



上海交通大学  
SHANGHAI JIAO TONG UNIVERSITY



# 人工神经网络基础





## 主要内容:

- BN与AN;
- 拓扑结构;
- 存储;
- 训练



**重点:** AN; 拓扑结构; 训练



**难点:** 训练

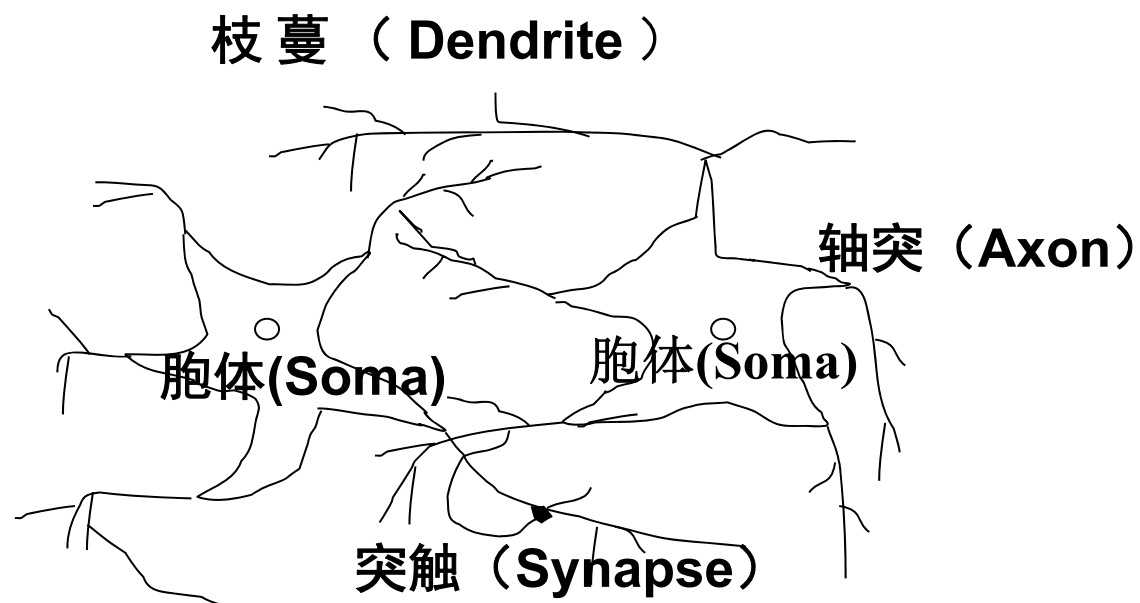
---



- 1 生物神经网络
  - 2 人工神经元
  - 3 人工神经网络的拓扑特性
  - 4 存储方式
  - 5 人工神经网络的训练
-

# 1 生物神经网络

## (1)、构成



## (2)、工作过程

# 1 生物神经网络

---

## (3)、六个基本特征：

- 1) 神经元及其联接；
  - 2) 神经元之间的联接强度决定信号传递的强弱；
  - 3) 神经元之间的联接强度是可以随训练改变的；
  - 4) 信号可以是起刺激作用的，也可以是起抑制作用的；
  - 5) 一个神经元接受的信号的累积效果决定该神经元的状态；
  - 6) 每个神经元可以有一个“阈值”。
-

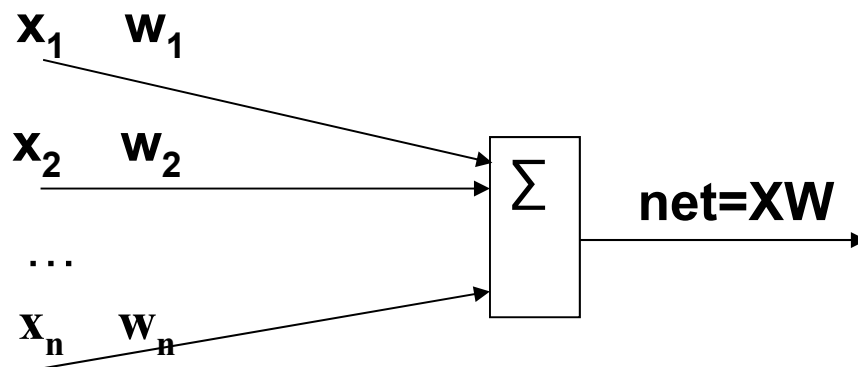
## 2 人工神经元

---

- ① 神经元是构成神经网络的最基本单元（构件）。
- ① 人工神经元模型应该具有生物神经元的六个基本特性。



## 2.1 人工神经元的基本构成



⊙ 人工神经元模拟生物神经元的一阶特性。

- 输入:  $X = (x_1, x_2, \dots, x_n)$
- 联接权:  $W = (w_1, w_2, \dots, w_n)^T$
- 网络输入:  $\text{net} = \sum x_i w_i$
- 向量形式:  $\text{net} = XW$

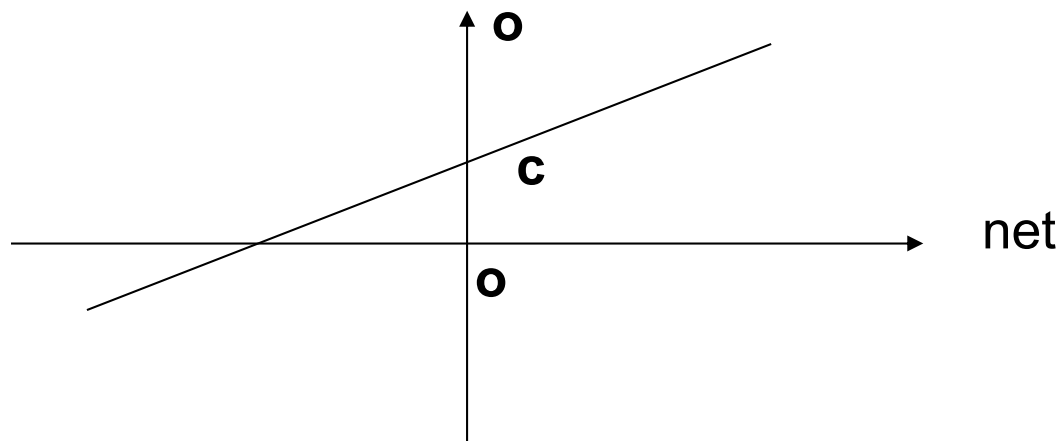


## 2.2 激活函数(Activation Function)

- 激活函数——执行对该神经元所获得的网络输入的变换，也可以称为激励函数、活化函数： $o=f(\text{net})$

- (1)、线性函数 (Liner Function)

