其他常见神经网络

前言

这部分仅作了解即可，因为这里是机器学习而不是深度学习。我们仅深入讨论机器学习，而对于深度学习，我会在以后深度学习部分单开一个专题，这里仅作为了解。

这部分我直接照搬了西瓜书机器学习的内容，我尽量见缝插针作个人注释。

RBF 网络

RBF(Radial Basis Function, 径向基函数) 网络 [Broomhead and Lowe, 1988] 是一种单隐层前馈神经网络（理论上来说可使用多个隐层，但常见的 RBF 设置是单隐层），它使用径向基函数作为隐层神经元激活函数，而输出层则是对隐层神经元输出的线性组合。假定输入为 d 维向量 $\mathbf{x}$ 输出为实值，则 RBF 网络可表示为

$$\varphi(\mathbf{x}) = \sum_{i=1}^{q} w_i \rho(\mathbf{x}, \mathbf{c}_i), \quad (5.18)$$

其中 q 为隐层神经元个数，$\mathbf{c}_i$ 和 w_i 分别是第 i 个隐层神经元所对应的中心和权重，$\rho(\mathbf{x}, \mathbf{c}_i)$ 是径向基函数，这是某种沿径向对称的标量函数，通常定义为样本 x 到数据中心 c_i 之间的欧氏距离的单调函数。常用的高斯径向基函数形如

$$\rho(\mathbf{x}, \mathbf{c}_i) = e^{-\beta_i \|\mathbf{x}-\mathbf{c}_i\|}, \quad (5.19)$$

[Park and Sandberg, 1991] 证明，***具有足够多隐层神经元的 RBF 网络能以任意精度逼近任意连续函数***。

通常采用两步过程来训练 RBF 网络：第一步，*确定神经元中心 c_i，常用的方式包括随机采样、聚类等*；第二步，*利用 BP 算法等来确认参数 w_i 和 β_i*。

ART 网络

竞争型学习(competitive learning)是神经网络中一种常用的无监督学习策略，在使用该策略时，网络的输出神经元相互竞争，每一时刻仅有一个竞争获胜的神经元被激活，其他神经元的状态被抑制。这种机制亦称 ***“胜者通吃”(winner-take-all)原则***。

ART(Adaptive Resonance Theory, 自适应谐振理论) 网络 [Carpenter and Grossberg, 1987] 是竞争型学习的重要代表。该网络由比较层、识别层、识别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。（模式类可认为是某类别的“子类”）

在接收到比较层的输入信号后，识别层神经元之间相互竞争以产生获胜神经元。**竞争的最简单方式是，计算输入向量与每个识别层神经元所对应的模式类的代表向量之间的距离，距离最小者胜（上**

述“胜者通吃”原则的体现）。获胜神经元将向其他识别层神经元发送信号，抑制其激活。若输入向量与获胜神经元所对应的代表向量之间的相似度大于识别阈值，则当前输入样本将被归为该代表向量所属类别，同时，网络连接权将会更新，使得以后在接收到相似输入样本时该模式类会计算出更大的相似度，从而使该获胜神经元有更大可能获胜；若相似度不大于识别阈值，则重置模块将在识别层增设一个新的神经元，其代表向量就设置为当前输入向量。

显然，识别阈值对 ART 网络的性能有重要影响。*当识别阈值较高时，输入样本将会被分成比较多、比较精细的模式类，而如果识别阈值较低，则会产生比较少、比较粗略的模式类。*

ART 比较好地缓解了竞争型学习中的“可塑性-稳定性窘境”(stabilityplasticity dilemma)，*可塑性是指神经网络要有学习新知识的能力，而稳定性则是指神经网络在学习新知识时要保持对旧知识的记忆。*这就使得 ART 网络具有一个很重要的优点：*可进行增量学习(incremental learning)或在线学习(online learning)。*

注：增量学习是指在学得模型后，再接收到训练样例时，*仅需根据新样例对模型进行更新，不必重新训练整个模型，并且先前学得的有效信息不会被“冲掉”*；在线学习是指每获得一个新样本就进行一次模型更新。显然，在线学习是增量学习的特例，而增量学习可视为“批模式”(batch-mode)的在线学习。

早期的 ART 网络只能处理布尔型输入数据，此后 *ART 发展成了一个算法族，包括能处理实值输入的 ART2 网络、结合模糊处理的 FuzzyART 网络，以及可进行监督学习的 ARTMAP 网络等。*

SOM 网络

*SOM(Self-Organizing Map, 自组织映射) 网络 [Kohonen, 1982] 是一种竞争学习型的无监督神经网络,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**。*

如👇图所示，SOM 网络中的输出层神经元以矩阵方式排列在二维空间中，每个神经元都拥有一个权向量，网络在接收输入向量后，将会确定输出层获胜神经元，它决定了该输入向量在低维空间中的位置。**SOM 的训练目标就是为每个输出层神经元找到合适的权向量，以达到保持拓扑结构的目的。**

