

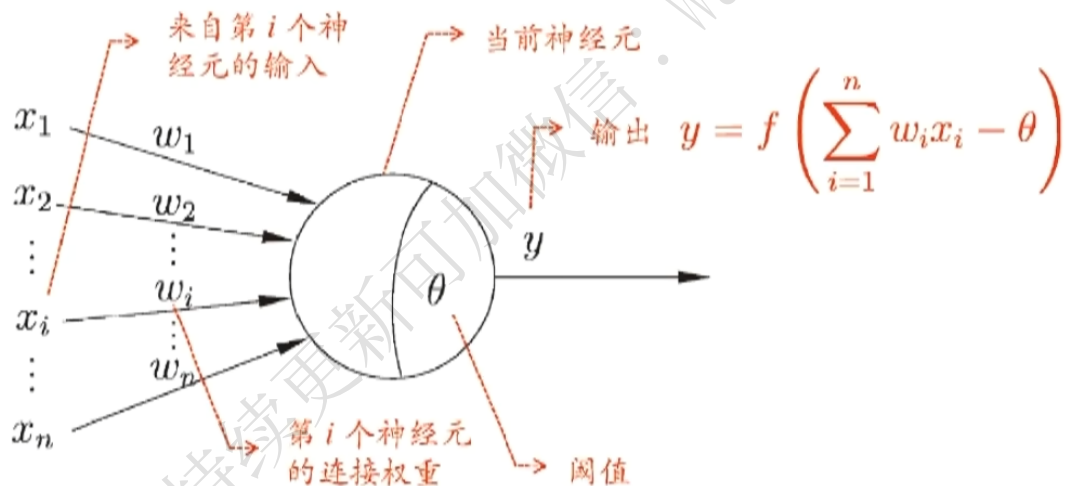
Lecture 5: 神经网络模型

神经网络基础概念

概念：神经网络基础概念

- 概念：
 - "神经网络是由具有适应性的简单单元组成的广泛并行互连的网络，它的组织能够模拟生物神经系统对真实世界物体所作出的交互反应”；
 - 神经网络是一个很大的学科领域，本课程仅讨论神经网络与机器学习的交集，即“神经网络学习”，亦称“连接主义 (connectionism)”学习；
- 概念：神经元模型
 - 图示：

M-P 神经元模型 [McCulloch and Pitts, 1943]



神经网络学得的知识蕴含在连接权与阈值中

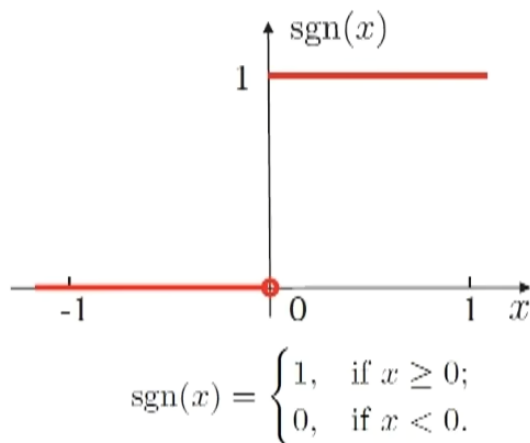
- 公式：

$$y = f \left(\sum_{i=1}^n w_i x_i - \theta \right)$$

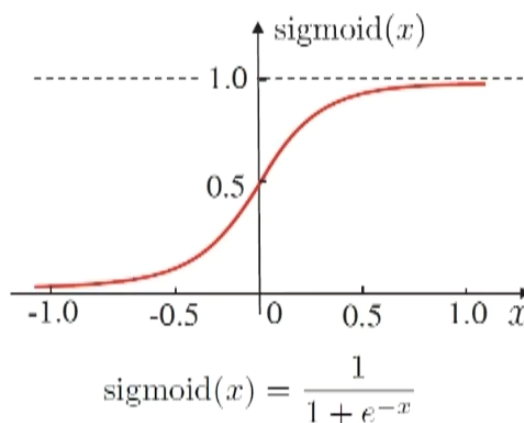
- x 为信号， w 为放大的权值，将所有信号乘以权值求和，如果大于阈值，则输出结果 y ；
- 其中的 $f()$ 就是激活函数；

概念：神经元的激活函数

- 概念：
 - 理想激活函数是阶跃函数，0 表示抑制神经元，而 1 表示激活神经元；
 - 阶跃函数具有不连续、不光滑等不好的性质，常用的是 Sigmoid 函数；
- 图示：典型的神经元激活函数