$$f(\text{net}) = k * \text{net} + c$$





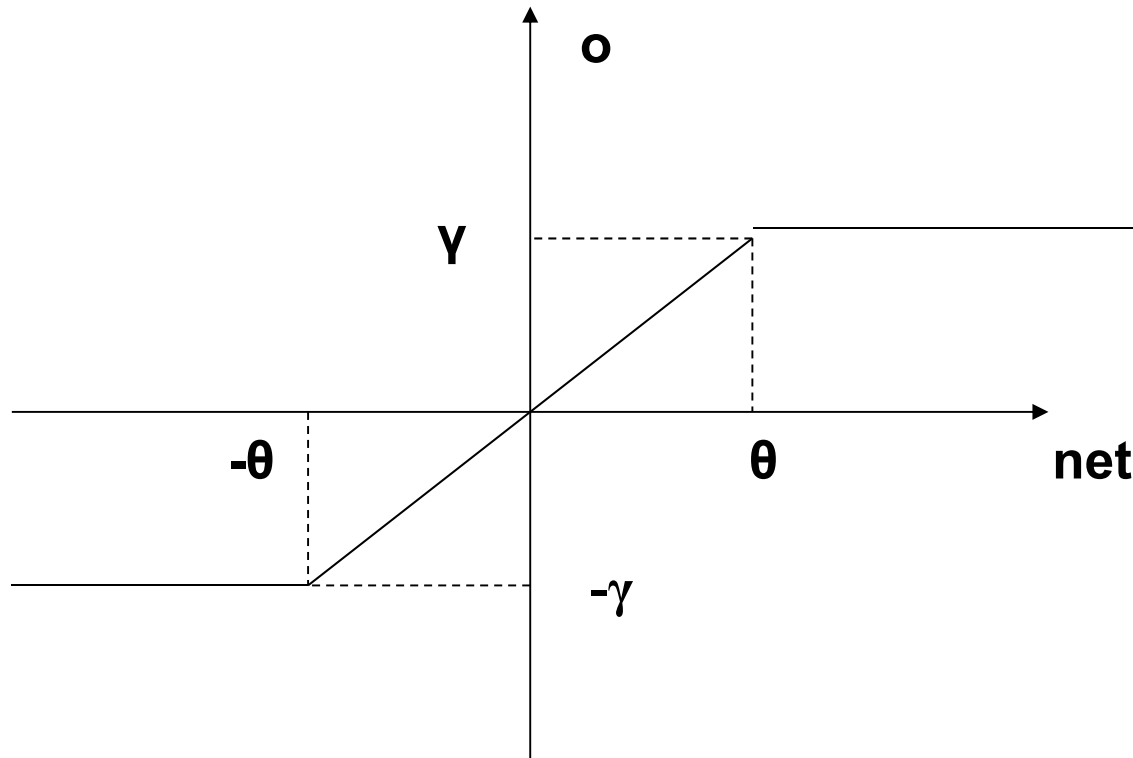
## (2)、非线性斜面函数(Ramp Function)

$$f(\text{net}) = \begin{cases} \gamma & \text{if } \text{net} \geq \theta \\ k * \text{net} & \text{if } |\text{net}| < \theta \\ -\gamma & \text{if } \text{net} \leq -\theta \end{cases}$$

- ④  $\gamma > 0$ 为一常数，被称为饱和值，为该神经元的最大输出。
-



# 非线性斜面函数 (Ramp Function)



### (3)、阈值函数 (Threshold Function) 阶跃函数

$$f(\text{net}) = \begin{cases} \beta & \text{if net} > \theta \\ -\gamma & \text{if net} \leq \theta \end{cases}$$

$\beta$ 、 $\gamma$ 、 $\theta$ 均为非负实数， $\theta$ 为阈值

二值形式:

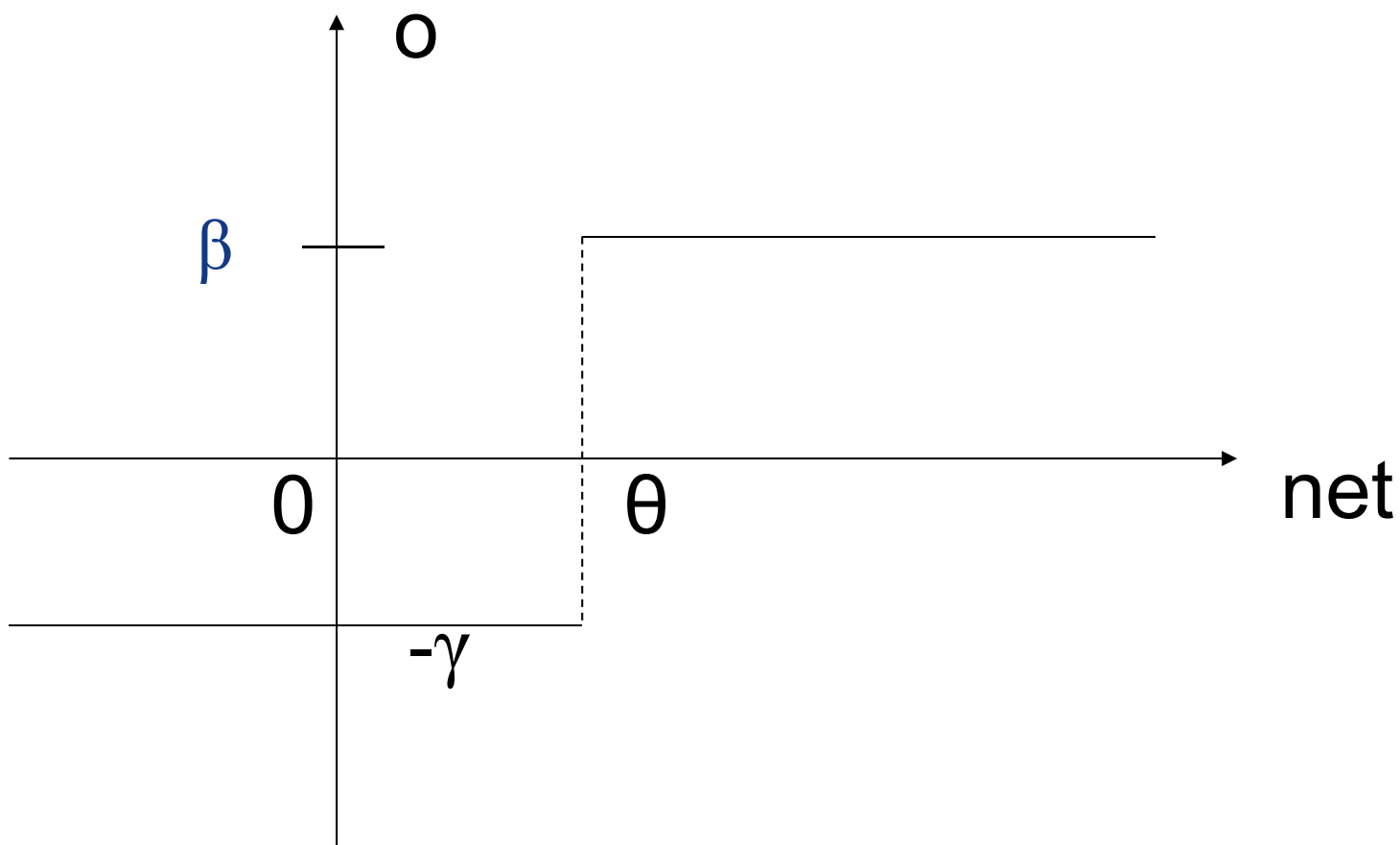
$$f(\text{net}) = \begin{cases} 1 & \text{if net} > \theta \\ 0 & \text{if net} \leq \theta \end{cases}$$

双极形式:

$$f(\text{net}) = \begin{cases} 1 & \text{if net} > \theta \\ -1 & \text{if net} \leq \theta \end{cases}$$