SOM 的训练过程很简单：**在接收到一个训练样本后，每个输出层神经元会计算该样本与自身携带的权向量之间的距离，距离最近的神经元成为竞争获胜者，称为*最佳匹配单元(best matching unit)*。然后，最佳匹配单元及其邻近神经元的权向量将被调整，以使得这些权向量与当前输入样本的距离缩小。**这个过程不断迭代，直至收敛。

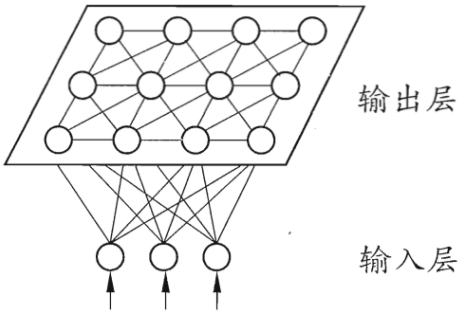

图 5.11 SOM 网络结构

级联相关网络

一般的神经网络模型通常假定网络结构是事先固定的，训练的目的是利用训练样本来确定合适的连接权、阈值等参数。与此不同，*结构自适应网络则将网络结构也当作学习的目标之一，并希望能在训练过程中找到最符合数据特点的网络结构（结构自适应神经网络亦称“构造性 construc-tive 神经网络”）。级联相关(Cascade-Correlation) 网络 [Fahlman and Lcbiere,1990] 是结构自适应网络的重要代表*。

注：前面介绍的 ART 网络由于隐层神经元数目可在训练过程中增长，因此也是一种结构自适应神经网络。

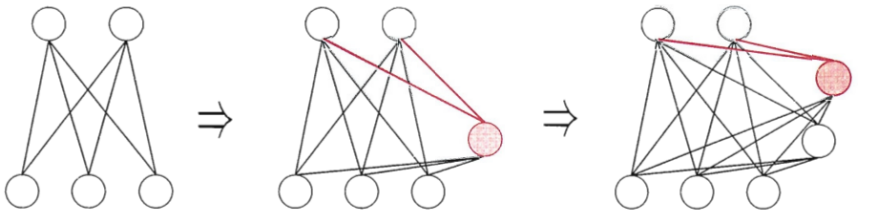

图 5.12 级联相关网络的训练过程. 新的隐结点加入时, 红色连接权通过最大化新结点的输出与网络误差之间的相关性来进行训练.

级联相关网络有两个主要成分：“级联”和“相关”。级联是指建立层次连接的层级结构。在开始训练时，网络只有输入层和输出层，处于最小拓扑结构；*随着训练的进行，如图👆所示，新的隐层神经元逐渐加入，从而创建起层级结构*。当新的隐层神经元加入时，*其输入端连接权值是冻结固定的。相关是指通过最大化新神经元的输出与网络误差之间的相关性(correlation)来训练相关的参数*。

与一般的前馈神经网络相比，级联相关网络无需设置网络层数、隐层神经元数目，且训练速度较快，但其在数据较小时易陷入过拟合。

Elman 网络（艾尔曼网络）

与前馈神经网络不同，*“递归神经网络”(recurrent neural networks, 亦称recursive neural networks) 允许网络中出现环形结构，从而可让一些神经元的输出反馈回来作为输入信号*。这样的结构与信息反馈过程，使得网络在时刻的输出状态不仅与时刻的输入有关，还与 $t-1$ 时刻的网络状态有关，从而能处理与时间有关的动态变化。

Elman 网络 [Elman, 1990] 是最常用的递归神经网络之一，其结构如👇图所示，它的结构与多层前馈网络很相似，但隐层神经元的输出被反馈回来，与下一时刻输入层神经元提供的信号一起，作为隐层神经元在下一时刻的输入。隐层神经元通常采用 Sigmoid 激活函数，而网络的训练则常通过推广的 BP 算法进行 [Pineda, 1987]。

