

【机器学习】神经网络浅讲：从神经元到深度学习（一）

计算机的潜意识 机器学习算法那些事 2018-11-08

作者：计算机的潜意识

链接：<http://www.cnblogs.com/subconscious/p>

神经网络是一门重要的机器学习技术。它是目前最为火热的研究方向--深度学习的基础。学习神经网络不仅可以让你掌握一门强大的机器学习方法，同时也可以更好地帮助你理解深度学习技术。

本文以一种简单的，循序的方式讲解神经网络，对神经元，单层感知器和多层感知器的知识讲解的浅显易懂，是一篇入门深度学习的好文。

神经网络是一种模拟人脑的神经网络以期能够实现类人工智能的机器学习技术。人脑中的神经网络是一个非常复杂的组织。成人的大脑中估计有1000亿个神经元之多。

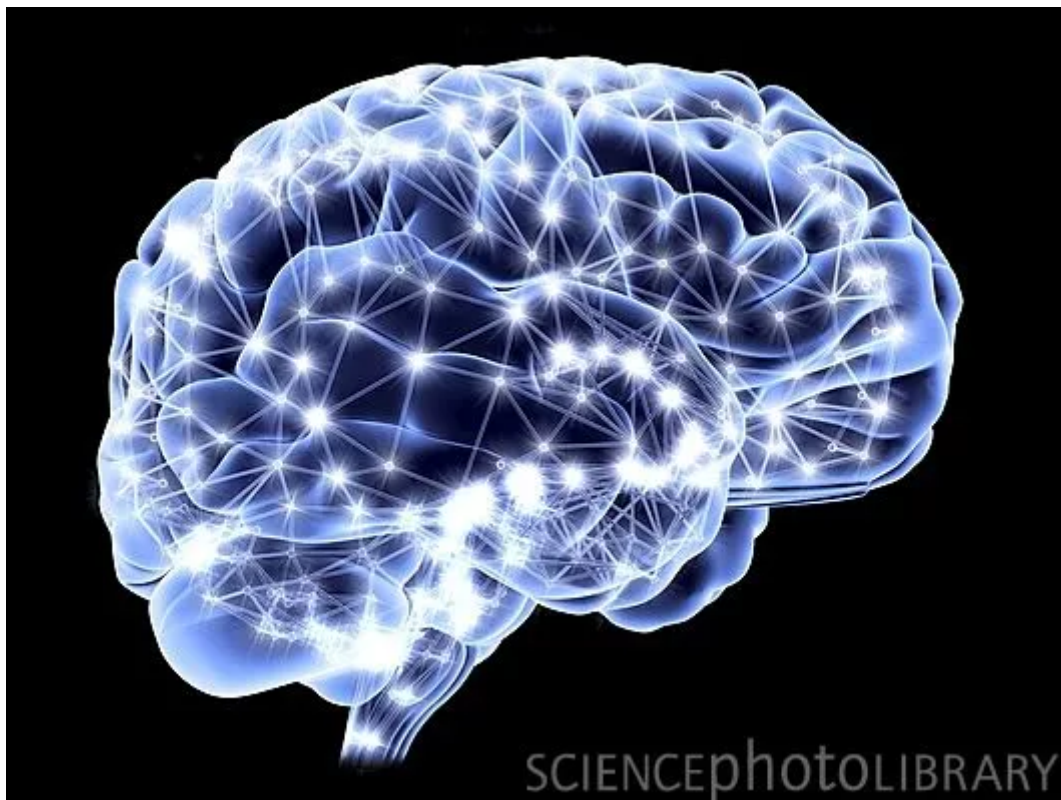


图1 人脑神经网络

那么机器学习中的神经网络是如何实现这种模拟的，并且达到一个惊人的良好效果的？通过本文，你可以了解到这些问题的答案，同时还能知道神经网络的历史，以及如何较好地学习它。

由于本文较长，为方便读者，以下是**本文的目录**：

- 1：前言
- 2：神经元
- 3：单层神经网络（感知器）
- 4：两层神经网络（多层感知器）
- 5：两层神经网络（深度学习）
- 6：回顾
- 7：总结
- 8：展望

前言

让我们来看一个经典的神经网络。这是一个包含三个层次的神经网络。红色的是**输入层**，绿色的是**输出层**，紫色的是**中间层**（也叫**隐藏层**）。输入层有3个输入单元，隐藏层有4个单元，输出层有2个单元。后文中，我们统一使用这种颜色来表达神经网络的结构。

