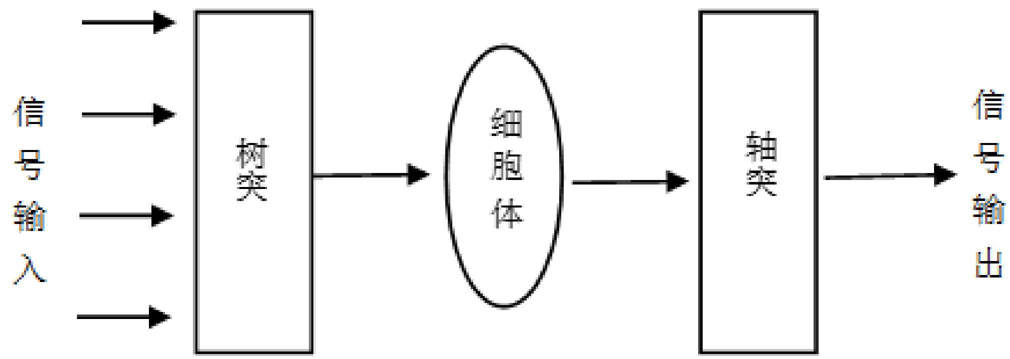
第十二章　　神经网络

人工神经网络（Artificial Neural Networks，简称ANNs）也称为神经网络（NNs）或称为连接模型（Connection Model）。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做出的交互反应。

12.1　人工神经网络概述

生物神经元

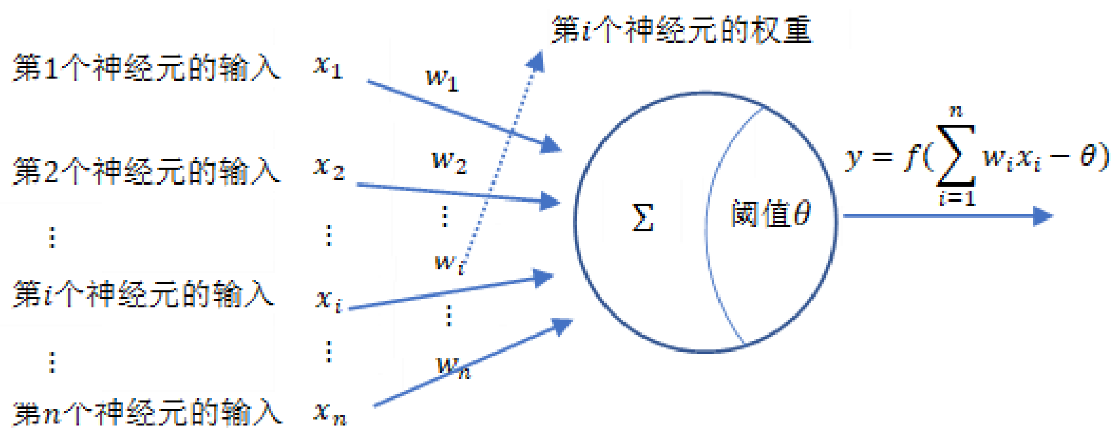
生物神经系统由大量名为神经元的基本单元通过某种方式构成，其工作原理如下图所示：



树突是神经元的分支，负责接收其他神经元的信号。轴突是神经元的传递通道，负责将信号传递给其他神经元。其工作原理维：轴突接收其他神经元的“信号”，当这些“信号”叠加达到一定“阈值”时，会导致神经元的电位发生变化，那么此神经元会“兴奋”起来，把新的“信号”通过轴突传递给其他神经元。