(a) 阶跃函数

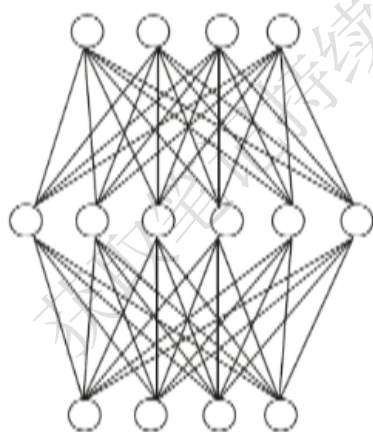


(b) Sigmoid 函数

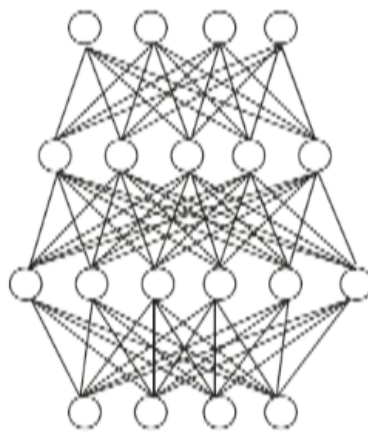
概念：多层前馈网络结构

- 概念：
 - 多层网络：
 - 包含隐层的网络；
 - 前馈网络：
 - 神经元之间不存在同层连接也不存在跨层连接，即不存在环；
 - 隐层和输出层神经元亦称“功能单元”(Functional Unit)；
 - 假设输入向量一共有十个类（十个维度），则最下层一共有十个结点；
 - 假设输出的结果一共两个可能的类，则最上层一共两个结点；
 - 中间称之为隐层神经元；

- 图示：



(a) 单隐层前馈网络



(b) 双隐层前馈网络

万有逼近能力

概念：万有逼近能力

- 概念：
 - 多层前馈网络有强大的表示能力（“万有逼近性”）；
 - 仅需一个包含足够多神经元的隐层，多层前馈神经网络就能以任意精度逼近任意复杂度的连续函数；

- 但是，如何设置隐层神经元数是未决问题 (Open Problem)；
- 实际常用“试错法”
- 意义：
 - 任何一个函数逼近的问题，都可以使用一个单隐层神经网络来解决它；
 - 而且：任何一个图灵可计算的问题，都可以找到一个神经网络来作为它的求解器；
 - 因此，万有逼近能力表示了神经网络学习的解存在，只是需要我们去寻找它；

BP 算法推导

概念： BackPropagation - 误差逆传播算法

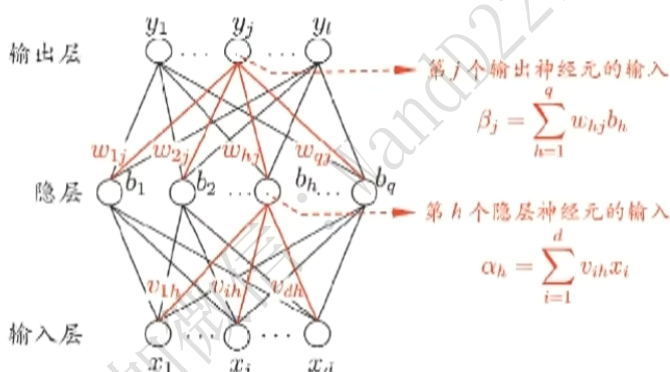
- 符号：

给定训练集 $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$, $x_i \in \mathbb{R}^d$, $y_i \in \mathbb{R}^l$

输入： d 维特征向量
输出： l 个输出值

隐层：假定使用 q 个隐层神经元

假定功能单元均使用
• Sigmoid 函数



- 推导：

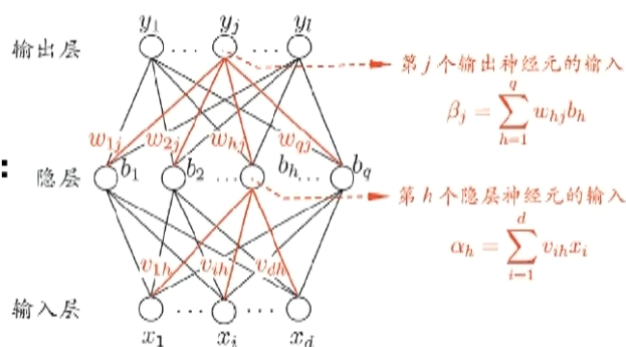
- 神经网络学习，就是确定网络中的参数；

对于训练例 (x_k, y_k) ，假定网络的实际输出为 $\hat{y}_k = (\hat{y}_1^k, \hat{y}_2^k, \dots, \hat{y}_l^k)$

