

隐马尔可夫模型

一、隐马尔可夫模型HMM

1. 隐马尔可夫模型（Hidden Markov model, HMM）是可用于序列标注问题的统计学模型，描述了由隐马尔可夫链随机生成观察序列的过程，属于生成模型。
2. 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察而产生观察随机序列的过程。
 - 隐藏的马尔可夫链随机生成的状态的序列称作状态序列。
 - 每个状态生成一个观测，而由此产生的观测的随机序列称作观测序列。
 - 序列的每一个位置又可以看作是一个时刻。

1.1 基本概念

1. 设 $\mathcal{Q} = \{\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_Q\}$ 是所有可能的状态的集合， $\mathcal{V} = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_V\}$ 是所有可能的观测的集合，其中 Q 是可能的状态数量， V 是可能的观测数量。
 - \mathcal{Q} 是状态的取值空间， \mathcal{V} 是观测的取值空间。
 - 每个观测值 \mathbf{v}_i 可能是标量，也可能是一组标量构成的集合，因此这里用加粗的黑体表示。状态值的表示也类似。
2. 设 $\mathbf{I} = (i_1, i_2, \dots, i_T)$ 是长度为 T 的状态序列， $\mathbf{O} = (o_1, o_2, \dots, o_T)$ 是对应的观测序列。
 - $i_t \in \{1, \dots, Q\}$ 是一个随机变量，代表状态 \mathbf{q}_{i_t} 。
 - $o_t \in \{1, \dots, V\}$ 是一个随机变量，代表观测 \mathbf{v}_{o_t} 。
3. 设 \mathbf{A} 为状态转移概率矩阵

$$\mathbf{A} = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,Q} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,Q} \\ \vdots & \vdots & \ddots & \vdots \\ a_{Q,1} & a_{Q,2} & \cdots & a_{Q,Q} \end{bmatrix}$$

其中 $a_{i,j} = P(i_{t+1} = j \mid i_t = i)$ ，表示在时刻 t 处于状态 \mathbf{q}_i 的条件下，在时刻 $t+1$ 时刻转移到状态 \mathbf{q}_j 的概率。

4. 设 \mathbf{B} 为观测概率矩阵

$$\mathbf{B} = \begin{bmatrix} b_1(1) & b_1(2) & \cdots & b_1(V) \\ b_2(1) & b_2(2) & \cdots & b_2(V) \\ \vdots & \vdots & \ddots & \vdots \\ b_Q(1) & b_Q(2) & \cdots & b_Q(V) \end{bmatrix}$$

其中 $b_j(k) = P(o_t = k \mid i_t = j)$ ，表示在时刻 t 处于状态 \mathbf{q}_j 的条件下生成观测 \mathbf{v}_k 的概率。

5. 设 $\vec{\pi}$ 是初始状态概率向量： $\vec{\pi} = (\pi_1, \pi_2, \dots, \pi_Q)^T$ ， $\pi_i = P(i_1 = i)$ 是时刻 $t=1$ 时处于状态 \mathbf{q}_i 的概率。
根据定义有： $\sum_{i=1}^Q \pi_i = 1$ 。

6. 隐马尔可夫模型由初始状态概率向量 π 、状态转移概率矩阵 \mathbf{A} 以及观测概率矩阵 \mathbf{B} 决定。因此隐马尔可夫模型 λ 可以用三元符号表示，即： $\lambda = (\mathbf{A}, \mathbf{B}, \pi)$ 。其中 $\mathbf{A}, \mathbf{B}, \pi$ 称为隐马尔可夫模型的三要素：

- 状态转移概率矩阵 \mathbf{A} 和初始状态概率向量 π 确定了隐藏的马尔可夫链，生成不可观测的状态序列。
- 观测概率矩阵 \mathbf{B} 确定了如何从状态生成观测，与状态序列一起确定了如何产生观测序列。

7. 从定义可知，隐马尔可夫模型做了两个基本假设：

- 齐次性假设：即假设隐藏的马尔可夫链在任意时刻 t 的状态只依赖于它在前一时刻的状态，与其他时刻的状态和观测无关，也与时刻 t 无关，即：

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

- 观测独立性假设，即假设任意时刻的观测值只依赖于该时刻的马尔可夫链的状态，与其他观测及状态无关，即：

$$P(o_t | i_T, o_T, \dots, i_{t+1}, o_{t+1}, i_t, i_{t-1}, o_{t-1}, \dots, i_1, o_1) = P(o_t | i_t), \quad t = 1, 2, \dots, T$$