12.1 人工神经网络概述

M-P神经元



M-P神经元“激活”动作通过“激活函数”（activation function）f实现，即输出的信号为

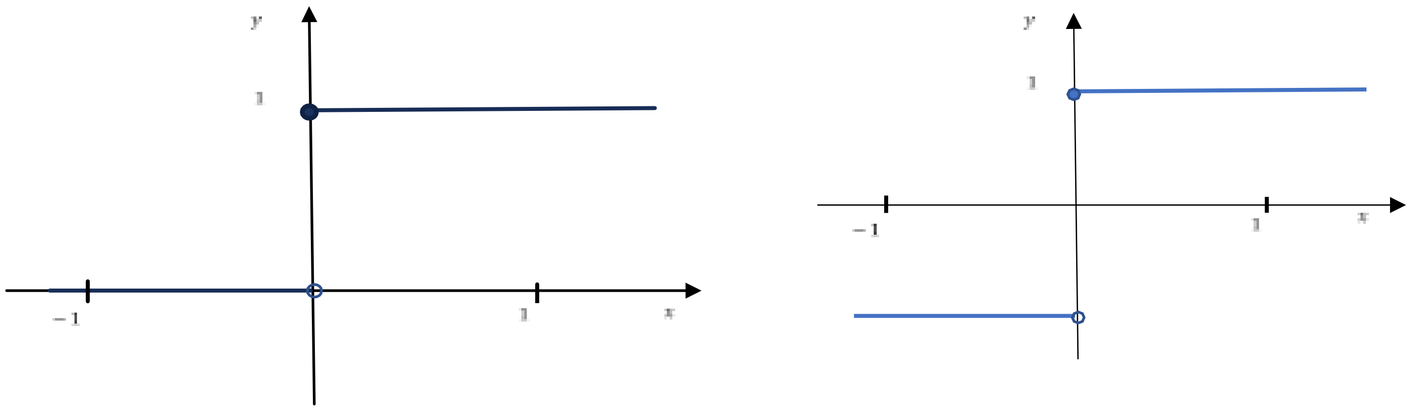
$$y = f(\sum_{i=1}^{n} w_i x_i - \theta)$$

12.1　人工神经网络概述

激活函数

①阶跃函数

$$y=\operatorname{sgn}(x)=\begin{cases}1, & x\geq 0;\\ 0, & x<0,\end{cases}\quad \text{or}\quad y=\operatorname{sgn}(x)=\begin{cases}1, & x\geq 0;\\ -1, & x<0.\end{cases}$$

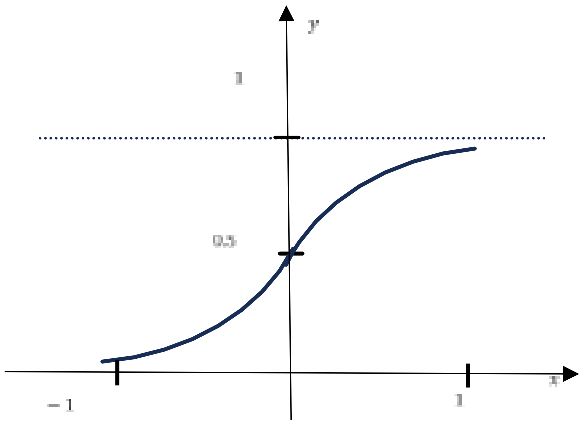


阶跃函数具有不连续，不光滑等性质，对后续模型的求解不利。

12.1　人工神经网络概述

②Sigmoid函数

$$y = \mathrm{Sigmoid}(x) = \frac{1}{1+e^{-x}} \quad \forall x \in R$$

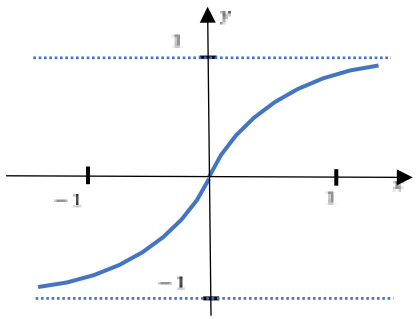


Sigmoid函数连续、可微，并且满足$y' = y(1-y)$，这些性质对于后续模型的求解非常有利。

12.1 人工神经网络概述

Sigmoid函数变型

$$y = 2\text{Sigmoid}(x) - 1 = \frac{1 - e^{-x}}{1 + e^{-x}} \quad \forall x \in R$$



此变型将Sigmoid函数值域扩展到(−1,1)上，保持连续性和可微性，并且其导数满足$y' = \frac{1-y^2}{2}$，这些性质对于后续模型的求解都是非常有利的。