# 阈值函数 (Threshold Function) 阶跃函数



## (4)、S形函数

---

压缩函数（Squashing Function）和逻辑斯特函数（Logistic Function）。

$$f(\text{net}) = a + b / (1 + \exp(-d * \text{net}))$$

a, b, d为常数。它的饱和值为a和a+b。

最简单形式为：

$$f(\text{net}) = 1 / (1 + \exp(-d * \text{net}))$$

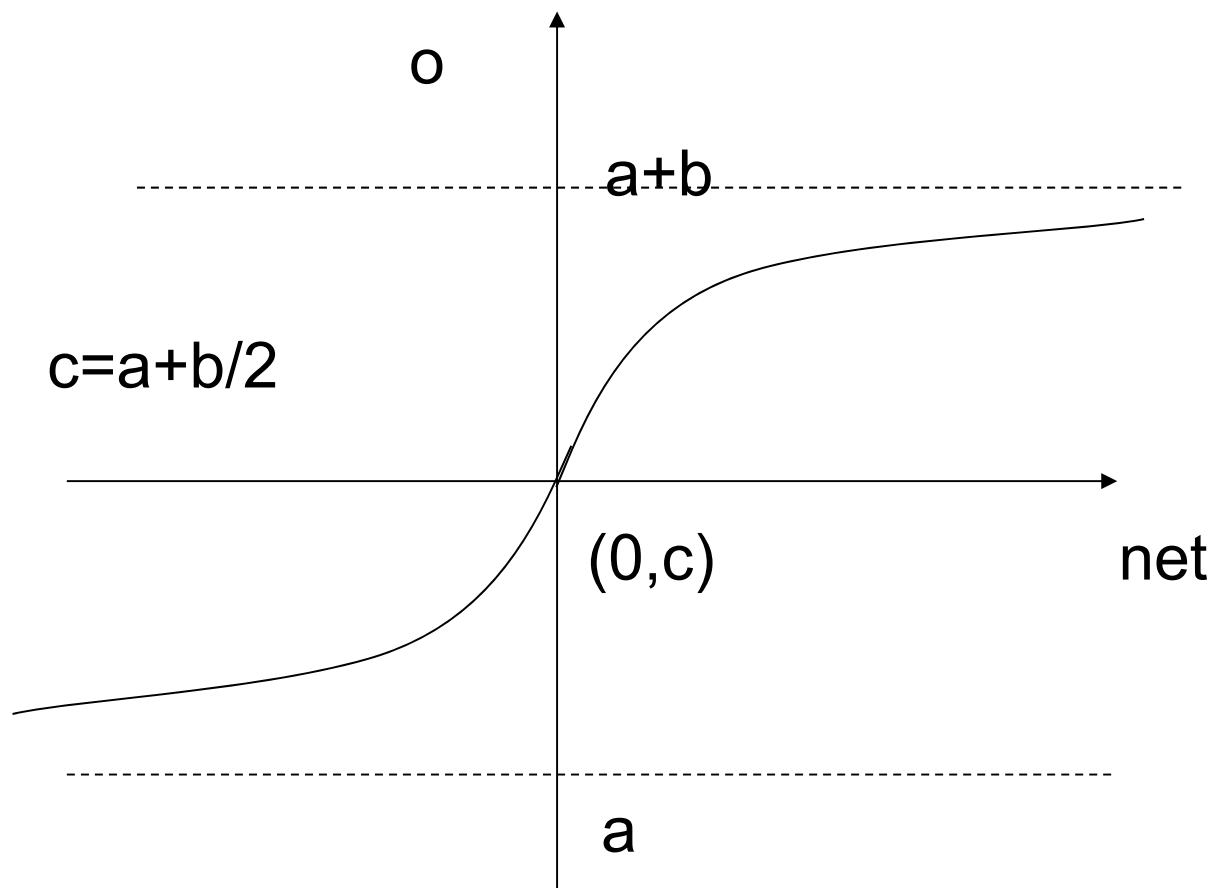
函数的饱和值为0和1。

⊙ S形函数有较好的增益控制

---



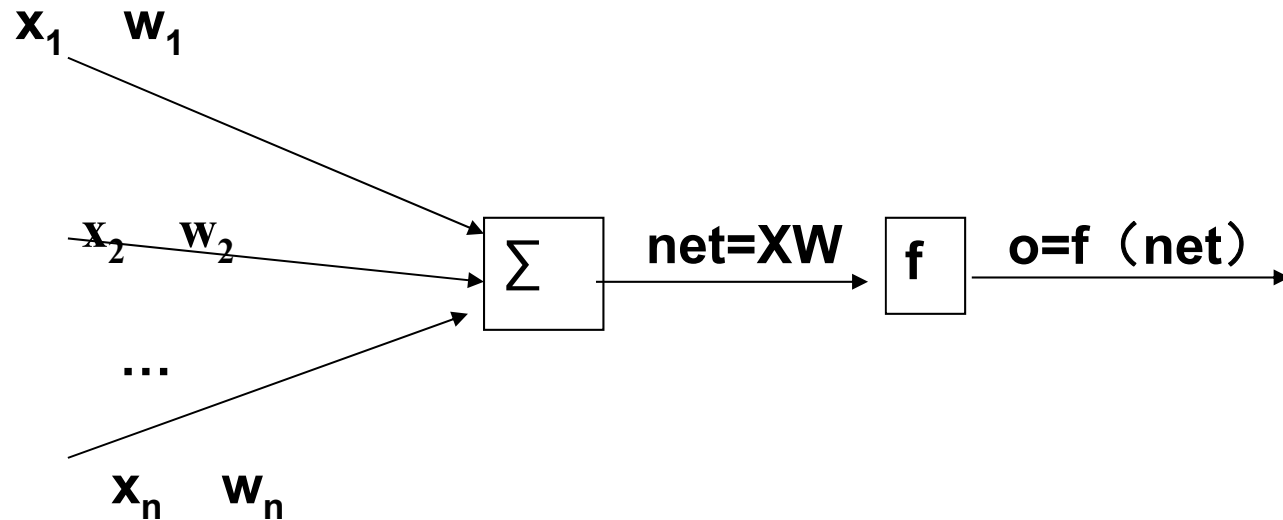
# S形函数





## 2.3 M-P模型

**McCulloch—Pitts (M—P) 模型，  
也称为处理单元 (PE)**



### 3 人工神经网络的拓扑特性

---

#### 连接的拓扑表示





## 3.1 联接模式

---

- ④ 用正号（“+”，可省略）表示传送来的信号起**刺激**作用，它用于增加神经元的活跃度；
  - ④ 用负号（“-”）表示传送来的信号起**抑制**作用，它用于降低神经元的活跃度。
  - ④ **层次**（又称为“级”）的划分，导致了神经元之间的三种不同的**互连模式**：
-