1.2 生成算法

1. 隐马尔可夫模型可以用于标注问题：给定观测的序列，预测其对应的状态序列。如：词性标注问题中，状态就是单词的词性，观测就是具体的单词。在这个问题中：

- 状态序列：词性序列。
- 观察序列：单词序列。
- 生成方式：
 - 给定初始状态概率向量 π ，随机生成第一个词性。
 - 根据前一个词性，利用状态转移概率矩阵 \mathbf{A} 随机生成下一个词性。
 - 一旦生成词性序列，则根据每个词性，利用观测概率矩阵 \mathbf{B} 生成对应位置的观察，得到观察序列。

2. 一个长度为 T 的观测序列的 HMM 生成算法：

- 输入：
 - 隐马尔可夫模型 $\lambda = (\mathbf{A}, \mathbf{B}, \pi)$
 - 观测序列长度 T
- 输出：观测序列 $\mathbf{O} = (o_1, o_2, \dots, o_T)$
- 算法步骤：
 - 按照初始状态分布 π 产生状态 i_1 。
 - 令 $t = 1$ ，开始迭代。迭代条件为： $t \leq T$ 。迭代步骤为：
 - 按照状态 i_t 的观测概率分布 $b_j(k)$ 生成 o_t ， $o_t \in \mathbb{V}$ 。
 - 按照状态 i_t 的状态转移概率分布 $a_{i,j}$ 产生状态 i_{t+1} ， $i_{t+1} \in \mathbb{Q}$ 。
 - 令 $t = t + 1$ 。

二、HMM 基本问题

1. 隐马尔可夫模型的 3 个基本问题：

- 概率计算问题：给定模型 $\lambda = (\mathbf{A}, \mathbf{B}, \vec{\pi})$ 和观测序列 $\mathbf{O} = (o_1, o_2, \dots, o_T)$ ，计算观测序列 \mathbf{O} 出现的概率 $P(\mathbf{O}; \lambda)$ 。即：评估模型 λ 与观察序列 \mathbf{O} 之间的匹配程度。
- 学习问题：已知观测序列 $\mathbf{O} = (o_1, o_2, \dots, o_T)$ ，估计模型 $\lambda = (\mathbf{A}, \mathbf{B}, \vec{\pi})$ 的参数，使得在该模型下观测序列概率 $P(\mathbf{O}; \lambda)$ 最大。即：用极大似然估计的方法估计参数。
- 预测问题（也称为解码问题）：已知模型 $\lambda = (\mathbf{A}, \mathbf{B}, \vec{\pi})$ 和观测序列 $\mathbf{O} = (o_1, o_2, \dots, o_T)$ ，求对给定观测序列的条件概率 $P(\mathbf{I} | \mathbf{O})$ 最大的状态序列 $\mathbf{I} = (i_1, i_2, \dots, i_T)$ 。即：给定观测序列，求最可能的对应的状态序列。

如：在语音识别任务中，观测值为语音信号，隐藏状态为文字。解码问题的目标就是：根据观测的语音信号来推断最有可能的文字序列。

2.1 概率计算问题

1. 给定隐马尔可夫模型 $\lambda = (\mathbf{A}, \mathbf{B}, \vec{\pi})$ 和观测序列 $\mathbf{O} = (o_1, o_2, \dots, o_T)$ ，概率计算问题需要计算在模型 λ 下观测序列 \mathbf{O} 出现的概率 $P(\mathbf{O}; \lambda)$ 。
2. 最直接的方法是按照概率公式直接计算：通过列举所有可能的、长度为 T 的状态序列 $\mathbf{I} = (i_1, i_2, \dots, i_T)$ ，求各个状态序列 \mathbf{I} 与观测序列 $\mathbf{O} = (o_1, o_2, \dots, o_T)$ 的联合概率 $P(\mathbf{O}, \mathbf{I}; \lambda)$ ，然后对所有可能的状态序列求和，得到 $P(\mathbf{O}; \lambda)$ 。

