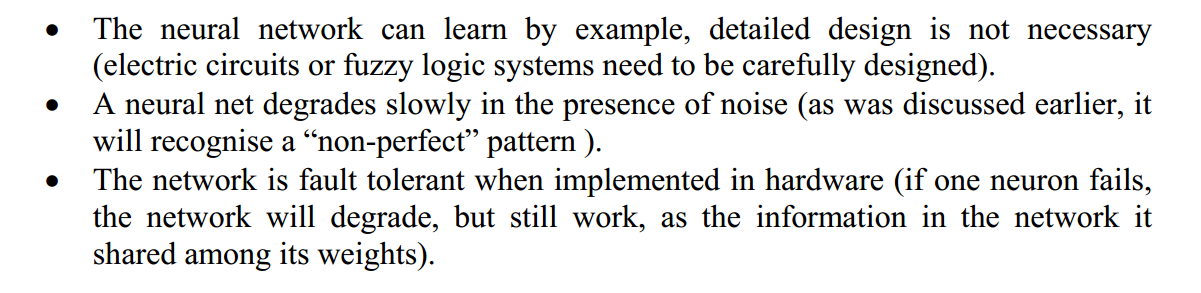


2.4单个神经元的局限性

对于一个神经元无法实现异或操作，在吴恩达的课有做过异或，需要两个神经元才行。所以要突破单个神经元的局限。



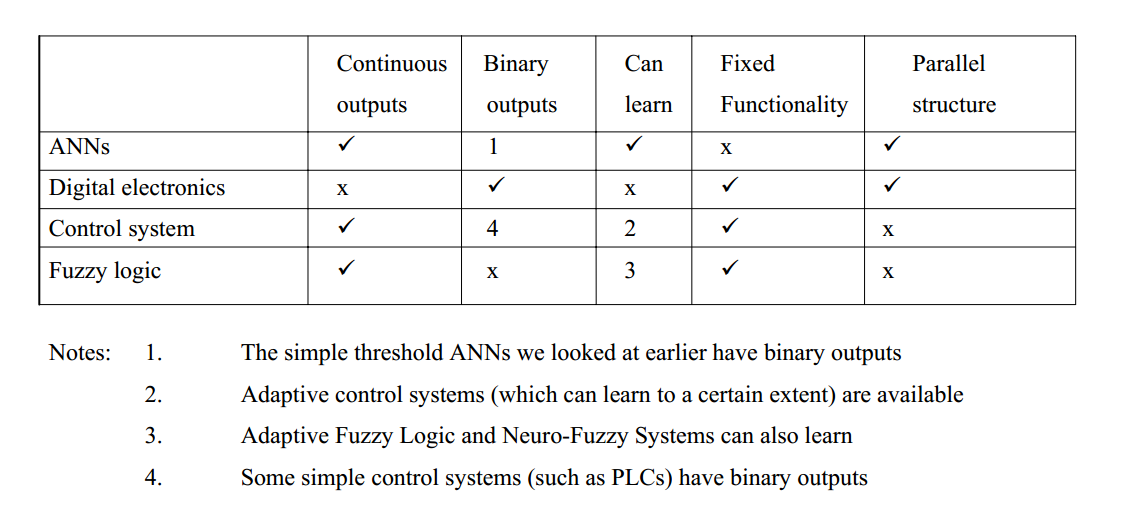
神经网络和数字电路的区别：



数字电路是基于硬件的，很明显有很大的局限性，不够灵活。

神经网络是可以训练和学习的，因此会比较灵活，容错性也比较好。

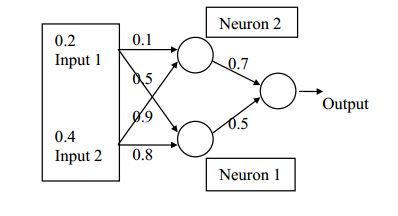
看一下更具体的对比：



在计算的时候尤其要注意，sigmod函数的定义，以及其导数的特殊性，每一层的输出都是一个sigmod函数啊。

### 正向传播

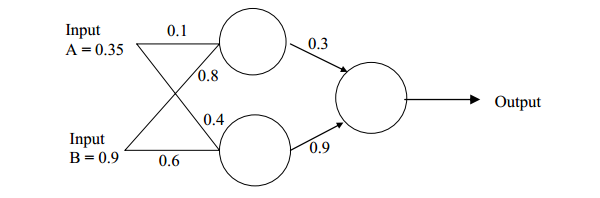
计算下面这个简单神经网络的正向传播：



代码参考python文件。

### 反向传播

计算下面正向加反向传播



### 预处理输入数据

之前看过类似的介绍，输入的数据尽量要集中到-1到1之间，这样sigmod函数不至于范围太小。

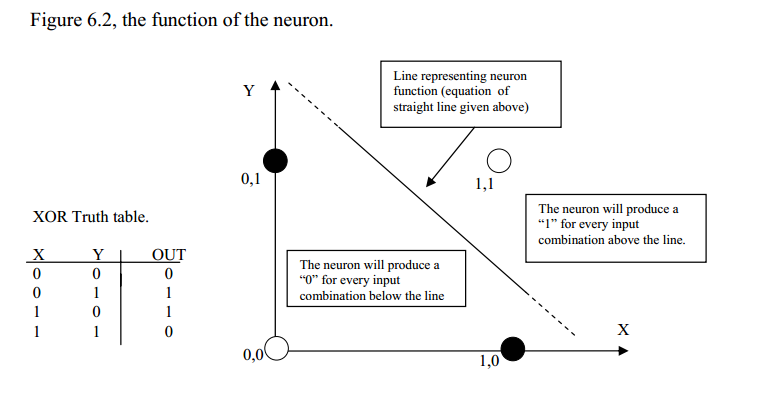
对于不同格式的图片和视频，需要进行统一处理，因为神经网络是不能处理这些原始数据的，这就是为什么caffe中只支持那几种数据，特定格式的数据更有利于训练。

