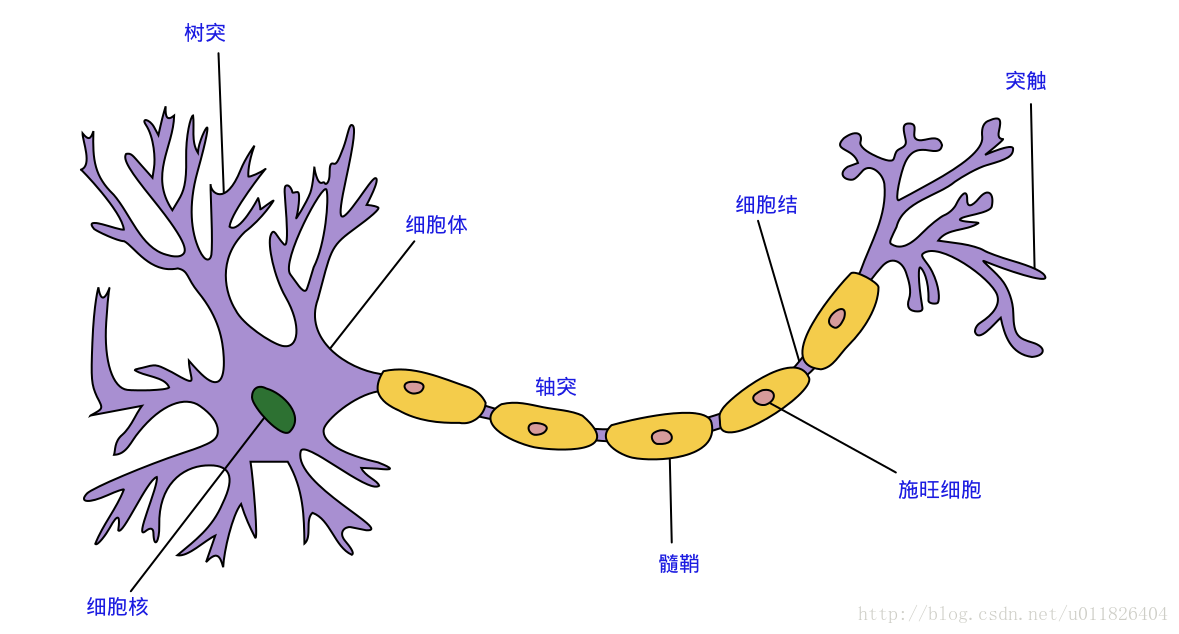
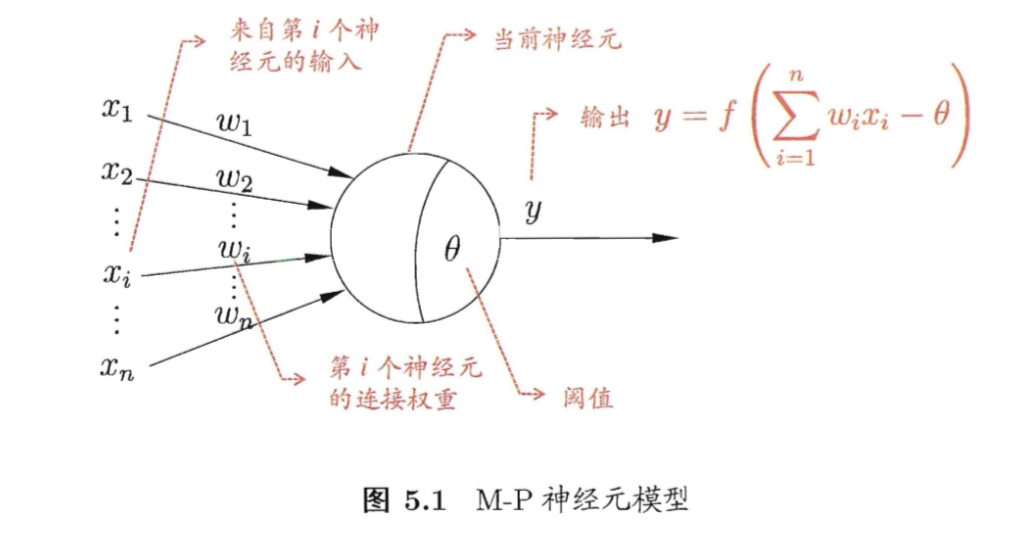
1.神经元模型



我们高中生物都学过，人体的信息传递大部分是由神经完成的，神经冲动的电信号到达神经末梢刺激释放神经递质给下一个神经元，当递质达到一定数目则下一个神经元产生一个新的电信号，这就是信息在人体传播的一个途径。