- 状态序列 $\mathbf{I} = (i_1, i_2, \dots, i_T)$ 的概率为：

$$P(\mathbf{I}; \lambda) = \pi_{i_1} a_{i_1, i_2} a_{i_2, i_3} \cdots a_{i_{T-1}, i_T}$$

- 给定状态序列 $\mathbf{I} = (i_1, i_2, \dots, i_T)$ ，观测序列 $\mathbf{O} = (o_1, o_2, \dots, o_T)$ 的条件概率为：

$$P(\mathbf{O} | \mathbf{I}; \lambda) = b_{i_1}(o_1) b_{i_2}(o_2) \cdots b_{i_T}(o_T)$$

- \mathbf{O} 和 \mathbf{I} 同时出现的联合概率为：

$$P(\mathbf{O}, \mathbf{I}; \lambda) = P(\mathbf{O} | \mathbf{I}; \lambda) P(\mathbf{I}; \lambda) = \pi_{i_1} a_{i_1, i_2} a_{i_2, i_3} \cdots a_{i_{T-1}, i_T} b_{i_1}(o_1) b_{i_2}(o_2) \cdots b_{i_T}(o_T)$$

- 对所有可能的状态序列 \mathbf{I} 求和，得到观测序列 \mathbf{O} 的概率：

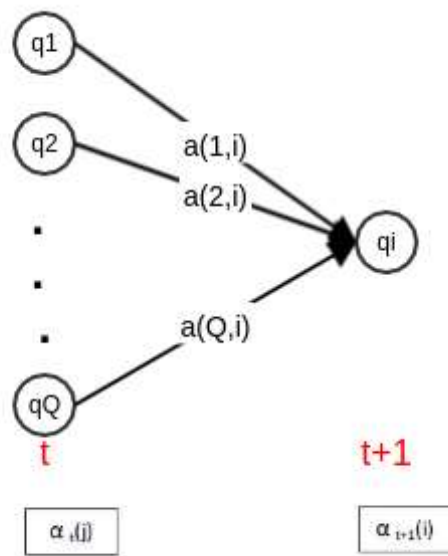
$$P(\mathbf{O}; \lambda) = \sum_{\mathbf{I}} P(\mathbf{O}, \mathbf{I}; \lambda) = \sum_{i_1, i_2, \dots, i_T} \pi_{i_1} a_{i_1, i_2} a_{i_2, i_3} \cdots a_{i_{T-1}, i_T} b_{i_1}(o_1) b_{i_2}(o_2) \cdots b_{i_T}(o_T)$$

- 上式的算法复杂度为 $O(T \times Q^T)$ ，太复杂，实际应用中不太可行。

2.1.1 前向算法

1. 给定隐马尔可夫模型 $\lambda = (\mathbf{A}, \mathbf{B}, \vec{\pi})$ ，定义前向概率：在时刻 t 时的观测序列为 o_1, o_2, \dots, o_t ，且时刻 t 时状态为 \mathbf{q}_i 的概率为前向概率，记作： $\alpha_t(i) = P(o_1, o_2, \dots, o_t, i_t = i; \lambda)$
2. 根据定义， $\alpha_t(j)$ 是在时刻 t 时观测到 o_1, o_2, \dots, o_t ，且在时刻 t 处于状态 \mathbf{q}_j 的前向概率。则有：
 - $\alpha_t(j) \times a_{j,i}$ ：为在时刻 t 时观测到 o_1, o_2, \dots, o_t ，且在时刻 t 处于状态 \mathbf{q}_j ，且在 $t+1$ 时刻处在状态 \mathbf{q}_i 的概率。
 - $\sum_{j=1}^Q \alpha_t(j) \times a_{j,i}$ ：为在时刻 t 观测序列为 o_1, o_2, \dots, o_t ，并且在时刻 $t+1$ 时刻处于状态 \mathbf{q}_i 的概率。
 - 考虑 $b_i(o_{t+1})$ ，则得到前向概率的地推公式：

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



3. 观测序列概率的前向算法：

◦ 输入：

- 隐马尔可夫模型 $\lambda = (\mathbf{A}, \mathbf{B}, \vec{\pi})$
- 观测序列 $\mathbf{O} = (o_1, o_2, \dots, o_T)$