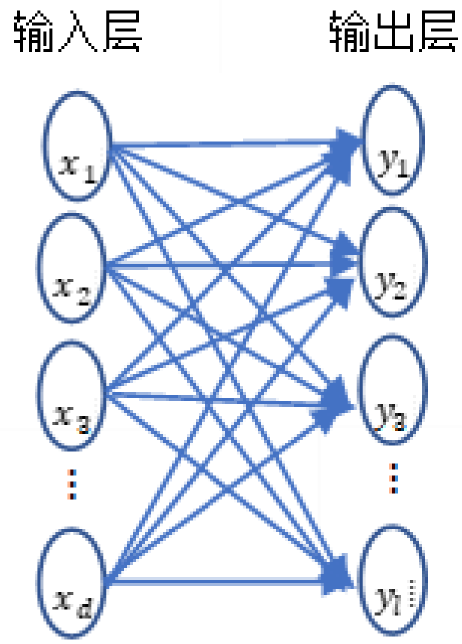
12.1　人工神经网络概述

人工神经网络

人工神经网络由相互连接的M-P神经元（也称为节点或者处理单元）构成。生物神经元的连接和连接的强弱，在人工神经网络中以节点间的连线以及连接权重来表示。根据网络的层数可分为两层神经网络、三层及以上的神经网络或多层神经网络。

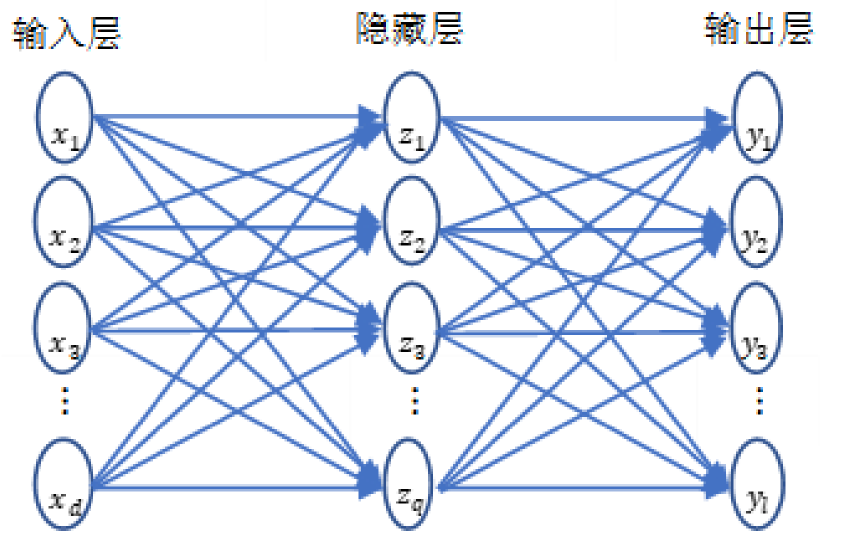
12.1　人工神经网络概述

如右图所示。图中椭圆表示节点，有向线段表示节点之间的连接。人工神经网络一般分为输入层、隐藏层、输出层，隐藏层可以有也可以没有，也可以有多层。如右图，只有输入层和输出层，没有隐藏层，即为最简单的神经网络结构，感知机（Perception）模型。



感知机能力有限，需要通过很多神经元协作完成复杂的功能。通过一定的链接方式或信息传递方式进行协作的多个神经元可以看作一个神经网络，称为人工神经网络，也简称为神经网络。到目前为止，研究者已经发明了各种各样的神经网络结构。本章主要介绍“多层前馈神经网络”（Multi-Layer Feedforward Neural Networks），也可称为多层感知机（Multiple Layers Perception，简记为MLP）。

12.1　人工神经网络概述



如左图所示，其中输入层负责接收外界信号输入，隐藏层和输出层负责对信号进行处理，最后由输出层输出，其中隐藏层可以是单层也可以是多层，分别称为“单隐层前馈网络”和“多隐层前馈网络”。
多层前馈神经网络能够解决复杂的分类和回归问题。

12.2　感知机

感知机是一种最基本的前馈式神经网络模型，仅由输入层和输出层构成。

感知机模型原理

以二分类为例，其输入为样本的属性向量，输出为样本的类别，取值为+1和−1。

设样本为$(x,y)\in R^{n+1}$，其中$x\in R^n$为样本的属性向量，$y\in\{+1,-1\}$为样本的标记。设$w=(w_1,w_2,\cdots,w_n)^T$为权重向量，激活函数为值域是$\{+1,-1\}$的阶跃函数。记$b=-\theta$，那么此时的模型为