然后就有人根据这个搞出来的神经网络，下面的图片就是一个简单的神经元模型（M-P神经元模型），是所有神经网络模型的一个基本的成分。

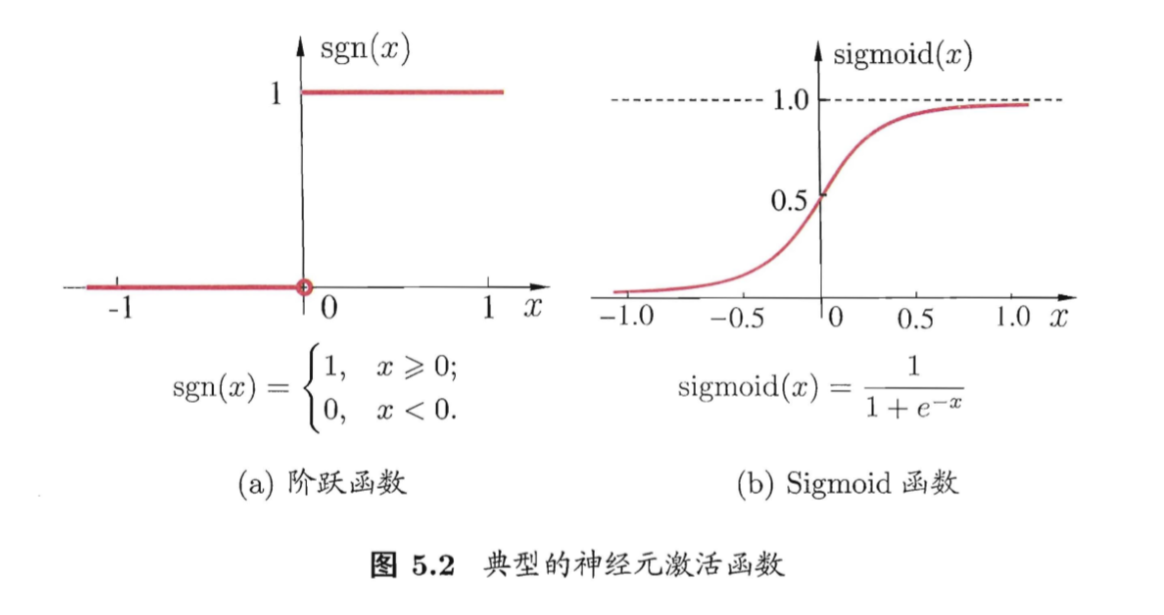


里面的几个参数根据图例我们可以看出，分别是输入、权重、阈值、输出；

这个简单的神经元的运转方式就是获得输入和权值，两个的乘积与阈值进行对比，再经过激活函数*f(-)*的作用下获得输出

激活函数：激活函数的作用就是在神经网络之中引入非线性的因素从而可以解决一些线性模型无法解决的问题，激活函数都是非线性的比如下面介绍的两种。

常见的激活函数包括：阶跃函数、sigmoid函数等机器学习书上列举出来了两种函数的图形。

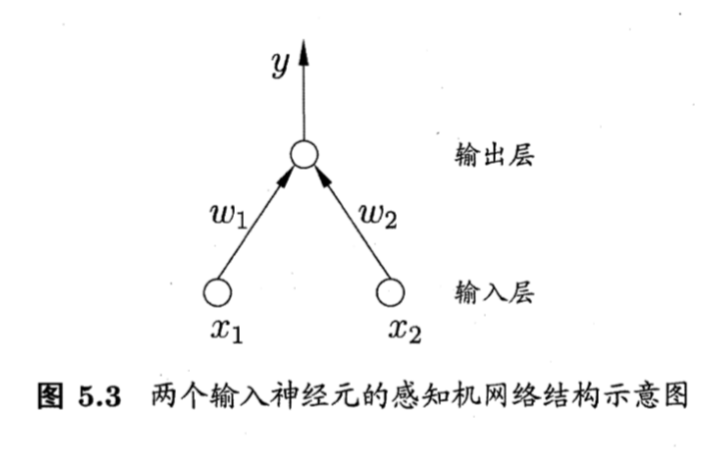


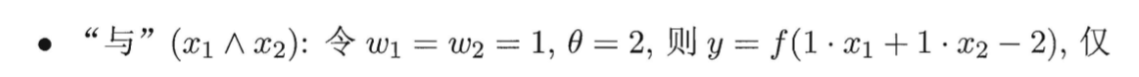
当多个神经元组合起来就形成了神经网络

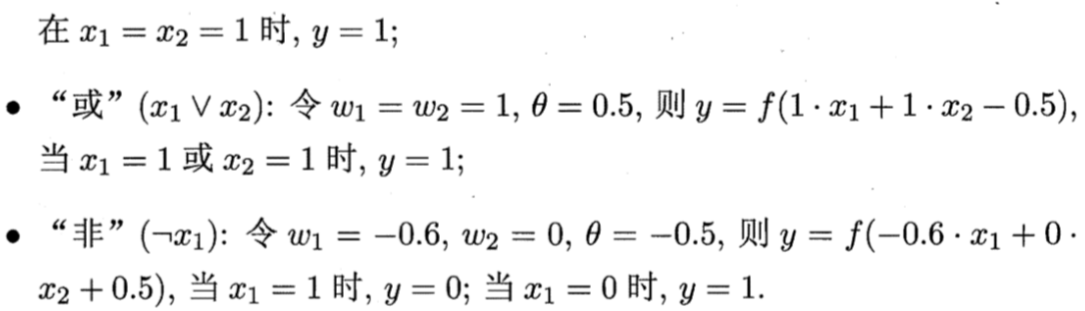
2.感知机与多层网络

先介绍一个简单的网络，甚至不能说是网络，因为组成这个模型的仅仅只有两个神经元模型，这个模型被称为感知机模型，两层模型分为输入层与输出层。只有输出神经元是M-P神经元，输入层仅有输入的作用，这个简单的感知机模型的作用非常有限，比如书中介绍的三种逻辑运算，与、或、非。

书上举了一个以阶跃函数为激活函数的感知机模型，



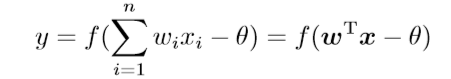




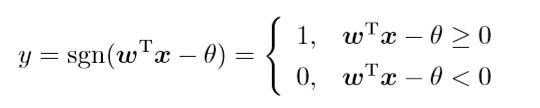
这个应该比较好理解，Xi是输入仅在全为1时与操作才会输出1

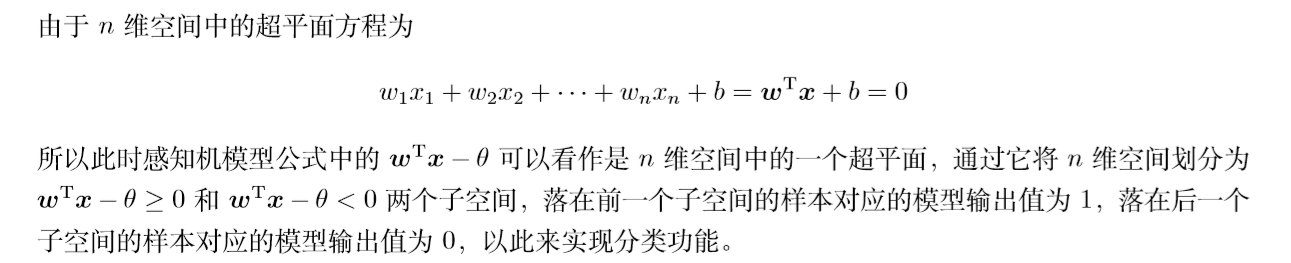
为了得到一个很好的感知机模型乃至是神经网络模型，那我们必须要学习到好的参数，这里指的参数就是权重和阈值。

一般的学习一个感知机模型，通过给定的训练集合，权重和阈值可以通过学习这个训练集来得到，例如我们想要一个分类操作的感知机模型，那么这个感知机模型的公式可表现为下面的式子，其中代表的是样本的一个属性是一个属于n维向量空间的特征向量，而则分别是n个权重的特征向量和阈值。

