

深度学习三步

神经网络

1.确定神经网络的连接方法

前馈神经网络 (Fully Connect Feedforward Network)

矩阵运算(Matrix Operation)

如何神经网络结构的设计

2.判断函数的好坏

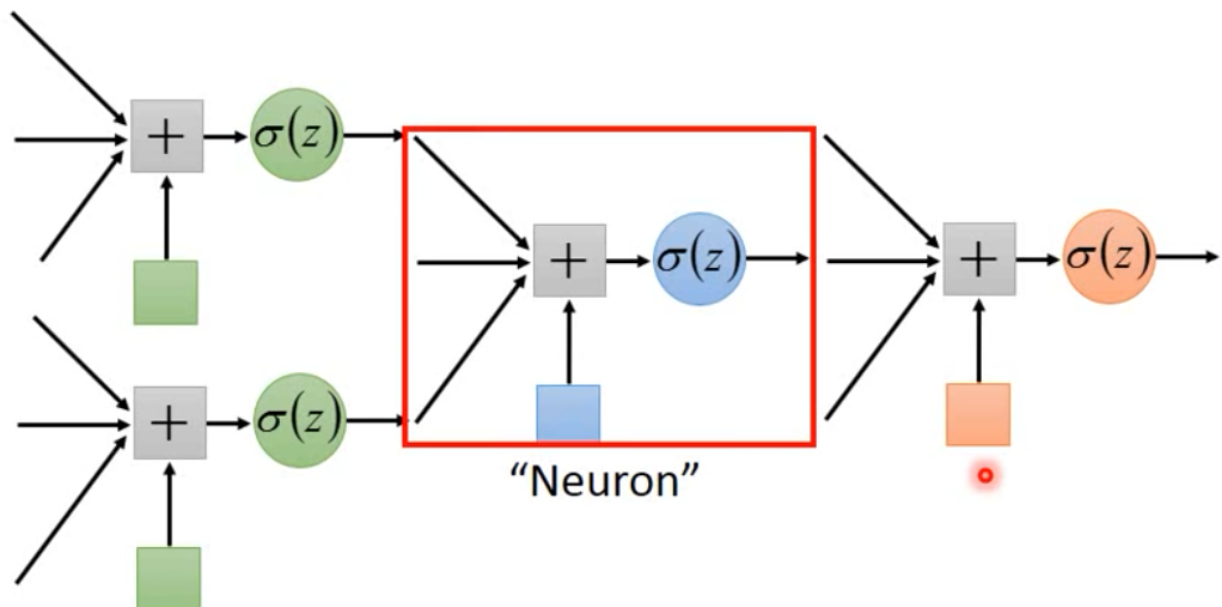
3.使用梯度下降

## 深度学习三步



而在第一步设置的函数大部分是神经网络 (Neural Network) 。

## 神经网络



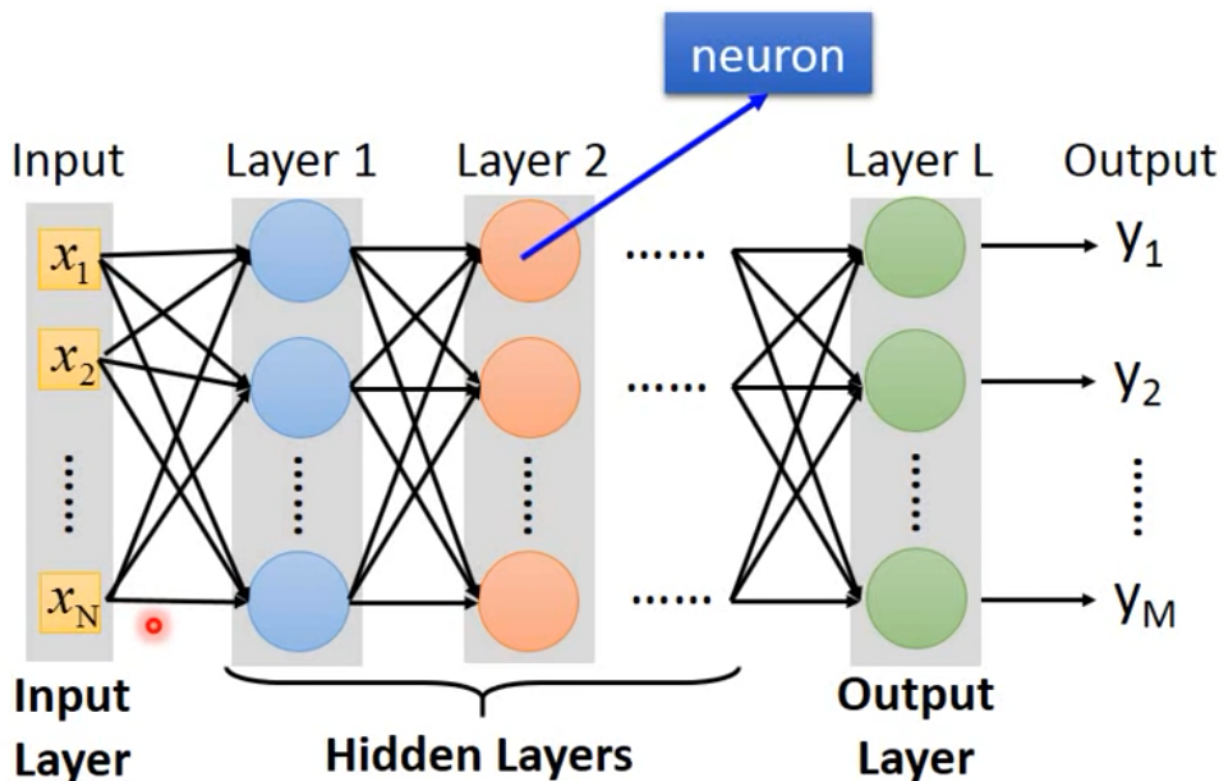