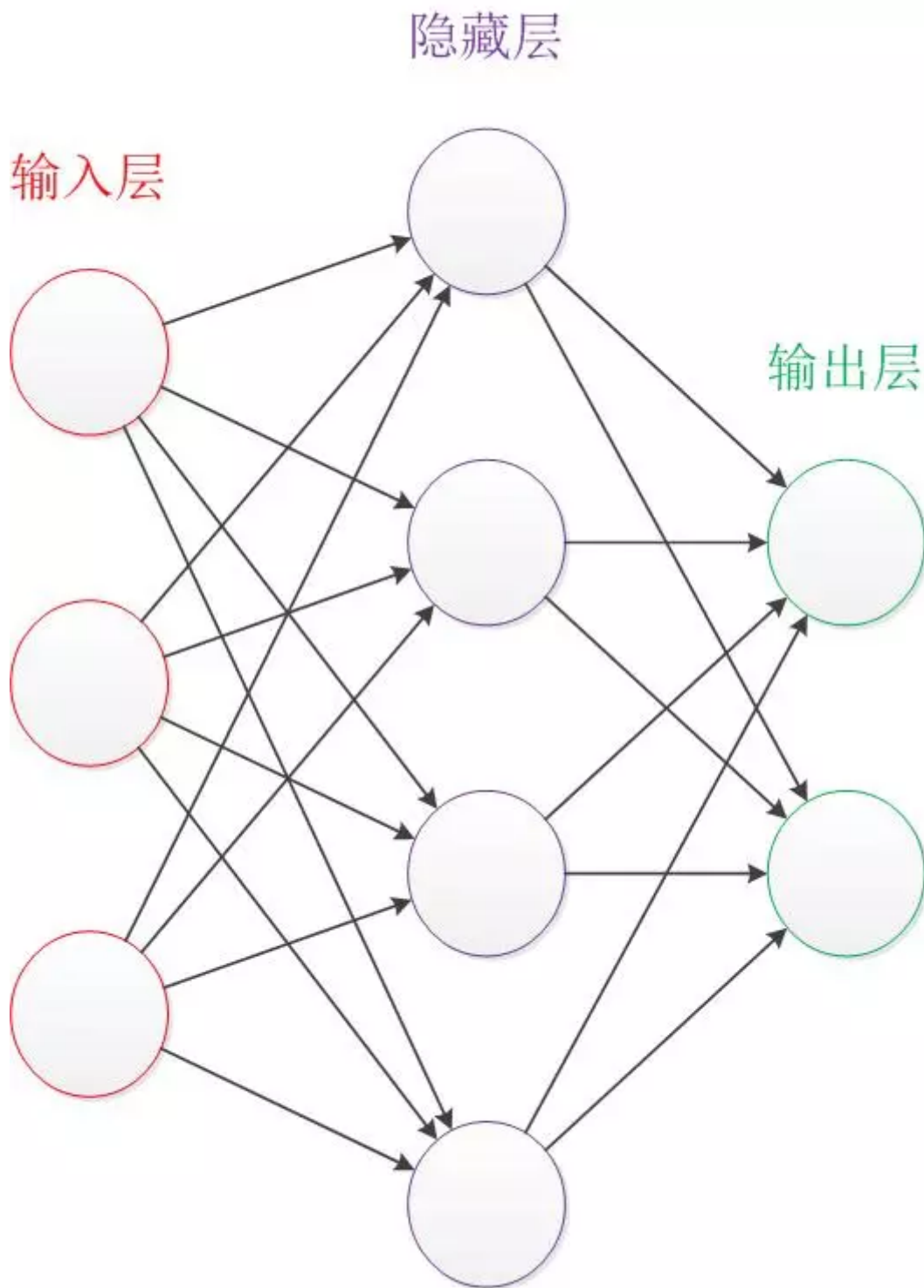


图2 神经网络结构图

在开始介绍前，有一些知识可以先记在心里：

- 1：设计一个神经网络时，输入层与输出层的节点数往往是固定的，中间层则可以自由指定；
- 2：神经网络结构图中的拓扑与箭头代表着**预测**过程时数据的流向，跟**训练**时的数据流有一定的区别；
- 3：结构图里的关键不是圆圈（代表“神经元”），而是连接线（代表“神经元”之间的连接）。每个连接线对应一个不同的**权重**（其值称为**权值**），这是需要训练得到的。

除了从左到右的形式表达的结构图，还有一种常见的表达形式是从下到上来表示一个神经网络。这时候，输入层在图的最下方。输出层则在图的最上方，如下图：

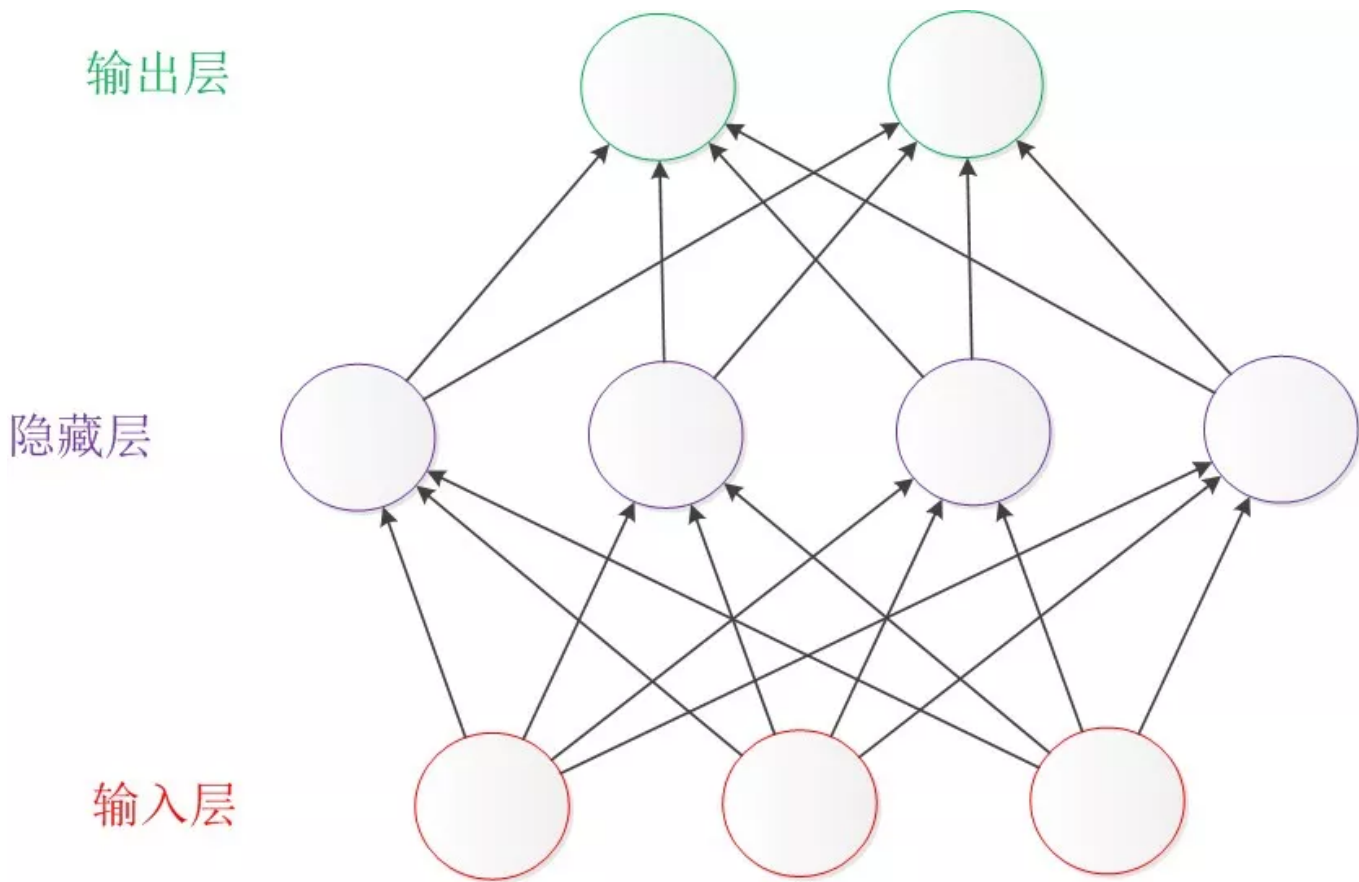


图3 从下到上的神经网络结构图

从左到右的表达形式以Andrew Ng和LeCun的文献使用较多，Caffe里使用的则是从下到上的表达。在本文中使用Andrew Ng代表的从左到右的表达形式。

下面从简单的神经元开始说起，一步一步介绍神经网络复杂结构的形成。

神经元

引子

对于神经元的研究由来已久，1904年生物学家就已经知晓了神经元的组成结构。

一个神经元通常具有多个**树突**，主要用来接受传入信息；而**轴突**只有一条，轴突尾端有许多轴突末梢可以给其他多个神经元传递信息。轴突末梢跟其他神经元的树突产生连接，从而传递信号。这个连接的位置在生物学上叫做“**突触**”。