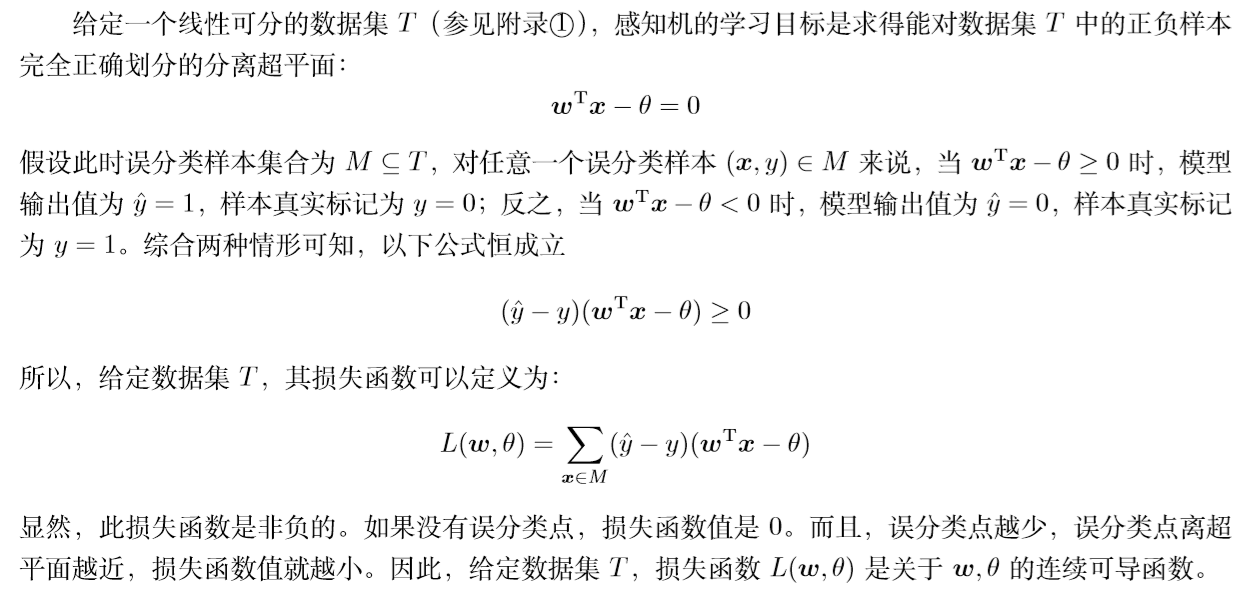


*f*是激活函数那么上式可表示为

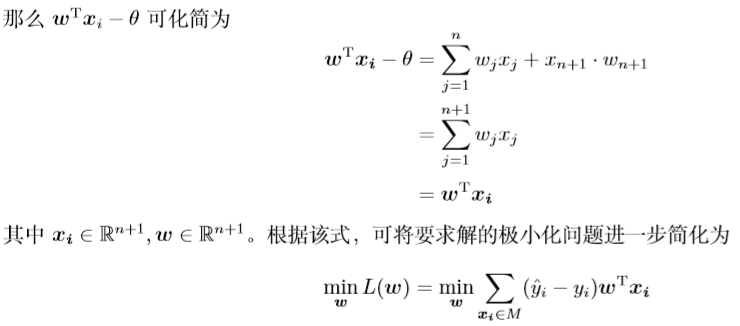




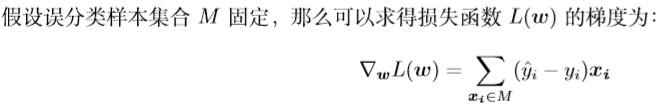
那么感知机是怎么学习的呢，我们运用一个函数来表示在训练过程中发生的误分类的损失，这个函数我们叫做损失函数，这个函数根据其表达式我们很容易看出，根据以下证明我们可以知道，损失函数越小，那么误差就越小，误分类点距离超平面就越近，我们训练的参数就越好。

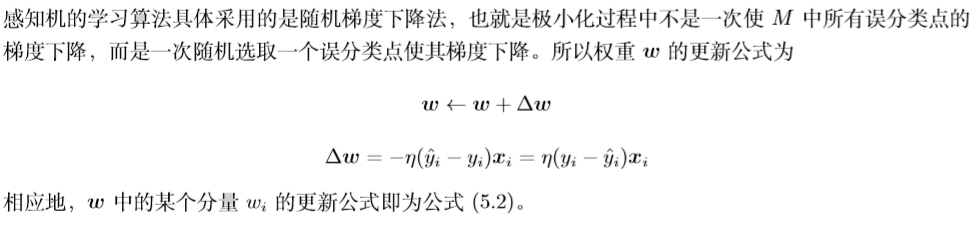


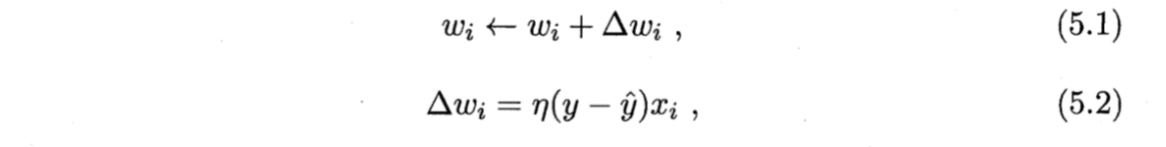
对于给定的训练集，我们要求得对应的使得损失函数尽可能的小为了方便计算我们可将看成一个输入恒为-1的哑节点，其权重为对应的我们可以将那么损失函数可以化简为



