

# 随机模型

## 原理部分——建模

模型分为确定性模型和随机模型。而随机模型主要分为以下三类

1. 概率模型
2. 马氏链模型
3. 回归模型(单独总结)

### 概率模型

主要用到概率的运算，概率分布期望方差等基本知识。以轧刀中的浪费为例。

#### 轧钢中的浪费

问题为：已知成品材的规定长度为  $l$  以及粗轧后的钢材均方差  $\sigma$ ，确定粗轧后的钢材长度均值为  $m$ ，使得当轧机以  $m$  粗轧，再进行精轧得到成品时浪费最少。

最简单的思路便是把两部分的损失加起来：

$$W = \int_l^\infty (x - l)p(x)dx + \int_{-\infty}^l xp(x)dx$$

利用概率密度函数的性质（积分为1），期望定义，上式可改为：

$$W = m - l \int_l^\infty p(x)dx = m - lP \quad (*)$$

但事实这并不合理，应该考虑没得到一根钢材平均浪费了多少而不是每轧一个浪费了多少。

粗轧  $N$  根浪费了  $mN - lPN$  长度，但是只有  $PN$  成了成品，所以

$$J = \frac{mN - lPN}{PN} = \frac{m}{P} - l$$

最小化  $J$  即可，其中

$$P(m) = \int_l^\infty p(x)dx$$
$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-m)^2}{2\sigma^2}}$$

## 马氏链模型

在一个离散时间集合  $T = \{0, 1, 2, \dots\}$  和一个有限或可列无穷的状态空间  $S = \{1, 2, \dots\}$  上, 一个随机过程在任一时刻从一个状态以一定的概率向其他状态转移 (或保持原状态不变)。记  $X_n$  为时刻  $n$  时时刻过程所处的状态, 假定:

1. 在时刻0, 过程所处的状态  $X_0$  是  $S$  上的一个随机变量;
2. 在任一时刻  $n$ , 给定  $X_0, X_1, \dots, X_n$  时,  $X_{n+1}$  的**条件分布**只与  $X_n$  有关, 而与  $X_0, X_1, \dots, X_{n-1}$  无关。

满足上述条件的**随机过程**为马尔可夫链, 简称马氏链。通常有以下两种类型:

### 1. 正则链: 从任意状态出发都可以经有限次转移到达任意状态

- 马氏链是正则链的充要条件是: 存在正整数  $N$ , 使得  $P^N > 0$ 。即每个元素都大于零
- 正则链存在唯一的极限状态概率  $w = (w_1, w_2, \dots, w_n)$ , 使得  $a_n \rightarrow w$ 。稳态状态与初始状态无关, 满足  $wP = w, \sum w_i = 1$
- 从状态  $i$  经  $n$  次转移第一次到  $j$  的概率称为首达概率, 记作  $f_{ij}(n)$ , 则转移的平均次数:

$$\mu_{ij} = \sum_{n=1}^{\infty} n f_{ij}(n)$$

特别的, 针对正则链,  $\mu_{ii} = 1/w_i$

### 2. 吸收链: 转移概率 $p_{ii} = 1$ 的状态 $i$ 称为吸收状态。如果某一个马氏链包含只招一个吸收状态并且从非吸收状态可经有限次转移达到吸收状态, 则称之为吸收链

- 吸收链的转移矩阵可以化为简单的标准形式 ( $r$  个吸收态全部提前)

$$P = \begin{bmatrix} I_{r \times r} & O \\ R & Q \end{bmatrix}$$

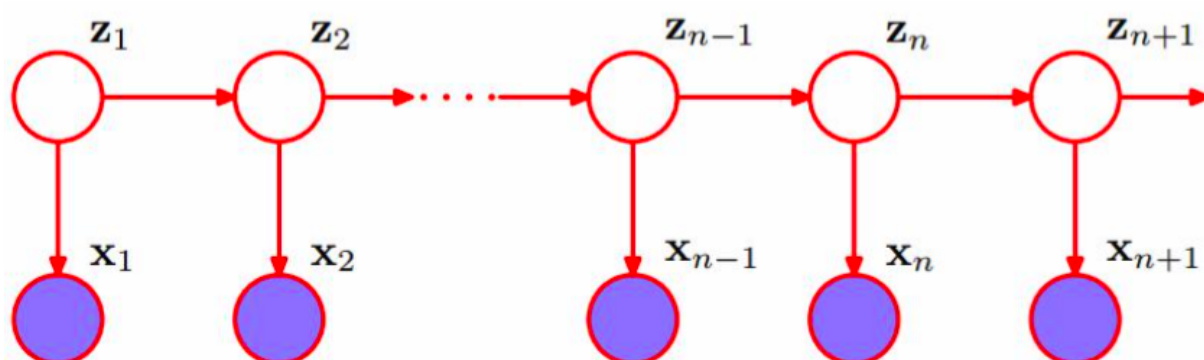
- 上述中  $I - Q$  可逆。且逆阵  $M = (I - Q)^{-1} = \sum_{s=0}^{\infty} Q^s$ 。记元素全为 1 的向量  $e = (1, 1, \dots, 1)$ 。则  $y = Me$  的第  $i$  个分量是第  $i$  个非吸收状态出发, 被某个吸收状态吸收的平均次数
- 首达概率矩阵  $F = MR$