◦ 输出：观测序列概率 $P(\mathbf{O}; \lambda)$

◦ 算法步骤：

- 计算初值： $\alpha_1(i) = \pi_i b_i(o_1), \quad i = 1, 2, \dots, Q$ 。

该初值是初始时刻的状态 $i_1 = i$ 和观测 o_1 的联合概率。

- 递推：对于 $t = 1, 2, \dots, T - 1$ ：

$$\alpha_{t+1}(i) = \left[\sum_{j=1}^Q \alpha_t(j) a_{j,i} \right] b_i(o_{t+1}), \quad i = 1, 2, \dots, Q$$

- 终止： $P(\mathbf{O}; \lambda) = \sum_{i=1}^Q \alpha_T(i)$ 。

因为 $\alpha_T(i)$ 表示在时刻 T ，观测序列为 o_1, o_2, \dots, o_T ，且状态为 \mathbf{q}_i 的概率。对所有可能的 Q 个状态 \mathbf{q}_i 求和则得到 $P(\mathbf{O}; \lambda)$ 。

4. 前向算法是基于 状态序列的路径结构 递推计算 $P(\mathbf{O}; \lambda)$ 。

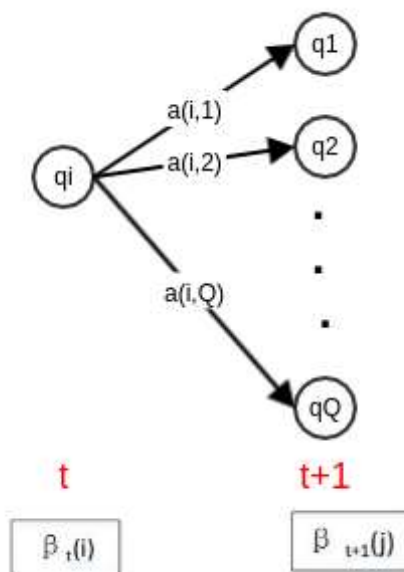
- 其高效的关键是局部计算前向概率，然后利用路径结构将前向概率“递推”到全局。
- 算法复杂度为 $O(TQ^2)$ 。

2.1.2 后向算法

1. 给定隐马尔可夫模型 $\lambda = (\mathbf{A}, \mathbf{B}, \vec{\pi})$ ，定义后向概率：在时刻 t 的状态为 \mathbf{q}_i 的条件下，从时刻 $t + 1$ 到 T 的观测序列为 $o_{t+1}, o_{t+2}, \dots, o_T$ 的概率为后向概率，记作： $\beta_t(i) = P(o_{t+1}, o_{t+2}, \dots, o_T \mid i_t = i; \lambda)$ 。
2. 在时刻 t 状态为 \mathbf{q}_i 的条件下，从时刻 $t + 1$ 到 T 的观测序列为 $o_{t+1}, o_{t+2}, \dots, o_T$ 的概率可以这样计算：

- 考虑 t 时刻状态 \mathbf{q}_i 经过 $a_{i,j}$ 转移到 $t+1$ 时刻的状态 \mathbf{q}_j 。
 - $t+1$ 时刻状态为 \mathbf{q}_j 的条件下, 从时刻 $t+2$ 到 T 的观测序列为观测序列为 $o_{t+2}, o_{t+3}, \dots, o_T$ 的概率为 $\beta_{t+1}(j)$ 。
 - $t+1$ 时刻状态为 \mathbf{q}_j 的条件下, 从时刻 $t+1$ 到 T 的观测序列为观测序列为 $o_{t+1}, o_{t+2}, \dots, o_T$ 的概率为 $b_j(o_{t+1}) \times \beta_{t+1}(j)$ 。
- 考虑所有可能的 \mathbf{q}_j , 则得到 $\beta_t(j)$ 的递推公式:

$$\beta_t(i) = \sum_{j=1}^Q a_{i,j} b_j(o_{t+1}) \beta_{t+1}(j)$$



3. 观测序列概率的后向算法:

- 输入:
 - 隐马尔可夫模型 $\lambda = (\mathbf{A}, \mathbf{B}, \pi)$
 - 观测序列 $\mathbf{O} = (o_1, o_2, \dots, o_T)$
- 输出: 观测序列概率 $P(\mathbf{O}; \lambda)$
- 算法步骤:
 - 计算初值: $\beta_T(i) = 1, \quad i = 1, 2, \dots, Q$
对最终时刻的所有状态 \mathbf{q}_i , 规定 $\beta_T(i) = 1$ 。
 - 递推: 对 $t = T-1, T-2, \dots, 1$:

$$\beta_t(i) = \sum_{j=1}^Q a_{i,j} b_j(o_{t+1}) \beta_{t+1}(j), \quad i = 1, 2, \dots, Q$$

- 终止: $P(\mathbf{O}; \lambda) = \sum_{i=1}^Q \pi_i b_i(o_1) \beta_1(i)$

$\beta_1(i)$ 为在时刻 1, 状态为 \mathbf{q}_i 的条件下, 从时刻 2 到 T 的观测序列为 o_2, o_3, \dots, o_T 的概率。对所有的可能初始状态 \mathbf{q}_i (由 π_i 提供其概率) 求和并考虑 o_1 即可得到观测序列为 o_1, o_2, \dots, o_T 的概率。

2.1.3 统一形式

1. 利用前向概率和后向概率的定义，可以将观测序列概率统一为：

$$P(\mathbf{O}; \lambda) = \sum_{i=1}^Q \sum_{j=1}^Q \alpha_t(i) a_{i,j} b_j(o_{t+1}) \beta_{t+1}(j), \quad t = 1, 2, \dots, T-1$$

◦ 当 $t = 1$ 时，就是后向概率算法；当 $t = T - 1$ 时，就是前向概率算法。

◦ 其意义为：在时刻 t ：

- $\alpha_t(i)$ 表示：已知时刻 t 时的观测序列为 o_1, o_2, \dots, o_t 、且时刻 t 时状态为 \mathbf{q}_i 的概率。
- $\alpha_t(i) a_{i,j}$ 表示：已知时刻 t 时的观测序列为 o_1, o_2, \dots, o_t 、且时刻 t 时状态为 \mathbf{q}_i 、且 $t + 1$ 时刻状态为 \mathbf{q}_j 的概率。
- $\alpha_t(i) a_{i,j} b_j(o_{t+1})$ 表示：已知时刻 $t + 1$ 时的观测序列为 o_1, o_2, \dots, o_{t+1} 、且时刻 t 时状态为 \mathbf{q}_i 、且 $t + 1$ 时刻状态为 \mathbf{q}_j 的概率。
- $\alpha_t(i) a_{i,j} b_j(o_{t+1}) \beta_{t+1}(j)$ 表示：已知观测序列为 o_1, o_2, \dots, o_T 、且时刻 t 时状态为 \mathbf{q}_i 、且 $t + 1$ 时刻状态为 \mathbf{q}_j 的概率。
- 对所有可能的状态 $\mathbf{q}_i, \mathbf{q}_j$ 取值，即得到上式。

2. 根据前向算法有： $\alpha_{t+1}(j) = \sum_{i=1}^Q \alpha_t(i) a_{i,j} b_j(o_{t+1})$ 。则得到：

$$\begin{aligned} P(\mathbf{O}; \lambda) &= \sum_{i=1}^Q \sum_{j=1}^Q \alpha_t(i) a_{i,j} b_j(o_{t+1}) \beta_{t+1}(j) \\ &= \sum_{j=1}^Q \left[\sum_{i=1}^Q \alpha_t(i) a_{i,j} b_j(o_{t+1}) \right] \beta_{t+1}(j) = \sum_{j=1}^Q \alpha_{t+1}(j) \beta_{t+1}(j) \\ &\quad t = 1, 2, \dots, T-1 \end{aligned}$$

由于 t 的形式不重要，因此有：

$$P(\mathbf{O}; \lambda) = \sum_{j=1}^Q \alpha_t(j) \beta_t(j), \quad t = 1, 2, \dots, T$$