对求偏导可得损失函数的梯度，沿着梯度方向函数值下降最快，其中的为学习率通常在（0，1）内

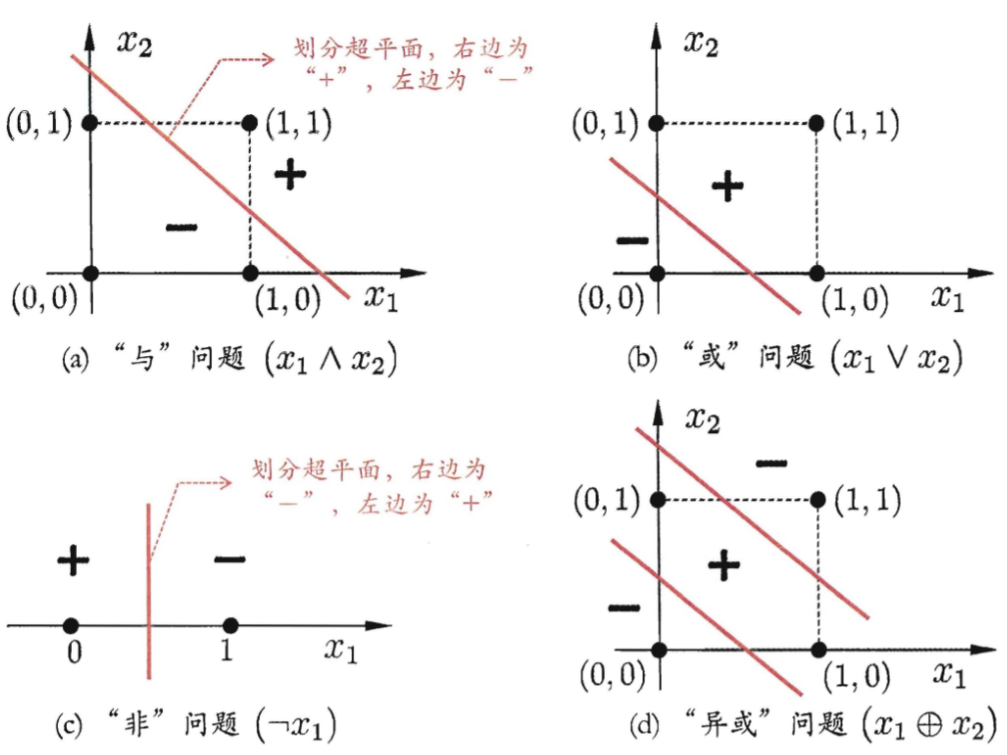


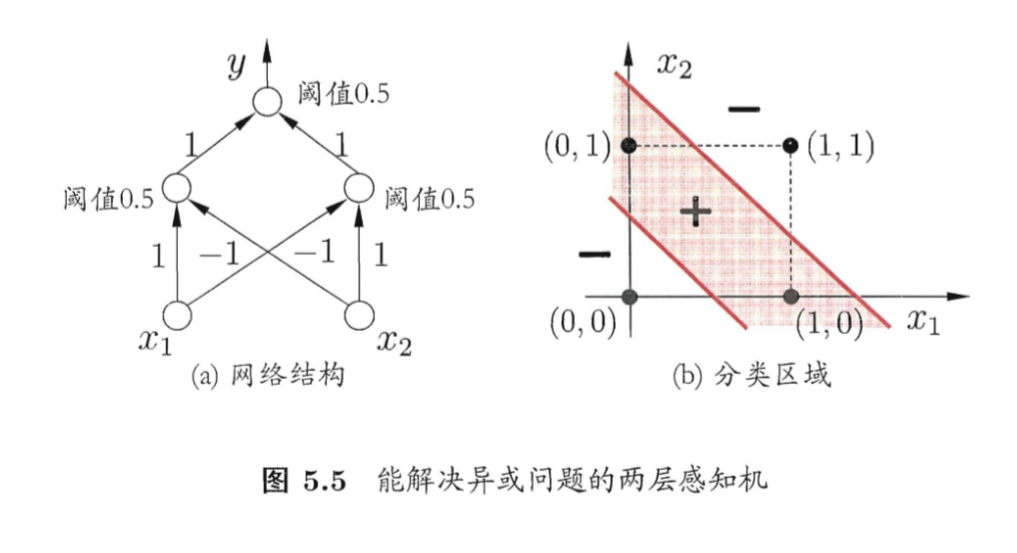




可以看出当输出与标记一样的时候权值是不变的，但是若有误差则对于误差的错误程度进行调整

对于与或非这种线性可分的问题只拥有一层功能神经元的感知机模型可以解决，能够找到一个超平面将不同的样本分开，如下图（a-c）所示



对于异或这种线性不可分的问题，一层感知机就不够看了，必须要用多层功能神经元，如两层感知机

这种两层感知机引出了一个隐层的概念，隐层就是输入层与输出层之间的一层神经元，与输出层一样都有激活函数，这个双层感知机实现异或也比较简单大家可以看书理解一下。

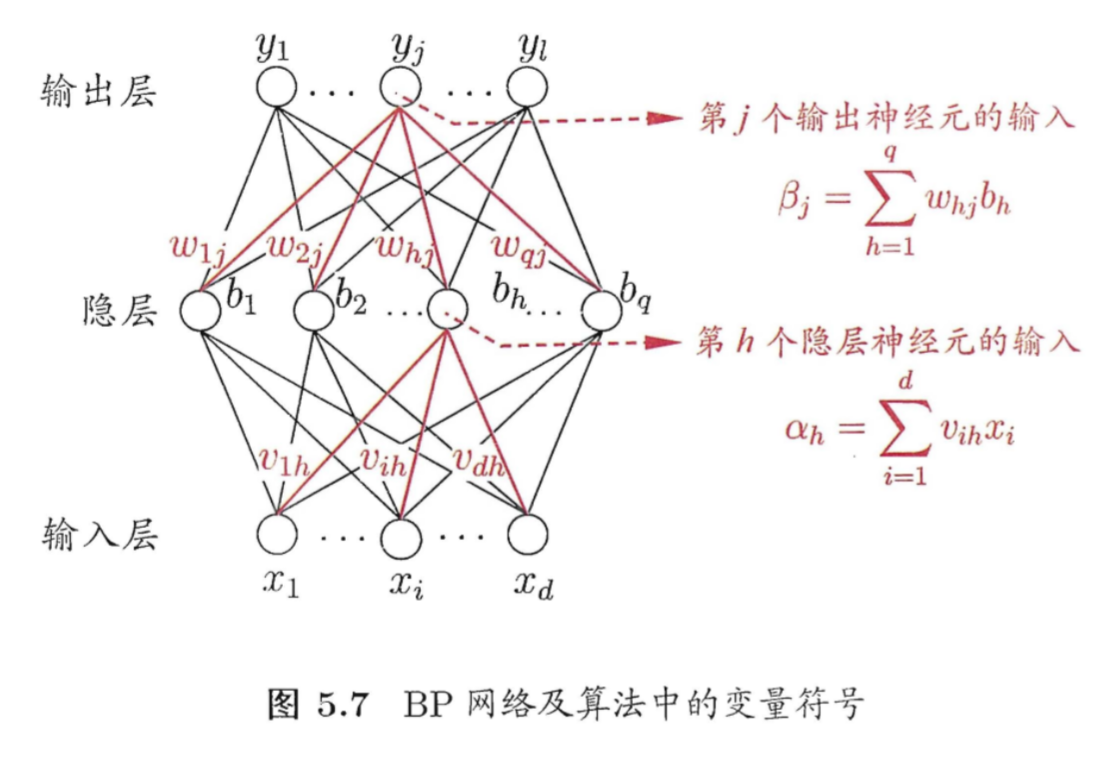
所以神经网络学习所学习到的就是神经元之间的连接权以及功能神经元的阈值。

下面就是介绍几种多层神经网络的学习算法：  
1.误差逆传播算法（BP算法）

BP算法是基于梯度下降公式对于损失函数cost（**x**）我们要求其最小值，如果能够找到一个序列x0，x1 ，x2 ，x3 ，x4….使得cost（**xt+1**）<cost（**xt**）那么那我我们肯定可以逐步收敛到最小值，那么根据泰勒公式cost（x+Δx）≈cost（x）+ΔxT▽cost（x）所以我们可以令Δx=-γ▽cost（x）那么ΔxT▽cost（x）将衡小于等于零，那么我们的变量更新公式也就随之而出了