## 3.1 联接模式

---



### 1、层（级）内联接

- 层内联接又叫做区域内（**Intra-field**）联接或侧联接（**Lateral**）。
- 用来加强和完成层内神经元之间的竞争



### 2、循环联接

- 反馈信号。
-

## 3.1 联接模式

---

### ③ 3、层（级）间联接

- 层间（**Inter-field**）联接指不同层中的神经元之间的联接。这种联接用来实现层间的信号传递
  - 前馈信号
  - 反馈信号
-

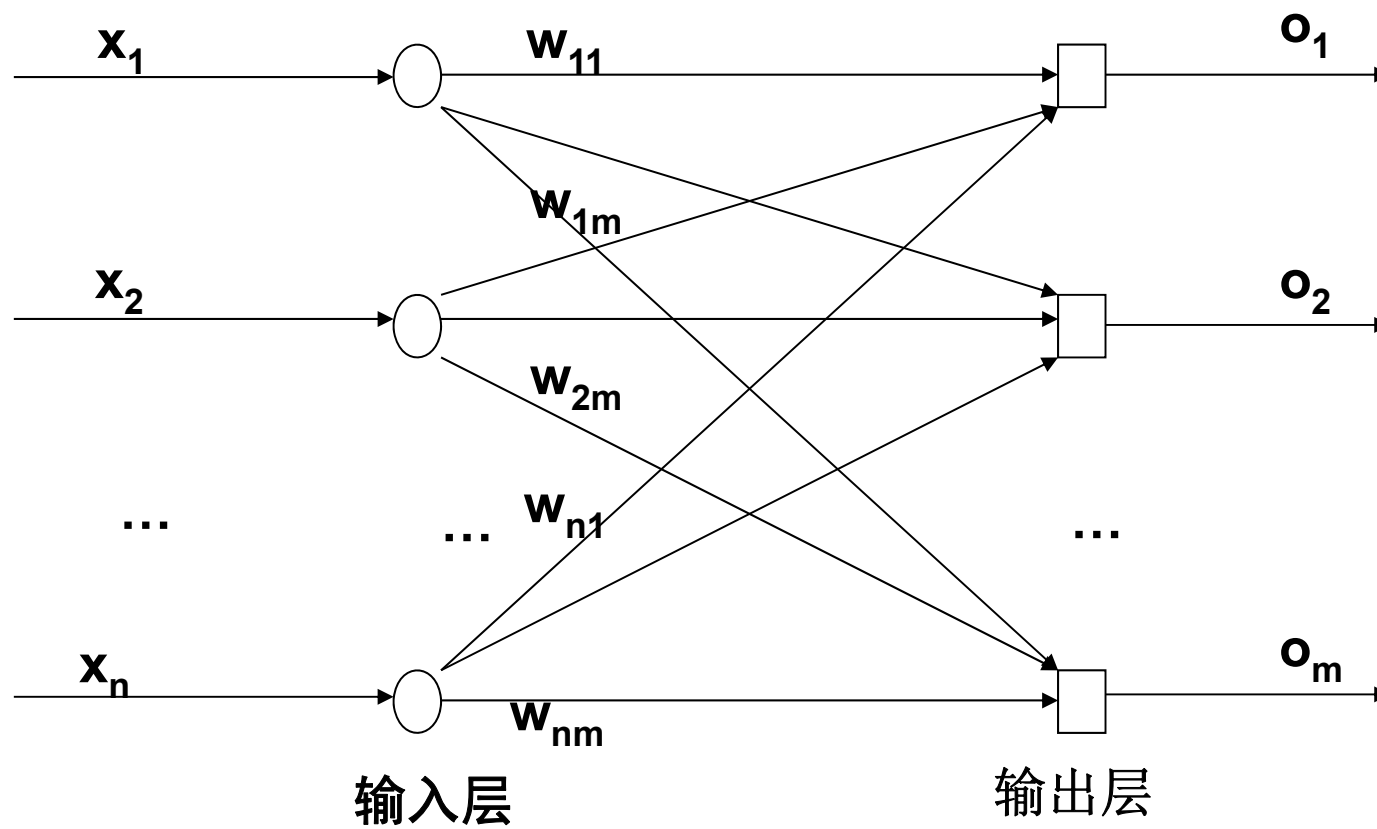


### 单级网

- 简单单级网



## 简单单级网



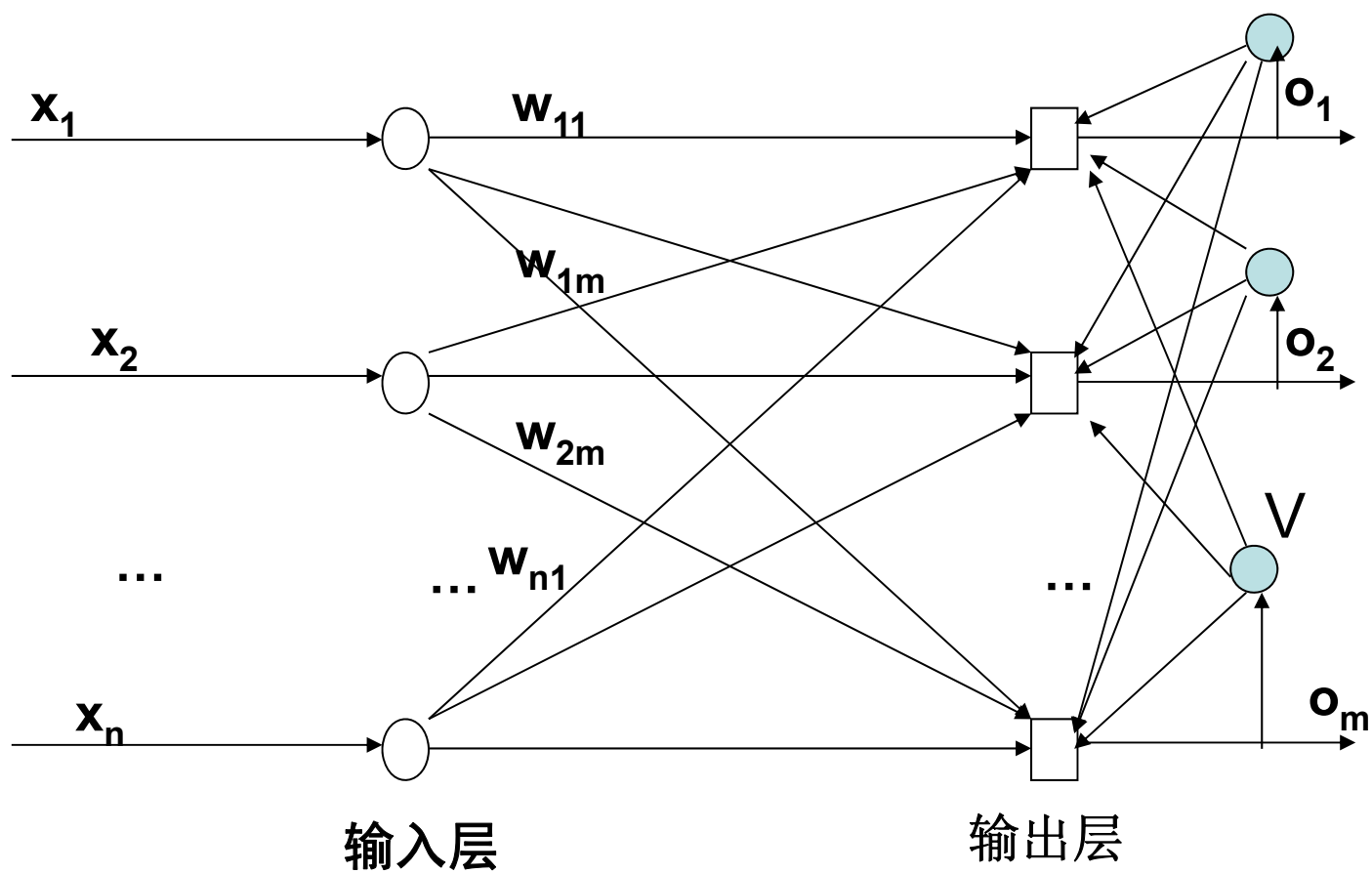


## 简单单级网

- $\mathbf{W} = (\mathbf{w}_{ij})$
- 输出层的第j个神经元的网络输入记为 $\mathbf{net}_j$ :
- $\mathbf{net}_j = x_1 \mathbf{w}_{1j} + x_2 \mathbf{w}_{2j} + \dots + x_n \mathbf{w}_{nj}$
- 其中,  $1 \leq j \leq m$ 。取