$$\hat{y}_j^k = f(\beta_j - \theta_j)$$

则网络在 (x_k, y_k) 上的均方误差为：

$$E_k = \frac{1}{2} \sum_{j=1}^l (\hat{y}_j^k - y_j^k)^2$$



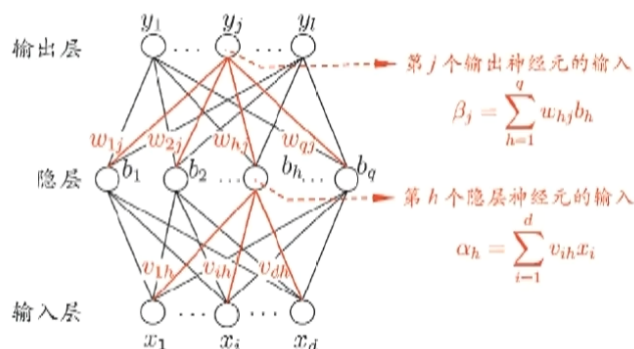
- 需通过学习确定的参数数目： $(d + l + 1)q + l$

- 其中：第一层和第二层连接为 $d \times q$ ，第二层和第三层连接为 $q \times l$ ，同时功能神经元还有 q 个，输出层还有 l 个；因此一共为 $(d+l+1)q+l$ 参数要学习；
- BP 是一个迭代学习算法，在迭代的每一轮中采用广义感知机学习规则；
- BP 算法基于梯度下降策略，以目标的负梯度方向对参数进行调整；

以 w_{hj} 为例

对误差 E_k , 给定学习率 η , 有:

$$\Delta w_{hj} = -\eta \frac{\partial E_k}{\partial w_{hj}}$$



- 得到误差之后, 要对其进行调整, 就对其求偏导, 而学习率为其梯度下降的幅度;

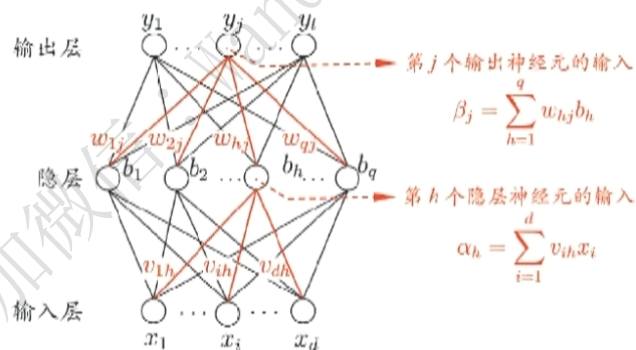
注意到 w_{hj} 先影响到 β_j , 再影响到 $\hat{y}_j^k$, 然后才影响到 E_k , 有:

$$\frac{\partial E_k}{\partial w_{hj}} = \frac{\partial E_k}{\partial \hat{y}_j^k} \cdot \frac{\partial \hat{y}_j^k}{\partial \beta_j} \cdot \frac{\partial \beta_j}{\partial w_{hj}}$$

$$\frac{\partial E_k}{\partial w_{hj}} = \boxed{\frac{\partial E_k}{\partial \hat{y}_j^k}} \cdot \boxed{\frac{\partial \hat{y}_j^k}{\partial \beta_j}} \cdot \boxed{\frac{\partial \beta_j}{\partial w_{hj}}}$$

$(\hat{y}_j^k - y_j^k)$

$= b_h$



注意到

$$\hat{y}_j^k = f(\beta_j - \theta_j)$$

$$f'(\beta_j - \theta_j)$$

$$\hat{y}_j^k (1 - \hat{y}_j^k)$$

对 $\text{sigmoid}(x) = \frac{1}{1 + e^{-x}}$, 有

$$f'(x) = f(x) (1 - f(x))$$

$$\text{令 } g_j = \frac{\partial E_k}{\partial \hat{y}_j^k} \cdot \frac{\partial \hat{y}_j^k}{\partial \beta_j}$$

$$= \hat{y}_j^k (1 - \hat{y}_j^k) (y_j^k - \hat{y}_j^k)$$

$$\text{于是, } \Delta w_{hj} = -\eta \frac{\partial E_k}{\partial w_{hj}} = \eta g_j b_h$$

似地，有：

$$\Delta\theta_j = -\eta g_j$$

$$\Delta v_{ih} = \eta e_h x_i$$

$$\Delta\gamma_h = -\eta e_h$$

中：

$$e_h = -\frac{\partial E_k}{\partial b_h} \cdot \frac{\partial b_h}{\partial \alpha_h}$$

$$= -\sum_{j=1}^l \frac{\partial E_k}{\partial \beta_j} \cdot \frac{\partial \beta_j}{\partial b_h} f'(\alpha_h - \gamma_h) = \sum_{j=1}^l w_{hj} g_j f'(\alpha_h - \gamma_h)$$

$$= b_h(1 - b_h) \sum_{j=1}^l w_{hj} g_j$$

学习率 $\eta \in (0, 1)$ 不能太大、不能太小

