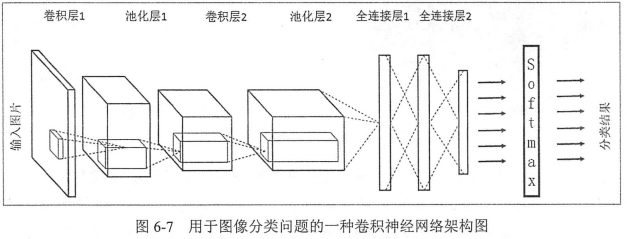
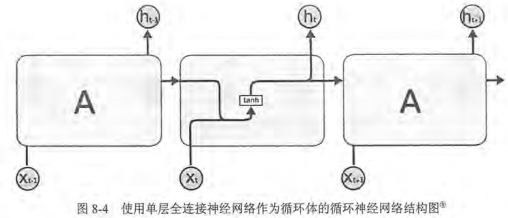
1. 卷积神经网络（CNN）

主要用于分类问题中。

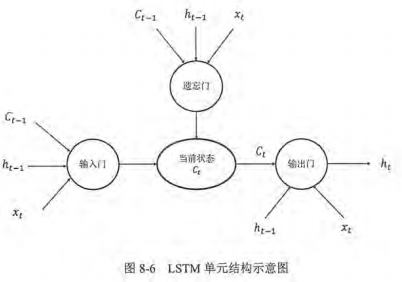


1. 循环神经网络

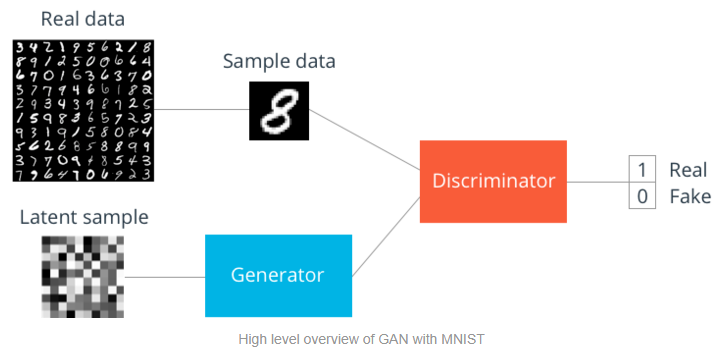
适用于与时间序列相关的问题。

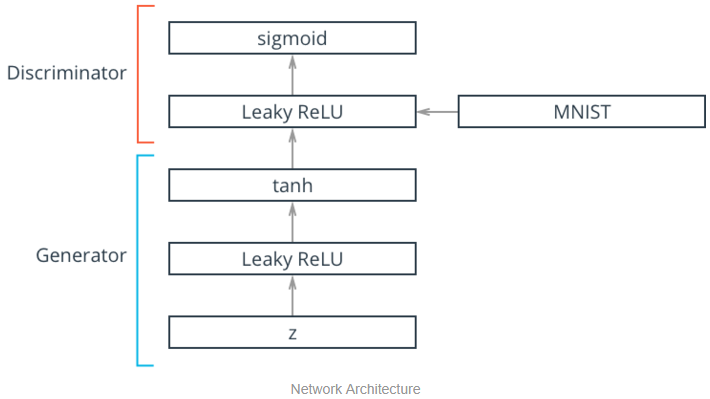


1. 长短时记忆网络（LTSM）



1. 生成式对抗网络（GAN）





大概流程：

1. 噪声矩阵通过生成器生成虚假图片数据
2. 辨别器（D）分别对虚假图片与真实图片计算损失函数，令假图趋近与0，真图趋近与1.
3. 生成器（G）计算损失函数
4. 开始训练
5. Softmax

用于优化分类结果，将神经网络前向传播得到的结果变成概率分布。



1. 卷积层/过滤器（filter）

将指定长宽（一般为3\*3或5\*5）的矩阵处理为指定深度的单位节点矩阵。（1\*1）

1. 池化层

缩小矩阵尺寸，减少最后全连接层中的参数，加快计算速度防止过拟合，有最大池化层和平均池化层两种。

1. 批规范化层（BN）

通过mini-batch来对相应的activation做规范化操作，使得结果（输出信号各个维度）的均值为0，方差为1，用于改善Covariate Shift问题。

1. 激活函数和偏置项（bias）

将神经网络转换为非线性结构。

1. 监督学习与无监督学习的区别：

监督学习(Supervised Learning)：就是高考前所做的练习题是有标准答案的。

非监督学习(Unsupervised Learning)：所做的练习题没有标准答案。

1. 交叉熵

计算预测的概率分布与真实答案的概率分布的距离。



1. 分类问题与回归问题

分类问题：结果为离散型数据，如数字识别、颜色分类等（损失函数：交叉熵）；

回归问题：结果为连续性数据，如价格等（损失函数：均方误差）；

1. 反向传播算法

根据定义好的损失函数优化神经网络的参数取值，使损失函数更小（如梯度下降算法）。

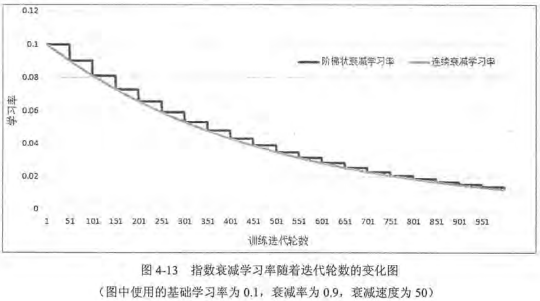
1. 学习率

控制训练时参数每次更新的幅度，如果过高容易造成过拟合，如果太低会增加学习时间。

1. 指数衰减法

动态控制学习率，初期使用较大的学习率，后期逐渐减小学习率。





1. 正则化（regularization）

在损失函数中添加刻画模型复杂程度的指标，用于避免过拟合问题。

1. 滑动平均模型

使模型在测试数据上更加健壮。

1. 迁移学习

将训练好的模型通过调整使其适应与一个新的模型。

1. 协变量转变（covariate shift）

在机器学习中，我们通常会做一种假设：训练样本独立同分布(iid)且训练样本与测试样本分布一致，如果真实数据符合这个假设则模型效果可能会不错，反之亦然，这个在学术上叫Covariate Shift，所以从样本（外部）的角度说，对于神经网络也是一样的道理。从结构（内部）的角度说，由于神经网络由多层组成，样本在层与层之间边提特征边往前传播，如果每层的输入分布不一致，那么势必造成要么模型效果不好，要么学习速度较慢，学术上这个叫InternalCovariate Shift。