多层神经网络的训练很复杂用简单的感知机学习规则是不够的，于是有人就提出了一种新的算法叫做BP算法，是训练多层神经网络的算法中比较杰出的一个例子。

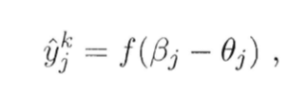
以一个多层前馈神经网络为例，多层前馈神经网络就是指的网络的拓扑结构中不存在环或者回路，如下图，由于是用的bp算法训练的网络我们把这个多层前馈神经网络叫做BP网络：



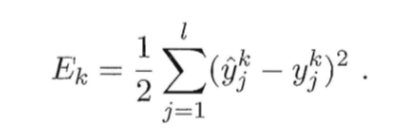
其中给定训练集D={(x1, y1),(x2,y2).....(Xm,Yn)},,,即输入示例由d个属性描述,输出l

维实值向量.为便于讨论,图5.7给出了一个拥有d个输入神经元、l个输出神经元、q个隐层神经元的多层前馈网络结构,其中输出层第j个神经元的阈值用表示，隐层第h个神经元的阈值用表示.输入层第i个神经元与隐层第h个神经元之间的连接权为,隐层第h个神经元与输出层第j个神经元之间的连接权为记隐层第h个神经元接收到的输入为,输出层第j个神经元接收到的输入为,其中为隐层第h个神经元的输出，激活函数均为sigmoid函数实质上是对率函数

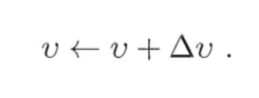
下式为输出的计算公式



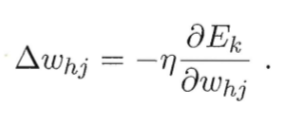
对于样本用均方误差来表示误差程度则其可表示为



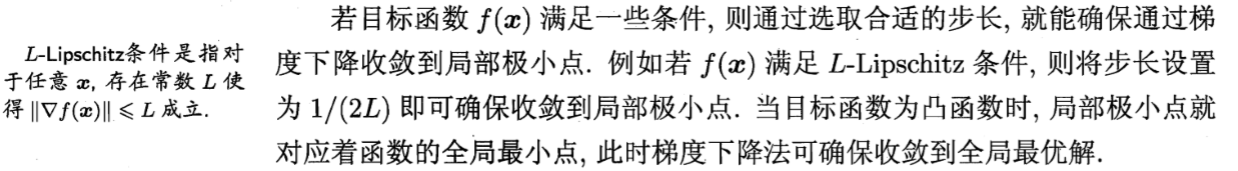
与感知机模型的权值更新公式一样对于多层神经网络其参数的更新也是一样的规则

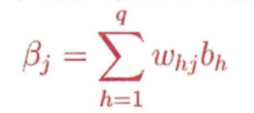
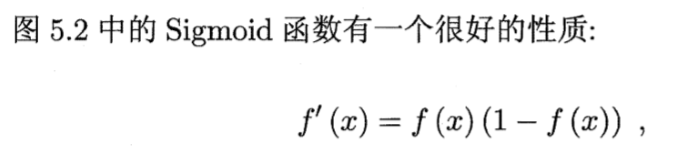
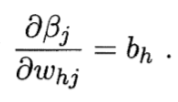
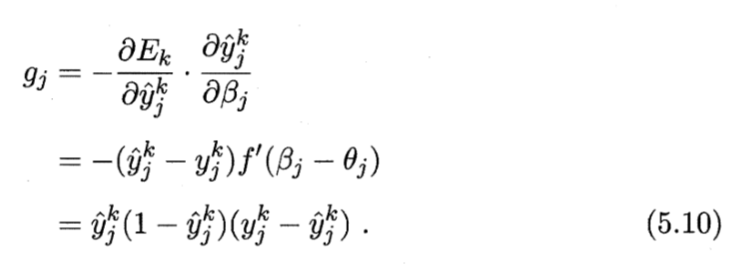


BP算法是基于梯度下降策略的，以调整目标的负梯度方向对目标进行调整，我们根据损失函数也就是之前的均方误差，我们的目标是让损失最小化也就是尽可能的小，根据梯度我们知道沿着梯度的反方向函数下降最快，因此我们可以得出下式



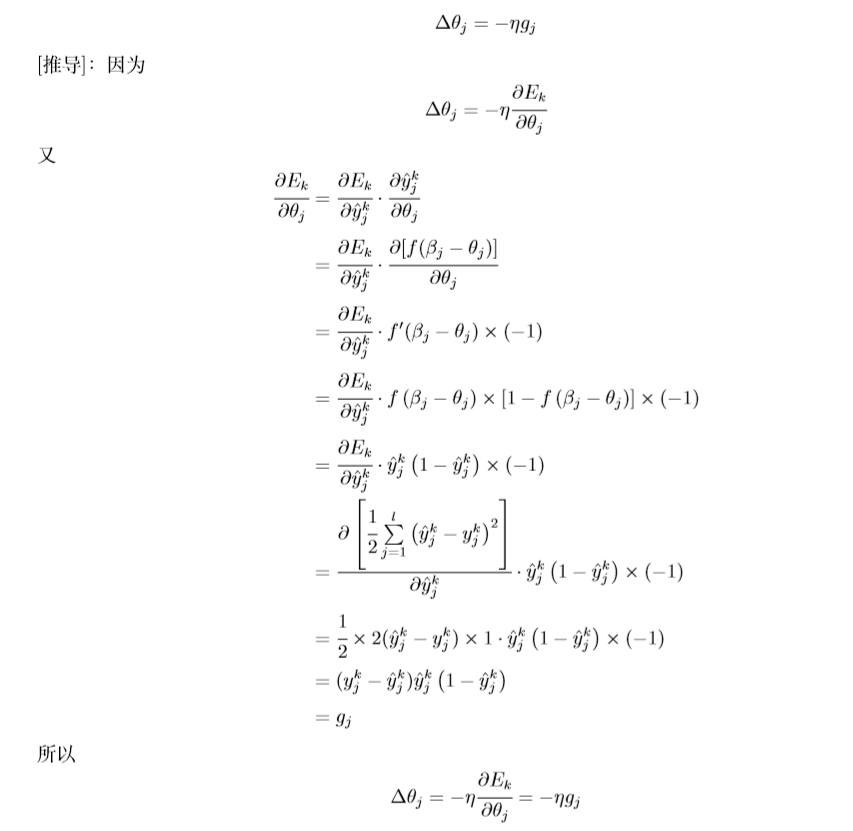
其中是学习率或者叫做步长，步长更好理解，可以这样理解我们要通过一步一步的逐步逼近最小值，而这个一步就是步长，我们走的步子大了可能会错过最小点也就是会走过，但是步子小了那我们走的时间就长了也许永远也到不了，所以一个合适的步长很重要，根据附录B中提到的梯度下降方法，