人脑中的神经元形状可以用下图做简单的说明：

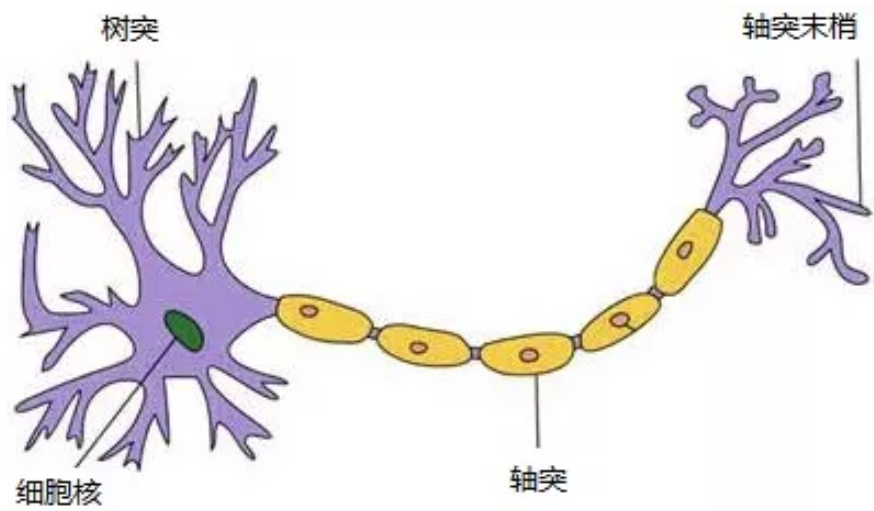


图4 神经元

1943年，心理学家McCulloch和数学家Pitts参考了生物神经元的结构，发表了抽象的神经元模型MP。在下文中，我们会具体介绍神经元模型。

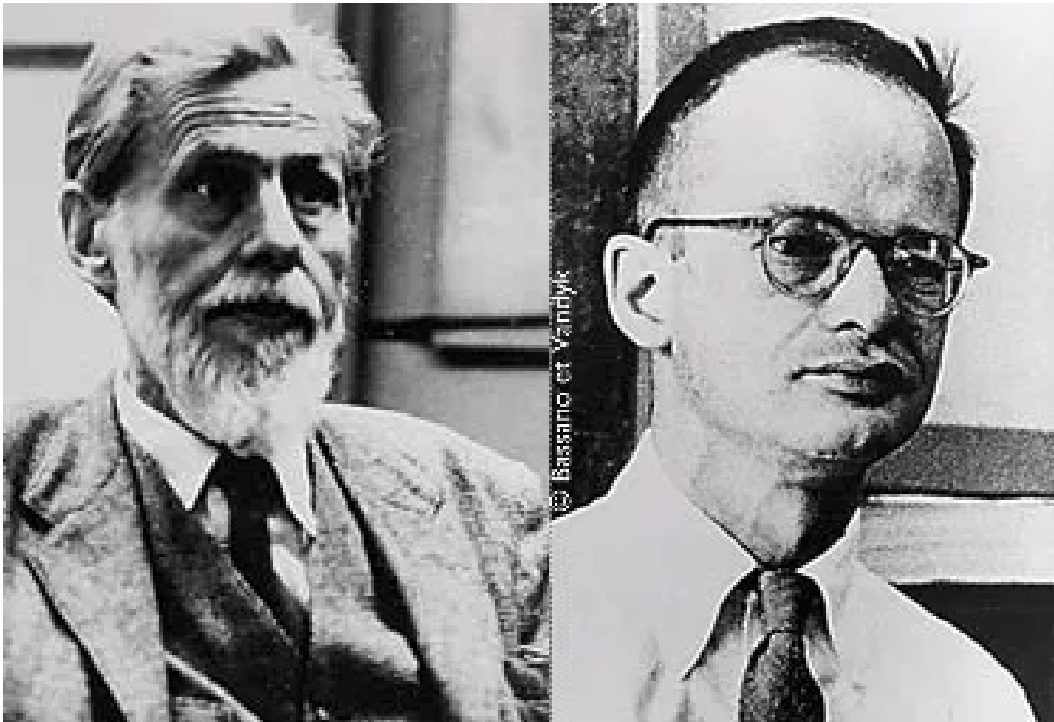


图5 Warren McCulloch（左）和 Walter Pitts（右）

结构

神经元模型是一个包含输入，输出与计算功能的模型。输入可以类比为神经元的树突，而输出可以类比为神经元的轴突，计算则可以类比为细胞核。

下图是一个典型的神经元模型：包含有3个输入，1个输出，以及2个计算功能。

注意中间的箭头线。这些线称为“连接”。每个上有一个“权值”。

