



倾听你的声音



利节
改编自汤志远版本



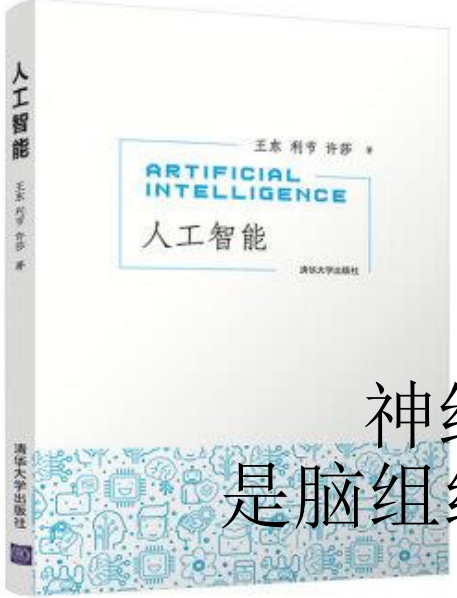
目录

- 神经网络
- Deep Clustering



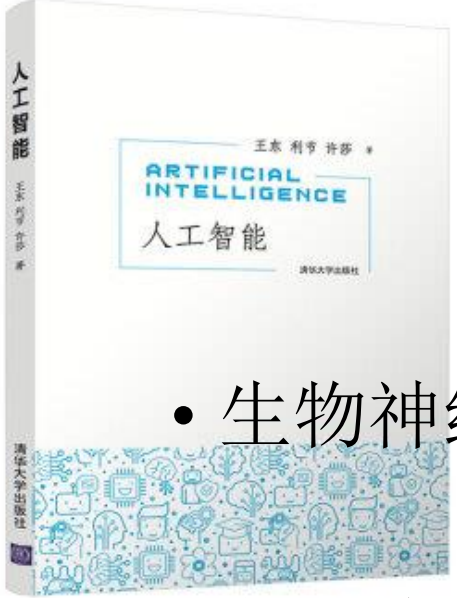
目录

- 神经网络
- Deep Clustering



神经生理学和神经解剖学的研究表明，神经元(Neuron)是脑组织的基本单元，是人脑信息处理系统的最小单元。

- 生物神经元
- 生物神经网络



- 生物神经元

生物神经元在结构上由:

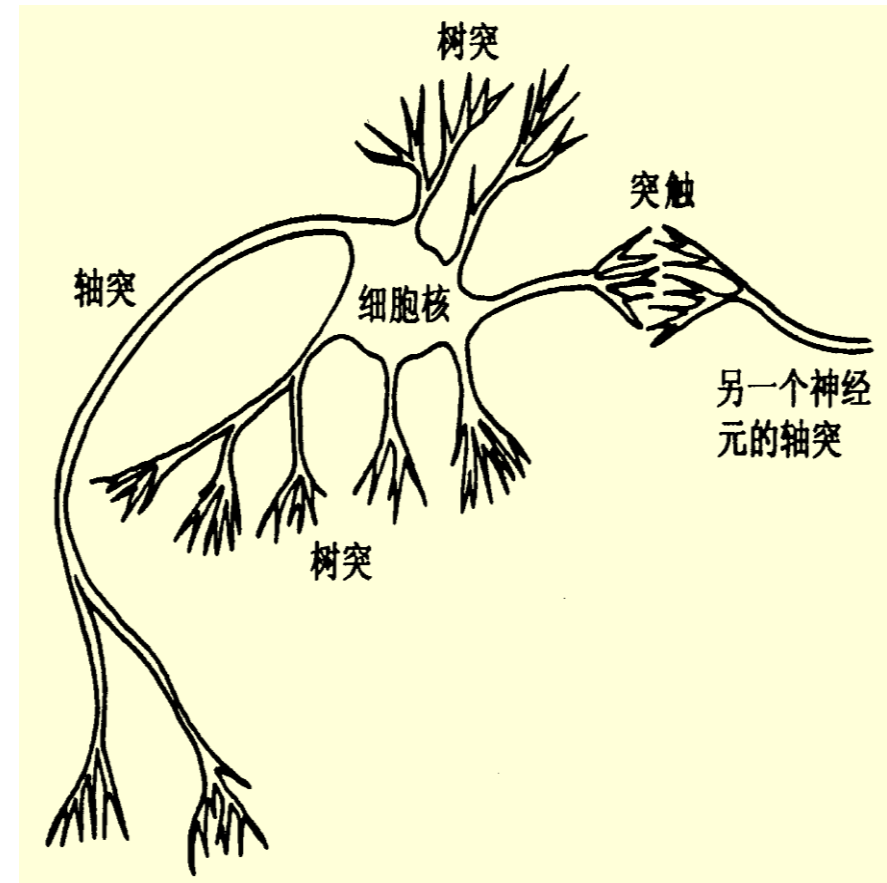
细胞体(Cell body)、

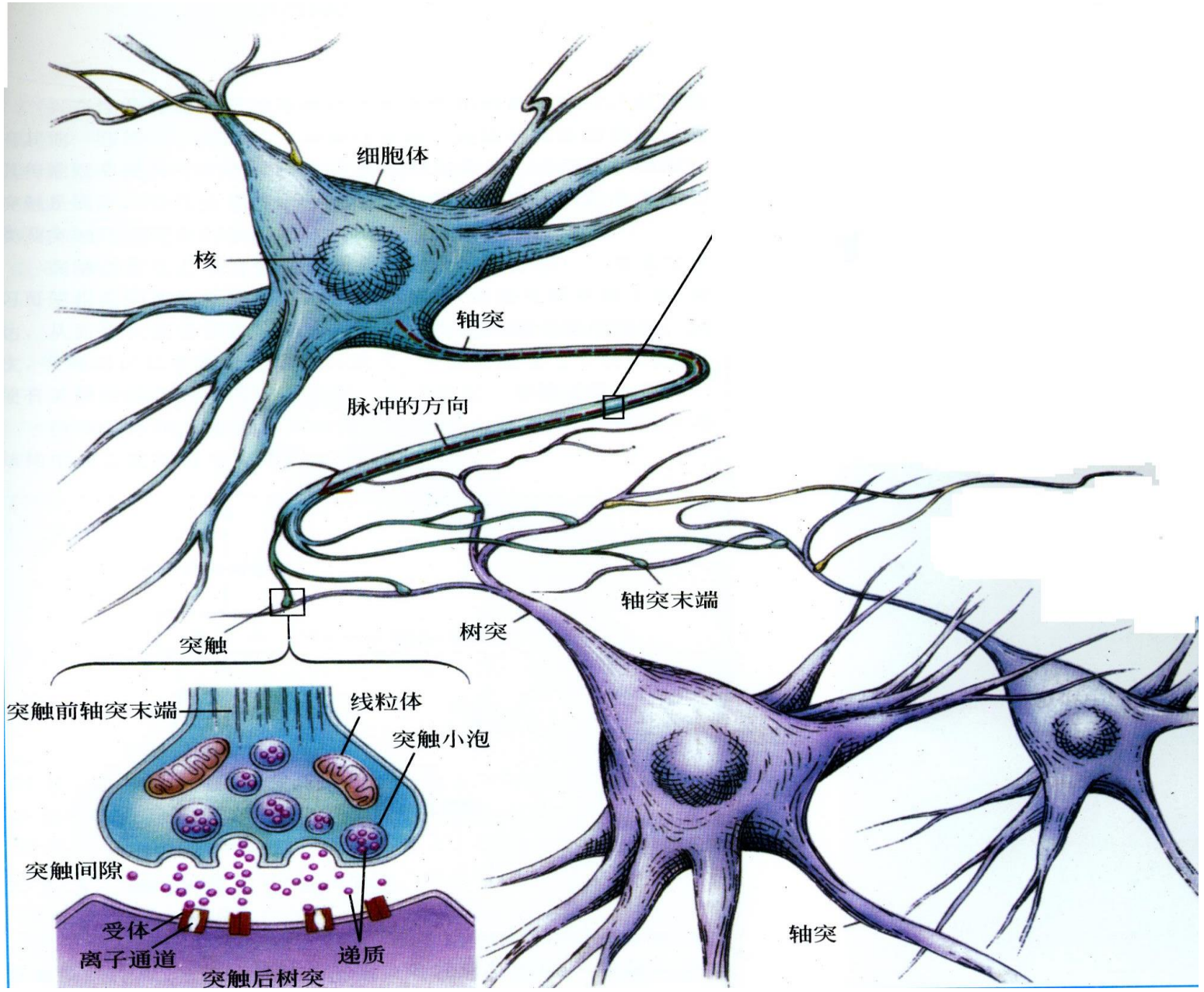
树突(Dendrite)、

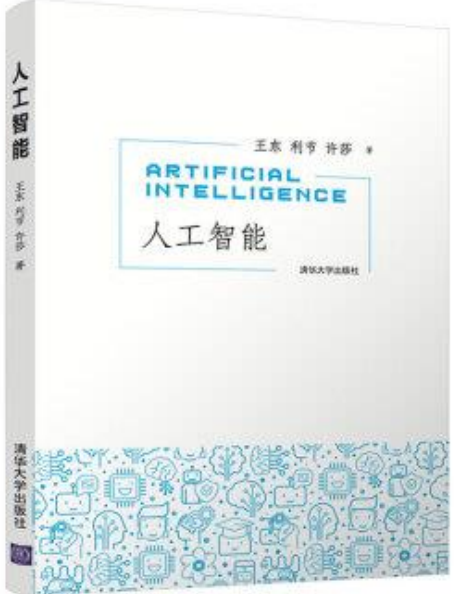
轴突(Axon)、

突触(Synapse)

四部分组成。用来完成神经元间信息的接收、传递和处理。







- **生物神经元:信息的产生**
 - 神经元间信息的产生、传递和处理是一种电化学活动。

神经元状态:

静息

兴奋

抑制

膜电位:

极 化

去极化

超极化

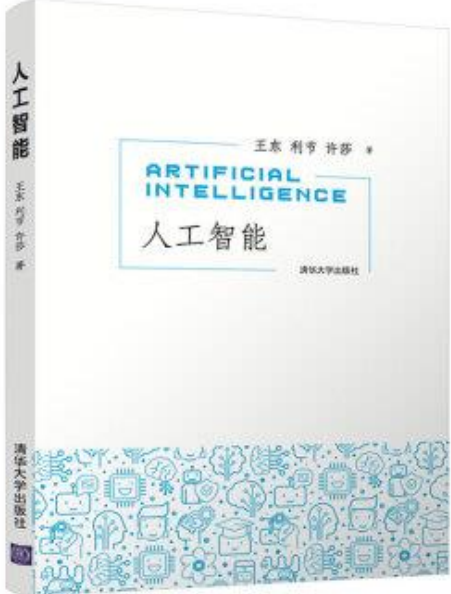




➤生物神经元:信息的整合

空间整合: 同一时刻产生的刺激所引起的膜电位变化, 大致等于各单独刺激引起的膜电位变化的代数和。

时间整合: 各输入脉冲抵达神经元的时间先后不一样。总的突触后膜电位为一段时间内的累积。

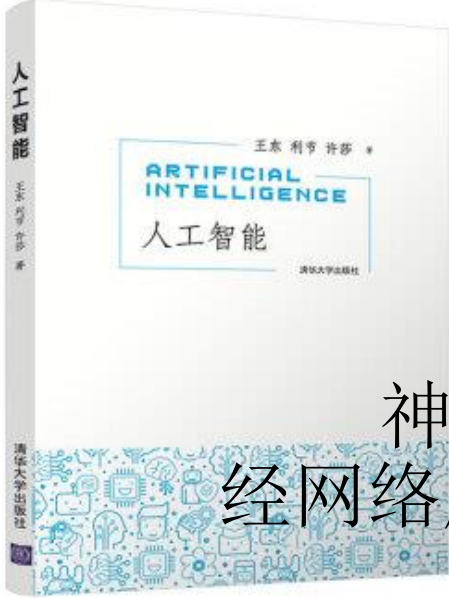


- 生物神经网络

- ← 由多个生物神经元以确定方式和拓扑结构相互连接即形成生物神经网络。

- ← 生物神经网络的功能不是单个神经元信息

- ← 处理功能的简单叠加。
神经元之间的突触连接方式和连接强度不同并且具有可塑性，这使神经网络在宏观呈现出千变万化的复杂的信息处理能力。

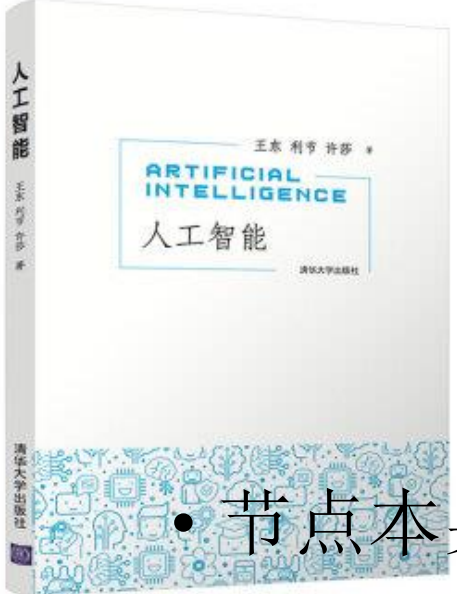


神经元及其突触是神经网络的基本器件。因此，模拟生物神经网络应首先模拟生物神经元 → 人工神经元(节点)

从三个方面进行模拟:

- 节点本身的信息处理能力
- 节点与节点之间连接(拓扑结构)
- 相互连接的强度(通过学习来调整)

决定人工神经网络整体性能的三大要素

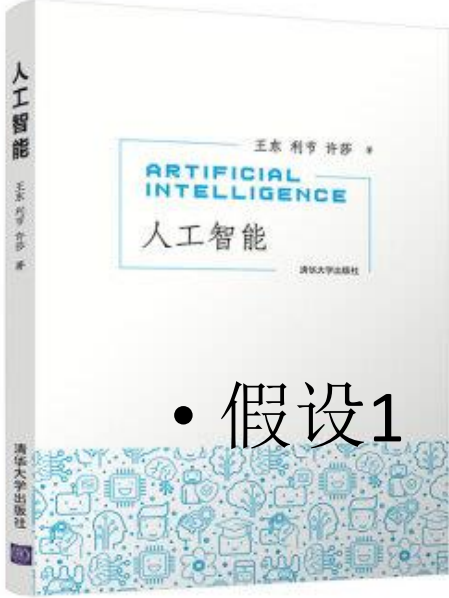


- 节点本身的信息处理能力(数学模型)
- 节点与节点之间连接(拓扑结构)
- 相互连接的强度(通过学习来调整)

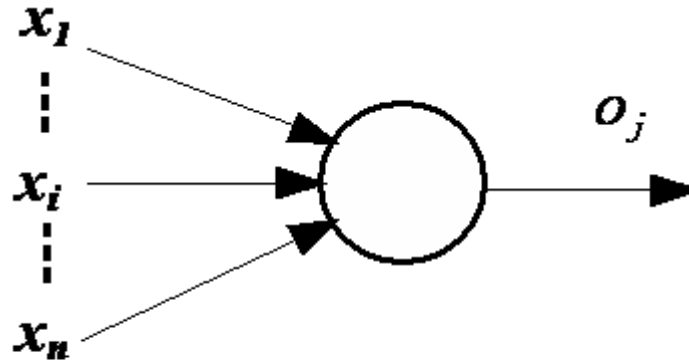


► 模型的六点假设：

- (1) 每个神经元都是一个多输入单输出的信息处理单元；
- (2) 神经元输入分兴奋性输入和抑制性输入两种类型；
- (3) 神经元具有空间整合特性和阈值特性；
- (4) 神经元输入与输出间有固定的时滞，主要取决于突触延搁；
- (5) 忽略时间整合作用和不应期；
- (6) 神经元本身是非时变的，即其突触时延和突触强度均为常数。

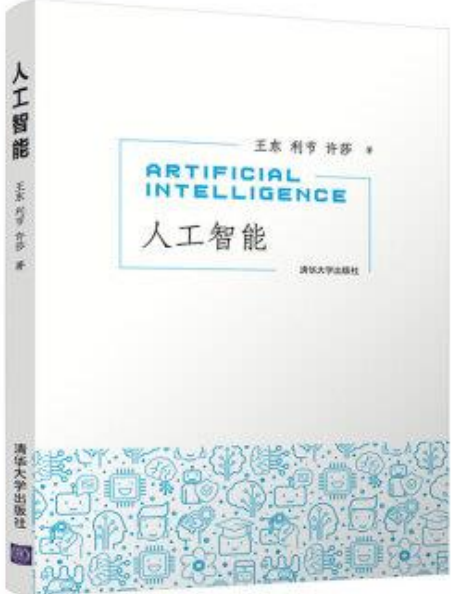


• 假设1

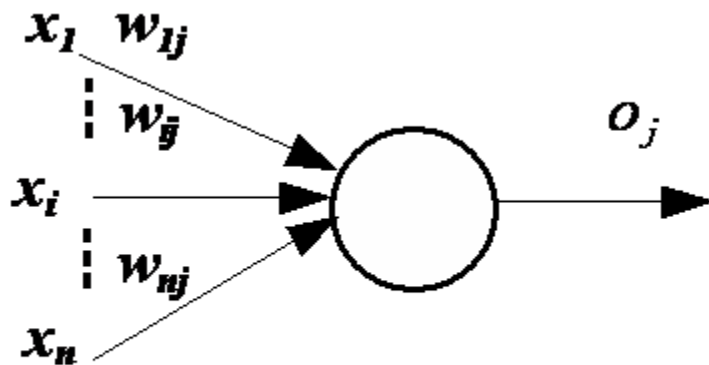


(a)多输入单输出

- 图(a) 表明，正如生物神经元有许多激励输入一样，人工神经元也应该有许多的输入信号，图中每个输入的大小用确定数值 x_i 表示，它们同时输入神经元 j ，神经元的单输出用 o_j 表示。



假设2

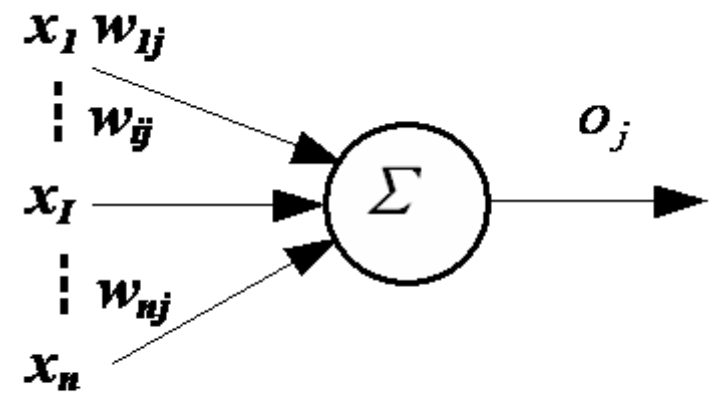


(b)输入加权

- ▶ 生物神经元具有不同的突触性质和突触强度，其对输入的影响是使有些输入在神经元产生脉冲输出过程中所起的作用比另外一些输入更为重要。图(b)中对神经元的每一个输入都有一个加权系数 w_{ij} ，称为权重值，其正负模拟了生物神经元中突触的兴奋和抑制，其大小则代表了突触的不同连接强度。

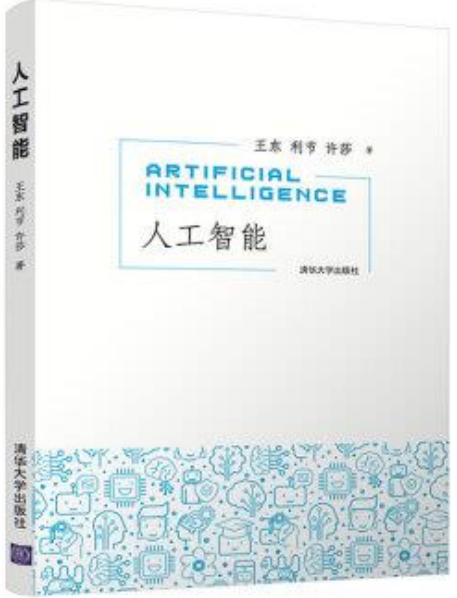


▶ 假设3

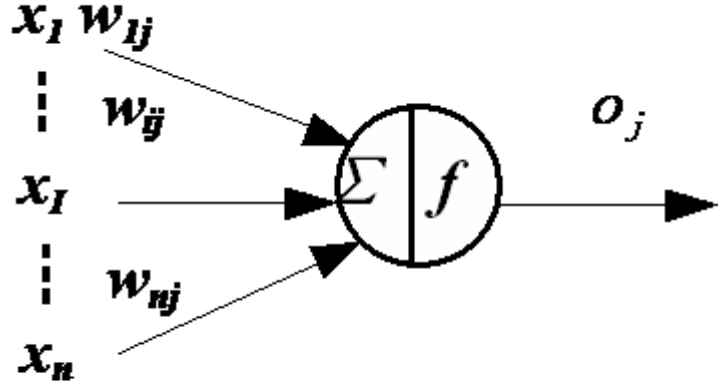


(c)输入加权求和

▶ 作为ANN的基本处理单元，必须对全部输入信号进行整合，以确定各类输入的作用总效果，图(c)表示组合输入信号的“总和值”，相应于生物神经元的膜电位。神经元激活与否取决于某一阈值电平，即只有当其输入总和超过阈值时，神经元才被激活而发放脉冲，否则神经元不会产生输出信号。