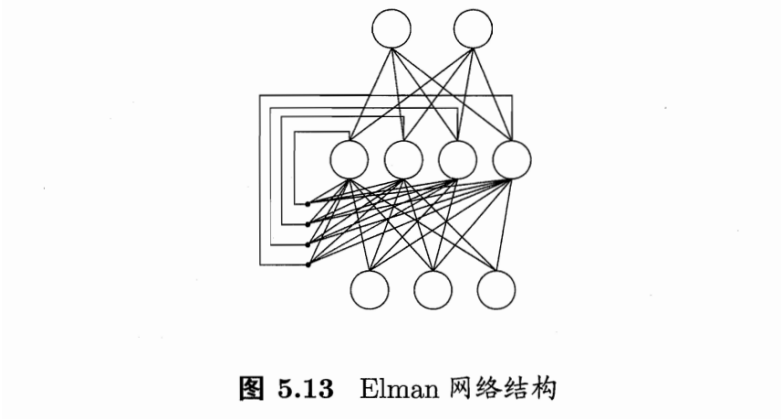

图 5.13 Elman 网络结构

Boltzmann 机（玻尔兹曼机，隐变量无向图模型）

神经网络中有一类模型是为网络状态定义一个“能量”(energy)，能量最小化时网络能达到理想状态，而网络的训练就是在最小化这个能量函数。Boltzmann 机 [Ackley et al., 1985] 就*是一种“基于能量的模型”(energy-based model)，常见结构如下图 a 所示（可以看出，Boltzmann 机是一种递归神经网络），其神经元分为两层：显层与隐层*。显层用于表示数据的输入与输出，隐层则被理解为数据的内在表达。*Boltzmann 机中的神经元都是布尔型的，即只能取 0、1 两种状态，状态 1 表示激活，状态 0 表示抑制*。令向量 $s \subseteq \{0,1\}^n$ 表示 n 个神经元的状态，w_{ij} 表示神经元 i 与 j 之间的连接权，θ 表示神经元 i 的阈值，则状态向量 s 所对应的 Boltzmann 机能量定义为

$$E(s) = -\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} w_{ij}s_i s_j - \sum_{i=1}^{n}\theta_i s_i, \quad (5.20)$$

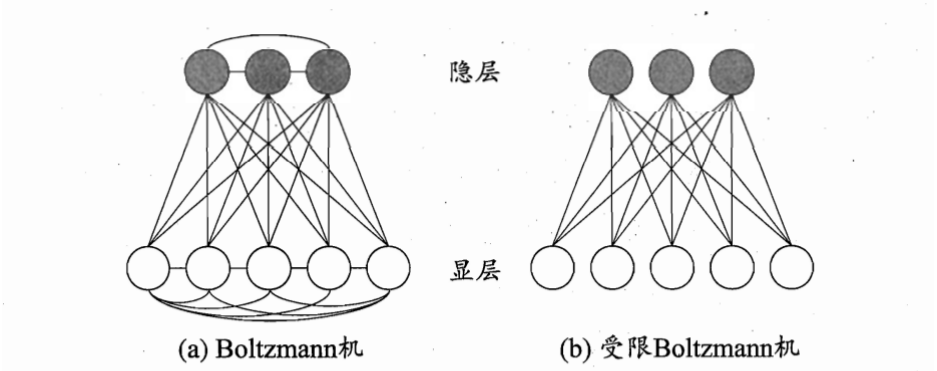

图 5.14 Boltzmann 机与受限 Boltzmann 机

若网络中的神经元以任意不依赖于输入值的顺序进行更新，则网络最终将达到 Boltzmann 分布（Boltzmann 分布亦称“平衡态equilibrium”或“平稳分布 stationary distribution”），此时状态向量 s 出现的概率将仅由其能量与所有可能状态向量的能量确定:

$$P(s)=\frac{e^{-E(s)}}{\sum_t e^{-E(t)}}, \quad (5.21)$$

Boltzmann 机的训练过程就是**将每个训练样本视为一个状态向量，使其出现的概率尽可能大。标准的 Boltzmann 机是一个全连接图，训练网络的复杂度很高，这使其难以用于解决现实任务。现实中常采用受限 Boltzmann 机(Restricted Boltzmann Machine, 简称 RBM)。如👆图 b 所示，受限 Boltzmann 机仅保留显层与隐层之间的连接，从而将 Boltzmann 机结构由完全图简化为二部图**。

受限 Boltzmann 机常用“对比散度”(Contrastive Divergence,简称 CD)算法 [Hinton, 2010] 来进行训练。假定网络中有 d 个显层神经元和 q 个隐层神经元，令 v 和 h 分别表示显层与隐层的状态向量，则由于同一层内不存在连接，有

$$P(\mathbf{v}|\mathbf{h})=\prod_{i=1}^{d} P(v_i|\mathbf{h}), \quad (5.22)$$

$$P(\mathbf{h}|\mathbf{v})=\prod_{j=1}^{q} P(h_j|\mathbf{v}), \quad (5.23)$$

CD 算法对每个训练样本 v *，先根据式(5.23)计算出隐层神经元状态的概率分布，然后根据这个概率分布采样得到* h *；此后，类似地根据式(5.22)从* $\mathbf{h}$ *产生* $\mathbf{v}'$ *，再从* $\mathbf{v}'$ *产生* $\mathbf{h}'$ *；连接权的更新公式为*

$$\Delta w=\eta\left(\mathbf{v}\mathbf{h}^{\top}-\mathbf{v}'\mathbf{h}'^{\top}\right)$$