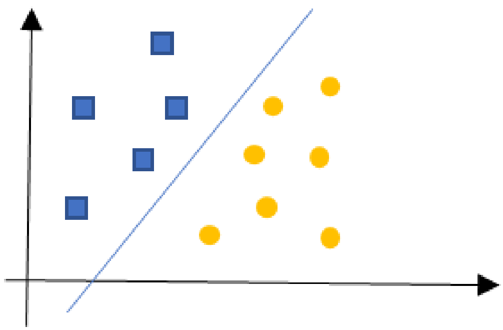
$$y=\mathrm{sgn}(w^Tx+b),$$

可以看到感知机模型是一种线性模型，其假设空间为R^n中所有的线性分类器，即集合$\{f\,|\,f(x)=w^Tx+b\}$。

12.2 感知机

感知机模型的几何解释

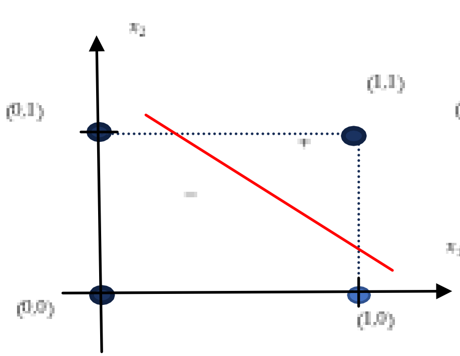
线性函数$w^T x + b = 0$为空间R^n中的超平面，它将空间划分为两个部分。其中，w为法向量，b为截距。对于二分类问题，不同的类别位于超平面的两边。

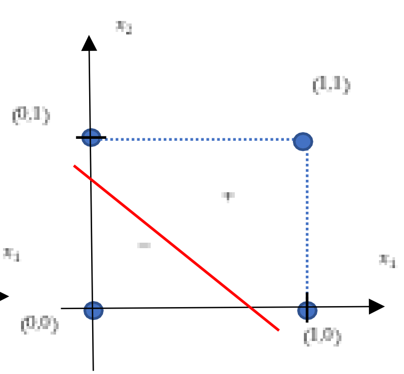


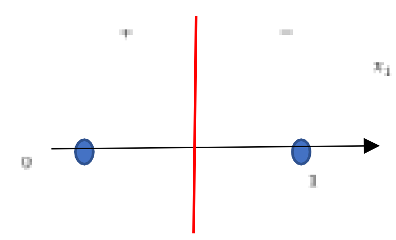
感知机模型的适用性

（1）对于线性可分问题

感知机为线性模型，可以解决线性可分问题，例如逻辑与、或、非运算

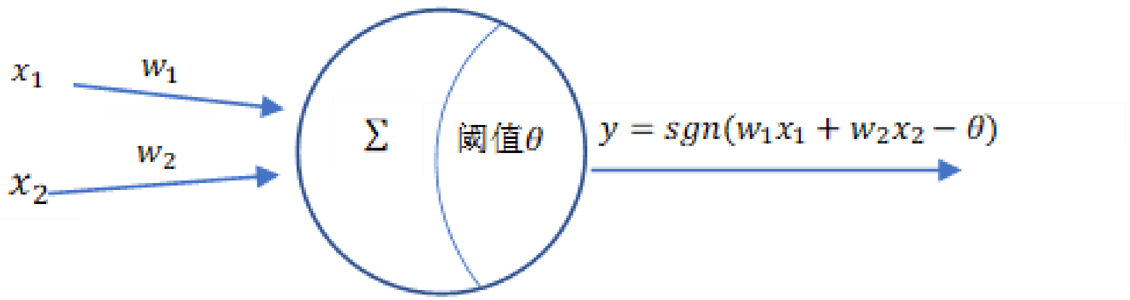






12.2　感知机

它们都可以通过只有两个输入节点的简单感知机模型实现：



"与"：设$w_1 = w_2 = 1$，$\theta = 2$，那么$y = \mathrm{sgn}(x_1 + x_2 - 2)$，则只有$x_1 = x_2 = 1$时，$y = 1$；