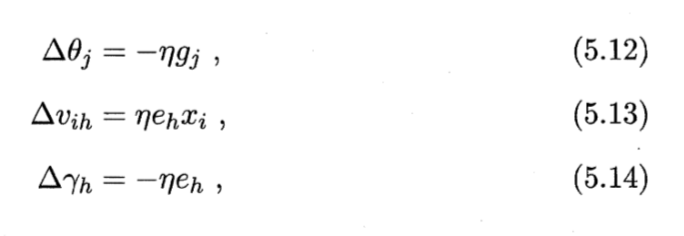
求解比较简单就是运用链式法则，由前面的可知道对求偏导可得出经过一系列的变换最终得到



具体的证明过程比较简单我就不再赘述了具体的过程与阈值更新公式一样

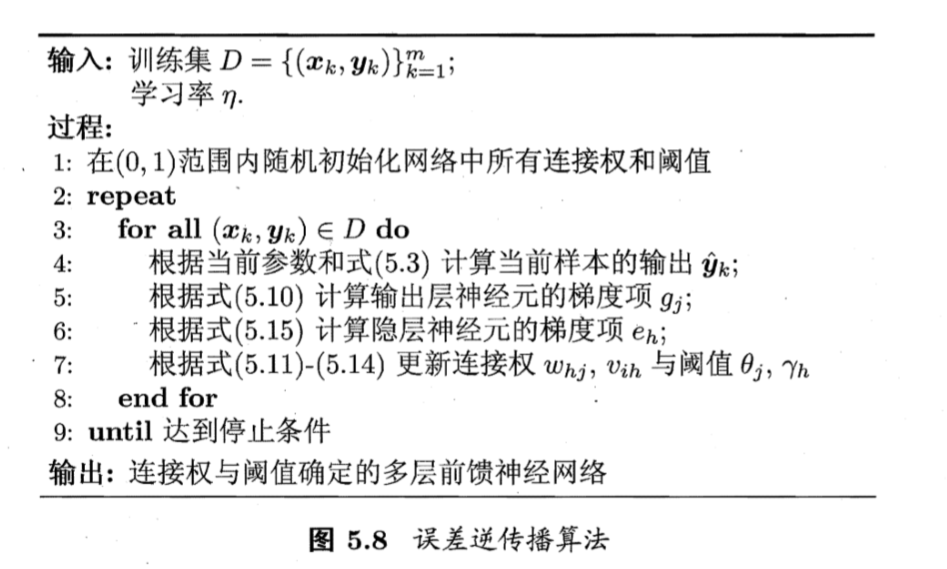


然后根据类似的过程也可证明



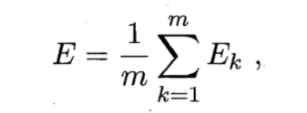
其中的分别是输出层和隐层的梯度项

下图为BP算法的工作流程，首先根据输入和输出计算误差（4-5）再将误差逆传播至隐层（6）然后来调整相应的连接权和阈值（7）整个过程一直迭代执行直到达到退出条件



BP算法的目的是为了最小化训练集D上的累积误差而不是仅仅最小化

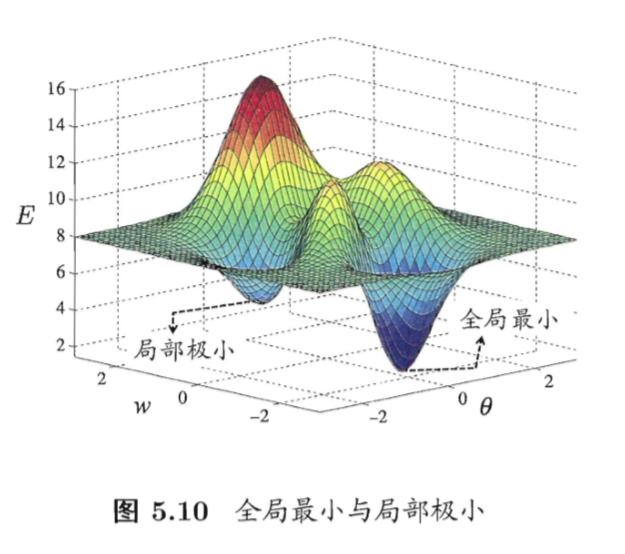
累计误差可以表示为



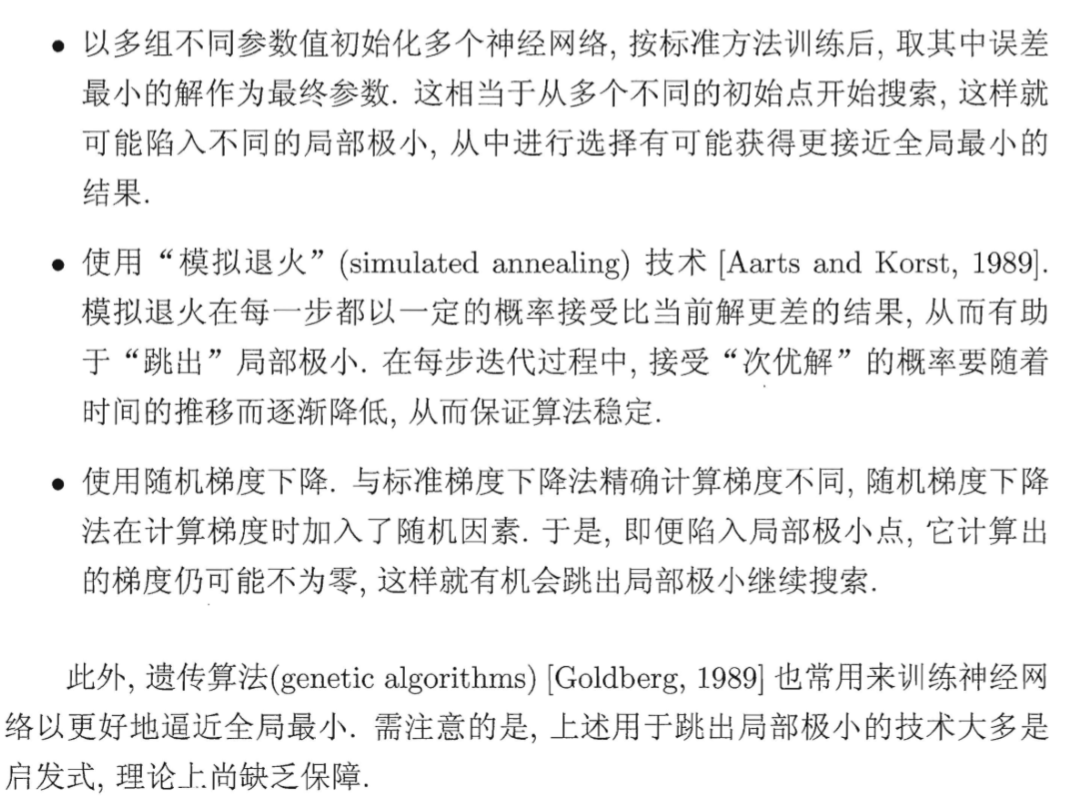
这是标准的BP算法是对每一个输入的样本都进行一次阈值和连接权的更新，还有一种累积BP算法与标准BP算法的区别就是累积BP算法是在整个训练集都已经输入完之后再对阈值和连接权进行调整不是用而是直接用*E*两种方法都常用根据不同的适应条件来进行选择。