各种与处理啊，目的是为了更好的训练网络。

盛

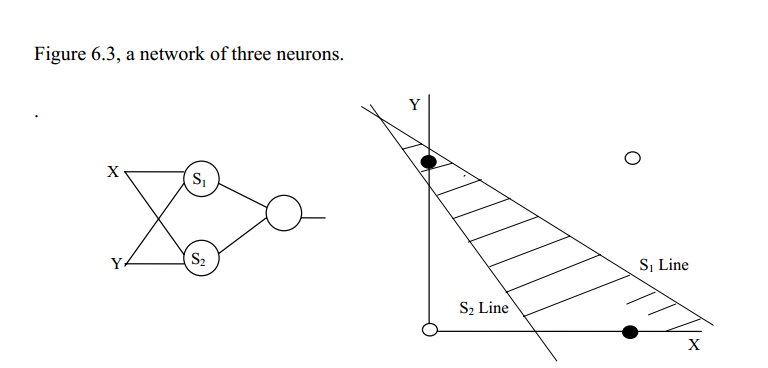
### 网络的层级和大小

单个神经元的弊端，无法实现异或功能。



单个神经元，实际上是一个直线，只能做线性可分的，对于异或的问题，无论这个线如何，都无法实现这个分割的功能。

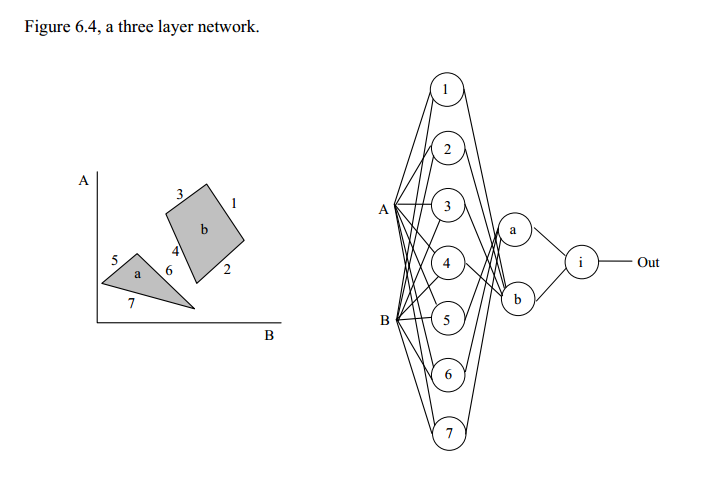
所以要想办法，使用多层。



这两个直线有很多种情况。只要满足条件就行。

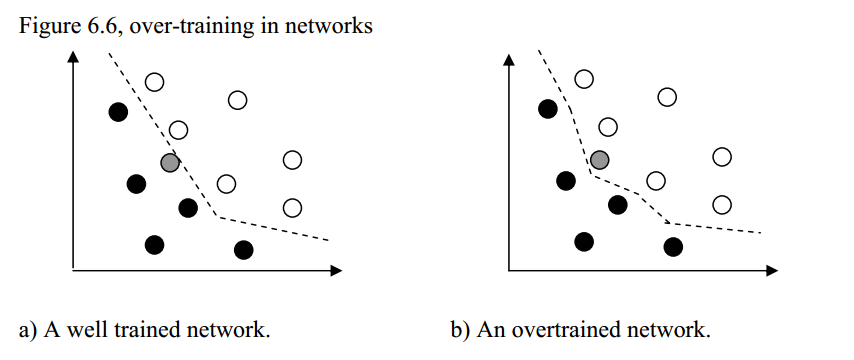
之前看到的，20 10 -10 这种只是一个，斜率是关键啊。

三层网络