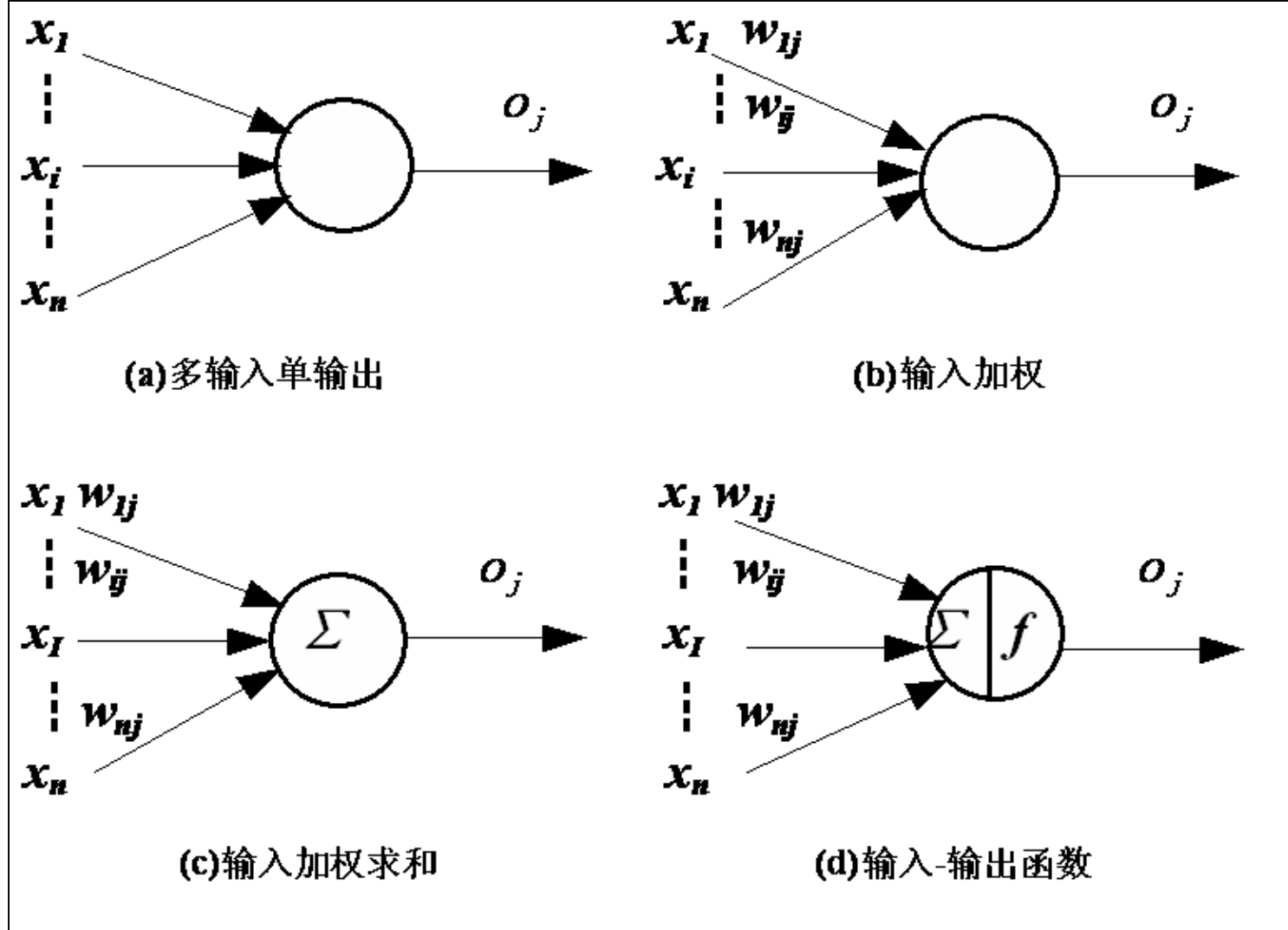
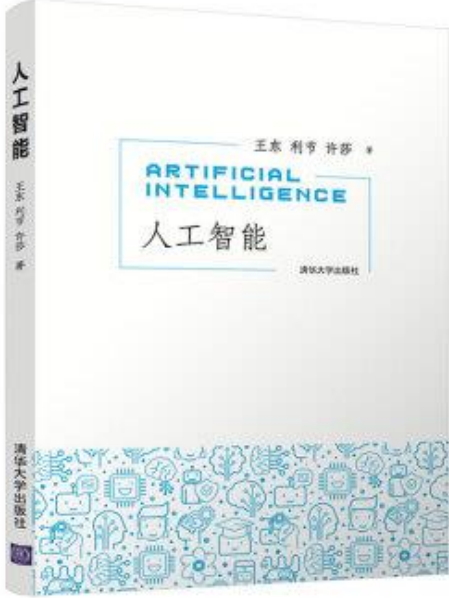


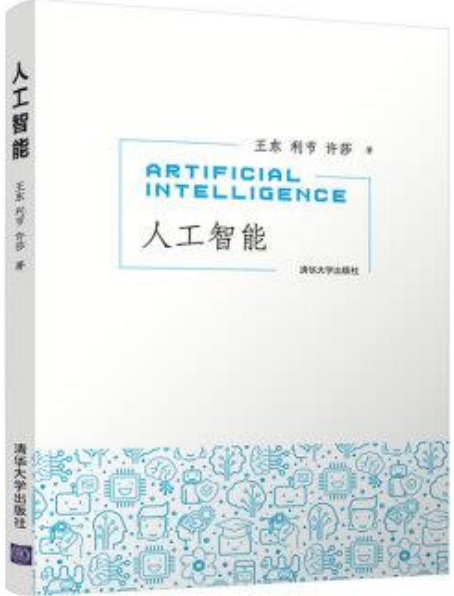
假设4



(d)输入-输出函数

图(d) 人工神经元的输出也同生物神经元一样仅有一个，如用 o_j 表示神经元输出，则输出与输入之间的对应关系可用图(d)中的某种非线性函数来表示，这种函数一般都是非线性的。





上述内容可用一个数学表达式进行抽象与概括。令 $x_i(t)$ 表示 t 时刻神经元 j 接收的来自神经元 i 的信息输入， $o_j(t)$ 表示 t 时刻神经元 j 的信息输出，则神经元 j 的状态可表达为1式。

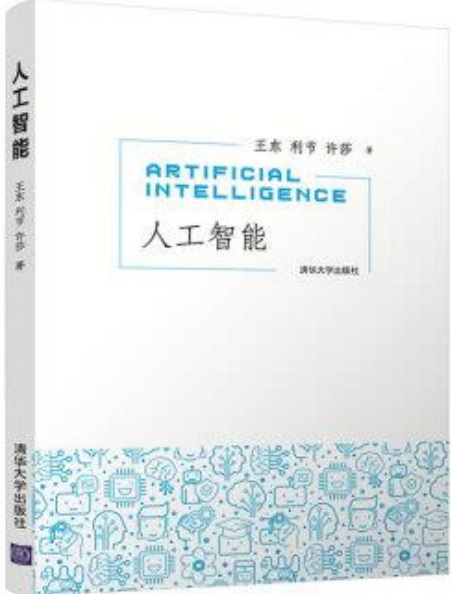
$$o_j(t) = f\left\{\left[\sum_{i=1}^n w_{ij} x_i(t - \tau_{ij})\right] - T_j\right\} \quad (1)$$

τ_{ij} ——输入输出间的突触时延；

T_j ——神经元 j 的阈值；

w_{ij} ——神经元 i 到 j 的突触连接系数或称 权重值；

$f(\cdot)$ ——神经元转移函数。



神经元的数学模型:

为简单起见, 将1上式中的突触时延取为单位时间, 则式(1)可写为2式。

$$o_j(t+1) = f\left\{\left[\sum_{i=1}^n w_{ij}x_i(t)\right] - T_j\right\} \quad (2)$$

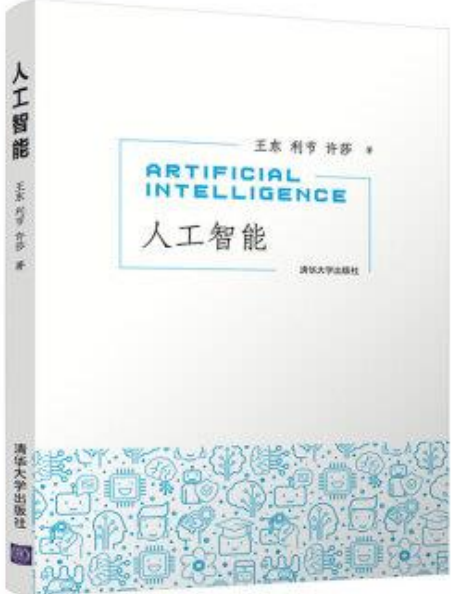
上式描述的神经元数学模型全面表达了神经元模型的6点假定。其中输入 x_i 的下标 $i=1,2,\dots,n$, 输出 o_j 的下标 j 体现了神经元模型假定(1)中的“多输入单输出”。权重值 w_{ij} 的正负体现了假定(2)中“突触的兴奋与抑制”。 T_j 代表假定(3)中神经元的“阈值”; “输入总和”常称为神经元在 t 时刻的净输入,