我们可以将神经网络看成，多个逻辑回归交叉转换形成的一个算法模型。单一的一个逻辑回归我们将其视为一个神经。不同的连接方法将会生成不同的神经网络结构。

**神经网络参数：**神经网络中所有的**权重 (weight)** 和**偏差(biases)**

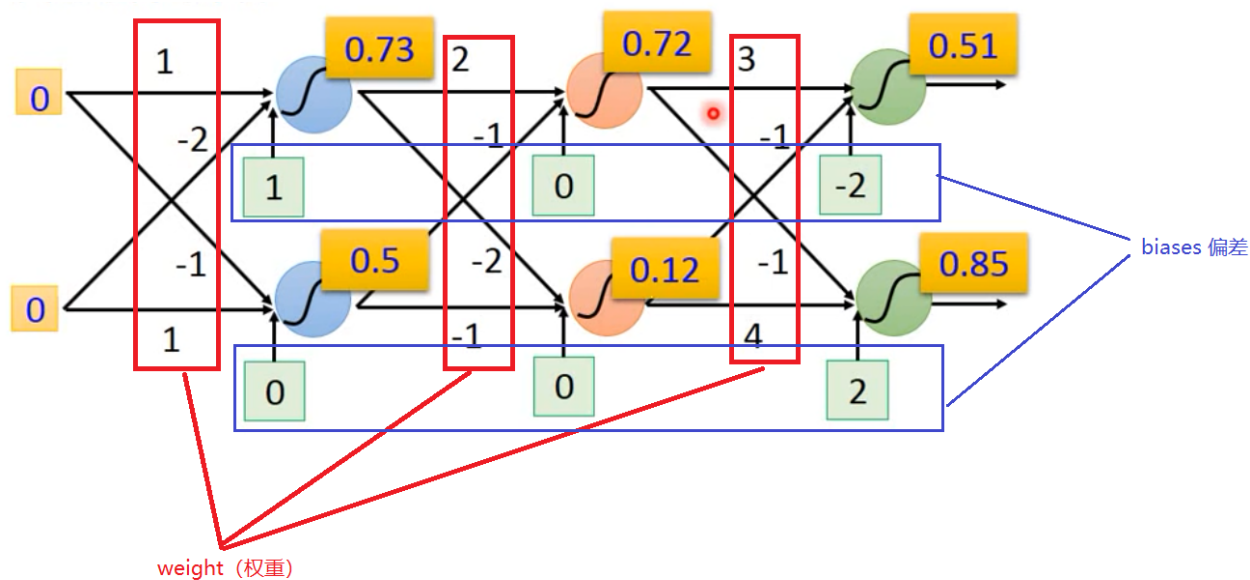
### 1.确定神经网络的连接方法

**前馈神经网络 (Fully Connect Feedforward Network)**

定义:

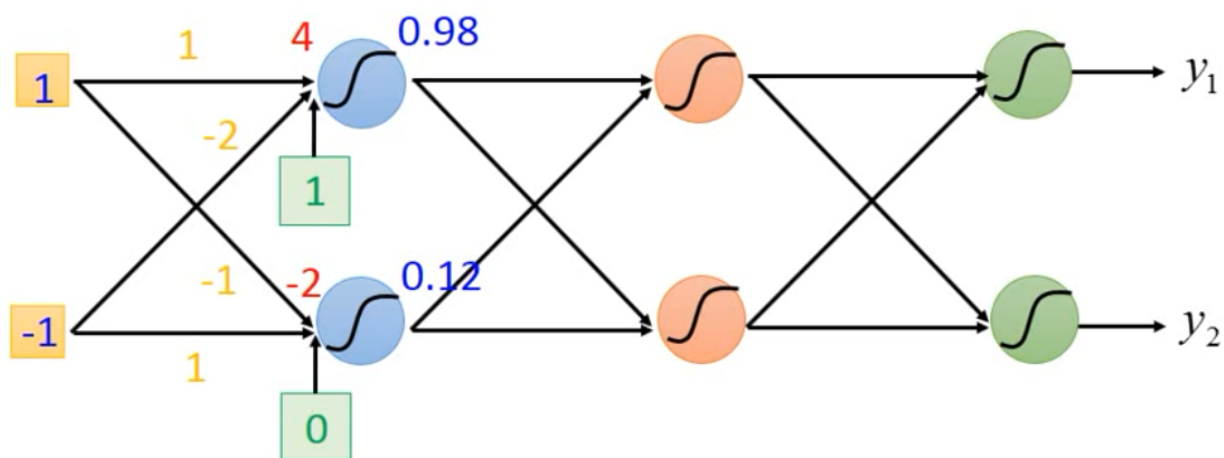


这个就是一个简单的函数模型。而一个简单的神经网络例子如下所示:



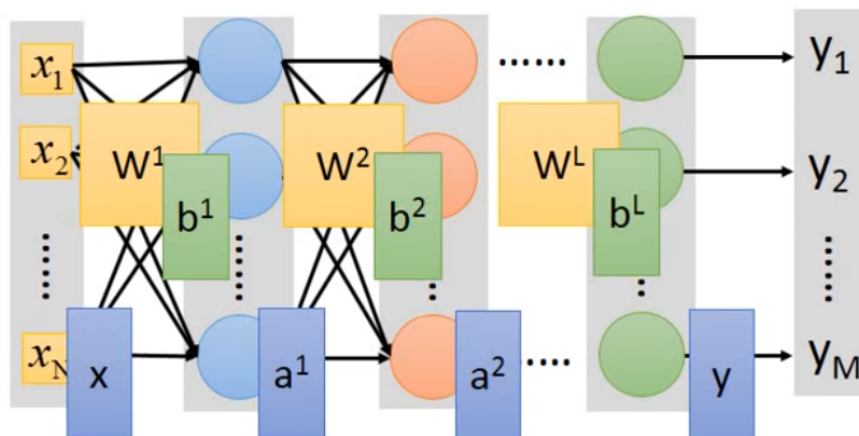
### 矩阵运算(Matrix Operation)

神经网络的运算我们往往会用矩阵运算来进行操作。



$$\sigma\left(\underbrace{\begin{bmatrix} 1 & -2 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix}}_{\begin{bmatrix} 4 \\ -2 \end{bmatrix}}\right) = \begin{bmatrix} 0.98 \\ 0.12 \end{bmatrix}$$

而对于整个神经网络的计算方式，可以视为下图的计算方式



$$\mathbf{y} = f(\mathbf{x})$$

Using parallel computing techniques to speed up matrix operation

$$= \sigma\left(\mathbf{W}^L \cdots \sigma\left(\mathbf{W}^2 \sigma\left(\mathbf{W}^1 \mathbf{x} + \mathbf{b}^1\right) + \mathbf{b}^2\right) \cdots + \mathbf{b}^L\right)$$

Created with EverCam.  
http://www.camdcmv.com

如何神经网络结构的设计

- Q: How many layers? How many neurons for each layer?