- $\mathbf{NET} = (\mathbf{net}_1, \mathbf{net}_2, \dots, \mathbf{net}_m)$
- $\mathbf{NET} = \mathbf{XW}$ , 其中  $\mathbf{X} = (x_1, x_2, \dots, x_n)$
- $\mathbf{O} = \mathbf{F}(\mathbf{NET}) = (f_1(\mathbf{net}_1), f_2(\mathbf{net}_2), \dots, f_n(\mathbf{net}_n))$



# 单级横向反馈网



## 单级横向反馈网

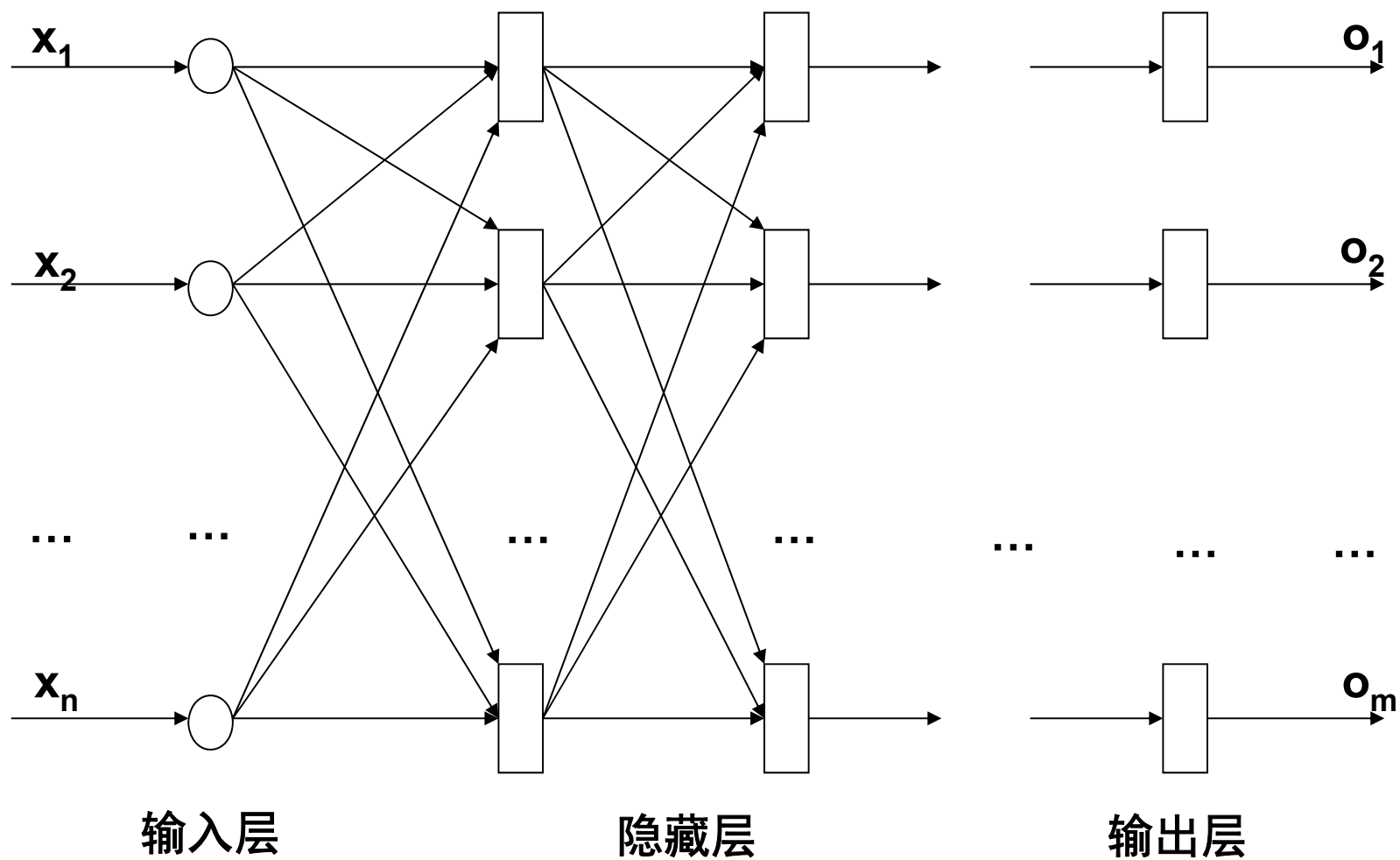
---

- ④  $V = (v_{ij})$
  - ④  $NET = XW + OV$
  - ④  $O = F(NET)$
  - ④ 时间参数——神经元的状态在主时钟的控制下同步变化
  - ④ 考虑X总加在网上的情况
    - $NET(t+1) = X(t)W + O(t)V$
    - $O(t+1) = F(NET(t+1))$
    - $O(0) = 0$
  - ④ 考虑仅在 $t=0$ 时加X的情况。
  - ④ 稳定性判定
-