神经元的数学模型:

$$net'_j(t) = \sum_{i=1}^n w_{ij} x_i(t) \quad (3)$$

$net'_j(t)$ 体现了神经元 j 的空间整合特性而未考虑时间整合，当 $net'_j - T_j > 0$ 时，神经元才能被激活。 $o_j(t+1)$ 与 $x_l(t)$ 之间的单位时差代表所有神经元具有相同的、恒定的工作节律，对应于假定(4)中的“突触延搁”； w_{ij} 与时间无关体现了假定(6)中神经元的“非时变”。



神经元的数学模型:

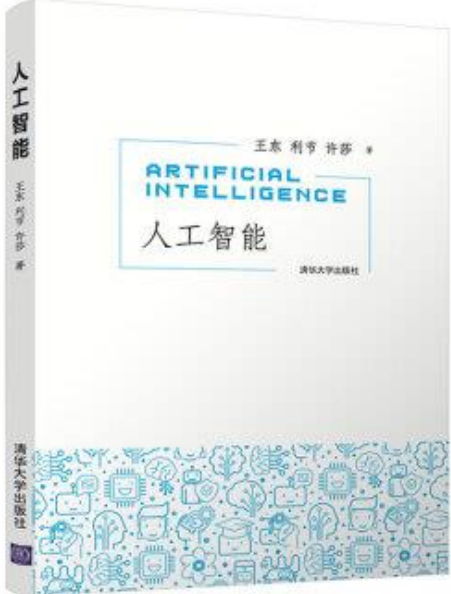
为简便起见,在后面用到式(3)时,常将其中的(t)省略。式(3)还可表示为权重向量 \mathbf{W}_j 和输入向量 \mathbf{X} 的点积 $\mathbf{W}^T\mathbf{X}$ 。其中 \mathbf{W}_j 和 \mathbf{X} 均为列向量,定义为

$$\mathbf{W}_j = (w_1 \ w_2 \ \dots \ w_n)^T, \quad \mathbf{X} = (x_1 \ x_2 \ \dots \ x_n)^T$$

如果令 $x_0 = -1$, $w_0 = T_j$, 则有 $-T_j = x_0 w_0$, 因此净输入与阈值之差可表达为

$$net'_j = \mathbf{W}_j^T \mathbf{X}$$

(4)

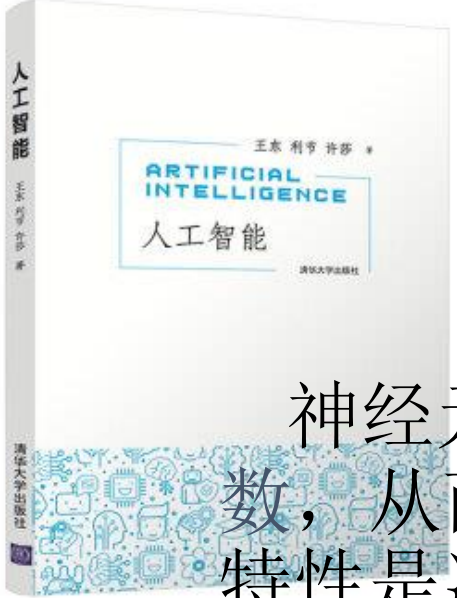


神经元的数学模型:

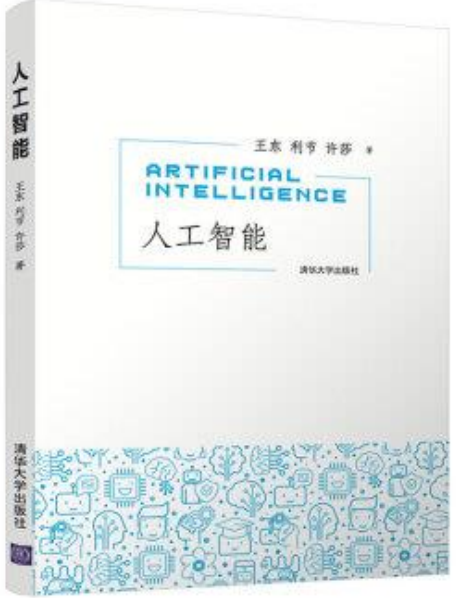
$$net'_j - T_j = net_j = \sum_{i=0}^n w_{ij} x_i = \mathbf{W}_j^T \mathbf{X} \quad (5)$$

显然，式(4)中列向量 \mathbf{W}_j 和 \mathbf{X} 的第一个分量的下标均从1开始，而式(5)中则从0开始。采用式(5)的约定后，净输入改写为 net_j ，与原来的区别是包含了阈值。综合以上各式，神经元模型可简化为

$$o_j = f(net_j) = f(\mathbf{W}_j^T \mathbf{X}) \quad (6)$$



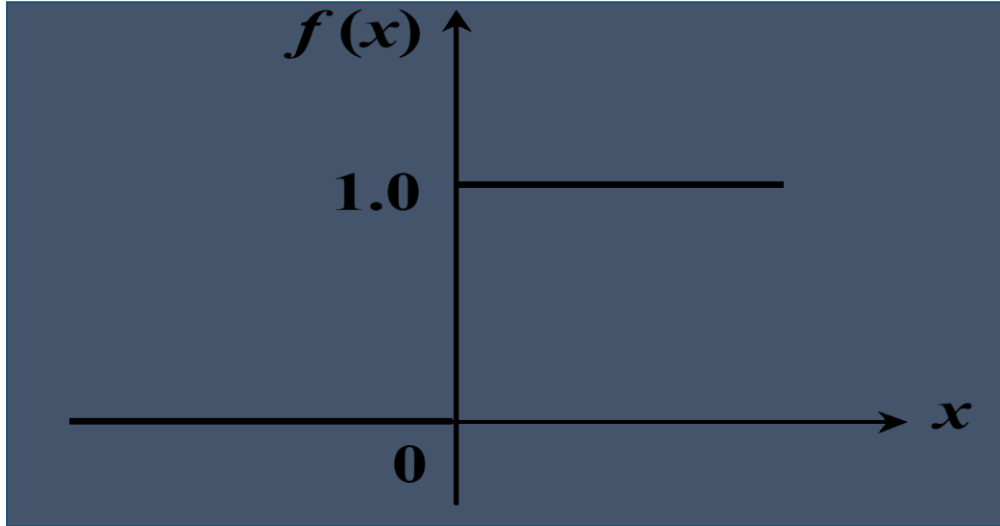
神经元各种不同数学模型的主要区别在于采用了不同的转移函数，从而使神经元具有不同的信息处理特性。神经元的特性是决定人工神经网络整体性能的三大要素之一，反映了神经元输出与其激活状态之间的关系，最常用的转移函数有4种形式。

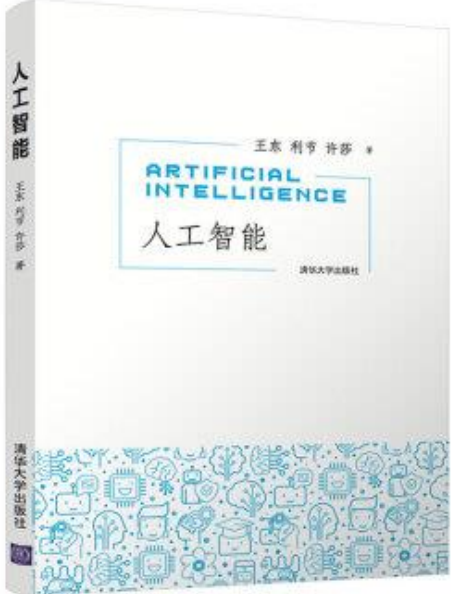


神经元的转移函数:

(1) 阈值型转移函数

$$f(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \quad (7)$$

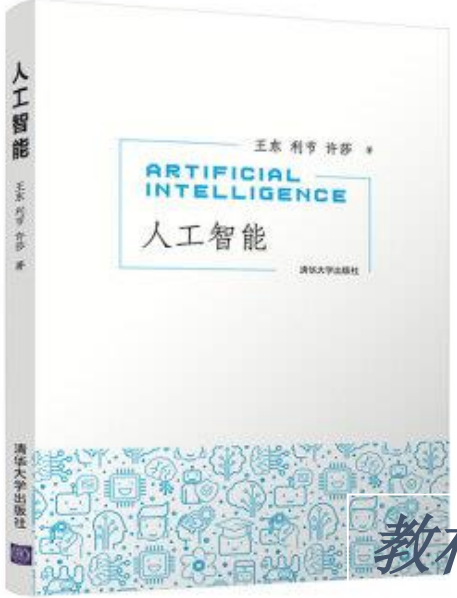




神经元的转移函数：

(2)非线性转移函数

非线性转移函数为实数域 \mathbb{R} 到 $[0,1]$ 闭集的非减连续函数，代表了状态连续型神经元模型。最常用的非线性转移函数是单极性的sigmoid函数曲线，简称S型函数。其特点是函数本身及其导数都是连续的，因而在处理上十分方便。S型函数函数又分为单极性和双极性两种，分别定义如下：