第一层，7个，都和输入连接在一起，每一个都是一条线，两条或者以上的线可以组成一个区域，，到了第二层，a b分别连接不同的，a对5 6 7，b对 1 2 3 4。最后的结果就可以把这两个区域合并在一起。

训练过度的问题。

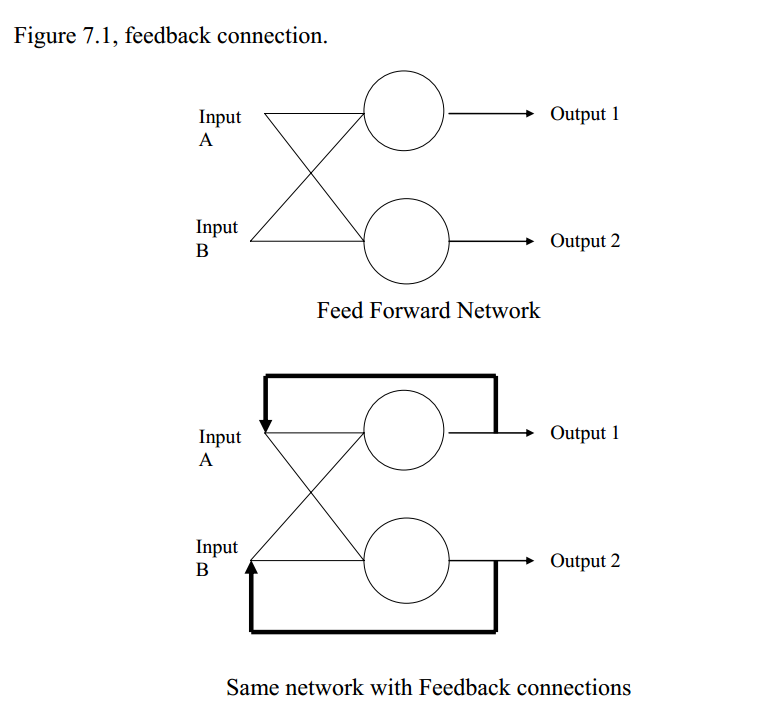


### Hopfield and recurrent networks

之前因为异或的问题，神经网络一下子停滞不前了。1980年代，Hopfield这哥们，灵感乍现啊。

然后突然开始进入了新方向。

现在看反向传播，那是前人花了好久才弄出来的啊。真是的不容易。



把输出当做输入直接输进去，直到输出不在改变，和输入一样，这是搞的啥，试试啊。