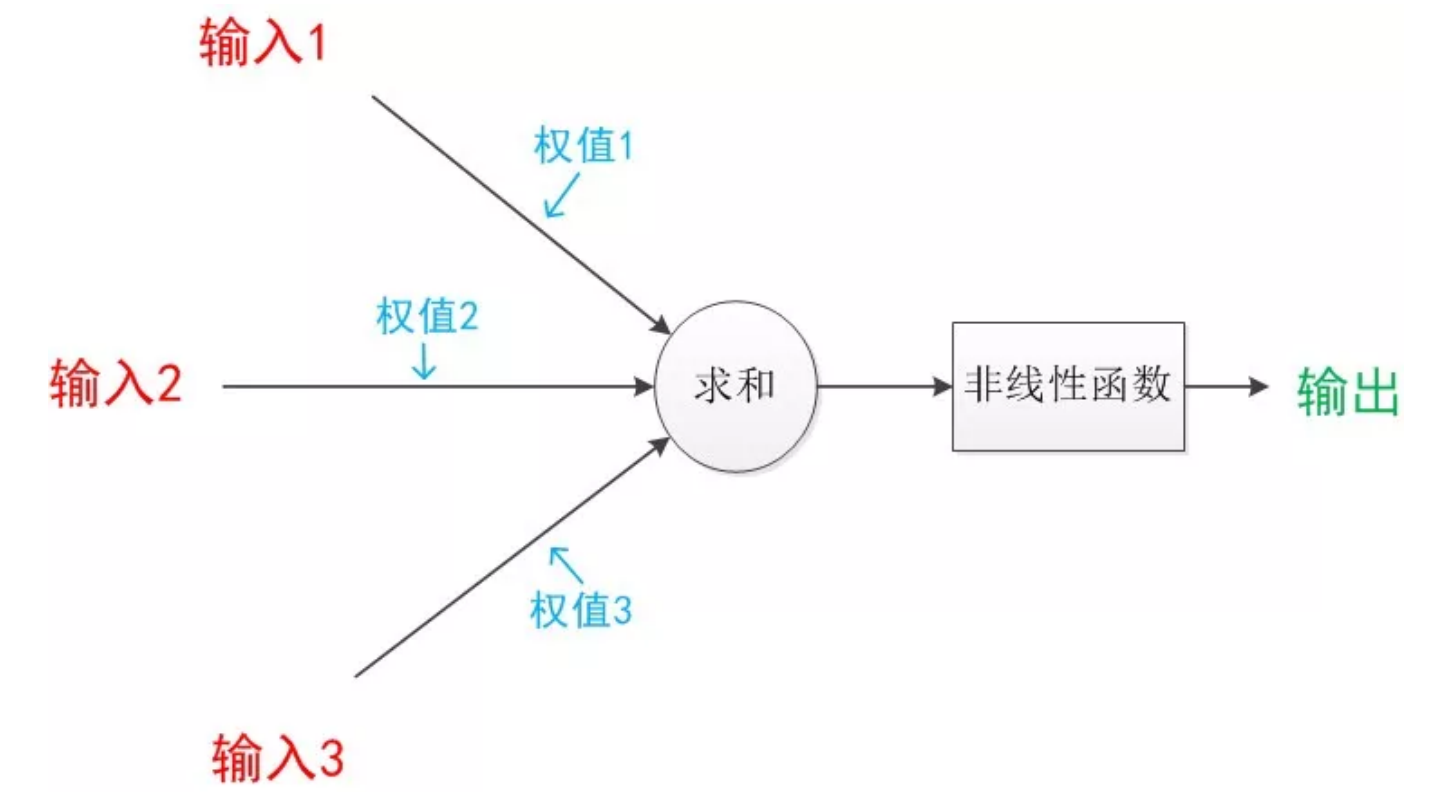


图6 神经元模型

连接是神经元中最重要的东西。每一个连接上都有一个权重。

一个神经网络的训练算法就是让权重的值调整到最佳，以使得整个网络的预测效果最好。

我们使用 a 来表示输入，用 w 来表示权值。一个表示连接的有向箭头可以这样理解：在初端，传递的信号大小仍然是 a ，端中间有加权参数 w ，经过这个加权后的信号会变成 $a * w$ ，因此在连接的末端，信号的大小就变成了 $a * w$ 。

在其他绘图模型里，有向箭头可能表示的是值的不变传递。而在神经元模型里，每个有向箭头表示的是值的加权传递。

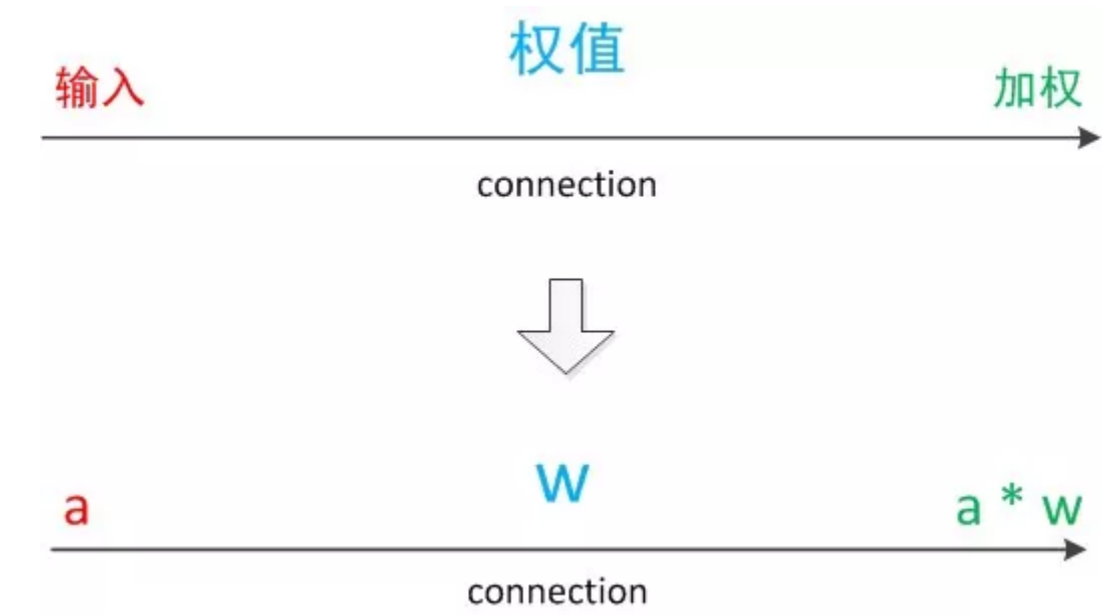
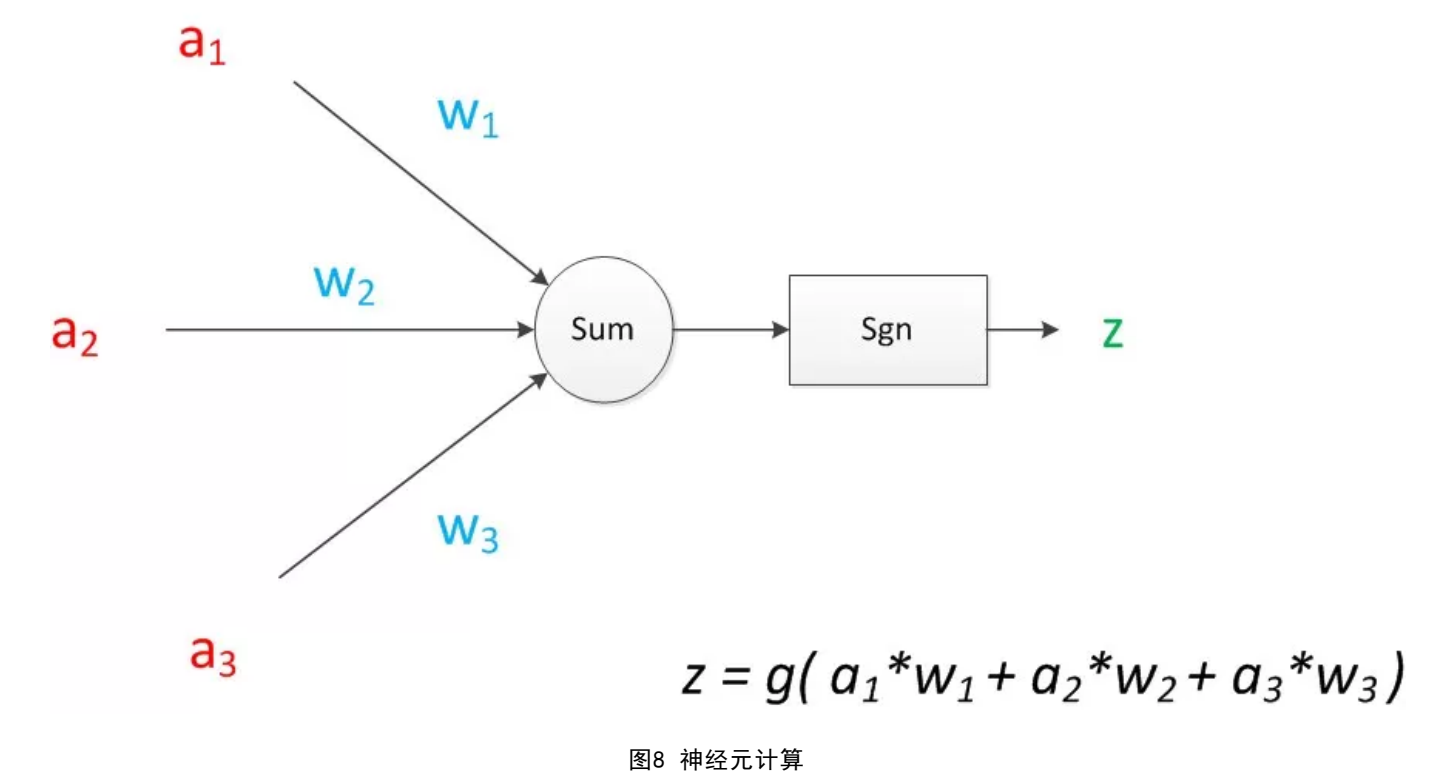