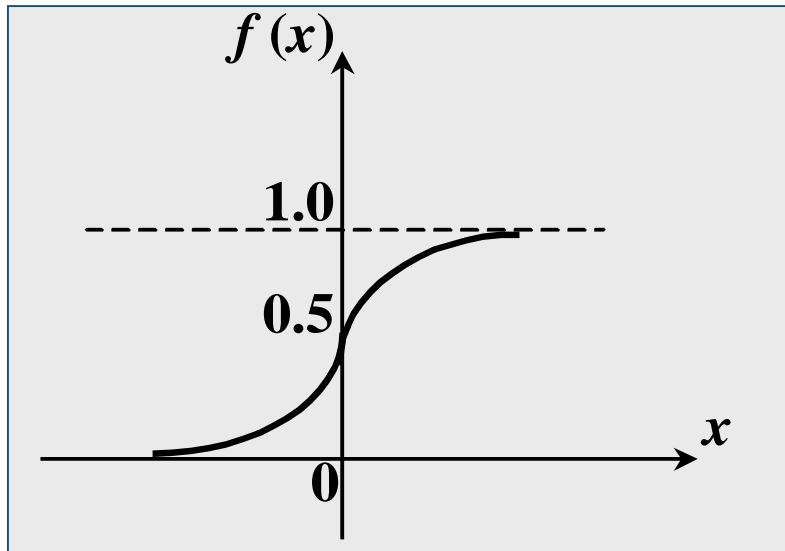


神经元的转移函数：

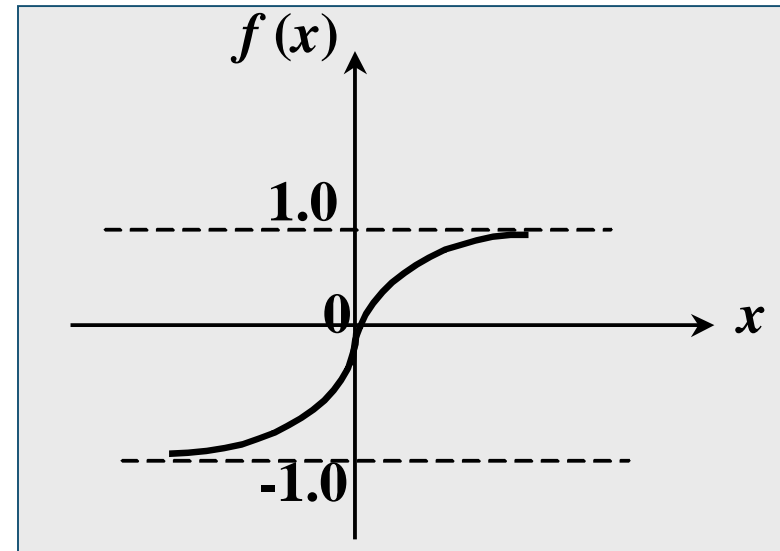
(2)非线性转移函数

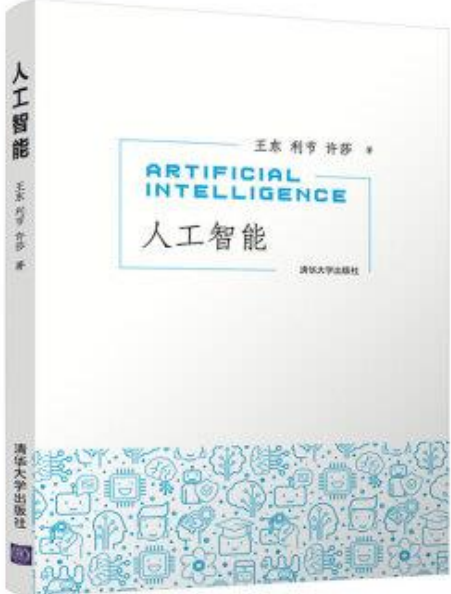
教材选用这个
转移函数

$$f(x) = \frac{1}{1 + e^{-x}}$$



$$f(x) = \frac{2}{1 + e^{-x}} - 1 = \frac{1 - e^{-x}}{1 + e^{-x}}$$

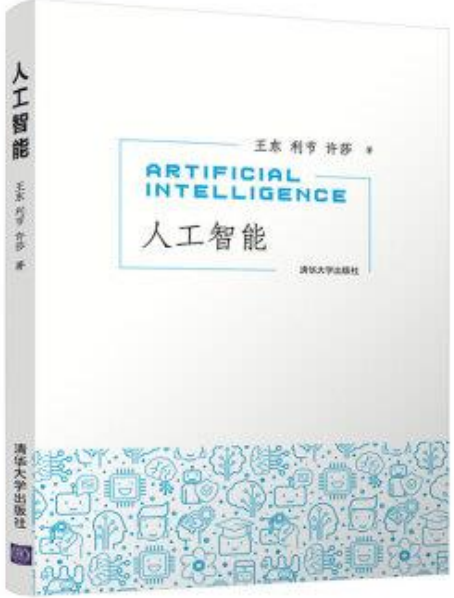




神经元的转移函数：

(3)分段线性转移函数

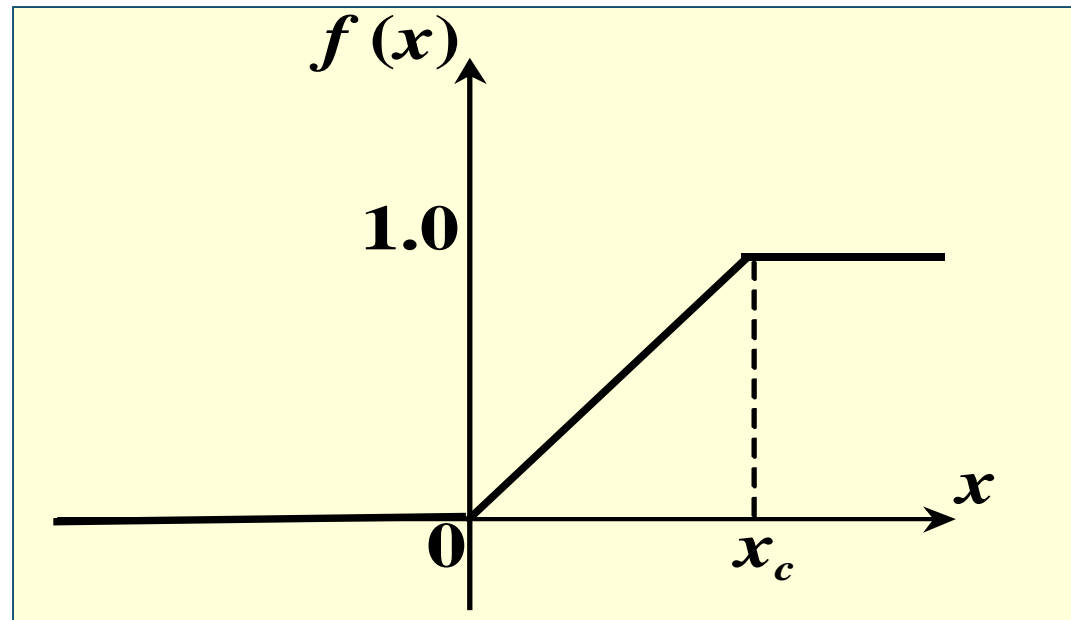
该函数特点是神经元的输入与输出在一定区间内满足线性关系，模拟了实际系统中的饱和特性。由于具有分段线性的特点，因而在实现上比较简单。这类函数也称为伪线性函数，表达式如下：

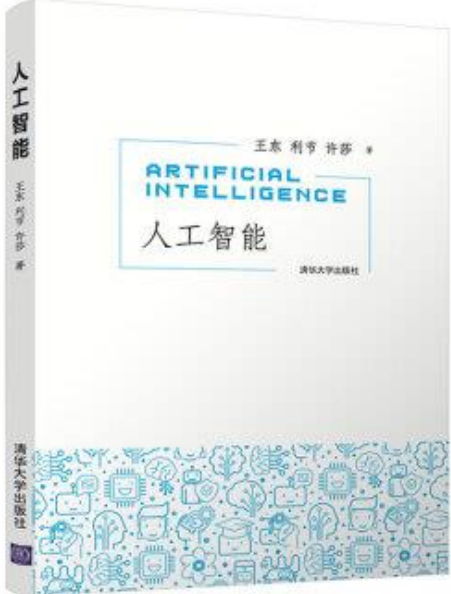


神经元的转移函数：

(3)分段线性转移函数

$$f(x) = \begin{cases} 0 & x \leq 0 \\ cx & 0 < x \leq x_c \\ 1 & x_c < x \end{cases} \quad (9)$$





神经元的转移函数：

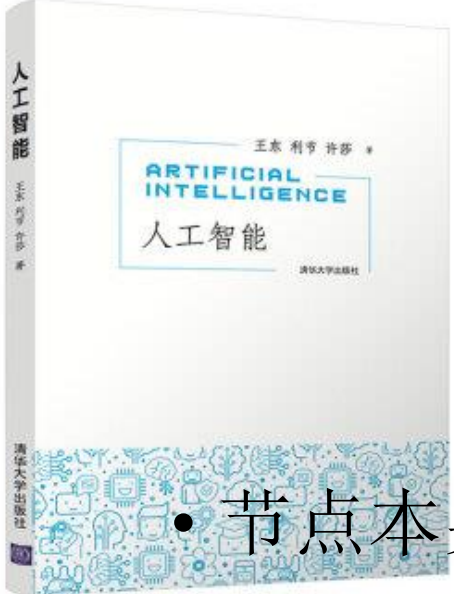
(4) 概率型转移函数

采用概率型转移函数的神经元模型其输入与输出之间的关系是不确定的，需用一个随机函数来描述输出状态为1或为0的概率。设神经元输出为1的概率为

$$P(1) = \frac{1}{1 + e^{-x/T}}$$

温度参数

由于采用该转移函数的神经元输出状态分布与热力学中的玻尔兹曼（Boltzmann）分布相类似，因此这种神经元模型也称为热力学模型。



- 节点本身的信息处理能力(数学模型)
- 节点与节点之间连接(拓扑结构)
- 相互连接的强度(通过学习来调整)



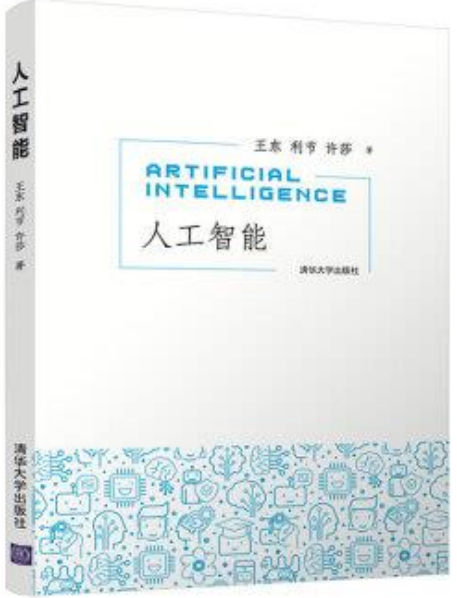
人工神经网络的模型很多，可以按照不同的方法进行分类。其中常见的两种分类方法是，按网络连接的拓扑结构分类和按网络内部的信息流向分类。