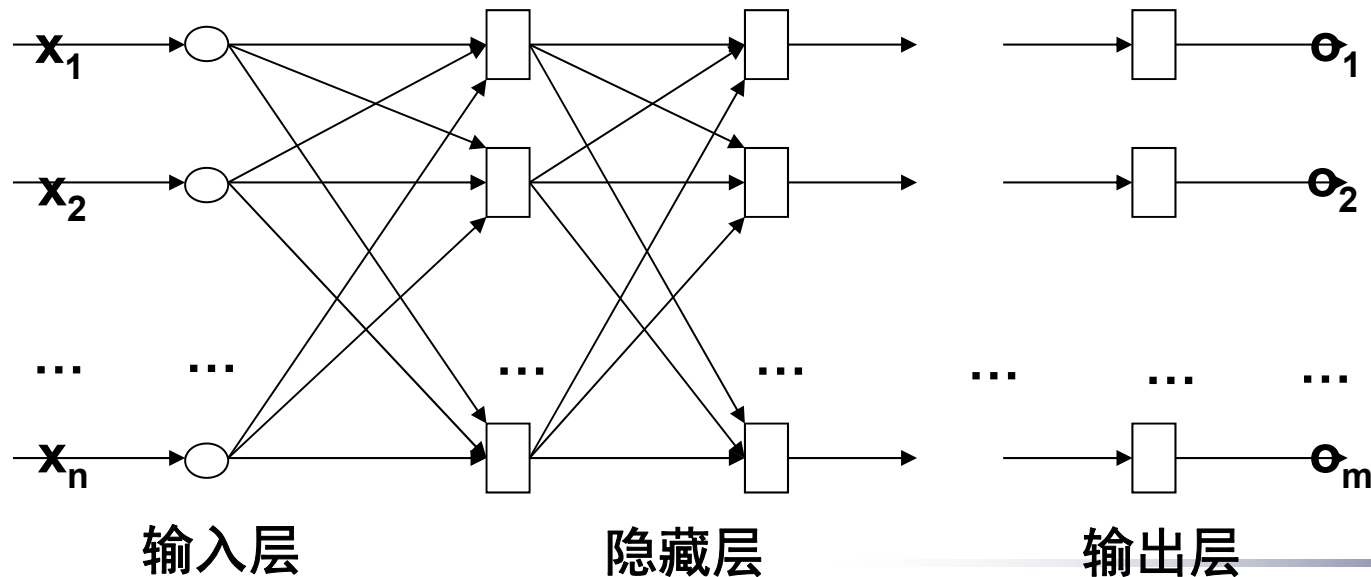
# 多级网



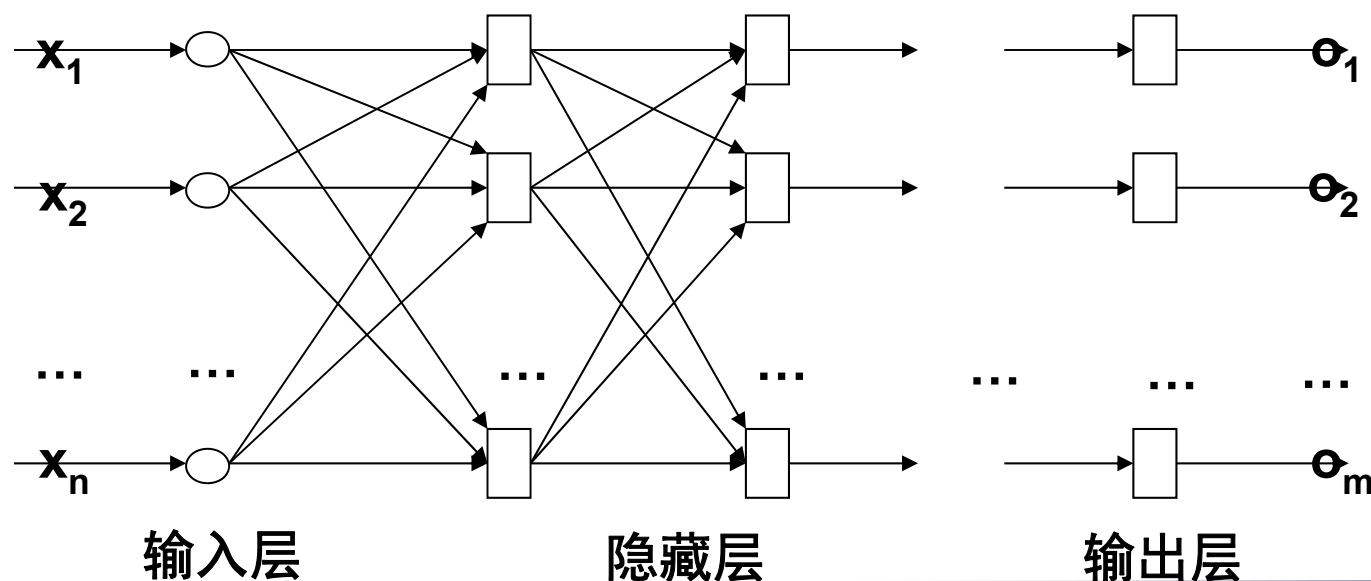


## 层次划分

- 信号只被允许从较低层流向较高层。
- 层号确定层的高低：层号较小者，层次较低，层号较大者，层次较高。
- **输入层**：被记作第0层。该层负责接收来自网络外部的信息

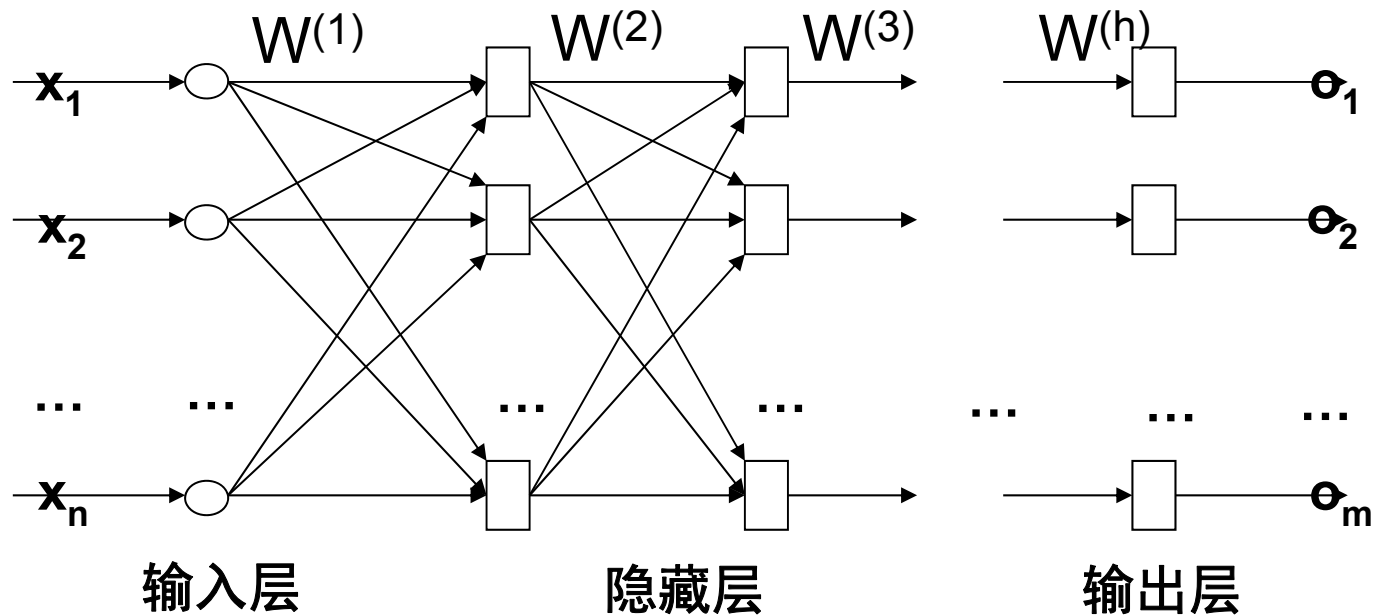


- **第j层：**第j-1层的直接后继层（ $j>0$ ），它直接接受第j-1层的输出。
- **输出层：**它是网络的最后一层，具有该网络的最大层号，负责输出网络的计算结果。
- **隐藏层：**除输入层和输出层以外的其它各层叫隐藏层。隐藏层不直接接受外界的信号，也不直接向外界发送信号。