图7 连接（connection）

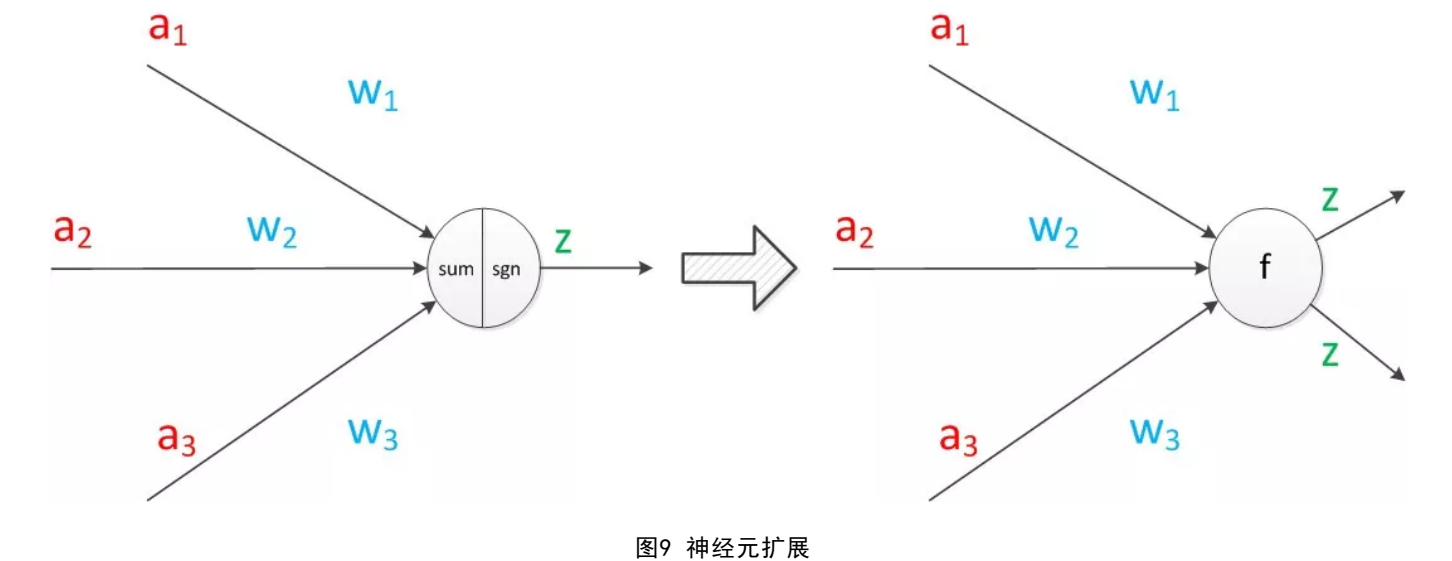
如果我们将神经元图中的所有变量用符号表示，并且写出输出的计算公式的话，就是下图。



可见z是在输入和权值的线性加权和叠加了一个函数g的值。在MP模型里，函数g是sgn函数，也就是取符号函数。这个函数当输入大于0时，输出1，否则输出0。

下面对神经元模型的图进行一些扩展。首先将sum函数与sgn函数合并到一个圆圈里，代表神经元的内部计算。其次，把输入a与输出z写到连接线的左上方，便于后面画复杂的网络。最后说明，一个神经元可以引出多个代表输出的有向箭头，但值都是一样的。

神经元可以看作一个计算与存储单元。计算是神经元对其的输入进行计算功能。存储是神经元会暂存计算结果，并传递到下一层。



当我们用“神经元”组成网络以后，描述网络中的某个“神经元”时，我们更多地会用“单元”（unit）来指代。同时由于神经网络的表现形式是一个有向图，有时也会用“节点”（node）来表达同样的意思。

效果

神经元模型的使用可以这样理解：

我们有一个数据，称之为样本。样本有四个属性，其中三个属性已知，一个属性未知。我们需要做的就是通过三个已知属性**预测**未知属性。

具体办法就是使用神经元的公式进行计算。三个已知属性的值是 a_1 , a_2 , a_3 ，未知属性的值是 z 。 z 可以通过公式计算出来。

这里，已知的属性称之为**特征**，未知的属性称之为**目标**。假设特征与目标之间确实是线性关系，并且我们已经得到表示这个关系的权值 w_1 , w_2 , w_3 。那么，我们就可以通过神经元模型预测新样本的目标。

影响

1943年发布的MP模型，虽然简单，但已经建立了神经网络大厦的地基。但是，MP模型中，权重的值都是预先设置的，因此不能学习。

1949年心理学家Hebb提出了Hebb学习率，认为人脑神经细胞的**突触**（也就是连接）上的强度上可以变化的。于是计算科学家们开始考虑用调整权值的方法来让机器学习。这为后面的学习算法奠定了基础。



图10 Donald Olding Hebb

尽管神经元模型与Hebb学习律都已诞生，但限于当时的计算机能力，直到接近10年后，第一个真正意义的神经网络才诞生。

单层神经网络

引子

1958年，计算科学家Rosenblatt提出了由两层神经元组成的神经网络。他给它起了一个名字--“感知器”（Perceptron）（有的文献翻译成“感知机”，下文统一用“感知器”来指代）。

1958年，计算科学家Rosenblatt提出了由两层神经元组成的神经网络。他给它起了一个名字--“感知器”（Perceptron）（有的文献翻译成“感知机”，下文统一用“感知器”来指代）。

人们认为已经发现了智能的奥秘，许多学者和科研机构纷纷投入到神经网络的研究中。美国军方大力资助了神经网络的研究，并认为神经网络比“原子弹工程”更重要。这段时间直到1969年才结束，这个时期可以看作神经网络的第一次高潮。



图11 Rosenblat与感知器

结构

下面来说明感知器模型。

在原来MP模型的“输入”位置添加神经元节点，标志其为“输入单元”。其余不变，于是我们就有了下图：从本图开始，我们将权值 w_1 , w_2 , w_3 写到“连接线”的中间。