- 分类：
 - 按网络连接的拓扑结构分类
 - 层次型结构
 - 互连型网络结构
 - 按网络内部的信息流向分类
 - 前馈型网络
 - 反馈型网络

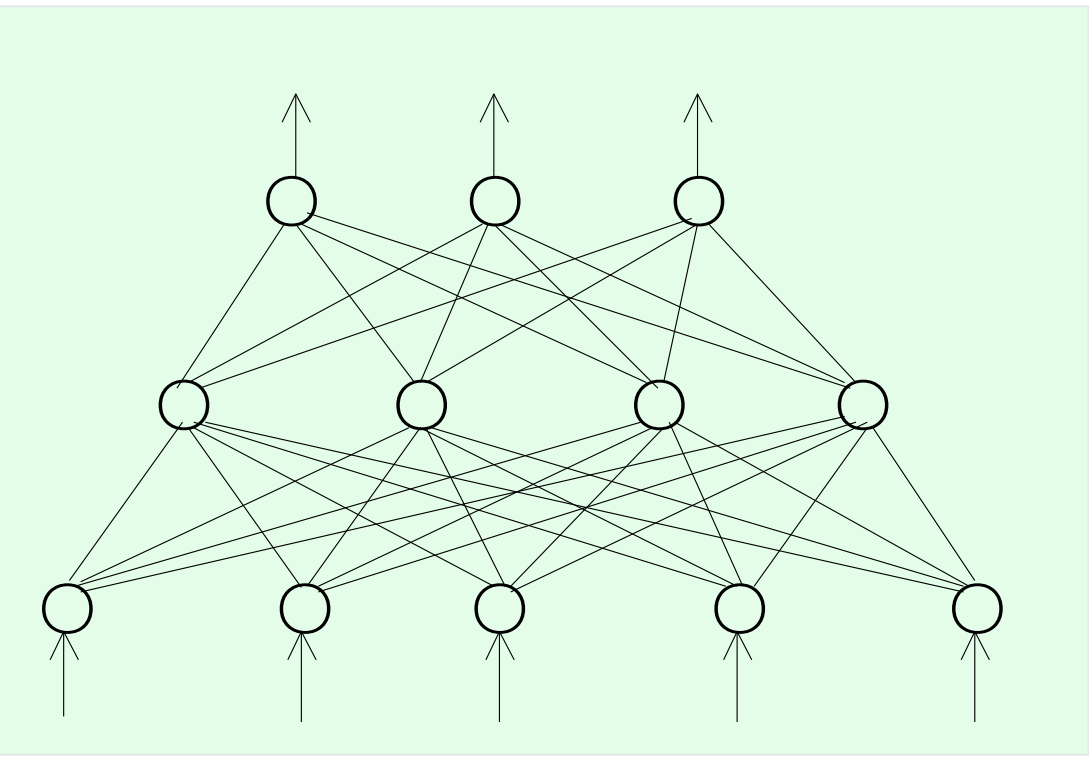


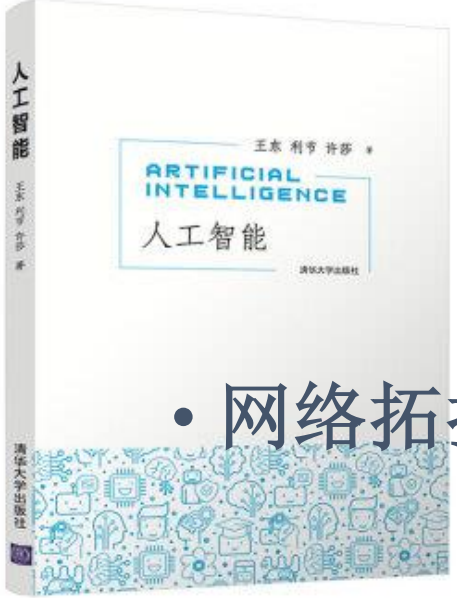
- 层次型结构：将神经元按功能分成若干层，如输入层、中间层（隐层）和输出层，各层**顺序**相连。
- 互连型网络结构：网络中**任意两个节点之间都可能存在连接**路径。