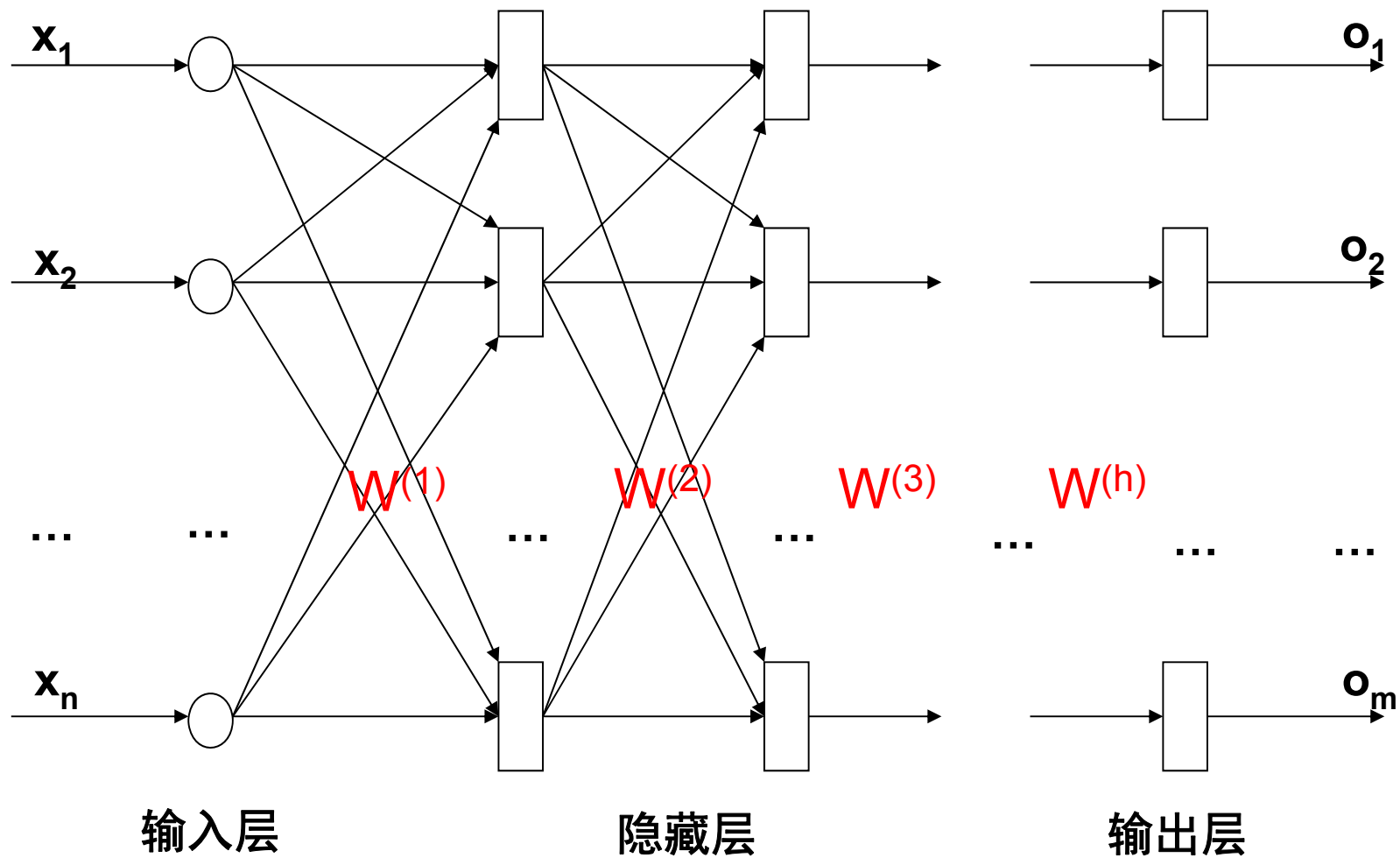
## 约定：

- 输出层的层号为该网络的层数：**n**层网络，或**n**级网络。
- 第**j-1**层到第**j**层的联接矩阵为第**j**层联接矩阵，输出层对应的矩阵叫输出层联接矩阵。今后，在需要的时候，一般我们用 $\mathbf{W}^{(j)}$ 表示第**j**层矩阵。





## 多级网——h层网络





## 非线性激活函数

- 使用线性激活函数则多级网的功能不会超过单级网的功能

输入向量:  $X = (x_1, x_2, \dots, x_l)$

各级联接矩阵:  $W^{(1)}, W^{(2)}, \dots, W^{(n)}$

各级输入向量:  $NET_1, NET_2, \dots, NET_n$

各级激活函数:  $F_1, F_2, \dots, F_n$



则有：

$$NET_1 = XW^{(1)}$$

$$NET_2 = F_1(NET_1)W^{(2)}$$

$$NET_{i+1} = F_i(NET_i)W^{(i+1)}, \quad i = 1, 2, \dots, n-1$$

• 设激活函数是线性函数：

$$F_i(NET_i) = K_i NET_i + A_i, \quad i = 1, 2, \dots, n$$



其中：

$$K_i = (k_1^{(i)}, k_2^{(i)}, \dots, k_{n_i}^{(i)})$$

$$NET_i = (net_1^{(i)}, net_2^{(i)}, \dots, net_{n_i}^{(i)})$$

$$K_i NET_i = (k_1^{(i)} net_1^{(i)}, k_2^{(i)} net_2^{(i)}, \dots, k_{n_i}^{(i)} net_{n_i}^{(i)})$$

$$A_i = (a_1^{(i)}, a_2^{(i)}, \dots, a_{n_i}^{(i)})$$

• 激活函数还可以写成：

$$F_i(NET_i) = NET_i K_i + A_i, \quad i = 1, 2, \dots, n$$



此时 $\mathbf{K}_i$ 为对角矩阵:

$$K_i = \text{diag}(k_1^{(i)}, k_2^{(i)}, \dots, k_{n_i}^{(i)}) = \begin{bmatrix} k_1^{(i)} & & & 0 \\ & k_2^{(i)} & & \\ & & \ddots & \\ 0 & & & k_{n_i}^{(i)} \end{bmatrix}$$



网络输出为:

$$\begin{aligned} O &= F_n(\dots F_3 \left( F_2 \left( F_1(NE T_1) W^{(2)} \right) W^{(3)} \right) \dots) \\ &= F_n(\dots F_3 \left( F_2 \left( F_1(XW^{(1)}) W^{(2)} \right) W^{(3)} \right) \dots) \\ &= F_n(\dots F_3 \left( F_2 \left( (XW^{(1)} K_1 + A_1) W^{(2)} \right) W^{(3)} \right) \dots) \\ &= F_n(\dots F_3 \left( F_2 \left( XW^{(1)} K_1 W^{(2)} + A_1 W^{(2)} \right) W^{(3)} \right) \dots) \\ &= F_n(\dots F_3 \left( \left( XW^{(1)} K_1 W^{(2)} K_2 + A_1 W^{(2)} K_2 + A_2 \right) W^{(3)} \right) \dots) \\ &= F_n(\dots F_3 \left( XW^{(1)} K_1 W^{(2)} K_2 W^{(3)} + A_1 W^{(2)} K_2 W^{(3)} + A_2 W^{(3)} \right) \dots) \end{aligned}$$