Trial and Error

+

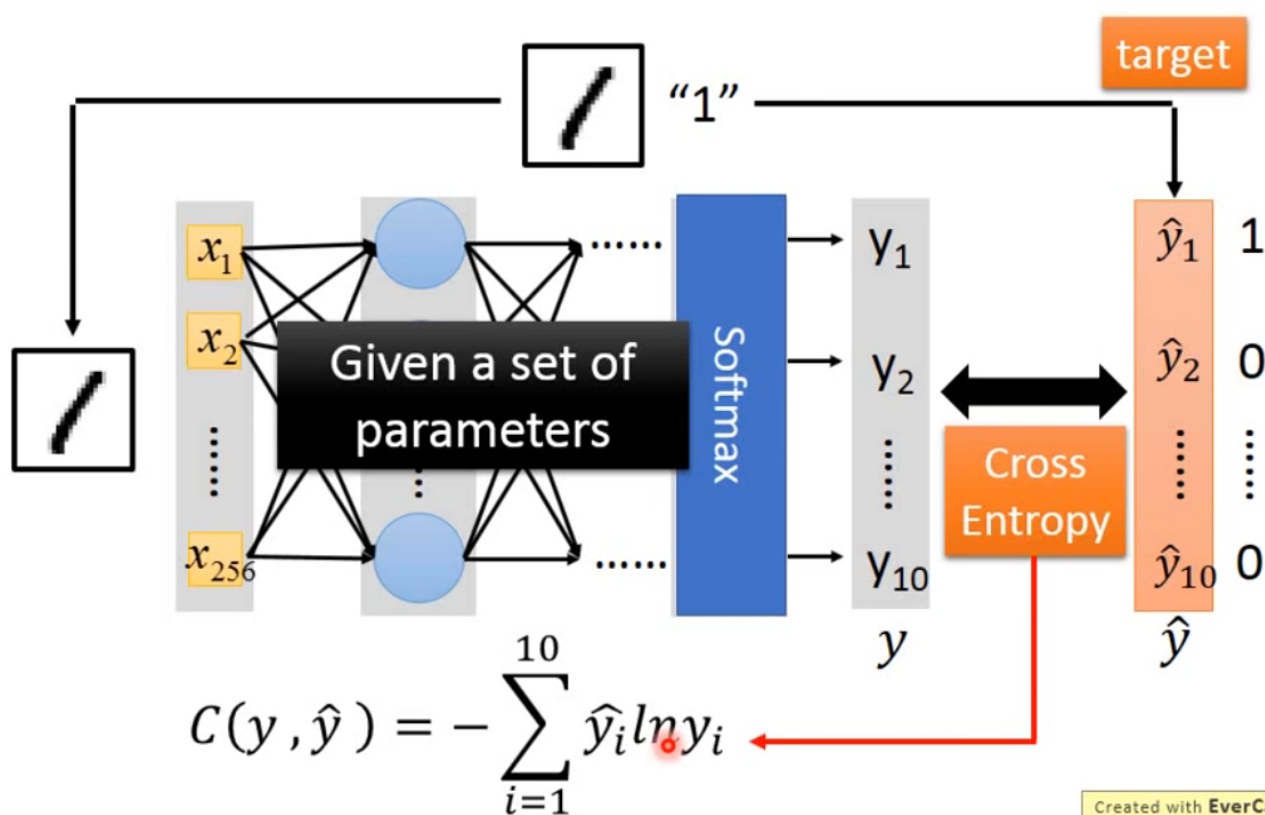
Intuition

- Q: Can the structure be automatically determined?
  - E.g. Evolutionary Artificial Neural Networks
- Q: Can we design the network structure?

