

## Chapter 1

# Linear Neurons and Their Limitations

Feed-forward networks（前馈网络）：

1. 同一层中，2个neurons之间没有连接；
2. 也没有从高层到低层的连接；
3. 只有从低层向高层传播的连接。

牢记：

1、大部分神奇的事情都发生在hidden layers，包括不得不花大量时间的有效特征的提取，隐层也会自动处理。

2、令每层的neurons 数量相同？这既不必要也不推荐。

往往，隐层的neurons数量 小于 input layer 的，以迫使神经网络习得原始输入的压缩的表现形式。

例如，我们的眼睛看到周遭事物，接受input 是非常大量的像素值，但我们的大脑反应出的是压缩后的边界和轮廓。

这是因为，对于每件事物的感知，大脑的隐层的生物神经元使我们联想出更好的描绘。

3、每个神经元的output，与下一层所有神经元的input，不一定要相连接。

怎么连接，主要靠经验。

4、input 和 output 以向量表示。

input：图像中很多像素，单个像素的RGB值，是以向量表示；

最后一层的output：

[1,0]-图像里有一只狗；

[0,1]-图像里有一只猫；

[1,1]-图像里有猫和狗；

[0,0]-图像里无猫无狗

以一系列向量和矩阵运算，我们也能数学地表示一个neural network

# Linear Neurons and Their Limitations

大多数neuron types是由它们的转换函数（激活函数） $f$  来定义的。比如

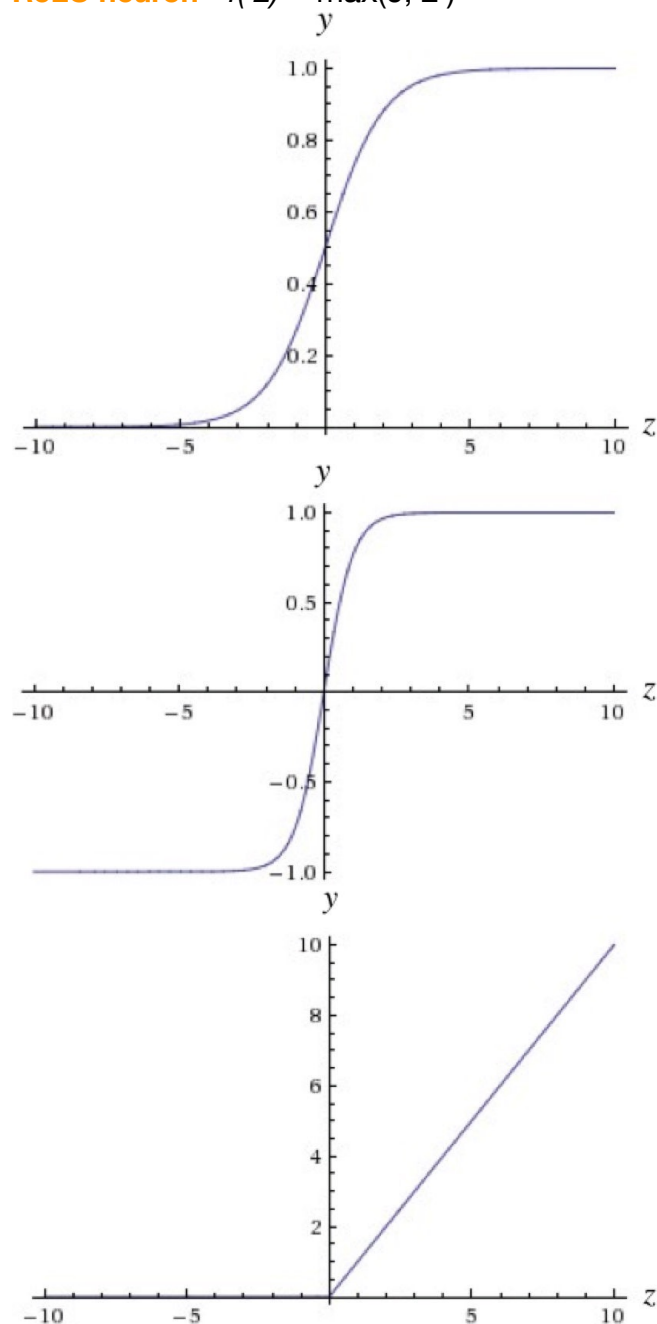
*use a linear function in the form of  $f(z) = az + b$  —— 输入或特征  $z$  在本书中被称为'logit'*

可以证明，任何 仅由 linear neurons 组成的 feed-forward neural network 可以表示成一个无隐层的网络。这是有问题的，因为正如我们前面所讨论的，隐藏层使我们能够从输入数据中学习重要的特性。换句话说，为了学习复杂的关系，我们需要使用一些非线性的神经元。

## Sigmoid, Tanh, and ReLU Neurons

当选用S型非线性neuron时，通常优先考虑tanh neuron > sigmoid，因为它以0为中心。

**ReLU neuron:**  $f(\mathbf{z}) = \max(0, \mathbf{z})$



## Softmax Output Layers

通常，我们希望output vector 是一组互斥labels 上的概率分布。

例如，手写数字识别。每个label (0-9) 互斥，但我们不可能有100%判别出一个数字。所以采用概率分布，比如预测为数字3时，我们有80%的自信这个预测是对的。

结果上，所需的output vector 形式如下：

$$\text{where } \sum_{i=0}^9 p_i = 1:$$

$$[p_0 \ p_1 \ p_2 \ p_3 \ \dots \ p_9]$$

A strong prediction 当是 向量中的某个分量比如  $p_9$  非常接近1，剩余分量均接近0；弱预测 则是每个分量值都差不多。

该 output layer 被称作 **softmax layer**。

不同与其他层，**softmax layer** 中的输出神经元 的output 取决于其层的所有其他神经元的输出。