1. 隐马尔可夫模型

用于生成观测序列，属于生成模型

**10.1 隐马尔可夫模型的基本概念**

10.1.1隐马尔可夫模型的定义

**隐马尔可夫模型**：隐马尔可夫模型是**关于时序**的概率模型，描述由一个隐藏的马尔可夫链随机生成不可观测的状态随机序列，再由各个状态生成一个观测从而产生观测随机序列的过程。隐藏的马尔可夫链随机生成的状态的序列，成为**状态序列**。每个状态生成一个观测，而由此产生的观测的随机序列，称为**观测序列**。序列的每一个位置又可以看作一个时刻。

隐马尔可夫模型的决定因素：**初始概率分布**、**状态转移概率分布**、**观测概率分布**。其中初始概率分布由初始状态概率向量Π来表示，Π是时刻t=1时处于状态qi的概率。状态转移概率矩阵A中的每一项aij描述了时刻t处于状态qi的条件下在时刻t+1转移到状态qj的概率。而观测概率矩阵B的每一项bj(k)描述了在时刻t处于状态qj的条件下生成观测vk的概率。

隐马尔可夫模型背后的两个基本假设：

①**齐次马尔可夫性假设**：假设隐藏的马尔可夫链在任意时刻t的状态只依赖于其前一时刻的状态，与其他时刻的状态及观测无关，也与时刻t无关。②**观测独立性假设**：假设任意时刻的观测只依赖于该时刻的马尔可夫链的状态，与其它观测及状态无关。

之后举用了盒子和球模型的例子，解释了一下这几个参量在实际场景中分别是什么意思。

10.1.2 观测序列的生成过程

其实就是根据状态转移的概率不断的产生下一个状态的过程，一直生成到有期望个数个观测状态为止。

10.1.3 隐马尔可夫模型的3个问题

有概率计算问题、学习问题、预测问题三个基本问题。

**10.2 概率计算算法**

10.2.1 直接计算法

核心是列举说有可能的长度为T的状态I，求各个状态序列I与观测序列O的联合概率，然后对所有可能的状态求和。由P(I|λ)与P(O,I|λ)计算出联合概率P(O,I|λ)，然后再对I求和，得到P(O|λ)

10.2.2 前向算法

**前向概率：**给定隐马尔可夫模型λ，定义到t时刻部分观察序列为o1,o2,...,ot且状态为qi的概率为前向概率at(i)=P(o1,o2,...,ot,it=qi|λ)

算法的核心是通过递推求得前向概率，迭代着得到最终的观测概率序列。每一步迭代中求得at+1(i)是由上一步为状态j的概率(at(j))乘上次状态j转到状态i的概率aji对所有上一次可能的状态j从1到N求和。求和得到的是在这一步为状态i的概率，求和后在与在这一步状态i下出现观察ot+1的概率，就可以得到这一步的前向概率at+1(i)。

而我们所需要的P(O|λ)只需要在t=T的那一刻对i求和即可。

可以看到，采用这个算法极大程度上降低了时间复杂度。

10.2.3 后向算法

**后向概率**：给定马尔可夫模型λ，定义在时刻t状态为qi的条件下，从t+1到T的部分观测序列ot+1,ot+2,...,oT的概率为后向概率，记作Bt(i)=P(ot+1,ot+2,...,oT|it=qi,λ)

算法的核心是通过递推求得后向概率，迭代着得到最终的观测概率序列。每一步迭代中求得Bt(i)是由这一步转移到下以状态j的概率、下一状态按照所求的观测序列得到ot+1的概率和Bt+1(j)相乘，再对所有可能的j从1到N累和求得的。

10.2.4 一些概率与期望值的计算

给出了一些用到前向/后向概率计算的概率与期望值。

这些概率包括：给定模型λ和观测O，在时刻t处于状态qi的概率，这个概率计算中主要是根据前向概率和后向概率的定义得出了等式at(i)Bt(i)=P(it=qi,O|λ)。给定模型λ和观测O，在时刻t处于状态qi且在时刻t+!处于qj的概率，属于上一个概率的扩充情况。且根据以上的两个概率，可以得到一些有用的期望。

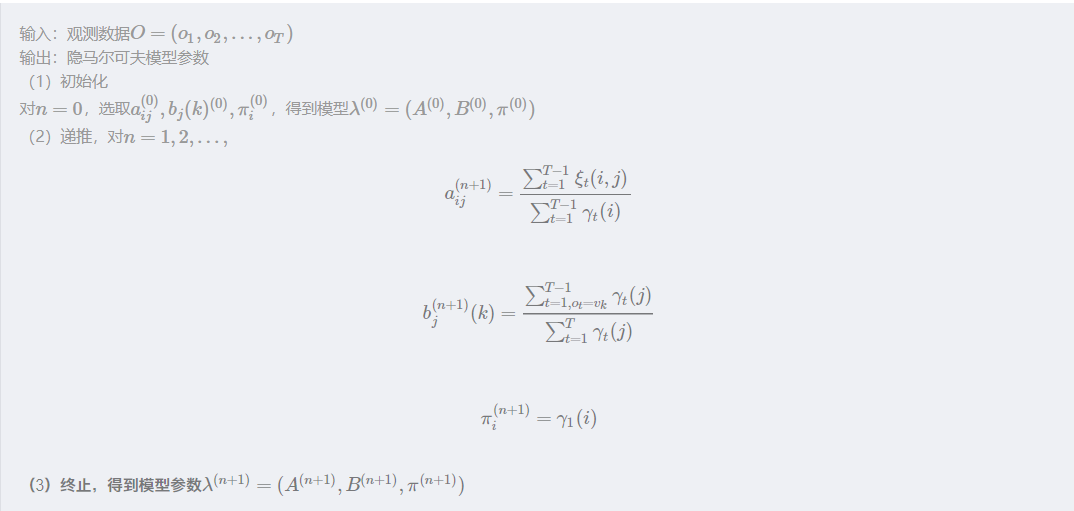
**10.3 学习算法**

10.3.1 监督学习算法

转移状态aij用i转移到j的状态个数Aij比上从状态i出发做出的转换的总个数求得。观测概率bj(k)也由类似的方法估算而成。初始状态则直接统计频率。

10.3.2 Baum-Welch算法

本质是采用EM算法处理隐马尔可夫模型中的参数估算问题。



**10.4 预测算法**

10.4.1 近似算法

近似算法的核心：在每个时刻t都选择**最有可能出现的状态**，从而得到最优的状态序列I\*。个人感觉这个算法近似一种贪婪算法的思想，可以确定每一步都是最优的选择，但不能确定这个序列从在整体上是最优的。

10.4.2 维特比算法

维特比算法的基础可以概括为下面三点：

1、如果概率最大的路径经过篱笆网络的某点，则从开始点到该点的子路径也一定是从开始到该点路径中概率最大的。

2、假定第i时刻有k个状态，从开始到i时刻的k个状态有k条最短路径，而最终的最短路径必然经过其中的一条。

3、根据上述性质，在计算第i+1状态的最短路径时，只需要考虑从开始到当前的k个状态值的最短路径和当前状态值到第i+1状态值的最短路径即可，如求t=3时的最短路径，等于求t=2时的所有状态结点x2i的最短路径加上t=2到t=3的各节点的最短路径。

简单来说就是：从开始状态之后每走一步，就记录下到达该状态的所有路径的概率最大值，然后以此最大值为基准继续向后推进。显然，如果这个最大值都不能使该状态成为最大似然状态路径上的结点的话，那些小于它的概率值（以及对应的路径）就更没有可能了。