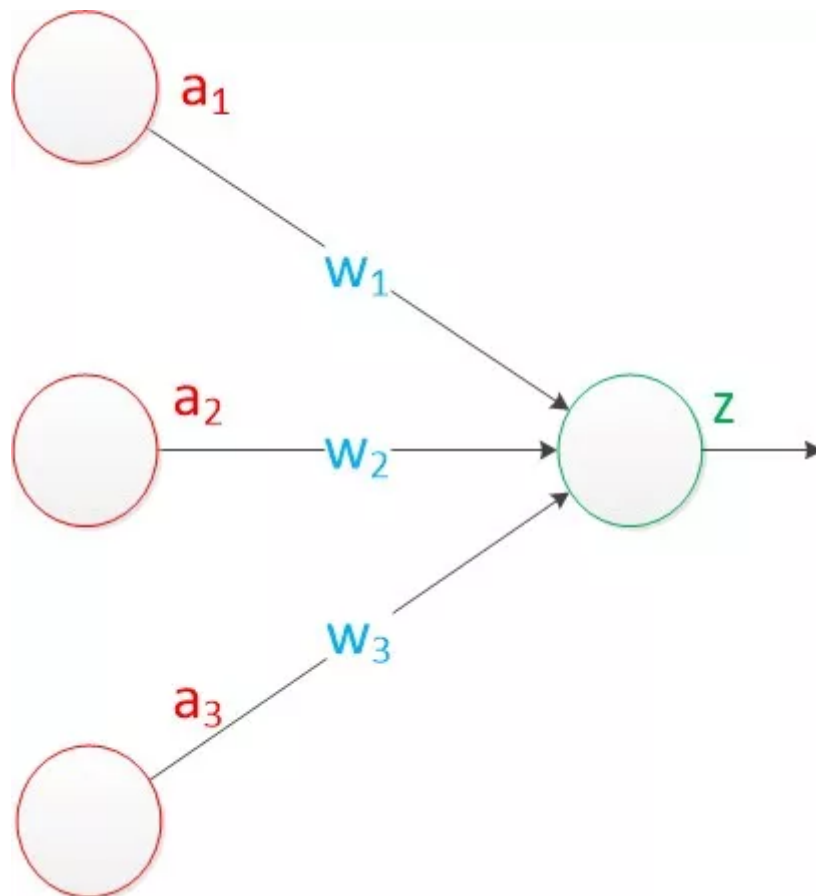


图12 单层神经网络

在“感知器”中，有**两个层次**。分别是输入层和输出层。输入层里的“输入单元”只负责传输数据，不做计算。输出层里的“输出单元”则需要对前面一层的输入进行计算。

我们把需要计算的层次称之为“计算层”，并把拥有一个计算层的网络称之为“单层神经网络”。有一些文献会按照网络拥有的层数来命名，例如把“感知器”称为两层神经网络。但在本文里，我们根据计算层的数量来命名。

假如我们要预测的目标不再是一个值，而是一个向量，例如 $[2,3]$ 。那么可以在输出层再增加一个“输出单元”。

下图显示了带有两个输出单元的单层神经网络，其中输出单元 z_1 的计算公式如下图。

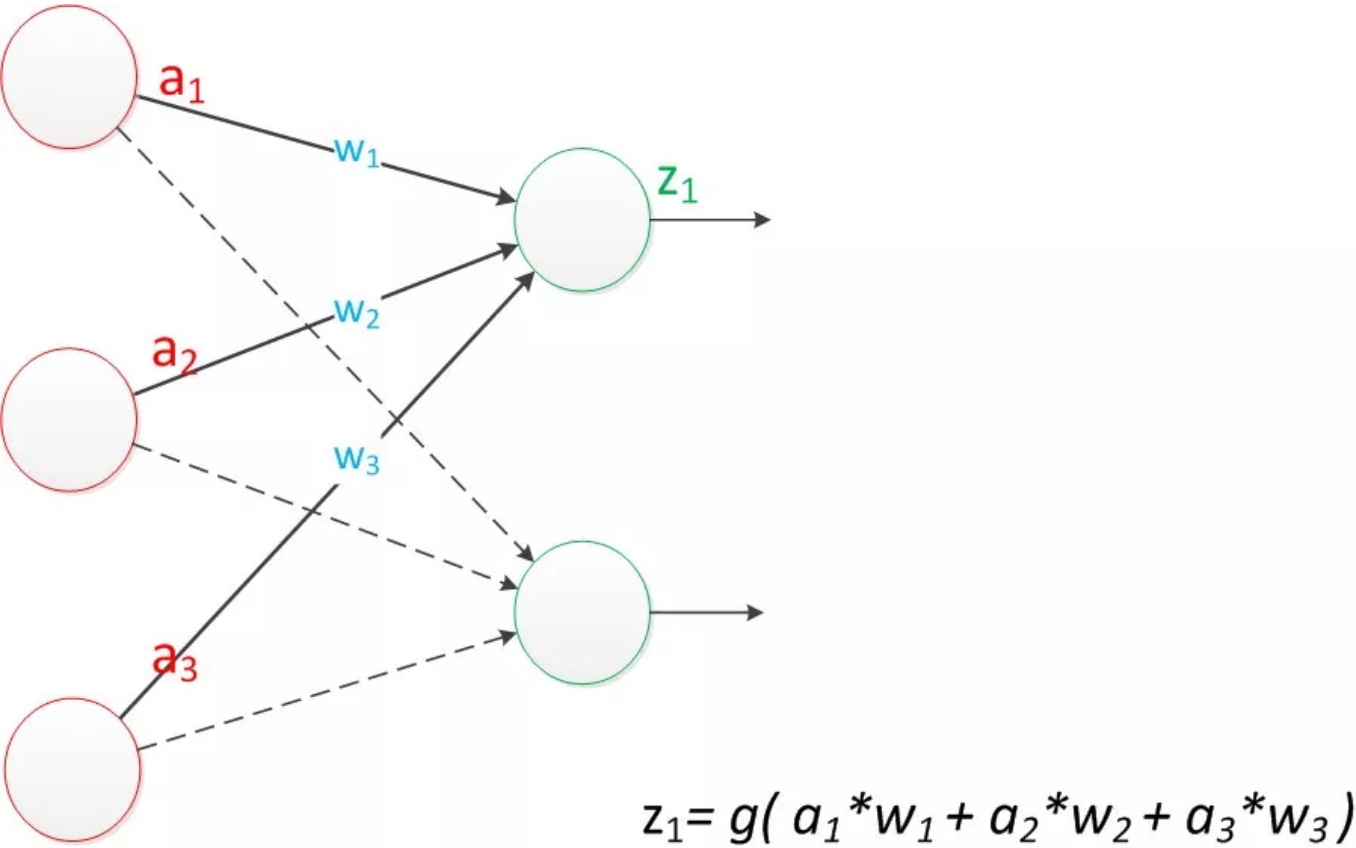


图13 单层神经网络 (Z₁)