# 隐马尔科夫模型 HMM

## 网络结构

HMM (Hidden Markov Model)是关于时序的概率模型,描述由一个**隐藏**的马尔科夫链生成**不可观测**的状态随机序列,再由各个状态生成**观测随机序列**的过程.

就是马尔科夫链本身状态不可观测，但是每个不可观测状态  $z$  又会生成一个可观测状态  $x$ 。



## 定义

参数:

1.  $Q = \{q_1, q_2, \dots, q_N\}$  为**可能**的状态集合,  $N$  是可能的状态数
2.  $V = \{v_1, v_2, \dots, v_M\}$  为可能的观测集合,  $M$  是观测集合的大小
3.  $I = \{i_1, i_2, \dots, i_T\}$  为长度为  $T$  的状态序列 (不可测),  $i_i \in Q$
4.  $O = \{o_1, o_2, \dots, o_T\}$  为对应的观测序列。  $o_i \in V$

## HMM的三要素

HMM由初始概率分布  $\pi$  (向量)、状态转移概率分布  $A$  (矩阵) 以及观测概率分布  $B$  (矩阵) 确定.  $\pi$  和  $A$  决定状态序列,  $B$  决定观测序列。因此, HMM可以用三元符号表示, 称为HMM的三要素:

$$\lambda = (A, B, \pi)$$

$A$  是转移概率矩阵,  $A = [a_{ij}]_{N \times N}$ , 其中  $a_{ij}$  是时刻  $t$  处于状态  $q_i$  的条件下时刻  $t + 1$  转移至  $q_j$  的概率

$$a_{ij} = P(i_{t+1} = q_j \mid i_t = q_i), \quad i = 1, 2, \dots, N; j = 1, 2, \dots, N$$

$B$  是观测转移概率矩阵: 其中,  $b_j(k)$  是在时刻  $t$  处于状态  $q_j$  的条件下生成观测  $v_k$  的概率:

$$B = [b_j(k)]_{N \times M}$$
$$b_j(k) = P(o_t = v_k \mid i_t = q_j), \quad k = 1, 2, \dots, M; j = 1, 2, \dots, N$$

$\pi$  是初始状态概率向量:  $\pi = (\pi)_{N \times 1}$ 。

$$\pi_i = P(i_1 = q_i), \quad i = 1, 2, \dots, N$$

一个HMM模型仅有以上三项确定, 而与观测序列无关, 因为本身就是随机的

## HMM的两个假设

齐次马尔可夫性假设: 任意时刻  $t$  的状态, 只依赖于其前一刻的状态, 与其他时刻的状态及观测无关, 也与时刻  $t$  无关.

$$P(i_t \mid i_{t-1}, o_{t-1}, \dots, i_1, o_1) = P(i_t \mid i_{t-1}), \quad t = 1, 2, \dots, T$$

观测独立性假设: 任何时刻的**观测**只依赖于该时刻的**马尔科夫链状态**. 与其他观测及状态无关.

$$P(o_t \mid i_T, o_T, \dots, i_t, o_t, \dots, i_1, o_1) = P(o_t \mid i_t), \quad t = 1, 2, \dots, T$$

## HMM的三个基本问题

详细介绍与实现可参考: [隐马尔科夫模型 \(HMM\) 及其Python实现](#)

可以的话可以试着改成 **matlab** 函数

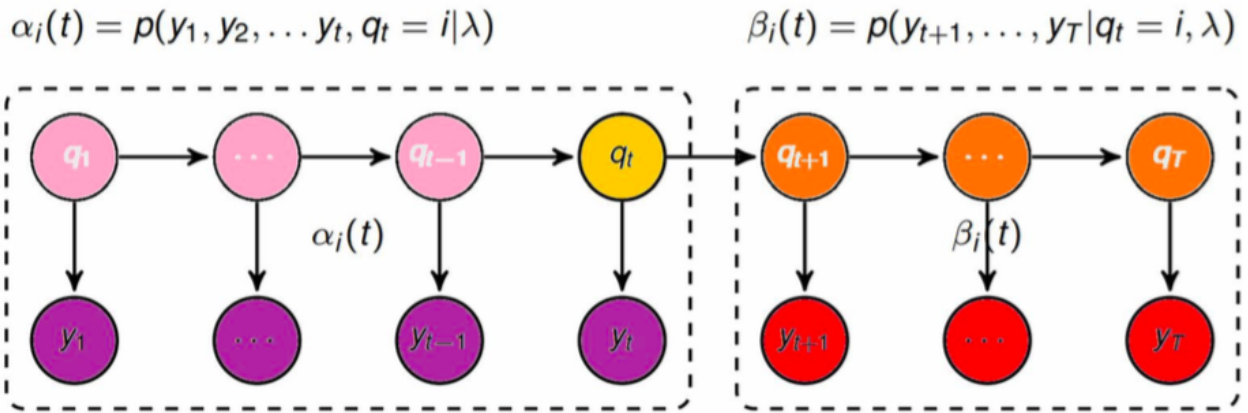
### 概率计算问题——前向后向算法 (动态规划)