"或"：设$w_1 = w_2 = 1$，$\theta = 1$，那么$y = \mathrm{sgn}(x_1 + x_2 - 1)$，则$x_1 = 1$或$x_2 = 1$时，$y = 1$；

"非"：设$w_1 = -1$，$w_2 = 0$，$\theta = -0.5$，那么$y = \mathrm{sgn}(-x_1 + 0.5)$，则$x_1 = 0$时，$y = 1$，$x_1 = 1$时，$y = -1$。

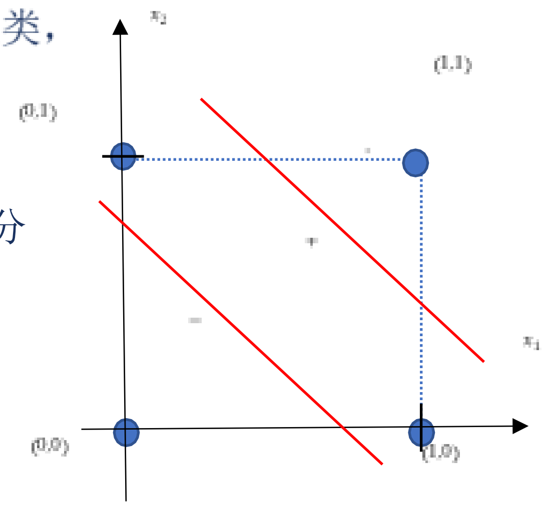
12.2　感知机

（2）对于非线性可分问题

感知机原理简单容易理解，但能力有限，不能解决非线性问题。比如逻辑异或。

“异或”：当x_1或x_2为1，且$x_1 \neq x_2$时，表示正类，否则为负类。

此问题非线性可分，不存在直线可以将正负类分开，需要多层神经网络解决。



12.2　感知机

感知机的学习策略

神经网络的学习，就是根据训练数据来调整神经元之间的**权重和阈值**。感知机的学习目标就是根据训练集，去寻找一个能够获奖正类和负类分开的**超平面**，这个超平面由权重向量w和阈值b（即$-\theta$）确定。感知机将其转化为一个优化问题，通过求解优化问题得到权重w和阈值b，进而得到分类超平面。

优化目标便是分类错误的样本“越少越好”。为了具体化，感知机采用误分类点到超平面的距离 $\frac{1}{\|w\|}|w^T x + b|$ 作为目标函数。其中$\|w\|$表示法向量的大小，一般设置为1，并且误分类点满足$y(w^T x + b) \leq 0$

设训练数据集为$D = [(x_1, y_1), (x_2, y_2)\cdots, (x_m, y_m)]$，集合$M$为误分类的样本点构成的集合。那么感知机的目标函数可以定义为：

$$\min_{w,b} L(w,b) := -\sum_{x_i \in M} y_i(w^T x_i + b)$$

12.2　感知机

1）我们可以通过随机梯度下降法极小化目标函数

对L关于参数w和b分别求偏导可得

$$\nabla_w L(w,b) = -\sum_{x_i \in M} y_i x_i，\ \nabla_b L(w,b) = -\sum_{x_i \in M} y_i。$$

设其中任意一个误分类点为(x_i, y_i)，则w和b按如下方式更新：

$$w \leftarrow w + \eta y_i x_i，\ b \leftarrow b + \eta y_i$$

其中$\eta(0 < \eta \leq 1)$为学习率，也称为步长。

12.2 感知机

2）我们可以通过对偶法极小化目标函数

由上面的讨论我们可以发现w和b分别为x_i和标记y_i的线性组合，即

$$w=\sum_{i=1}^{m}\alpha_i y_i x_i,\quad b=\sum_{i=1}^{m}\alpha_i y_i$$

其中，$\alpha_i \geq 0(i=1,2,\cdots,m)$，记$\alpha=(\alpha_1,\alpha_2,\cdots,\alpha_m)$，称为系数向量。

此时分离超平面为 $\sum_{j=1}^{m}\alpha_j y_j x_j^T x+\sum_{j=1}^{m}\alpha_j y_j=0$

误分类点的判定条件可以改写为：$y(\sum_{j=1}^{m}\alpha_j y_j x_j^T x+\sum_{j=1}^{m}\alpha_j y_j)\leq 0$