2.全局最小和局部最小

全局最小和局部最小的概念很容易理解其实就是我们学函数的极值的时候学到的极小值和最小值，但是这两个东西不一定是在二维的直角坐标系里面了，可以到三维四维，极小的概念就是在某个点的邻域里面没有其它点比这个点更小那么这个点就是局部最小，当在整个参数空间中都没有比这个点更小的点的时候那么这个点就是全局最小的。



对于误差函数梯度为零的点是局部最小点，其中只有一个全局最小点，也就是说全局最小一定是局部最小，而局部最小不一定是全局最小，因为我们想要找到全局最小，所以当我们找到的是局部最小的时候就称我们陷入了局部最小，对于这种情况有几种策略可以解决，分别是：

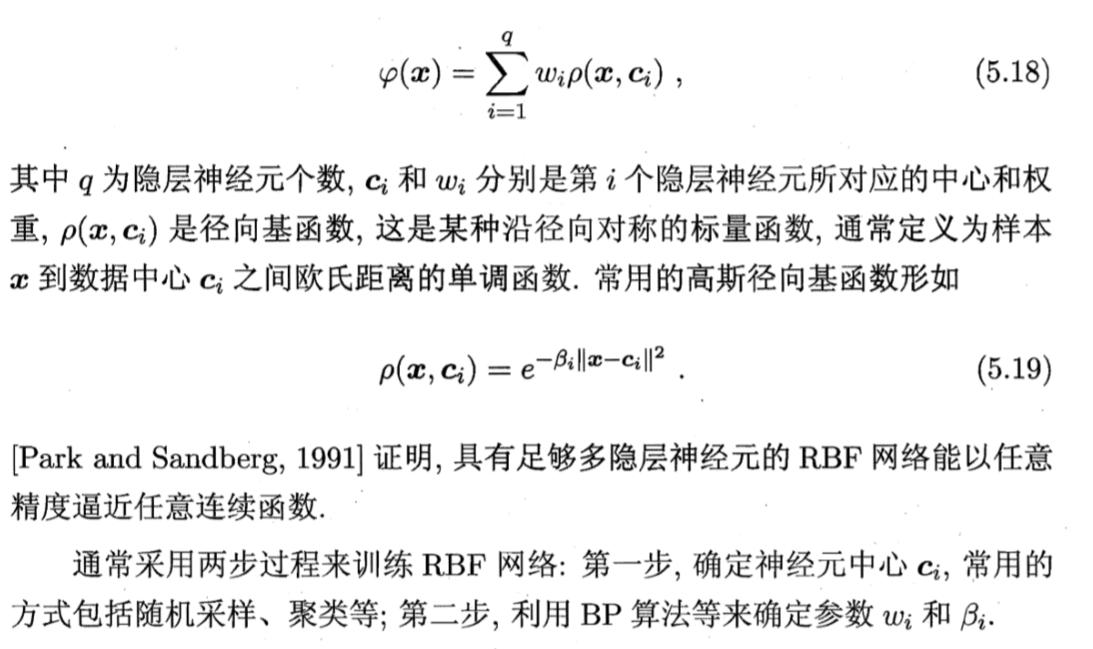


这几种解决办法是根据经验总结来的，无法保证一定能找到最小。

3.其它神经网络

（1）RBF网络

RBF神经网络与普通神经网络的区别在于，它是一个单隐层神经网络，并且隐层神经网络用的激活函数是一个径向基函数，而输出层则是是对隐层神经元输出的线性组合，可将其表示为：



（2）ART网络

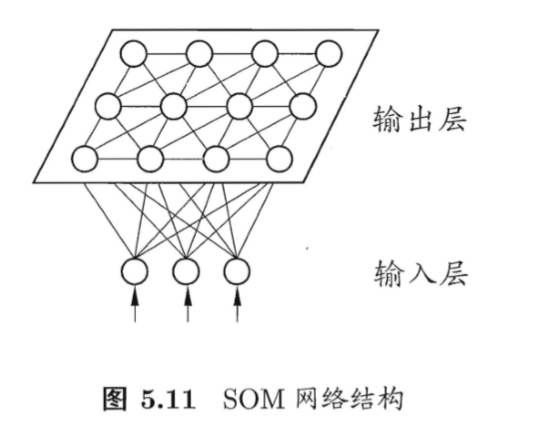
竞争型学习是神经网络中一种常用的无监督学习策略，ART网络是里面的重要代表之一，这种网络由比较层、识别层、识别阈值、和重置模块组成，阈值和重置模块构成.其中，比较层负责接收输入样本,并将其传递给识别层神经元.识别层每个神经元对应一个模式类,神经元数目可在训练过程中动态增长以增加新的模式类.

ART网络的实现模式就是，识别层在接受到比较层的输入后，识别层神经元之间进行竞争，竞争的简单方式是，计算输入向量与识别层神经元的对应模式类的代表向量之间的距离，距离小的就获胜，获胜的神经元就抑制其它神经元激活，若输入值与获胜神经元的代表向量的相似度大于识别阈值，则当前向量归入该代表向量所属类别同时网络连接权将更新，在接受到相似样本时该模式类会计算出更大的相似度，从而使该神经元更容易获胜，若相似度不大于识别阈值则重置模块会在识别层增设一个新的神经元，其代表向量就会设置为当前输入向量。

识别阈值高则分类细，阈值低分类就比较粗，ART网络的优点在于缓解了竞争性学习的可塑性稳定性窘境，可塑性是指学习新知识的能力，稳定性则是指在学习新知识的同时能够保持对旧知识的记忆，所以ART就有增量学习的能力

