

感知机与多层网络

前言

上一篇文章[神经元模型](#)中，我们已经介绍了关于神经元的基本信息，其中我们知道“权重”和“阈值”是神经元的根本，还有一个用于是否传递给下一个神经元的函数判断——“激活函数”。这一讲我们来深入，说说在这套理论之上的感知机。

这部分内容将会参考西瓜书（大面积的摘抄、个人吐槽与详细注解/图解）[（西瓜书真的太好看了！！！）](#)

感知机与多层神经网络

感知机(Perceptron)由[两层神经元组成](#)，如下图所示，输入层接收外界输入信号后传递给[输出层](#)，[输出层是 M-P 神经元，亦称“值逻辑单元”\(threshold logic unit\)](#)。

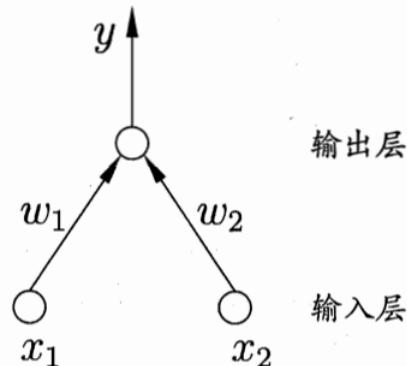


图 5.3 两个输入神经元的感知机网络结构示意图

感知机能通过激活函数、权重与阈值很容易地实现与、或、非运算，也就是 $y = f(\sum_i w_i x_i - \theta)$ ，假定 $f(x)$ 为阶跃函数，则有：

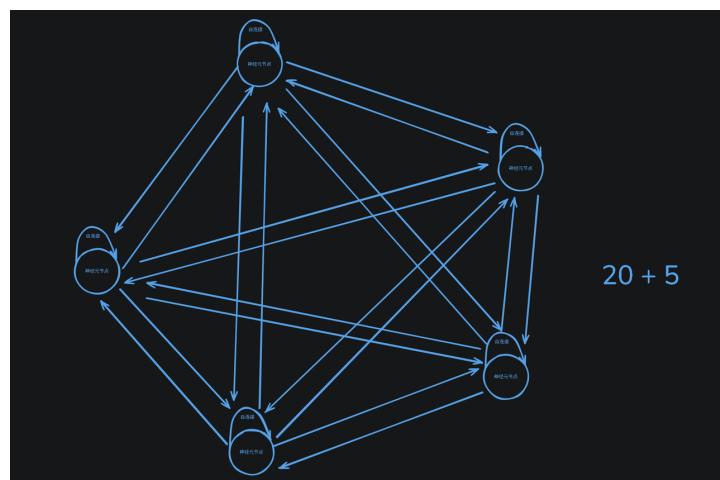
- 与 $(x_1 \wedge x_2)$ ：令 $w_1 = w_2 = 1, \theta = 2$ ，则 $y = f(1 \cdot x_1 + 1 \cdot x_2 - 2)$ ，仅在 $x_1 = x_2 = 1$ 时， $y = 1$
- 或 $(x_1 \vee x_2)$ ：令 $w_1 = w_2 = 1, \theta = 0.5$ ，则 $y = f(1 \cdot x_1 + 1 \cdot x_2 - 0.5)$ ，当 $x_1 = 1$ 或 $x_2 = 1$ 时， $y = 1$
- 非 $(\neg x_1)$ ：令 $w_1 = 0.6, w_2 = 0, \theta = -0.5$ ，则 $y = f(-0.6 \cdot x_1 + 0 \cdot x_2 + 0.5)$ ，当 $x_1 = 1$ 时， $y = 0$ ；当 $x_1 = 0$ 时， $y = 1$

当然这不是重点不在与或非，而是节点当中的“阈值”。我在[神经元模型](#)提到了10个节点，其中有一个问题没有解决，就是：[什么是哑](#)

节点？ 一般地，给定训练数据集，权重 $w_i (i = 1, 2, 3, \dots, n)$ 以及阈值 θ 可以通过学习得到。阈值 θ 可以看作一个固定输入为 -1.0 的“哑节点”（dummy node）所对应的连接权重 w_{n+1} ，这样，权重和阈值的学习就可统一为权重的学习。说白了，哑节点就是你们在线性回归当中学过的 $y = kx + b$ 的 b （我们一般把 b 塞进关于 k 的矩阵中，也就是 w_{n+1} ），再或者你可以把它理解为连接图中的节点自连接（你可以喂给 AI，让他为你详细解释），如👉图。



所以我们再来看这个问题，简化一下为 5 个节点（有向全连接），那么有：20 个连接权（自连接 + 与其他节点有向连接）和 5 个阈值 (θ)，总共 25 个参数。