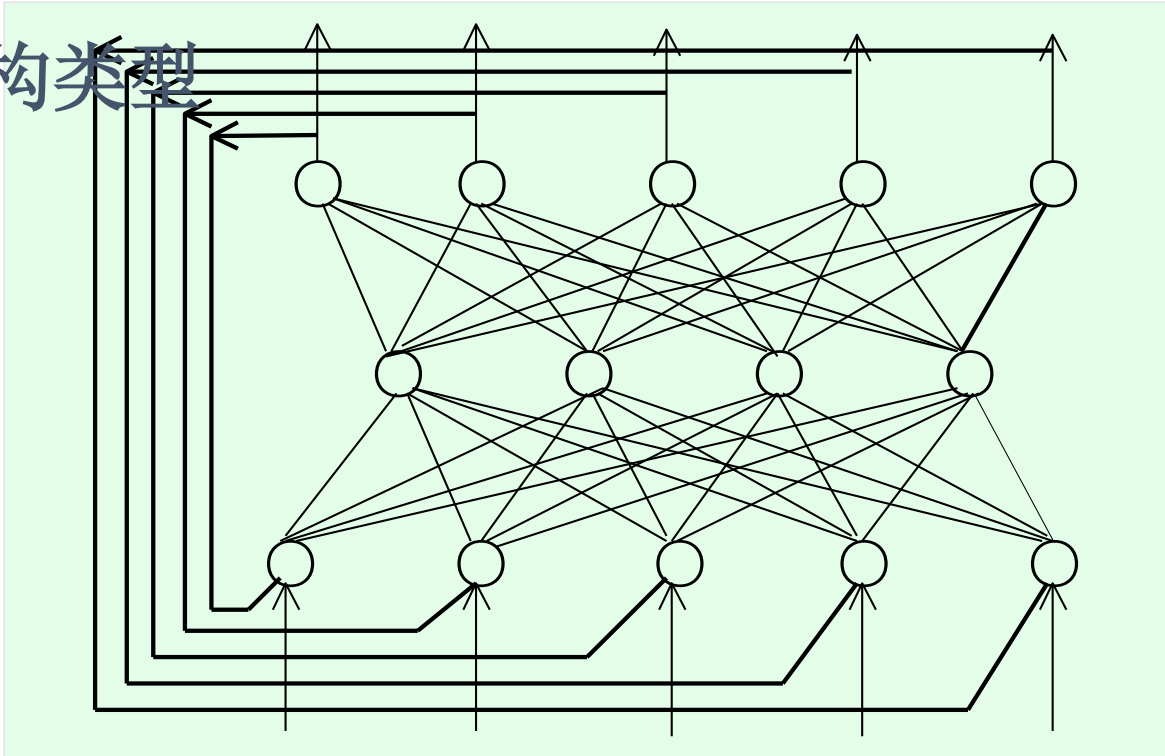
网络拓扑结构类型

层次型结构

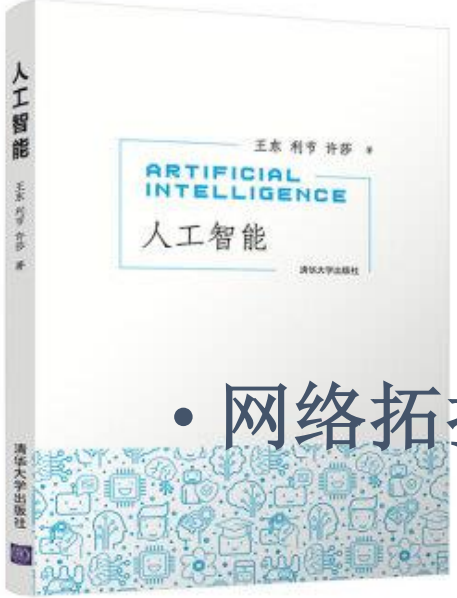




- 网络拓扑结构类型

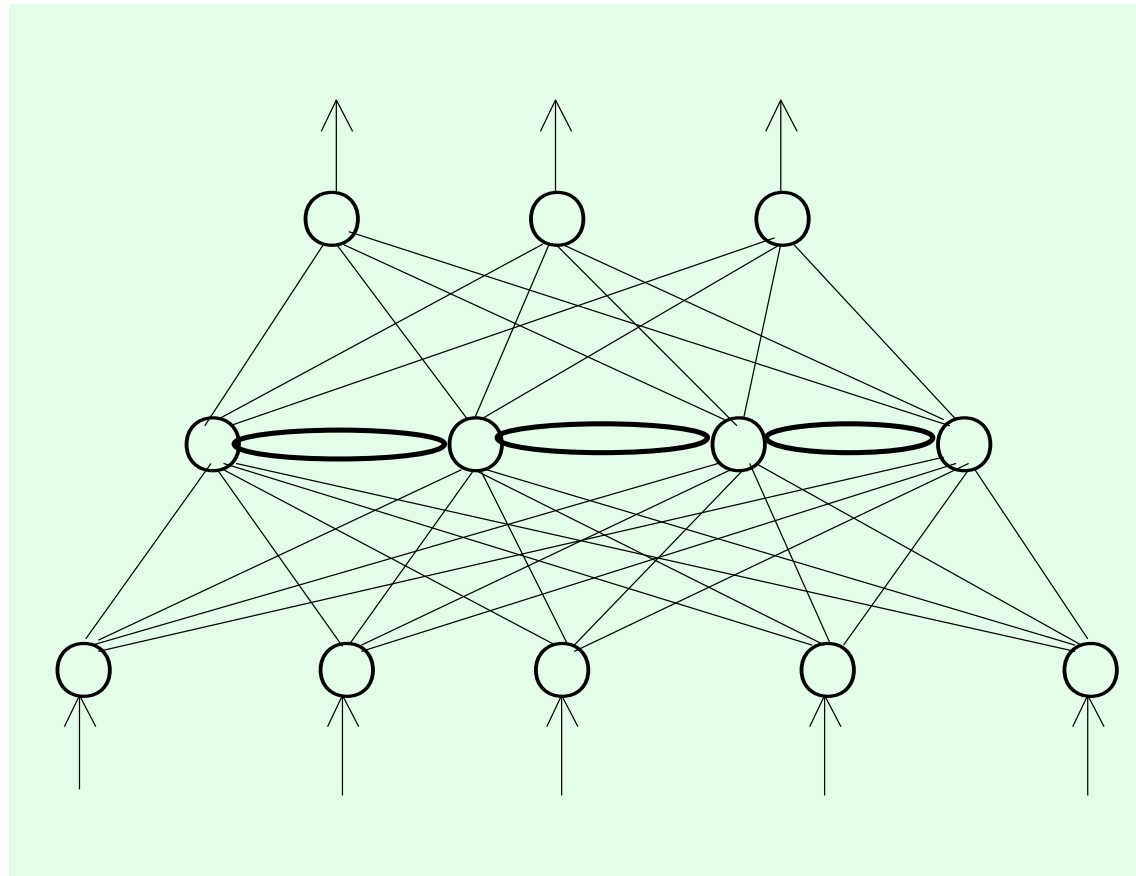


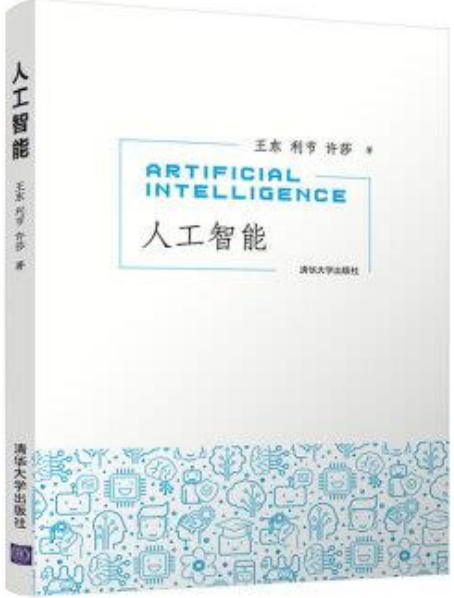
输出层到输入层有连接



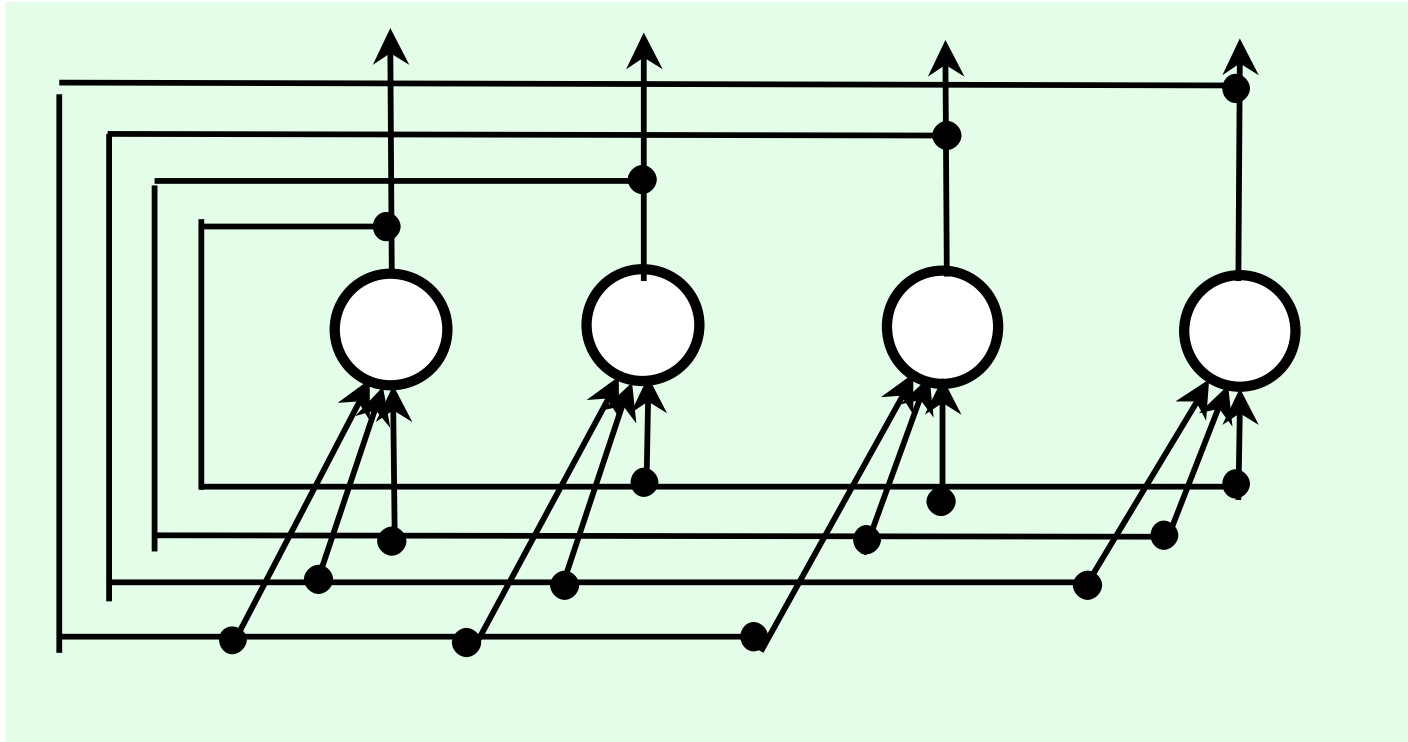
• 网络拓扑结构类型

层状有连接层次型结构

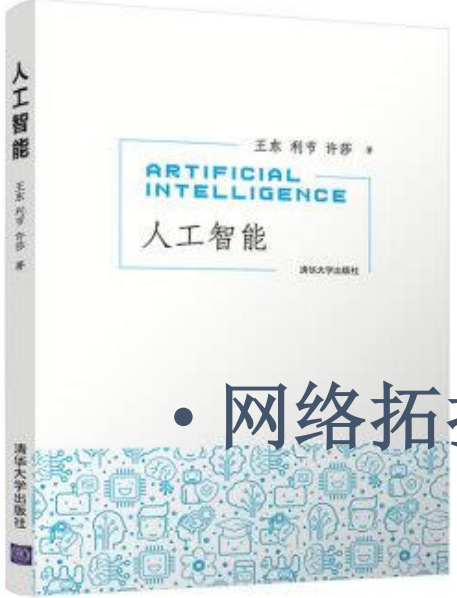




• 网络拓扑结构类型

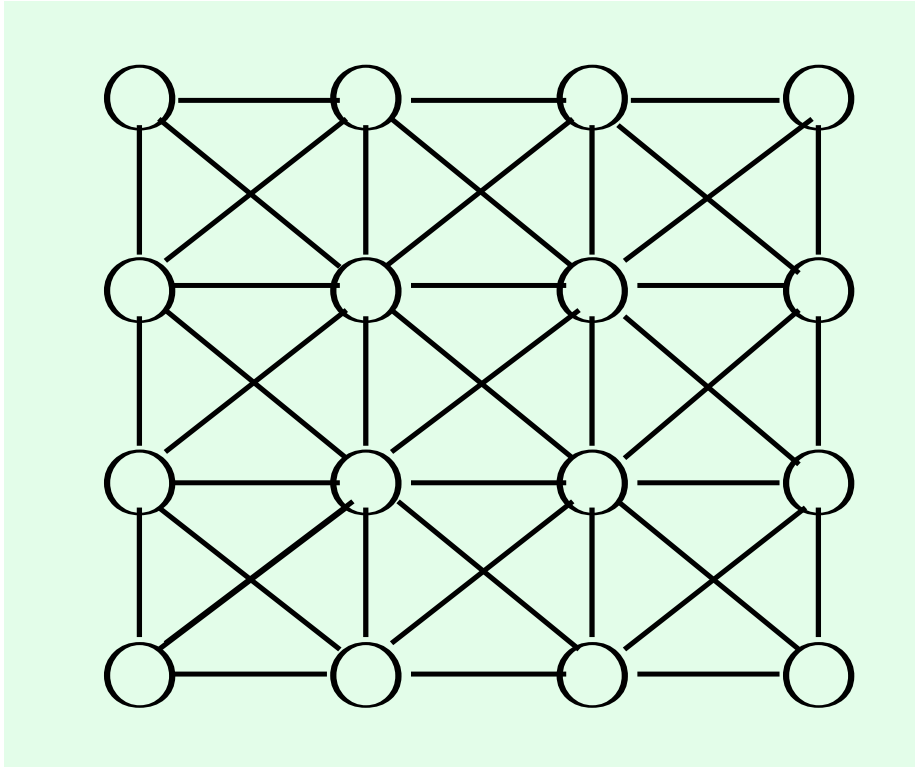


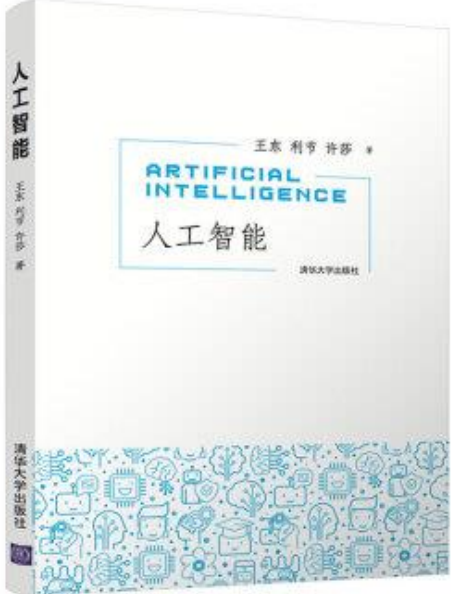
全互连型结构



• 网络拓扑结构类型

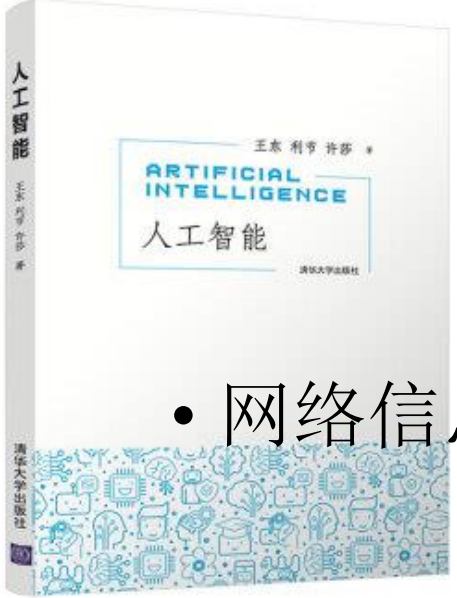
局部互连型网络结构



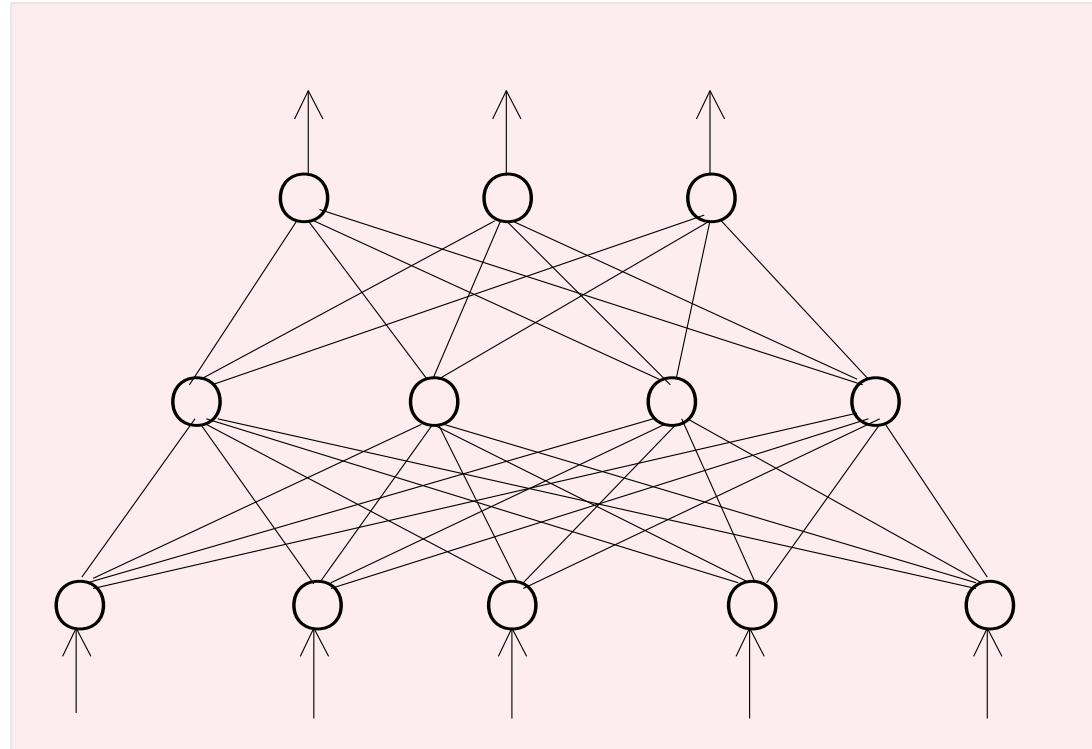


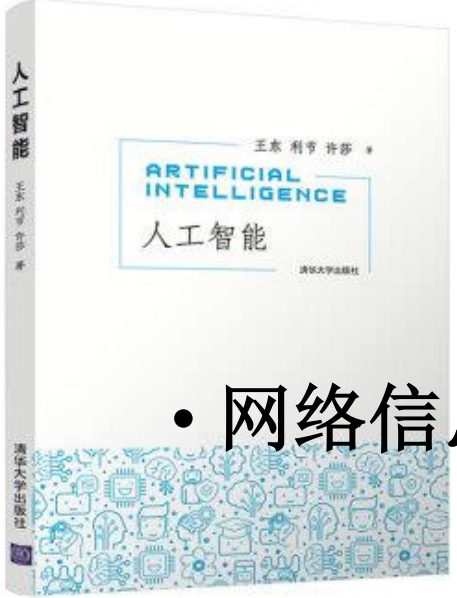
网络信息流向类型

- 前馈型网络
 - 前馈:网络信息处理的方向是从输入层到各隐层再到输出层逐层进行
- 反馈型网络
 - 在反馈网络中所有节点都具有信息处理功能，而且每个节点既可以从外界接收输入，同时又可以向外界输出。

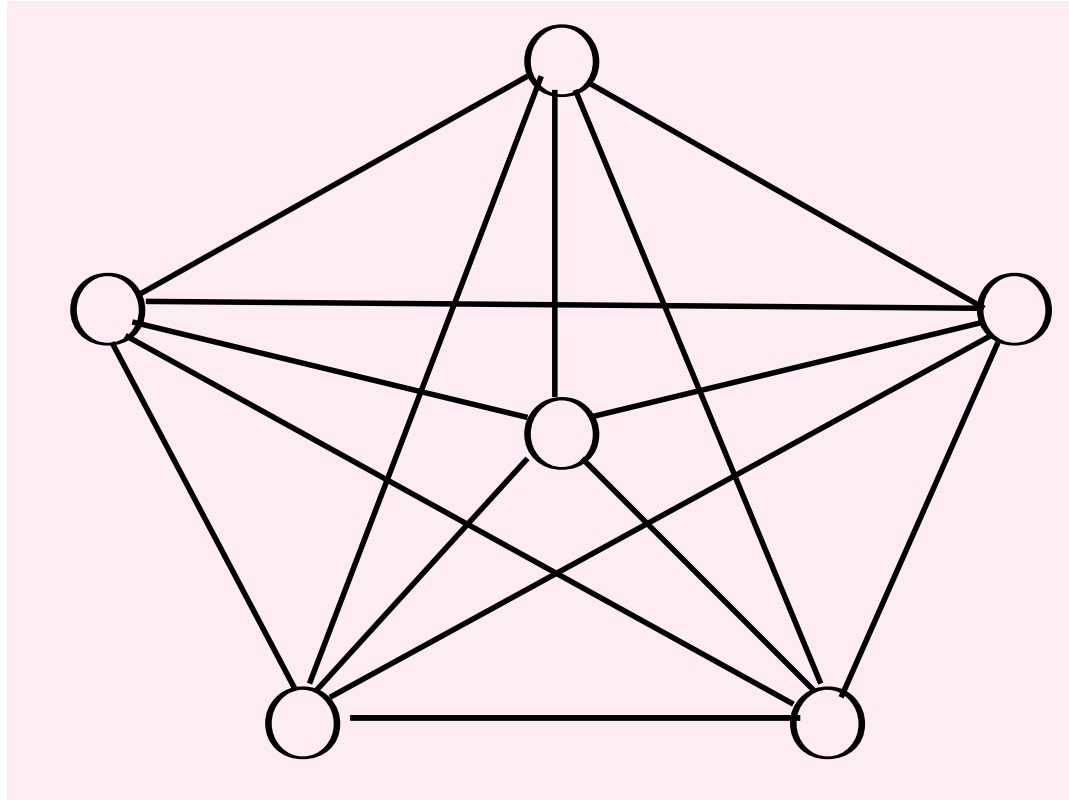


- 网络信息流向类型
前馈型网络



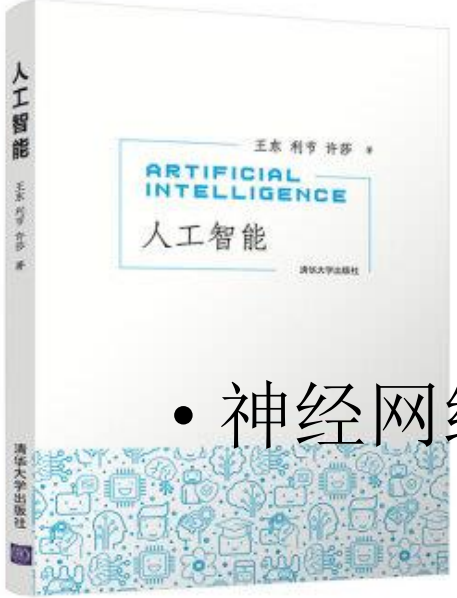