因此，我们也可以通过其系数$\alpha_i \geq 0(i=1,2,\cdots,N)$的更新达到对其更新的目的。

此外，误分点的判定条件中，样本的属性向量x均以内积的形式出现，因此在具体实现算法时，可以预先将训练样本的内积计算出来，以矩阵的形式存储，称为Gram矩阵（Gram matrix），记为：

$$G:=[x_i^T x_j]_{m\times m}。$$

12.2　感知机

感知机随机梯度下降法步骤

设训练集$D=\{(x_1,y_1),(x_2,y_2),\cdots,(x_m,y_m)\}$，其中$x_i\in R^n$，$y_i\in\{-1,1\}$，$i=1,2,\cdots,m$。学习率为$\eta(0<\eta\le 1)$。

1）任意选取初始w_0和b_0（一般选0作为初值）；
2）随机选取样本点(x_i,y_i)；
3）如果$y_i(w^Tx_i+b)\le 0$，更新w和b；
4）否则转至2），直至训练集中没有误分类点。
5）输出w和b，得到感知机$y=\mathrm{sgn}(w^Tx+b)$。

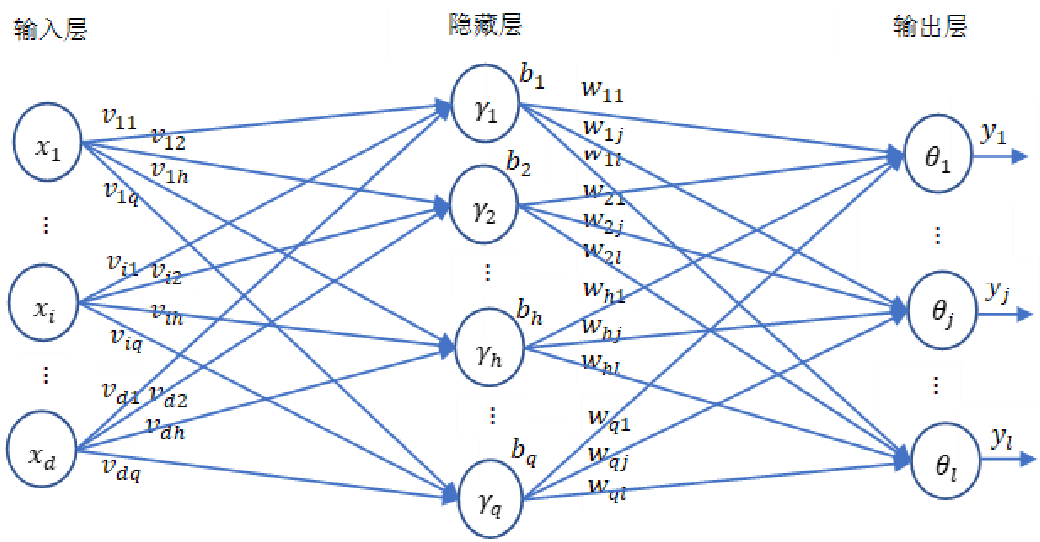
12.2　感知机

感知机对偶算法步骤

1）赋初值$\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_m) = 0$；
2）计算Gram矩阵；
3）选取样本点(x_i, y_i)
4）若不等式$y_i(\sum_{j=1}^{m} \alpha_j y_j x_j^T x_i + \sum_{j=1}^{m} \alpha_j y_j) \leq 0$成立，则更新$\alpha_i \leftarrow \alpha_i + \eta$；
5）否则转至3），直至训练集中没有误分类点；
6）输出α，得到感知机$y = \mathrm{sgn}(\sum_{j=1}^{m} \alpha_j y_j x_j^T x + \sum_{j=1}^{m} \alpha_j y_j)$。

12.3　多层前馈神经网络

基本结构



其中$x_i(i=1,2,\cdots,d)$表示输入层第i个神经元的输入信号，$b_h(h=1,2,\cdots,q)$表示隐藏层第h个神经元的输出信号，$y_j(j=1,2,\cdots,l)$表示输出层第j个神经元的输出信号，即整个网络的输出。隐藏层和输出层的激活函数记为f，这里采用Sigmoid函数。