## 网络输出为:

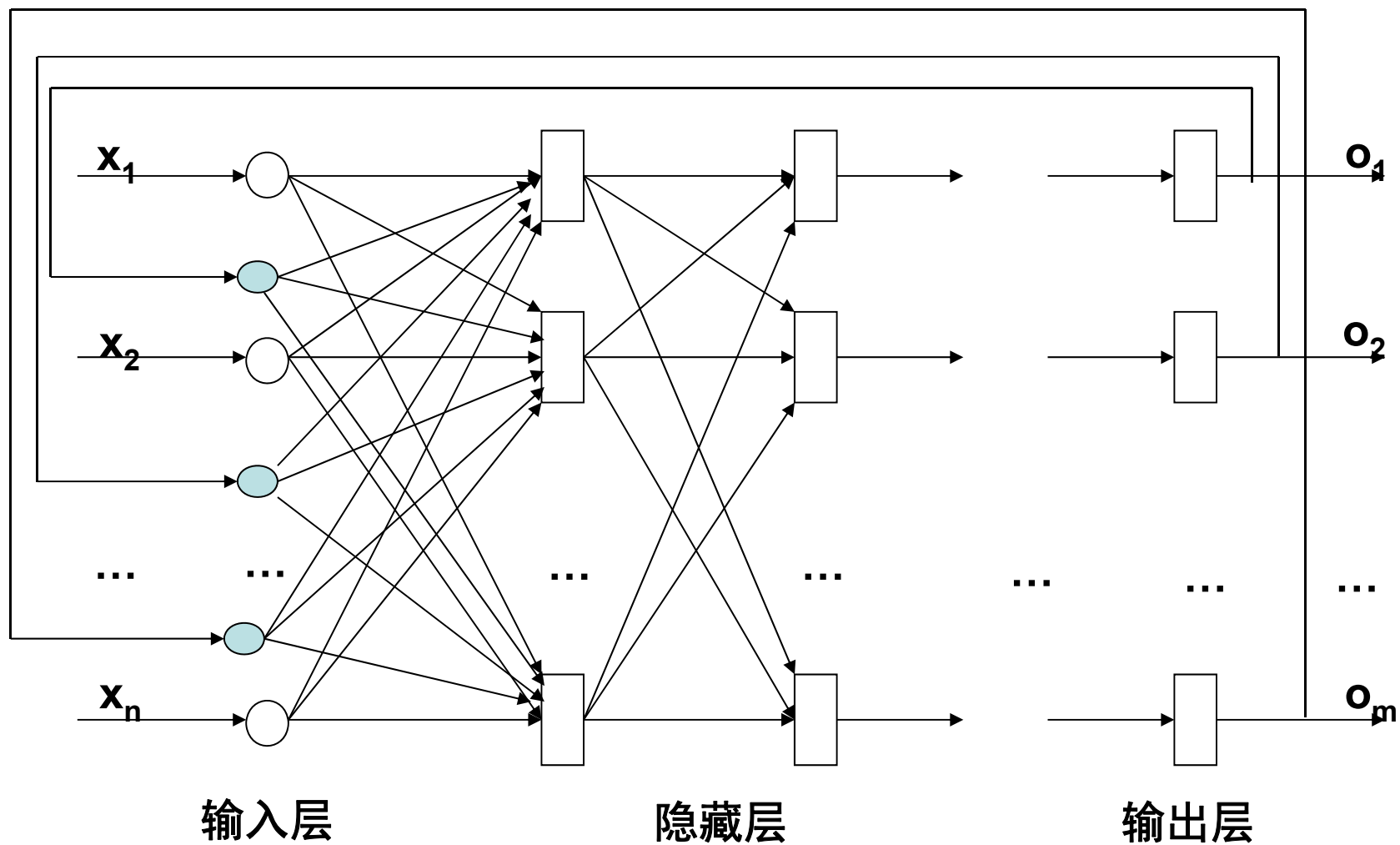
$$\begin{aligned} &= XW^{(1)}K_1W^{(2)}K_2W^{(3)}K_3\dots W^{(n)}K_n + \\ &A_1W^{(2)}K_2W^{(3)}K_3\dots + A_2W^{(3)}K_3\dots + A_3\dots + \dots \\ &= XWK + A \end{aligned}$$

## • 单级网络输出为:

$$\begin{aligned} O &= F_1(NE T_1) = F_1(XW^{(1)}) \\ &= XW^{(1)}K_1 + A_1 \end{aligned}$$



# 循环网





# 循环网

---

- ④ 如果将输出信号反馈到输入端,就可构成一个多层的循环网络。
  - ④ 输入的原始信号被逐步地“加强”、被“修复”。
  - ④ 大脑的**短期记忆特征**——看到的東西不是一下子就从脑海里消失的。
  - ④ **稳定**: 反馈信号会引起网络输出的不断变化。我们希望这种变化逐渐减小,并且最后能消失。当变化最后消失时,网络达到了平衡状态。如果这种变化不能消失,则称该网络是不稳定的。
-



## 4 存储方式

---

- ④ 空间模式 (Spatial Model)
  - ④ 时空模式 (Spatiotemporal Model)
  - ④ 空间模式三种存储类型
    - ④ 1、 RAM方式 (Random Access Memory)
      - 随机访问方式是将地址映射到数据。
    - ④ 2、 CAM方式 (Content Addressable Memory)
      - 内容寻址方式是将数据映射到地址。
    - ④ 3、 AM方式 (Associative Memory)
      - 相联存储方式是将数据映射到数据。
-



## 4 存储方式

---

- 后续的两种方式是人工神经网络的工作方式。
  - 在学习/训练期间，人工神经网络以**CAM**方式工作；权矩阵又被称为网络的**长期存储**（**Long Term Memory**，简记为**LTM**）。
  - 网络在正常工作阶段是以**AM**方式工作的；神经元的状态表示的模式为**短期存储**（**Short Term Memory**，简记为**STM**）。
-

## 5 人工神经网络的训练

---

- ④ 人工神经网络最具有吸引力的特点是它的学习能力。
  - ④ 1962年，**Rosenblatt**给出了人工神经网络著名的学习定理：人工神经网络可以学会它可以表达的任何东西。
  - ④ 人工神经网络的表达能力大大地限制了它的学习能力。
  - ④ 人工神经网络的学习过程就是对它的训练过程。
-



## 5.1 无导师学习

---

- ① 无导师学习(Unsupervised Learning)与无导师训练(Unsupervised Training)相对应
  - ② 抽取样本集合中蕴含的统计特性，并以神经元之间的联接权的形式存于网络中。
-



## 5.1无导师学习

---

- Hebb 学习律、竞争与协同（Competitive and Cooperative）学习、随机联接系统（Randomly Connected Learning）等。
  - Hebb算法[D. O. Hebb在1961年]的核心：
    - 当两个神经元同时处于激发状态时被加强，否则被减弱。
    - 数学表达式表示：
      - $\mathbf{W}_{ij} (t+1) = \mathbf{W}_{ij} (t) + \alpha \mathbf{o}_i (t) \mathbf{o}_j (t)$
-

## 5.2 有导师学习

---

- ① 有导师学习(Supervised Learning)与有导师训练(Supervised Training)相对应。
  - ② 输入向量与其对应的输出向量构成一个“训练对”。
  - ③ 有导师学习的训练算法的主要步骤包括:
    - 1) 从样本集合中取一个样本 ( $A_i, B_i$ ) ;
    - 2) 计算出网络的实际输出  $O$ ;
    - 3) 求  $D=B_i-O$ ;
    - 4) 根据  $D$  调整权矩阵  $W$ ;
    - 5) 对每个样本重复上述过程, 直到对整个样本集来说, 误差不超过规定范围。
-