给定模型  $\lambda = (A, B, \pi)$  和观测序列  $O = (o_1, o_2, \dots, o_T)$ , 计算模型  $\lambda$  下观测序列  $O$  出现的概率  $P(O \mid \lambda)$

显然给定模型可以直接计算, 即对所有的可能状态序列下的**该** 观测序列求和

$$P(O \mid \lambda) = \sum_I P(O \mid I, \lambda) P(I \mid \lambda)$$
$$= \sum_{\underbrace{i_1, i_2, \dots, i_T}_{N \times N \times \dots \times N}} \pi_{i_1} a_{i_1 i_2} b_{i_1}(o_1) \cdots a_{i_{T-1} i_T} b_{i_T}(o_T)$$

但是上述算法的复杂度达到了  $O(TN^T)$ ，不可取。



1. 前向概率：给定隐马尔科夫模型  $\lambda$ ，定义为到时刻  $t$  部分观测序列为  $o_1, o_2, \dots, o_t$  且状态为  $q_i$  的概率

$$\alpha_t(i) = P(o_1, o_2, \dots, o_t, i_t = q_i | \lambda)$$

2. 后向概率：给定隐马尔科夫模型  $\lambda$ ，定义为到时刻  $t$  状态为  $q_i$  的条件下，从  $t+1$  到  $T$  的部分观测序列为  $o_{t+1}, o_{t+2}, \dots, o_T$  的概率。

本处给出前向算法的步骤：

1. 给定初值：  $\alpha_1(i) = \pi_i b_i(o_1), i = 1, 2, \dots, N$
2. 递推：对于  $t = 1, 2, \dots, T-1$

$$\alpha_{t+1}(i) = \left[ \sum_{j=1}^N \alpha_t(j) a_{ji} \right] b_i(o_{t+1})$$

3. 最终：  $P(O | \lambda) = \sum_{i=1}^N \alpha_T(i)$

显然通过前向概率的引入作为中间变量从而实现了递归，大大降低了算法复杂度

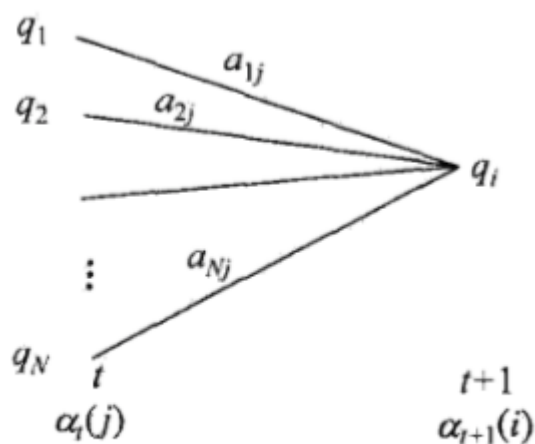


图 10.1 前向概率的递推公式

### 学习问题——Baum-Welch算法

已知观测序列  $O = o_1, o_2, \dots, o_T$ 。估计模型  $\lambda = (A, B, \pi)$ , 使得  $P(O | \lambda)$  最大, 即用极大似然法估计参数

#### 有监督的学习算法：极大似然法

假设有监督意味着还知道每一个状态序列对应的状态序列  $I = i_1, i_2, \dots, i_T$ 。

那么给定  $i_j$  下出现观测序列  $o_k$  的概率为：

$$P(o_k | i_j) = b_k(j)$$

所以似然函数可以构造为：

#### 无监督的学习算法：Baum-Weich算法

此时我们仅仅有一堆观测数据，这种情况下我们可以使用 **EM** 算法，将状态变量视为隐变量。使用 EM 算法学习HMM的参数的算法称为 BW 算法

### 预测算法——维特比 (Viterbi) 算法

也称为解码问题。已知观测序列  $O = o_1, o_2, \dots, o_T$  和模型  $\lambda = (A, B, \pi)$ ，求给定观测序列使得条件概率  $P(I | O)$  最大的状态序列  $I$ 。相当于预测出一个观测序列对应的**状态向量**