可以看到，z₁的计算跟原先的z并没有区别。

我们已知一个神经元的输出可以向多个神经元传递，因此z₂的计算公式如下图。

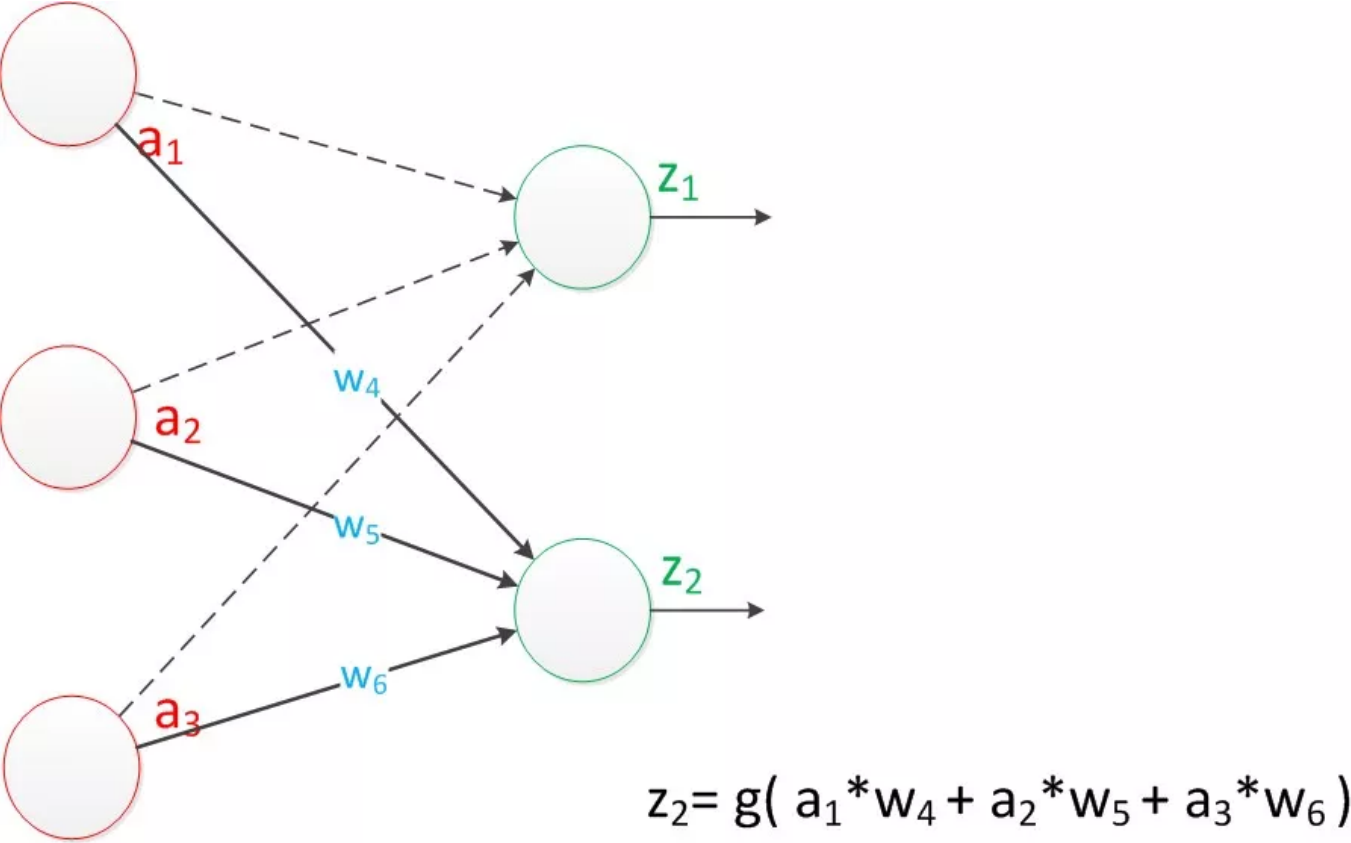


图14 单层神经网络 (Z₂)

可以看到， z_2 的计算中除了三个新的权值： w_4 , w_5 , w_6 以外，其他与 z_1 是一样的。

整个网络的输出如下图。

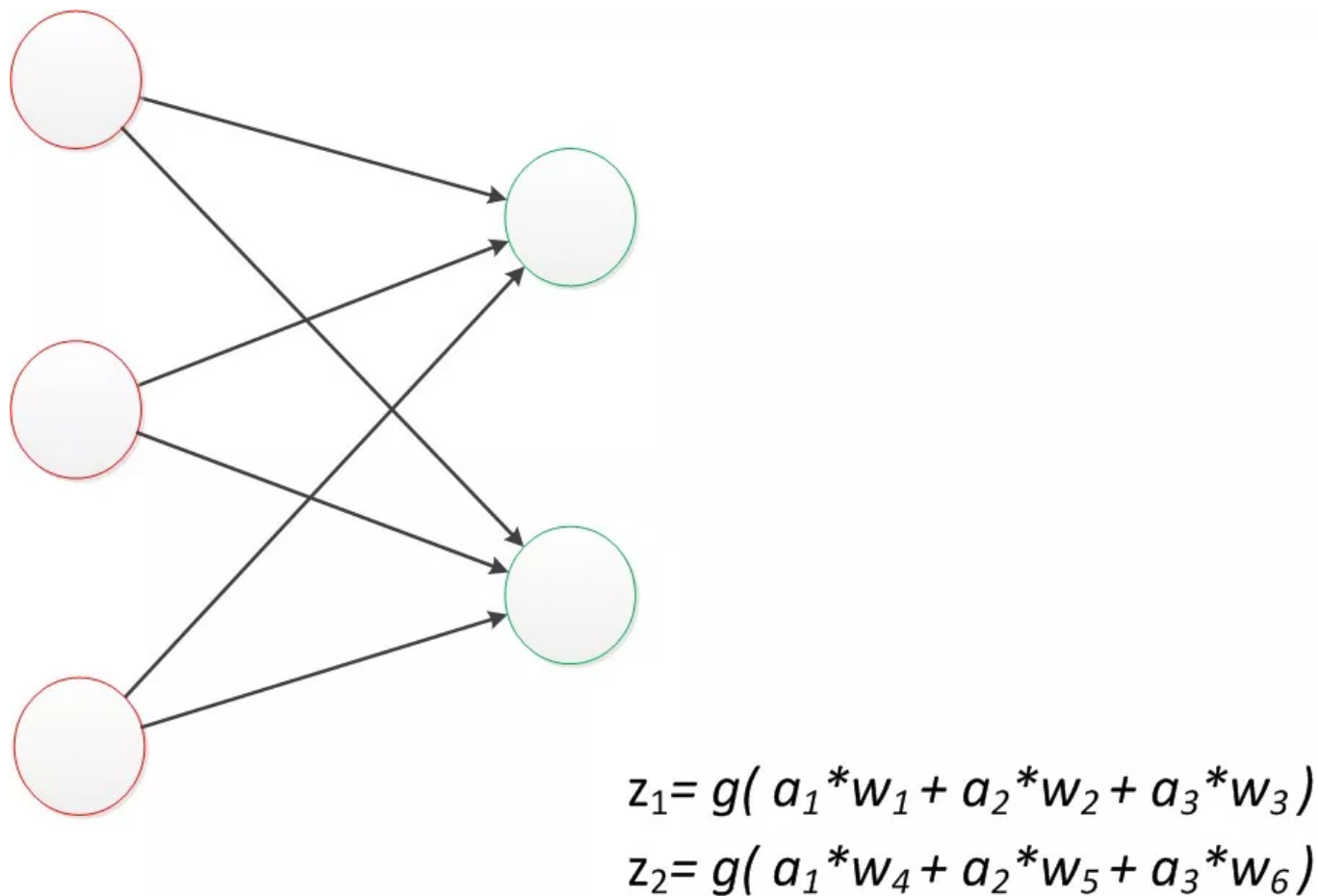
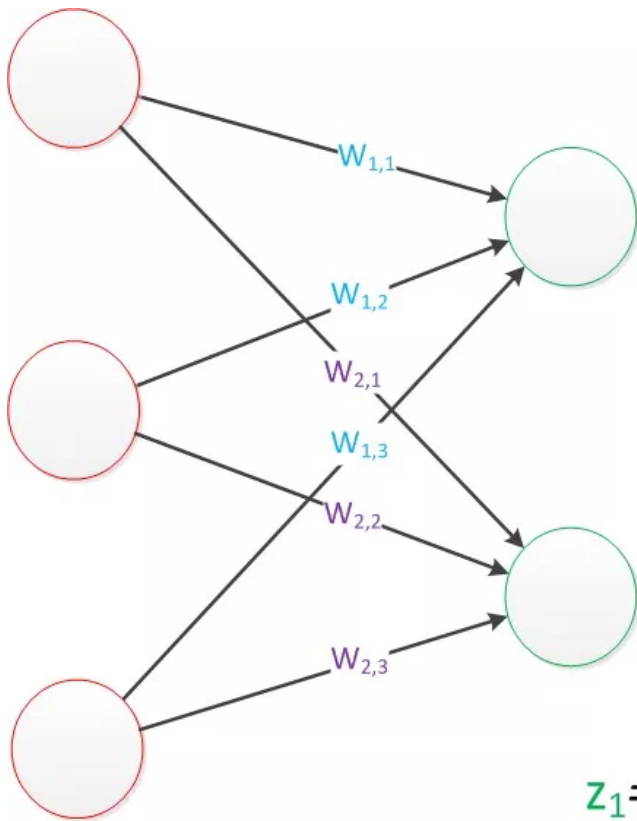