感知机学习规则非常简单，对训练样例 (x, y) ，若当前感知机的输出为 \hat{y} ，则感知机权重将这样调整（公式来自西瓜书）：

$$w_i \leftarrow w_i + \Delta w_i, \quad (5.1)$$

$$\Delta w_i = \eta(y - \hat{y})x_i, \quad (5.2)$$

其中 $\eta(0, 1)$ 称为学习率 (learning rate) (通常设置为一个小于一的数，比如 0.1)。从式 (5.1) 可看出，若感知机对训练样例 (x, y) 预测正确，即 $\hat{y} = y$ ，则感知机不发生变化，否则将根据错误的程度进行权重调整。

需注意的是，**感知机只有输出层神经元进行激活函数处理，即只拥有一层功能神经元 (functional neuron)，其学习能力非常有限**。事实上，上述与、或、非问题都是线性可分

(linearly separable) 的问题。可以证明，若两类模式是线性可分的，即存在一个线性超平面能将它们分开，如图下所示，则感知机的学习过程一定会收敛(*converge*)而求得适当的权向量 $w = (w_1; w_2; \dots; w_{n+1})$ ；否则感知机学习过程将会发生振荡(*fuctuation*)， w 难以稳定下来，不能求得合适解，例如感知机甚至不能解决如下图 4 所示的异或这样简单的非线性可分问题。

注：“非线性可分”意味着用线性超平面无法划分。

要解决非线性可分问题，需考虑使用多层功能神经元。例如下图中这个简单的两层感知机就能解决异或问题。在下图左边，输出层与输入层之间的一层神经元，被称为隐层或隐含层 (*hidden layer*)，隐含层和输出层神经元都是拥有激活函数的功能神经元。

注意，这里没有使用核哦~我在我的[博客](#)里也有说过。不过可以清晰地知道，当隐层达到一定数量，就可以实现非线性的功能！

更一般的，常见的神经网络是形如图右👉所示的层级结构，每层神经元与下一层神经元全互连，神经元之间不存在同层连接，也不存在跨层连接。这样的神经网络结构通常称为“**多层前馈神经网络**” (*multi-layer feedforward neural networks*)，其中输入层神经元接收外界输入，隐层与输出层神经元对信号进行加工，最终结果由输出层神经元输出；换言之，输入层神经元仅是接受输入，不进行函数处理，隐层与输出层包含功能神经元，因此，左图通常被称为“**两层网络**”，为避免歧义，讲称其为“**单隐层网络**”，只需包含隐层，即可称为**多层网络**，神经网络的学习过程，就是根据训练数据来调整神经元之间的“**连接权**” (*connection weight*) 以及每个功能神经元的**阈值**；换言之，神经网络“学”到的东西，蕴涵在连接权与阈值中（这就是我说的：**权重和阈值**）。

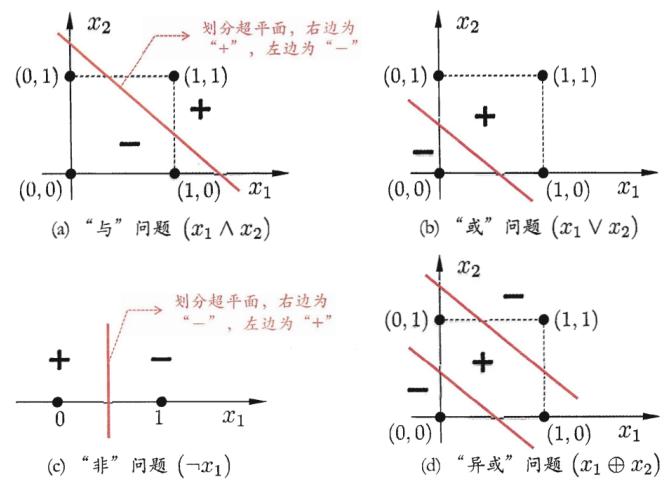


图 5.4 线性可分的“与”“或”“非”问题与非线性可分的“异或”问题

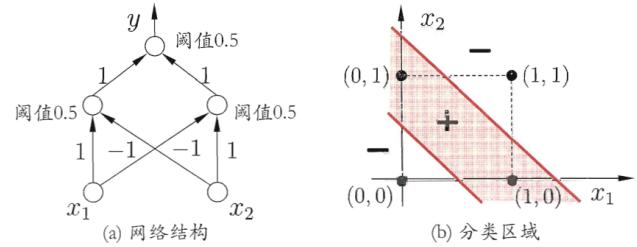


图 5.5 能解决异或问题的两层感知机

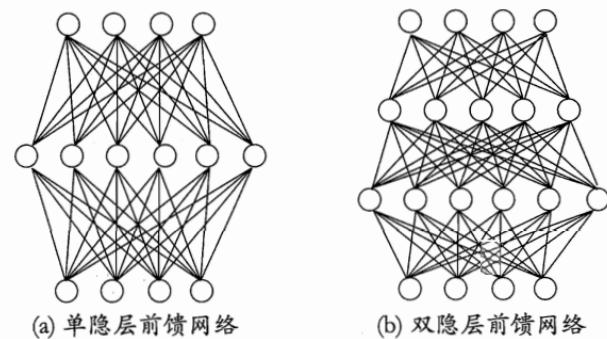


图 5.6 多层前馈神经网络结构示意图

