希望权值改变一点点，输出结果也跟着改变一点点，这样就可以调整权值来学习网络，但Perceptron不能满足这个要求，Sigmoid neuron可以完成这个任务。Sigmoid函数：



Sigmoid函数：



输出改变：



表示输出改变和权值改变以及偏差改变是线性的。

激活函数

MLP：Multilayer Perceptron

循环神经网络：recurrent neural networks可以允许loops

Cost function和梯度下降法（复杂度高）。

统计梯度下降法，用一部分样本的梯度均值代替整体的梯度，降低复杂度。虽然精度下降，但我们只需要一个大致的下降方向，而不需要准确的梯度值。实际情况用得最多

Backpropagation，两个假设

1，代价函数：



实际上只要求 即可

2，

Hadamard Product即元素点乘

**反向传播的四个基本方程**：

权重及激励：

表示第层的第个神经元到第层第个神经元的权重

表示第层第个神经元的激活值，且

矩阵形式为：

，第k行j列的元素为，，则定义：





代价函数：



其中n为样本个数，x表示样本。后面将省略索引x。

误差函数：



四个基本函数：









上式的矩阵形式：





方向传递的证明，因为有：



有因为有：



带入上式得到：