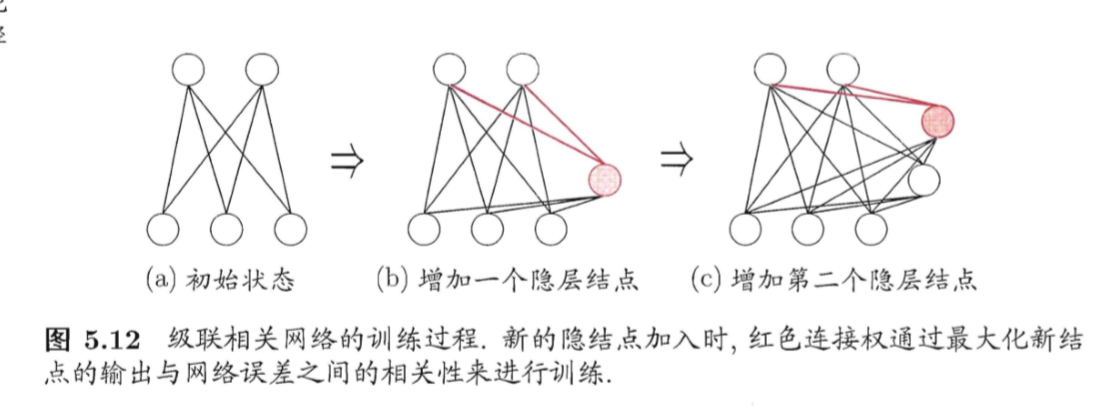
（3）SOM网络

SOM神经网络也是一种非监督的竞争性神经网络，这种神经网络的特点就是，它能将高维输入数据映射到低维空间(通常为二维),同时保持输入数据在高维空间的拓扑结构,即将高维空间中相似的样本点映射到网络输出层中的邻近神经元.



（4）级联相关网络

级联相关网络是结构自适应神经网络的一种，结构自适应网络与一般神经网络的区别在于，一般的神经网络都是先固定网络结构，然后通过训练样本来学习对应的连接权和阈值，而对于结构自适应神经网络来说，网络结构也是学习的目标之一

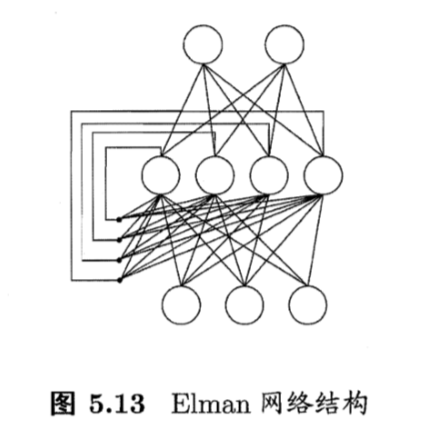


级联相关网络包括了级联和相关，级联就是建立层次连接的层级结构，在一开始的时候网络只有输入层与输出层，随着训练进行新的隐层神经元会逐渐加入建立起相应的网络结构，而相关是指最大化新结点的输出与网络误差之间的相关性。

级联相关网络的有点在于无需设置网络层数、隐层神经元数目，并且训练速度比较快，但是在训练数据少的时候容易陷入过拟合。

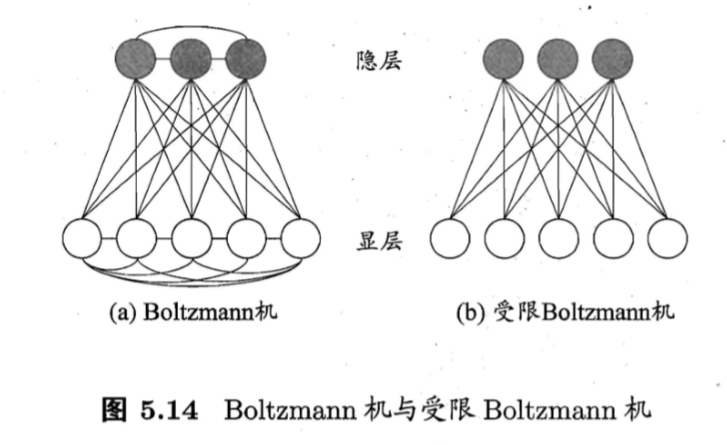
（5）Elman网络

与前馈神经网络不同，递归神经网络（RNN）允许网络中出现环型结构，这些环形结构的作用是用来将一些神经元的输出作为反馈的信号。这样的结构好处在于使得网络在t时刻的输出不仅与t时刻的输入有关，还会与t-1时刻的网络状况有关，这样就能处理与时间有关的动态变化。

Elman网络是递归神经网络的代表，其网络结构如下图：  


（6）Boltzmann机

玻尔兹曼机为网络状态定义一个能量，这个能量越小网络状态越好，玻尔兹曼机的训练就是基于这个能量函数，玻尔兹曼机只有两层显层和隐层，显层用于输入，隐层为数据的内在表达，



由于标准的玻尔兹曼机是一个全连接图，网络结构很复杂，难以用来解决现实任务，所以现实中经常采用的是受限玻尔兹曼，如上图将玻尔兹曼机简化为了二部图

4.深度学习

为了完成更为复杂的学习任务，简单的几层网络结构已经无法满足要求了，那么就需要提高模型的容量，模型的复杂度越高容量也就越高，那么对于神经网络增加模型复杂度最直接的办法就是增加隐层的数目，隐层数目越多连接权和阈值就更多，而且对比增加隐层神经元数目，增加隐层还增加了激活函数嵌套的层数，但是这时候出现的问题就是深度神经网络的训练问题，用之前的BP算法是无法完成训练的，因为误差在逆传播时会发生发散导致无法收敛到稳定状态。

无监督逐层训练便成了深度神经网络训练的方法，有效手段，其基本思想是每次训练一层隐结点,训练时将上一层隐结点的输出作为输入,而本层隐结点的输出作为下一层隐结点的输入,这称为“预训练"(pre-training);在预训练全部完成后，再对整个网络进行“微调”(ine-tuning)训练.例如，在深度信念网络(deep belief network,简称DBN) [Hinton, 2006]中每层都是一个受限Boltzmann机,即整个网络可视为若干个RBM堆叠而得，在使用无监督逐层训练时,首先训练第一层, 这是关于训练样本的RBM模型，可按标准的RBM训练;然后,将第一层预训练好的隐结点视为第二层的输入结点,对第二层进行预训练;....各层预训练完成后,再利用BP算法等对整个网络进行训练.

另外一种办法就是权共享在卷积神经网络之中用的比较多，卷积神经网络的的每一组神经元都是用相同的连接权，这样就大幅减少了需要训练的参数数目。

深度学习其实就是把输入的信号逐层进行加工，把初始的与输出目标之间联系不密切的输入表示，转化为与输出目标联系更紧密的表示，使得原来仅基于最后一层输出映射难以完成的任务成为可能，简而言之就是把低层的特征表示转换为高层的特征表示，用简单模型即可完成的分类等学习任务，所以深度学习又叫特征学习或表示学习。