图15 单层神经网络 (z_1 和 z_2)

目前的表达公式有一点不让人满意的就是： w_4 , w_5 , w_6 是后来加的，很难表现出跟原先的 w_1 , w_2 , w_3 的关系。

因此我们改用二维的下标，用 $w_{x,y}$ 来表达一个权值。下标中的 x 代表后一层神经元的序号，而 y 代表前一层神经元的序号（序号的顺序从上到下）。

例如， $w_{1,2}$ 代表后一层的第1个神经元与前一层的第2个神经元的连接的权值（这种标记方式参照了Andrew Ng的课件）。根据以上方法标记，我们有了下图。



$$z_1 = g(a_1 * w_{1,1} + a_2 * w_{1,2} + a_3 * w_{1,3})$$

$$z_2 = g(a_1 * w_{2,1} + a_2 * w_{2,2} + a_3 * w_{2,3})$$

图16 单层神经网络(扩展)

如果我们仔细看输出的计算公式，会发现这两个公式就是线性代数方程组。因此可以用矩阵乘法来表达这两个公式。

例如，输入的变量是 $[a_1, a_2, a_3]^T$ （代表由 a_1, a_2, a_3 组成的列向量），用向量 \mathbf{a} 来表示。方程的左边是 $[z_1, z_2]^T$ ，用向量 \mathbf{z} 来表示。系数则是矩阵 \mathbf{W} （2行3列的矩阵，排列形式与公式中的一样）。于是，输出公式可以改写成：

$$g(\mathbf{W} * \mathbf{a}) = \mathbf{z};$$

这个公式就是神经网络中从前一层计算后一层的**矩阵运算**。

效果

与神经元模型不同，感知器中的权值是通过训练得到的。因此，根据以前的知识我们知道，感知器类似一个**逻辑回归**模型，可以做线性分类任务。

我们可以用**决策分界**来形象的表达分类的效果。决策分界就是在二维的数据平面中划出一条直线，当数据的维度是3维的时候，就是划出一个平面，当数据的维度是n维时，就是划出一个n-1维的超平面。

下图显示了在二维平面中划出决策分界的效果，也就是感知器的分类效果。

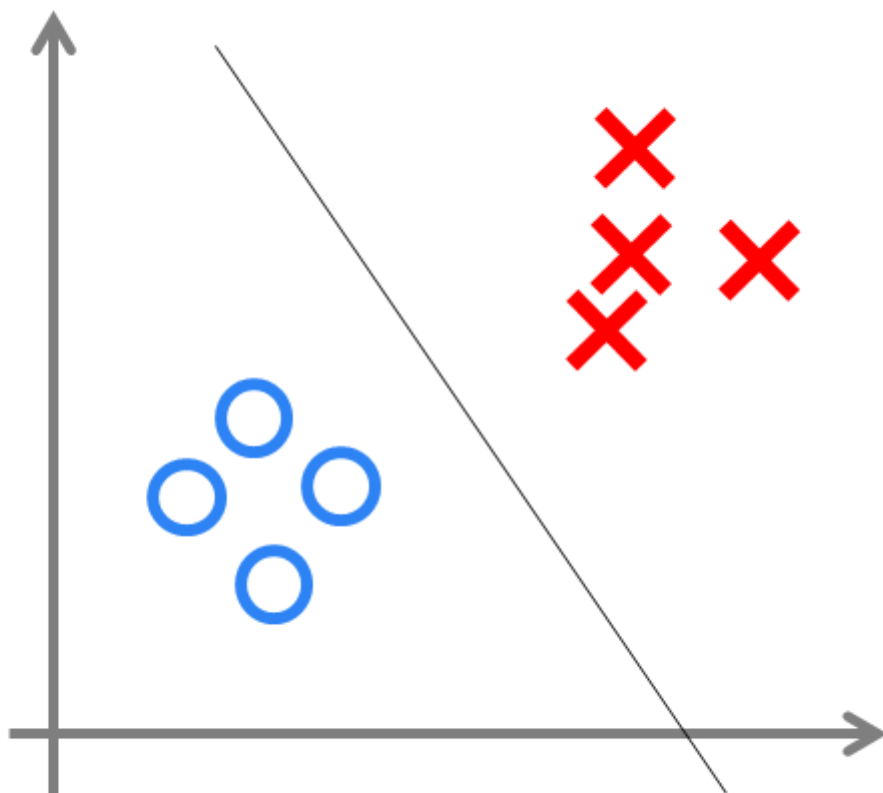


图17 单层神经网络（决策分界）

影响

感知器只能做简单的线性分类任务。但是当时的人们热情太过于高涨，并没有人清醒的认识到这点。于是，当人工智能领域的巨擘Minsky指出这点时，事态就发生了变化。

Minsky在1969年出版了一本叫《Perceptron》的书，里面用详细的数学证明了感知器的弱点，尤其是感知器对XOR（异或）这样的简单分类任务都无法解决。

Minsky认为，如果将计算层增加到两层，计算量则过大，而且没有有效的学习算法。所以，他认为研究更深层的网络是没有价值的。（本文成文后一个月，即2016年1月，Minsky在美国去世。谨在本文中纪念这位著名的计算机研究专家与大拿。）



图18 Marvin Minsky

由于Minsky的巨大影响力以及书中呈现的悲观态度，让很多学者和实验室纷纷放弃了神经网络的研究。神经网络的研究陷入了冰河期。这个时期又被称为“AI winter”。

接近10年以后，对于两层神经网络的研究才带来神经网络的复苏，关于两层神经网络，后面文章会总结。

推荐阅读文章

比较全面的L1和L2正则化的解释

线性分类模型（二）：logistic回归模型分析

为什么梯度是函数变化最快的方向

梯度下降法的三种形式BGD、SGD以及MBGD



长按二维码关注

机器学习算法那些事

微信：beautifulife244

砥砺前行 不忘初心