**M-P神经元，感知机与神经网络**

# 发展历史

1943年，美国心理学家McCulloch和数学家Pitts提出M-P神经元模型。

1956年，Dartmouth会议，首次提出人工智能（Artificial Intelligence, AI）

1957年，心理学家Frank Rosenblatt提出感知机（Perceptron）；

1968年，认知科学家Minsky指出感知机模型不能解决异或问题；

1982年，生物物理学家Hopfiled提出Hopfiled神经网络；

1986年，Hinton与Sejnowski提出玻尔兹曼机，其运行和学习过程使用模拟退火算法，克服了Hopfiled网络存在的能量局部极小值问题；

1986年，心理学家David Rumelhart和计算机科学家James McCleland提出反向传播算法，成为影响最大的人工神经网络学习方法；

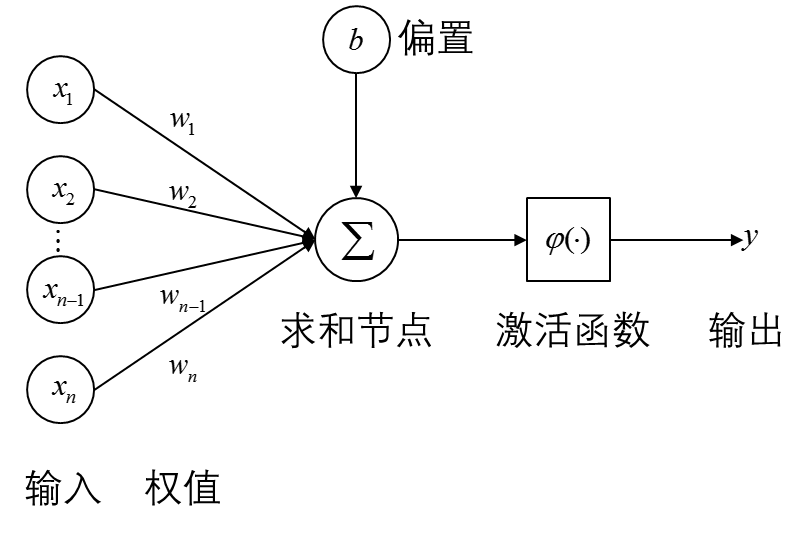
1995年，计算机科学家Vapnik提出支持向量机；

2006年，Hinton提出深度信念网络；

2012年，以Hinton为首的研究人员在ImageNet 2012图像识别大赛上使用深度卷积神经网络AlexNet夺冠；

2016年，AlphaGo战胜李世石。

# M-P神经元模型



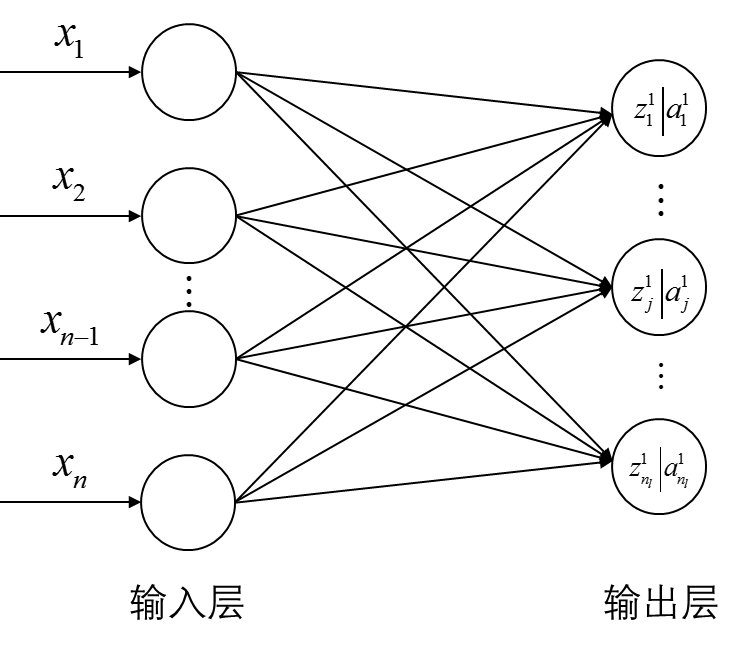
1. 每个神经元都是一个多输入单输出的信息处理单元；
2. 激活函数是阶跃函数，因此每个神经元的输出状态均为0或1；
3. 连接权值是初始设定的，不能调整，因此没有学习能力。