## Convolutional Neural Network (CNN)

### 2.判断函数的好坏

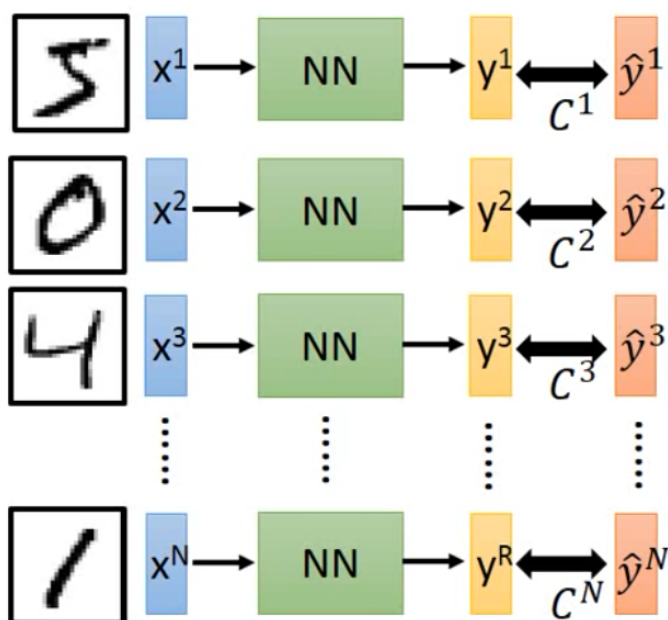
通过使用损失函数，计算预测结果与样本标签的交叉熵。从而判断函数的好坏。



这是计算一个样本的损失函数，而整个样本集的损失函数则如下图所示

# Total Loss

For all training data ...



Total Loss:

$$L = \sum_{n=1}^N C^n$$

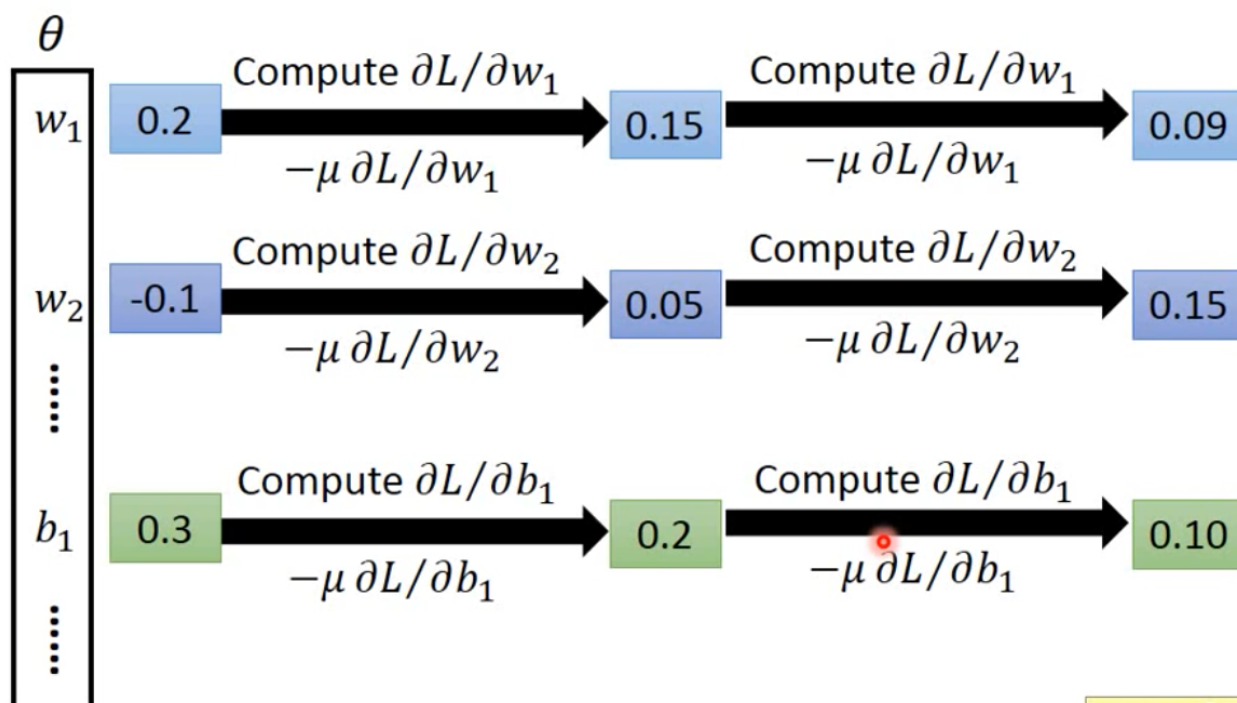
Find a function in function set that minimizes total loss  $L$

Find the network parameters  $\theta^*$  that minimize total loss  $L$

Created with EverCam.  
<http://www.camdemy.com>

## 3.使用梯度下降

在选择确定了函数之后，则进行梯度下降来确定最优参数。



Created with EverCam.