- 网络信息流向类型
反馈型网络



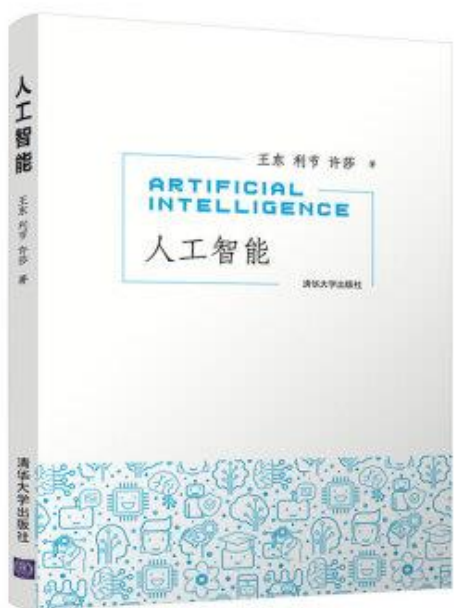


- 节点本身的信息处理能力(数学模型)
- 节点与节点之间连接(拓扑结构)
- 相互连接的强度(通过学习来调整)



- 神经网络学习

神经网络能够通过通过对样本的学习训练，不断改变网络的连接权值以及拓扑结构，以使网络的输出不断地接近期望的输出。这一过程称为神经网络的学习或训练，其本质是可变权值的动态调整。



目录

- 神经网络
- Deep Clustering

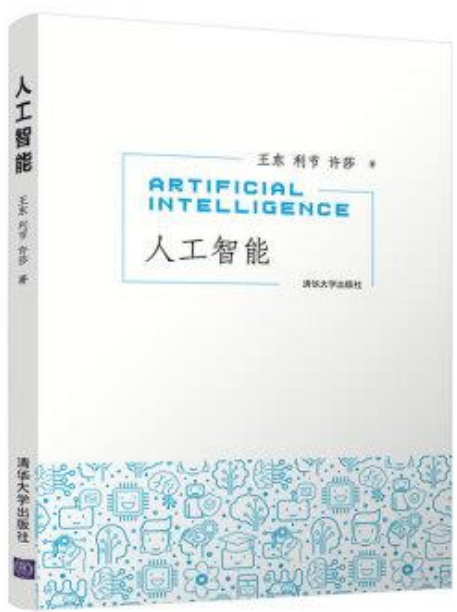


在基于掩蔽的深度学习语音分离框架中，对于说话人相关的任务效果不错，但是对于说话人无关的任务，效果很差，会发生置换问题，即无法确定哪些维度上的信号属于目标声源，哪些维度属于干扰声源。

所以针对这个情况，**Hershey** 等人提出了一种深度聚类**DPCL (Deep Clustering)**方法。**DPCL**没有直面标签置换这个问题，而是另辟思路，尝试将每个**TF-bin**结合它的前后相关信息映射到一个高维的特征上生成谱图嵌入，使得其可以更好的被区分开来。这是一种基于深度神经网络的模拟频谱聚类的方法。



传统的频谱聚类的方法都是基于对归一化的亲和矩阵的本征值分级，到那时这些方法都具有很高的计算代价和浅层学习问题，如果是使用简单的核函数从而产生较为稀疏的亲和矩阵，那么就需要大量计算从而进行后续的频谱聚类。复杂的核函数会使得前段的处理变得困难。而基于深度聚类的方法通过深度神经网络学习频谱的每个时频区域的嵌入向量，使得同一个源信号占主要地位的时频单元之间的距离最小而不同源信号占主要地位的时频单元之间的距离最大，有效的实现了频谱聚类。



The end !