# 感知机

## 单输出

相当于神经元模型，但具有学习能力，可根据感知器学习算法调整权值。

## 多输出



# 人工神经网络分类

## 前馈神经网络

### 感知机

#### 单层感知机（感知机学习算法）

#### 多层感知机（BP学习算法）

### 径向基函数神经网络

## 反馈神经网络

### Hopfield神经网络

### Boltzmann机

## 自组织竞争型神经网络

### 自组织共振理论神经网络（Adaptive Resonance Theory, ART）

#### ARTI

#### ARTII

#### ARTIII

### 自组织特征映射神经网络（Self-Organizing Feature Map, SOM）

### 对抗传播神经网络（Counter Propagation Network, CPN）

# 神经网络学习算法

## Hebb学习规则（无监督）

第个神经元与第个神经元之间的权重：



与分别为第个神经元与第个神经元的输出结果。

## 学习规则



## 最小均方学习规则



## 相关学习规则

## 竞争学习规则

## 随机学习规则

# 梯度消失与梯度爆炸

1. 如果采用logistic或者tanh等S型非线性激活函数，损失函数采用均方差损失函数，当输入的绝对值很大时，会出现“饱和”现象，即该处的导数接近于0，此时计算损失函数对网络权重的导数的绝对值越来越小，非常接近于0，称为“梯度消失”，这会导致网络的训练停滞不前。分类问题采用交叉熵损失函数
2. 如果初始化网络时使用了过大的初始值或者网络的权重随着训练越来越大，会使得网络不稳定，性能崩溃，称为梯度爆炸，对于循环神经网络或者生成式对抗网络（Generative Adversarial Networks, GAN）容易出现这种现象。

# 欠拟合与过拟合

## 欠拟合

### 原因：

1. 网络的容量不足，这往往由于网络的广度和深度不够；
2. 训练网络陷入了局部最优（但对于大规模的神经网络，神经网络的广度和深度足够大，往往最后会得到想当、、相当靠近局部最优值的解）[16]；
3. 初始化方法不佳或者学习率过大或者过小；
4. 数据本身的噪声过大或者分布过于极端；

## 过拟合

### 原因：

1. 网络太过复杂，参数过多；

### 解决方法：

1. 提早停止
2. 修改损失函数：L2正则化与L1正则化
3. Dropout（等价于训练多个不同结构的神经网络）
4. 每次训练时，随机隐藏部分隐层神经元。
5. 根据样本值，修改未隐藏的神经元的参数。隐藏的神经元的参数保持不变。
6. 下次训练时，重新随机选择需要隐藏的神经元。

由于神经网络的非线性，Dropout的理论证明尚属空白，这里只有一些直观解释。

1. dropout掉不同的隐藏神经元就类似在训练不同的网络，整个dropout过程就相当于对很多个不同的神经网络取平均。而不同的网络产生不同的过拟合，一些互为“反向”的拟合相互抵消就可以达到整体上减少过拟合。这实际上就是bagging的思想。
2. 因为dropout程序导致两个神经元不一定每次都在一个dropout网络中出现。这会迫使网络去学习更加鲁棒的特征。换句话说，假如我们的神经网络是在做出某种预测，它不应该对一些特定的线索片段太过敏感，即使丢失特定的线索，它也应该可以从众多其它线索中学习一些共同的模式（鲁棒性）。
3. 集成方法
4. 训练多个相同结构的神经网络，由于每次的初始化权重是随机的，且随机梯度下降算法也带有随机性，因此实际上会得到多个不同的网络，最后对这些网络的输出做平均；
5. 训练多个不同结构的网络，让网络之间的差异更大，会学习到数据的不同方面。

因为神经网络的损失函数一般是非凸的，容量小的网络更容易陷入局部极小点而达不到最优的效果，同时这些局部最小点的方差特别大，换句话说，也就是每个局部最优点的差异都特别大，所以你在训练网络的时候训练10 次时能得到的结果有很大的差异。但是对于容量更大的神经网络，它的局部极小点的方差特别小，也就是说训练多次虽然可能陷入不同的局部极小点，但是它们之间的差异是很小的，这样训练就不会完全依靠随机初始化。所以我们更希望使用大容量的网络去训练模型，同时运用一些方法来控制网络的过拟合。