3. 给定模型 $\lambda = (\mathbf{A}, \mathbf{B}, \vec{\pi})$ 和观测序列 \mathbf{O} 的条件下，在时刻 t 处于状态 \mathbf{q}_i 的概率记作：

$$\gamma_t(i) = P(i_t = i \mid \mathbf{O}; \lambda)$$

◦ 根据定义：

$$\gamma_t(i) = P(i_t = i \mid \mathbf{O}; \lambda) = \frac{P(i_t = i, \mathbf{O}; \lambda)}{P(\mathbf{O}; \lambda)}$$

◦ 根据前向概率和后向概率的定义，有： $\alpha_t(i) \beta_t(i) = P(i_t = i, \mathbf{O}; \lambda)$ ，则有：

$$\gamma_t(i) = \frac{P(i_t = i, \mathbf{O}; \lambda)}{P(\mathbf{O}; \lambda)} = \frac{\alpha_t(i) \beta_t(i)}{P(\mathbf{O}; \lambda)} = \frac{\alpha_t(i) \beta_t(i)}{\sum_{j=1}^Q \alpha_t(j) \beta_t(j)}$$

4. 给定模型 $\lambda = (\mathbf{A}, \mathbf{B}, \vec{\pi})$ 和观测序列 \mathbf{O} ，在时刻 t 处于状态 \mathbf{q}_i 且在 $t + 1$ 时刻处于状态 \mathbf{q}_j 的概率记作：

$$\xi_t(i, j) = P(i_t = i, i_{t+1} = j \mid \mathbf{O}; \lambda)$$

◦ 根据

$$\begin{aligned} \xi_t(i, j) &= P(i_t = i, i_{t+1} = j \mid \mathbf{O}; \lambda) = \frac{P(i_t = i, i_{t+1} = j, \mathbf{O}; \lambda)}{P(\mathbf{O}; \lambda)} \\ &= \frac{P(i_t = i, i_{t+1} = j, \mathbf{O}; \lambda)}{\sum_{u=1}^Q \sum_{v=1}^Q P(i_t = u, i_{t+1} = v, \mathbf{O}; \lambda)} \end{aligned}$$

- 考虑到前向概率和后向概率的定义有： $P(i_t = i, i_{t+1} = j, \mathbf{O}; \lambda) = \alpha_t(i) a_{i,j} b_j(o_{t+1}) \beta_{t+1}(j)$ ，因此有：

$$\xi_t(i, j) = \frac{\alpha_t(i) a_{i,j} b_j(o_{t+1}) \beta_{t+1}(j)}{\sum_{u=1}^Q \sum_{v=1}^Q \alpha_t(u) a_{u,v} b_v(o_{t+1}) \beta_{t+1}(v)}$$

5. 一些期望值：

- 在给定观测 \mathbf{O} 的条件下，状态 i 出现的期望值为： $\sum_{t=1}^T \gamma_t(i)$ 。
- 在给定观测 \mathbf{O} 的条件下，从状态 i 转移的期望值： $\sum_{t=1}^{T-1} \gamma_t(i)$ 。
 - 这里的转移，表示状态 i 可能转移到任何可能的状态。
 - 假若在时刻 T 的状态为 \mathbf{q}_i ，则此时不可能再转移，因为时间最大为 T 。
- 在观测 \mathbf{O} 的条件下，由状态 i 转移到状态 j 的期望值： $\sum_{t=1}^{T-1} \xi_t(i, j)$ 。

2.2 学习问题

1. 根据训练数据的不同，隐马尔可夫模型的学习方法也不同：

- 训练数据包括观测序列和对应的状态序列：通过监督学习来学习隐马尔可夫模型。
- 训练数据仅包括观测序列：通过非监督学习来学习隐马尔可夫模型。

2.2.1 监督学习

1. 假设数据集为 $\mathbb{D} = \{(\mathbf{O}_1, \mathbf{I}_1), (\mathbf{O}_2, \mathbf{I}_2), \dots, (\mathbf{O}_N, \mathbf{I}_N)\}$ 。其中：

- $\mathbf{O}_1, \dots, \mathbf{O}_N$ 为 N 个观测序列； $\mathbf{I}_1, \dots, \mathbf{I}_N$ 为对应的 N 个状态序列。
- 序列 $\mathbf{O}_k, \mathbf{I}_k$ 的长度为 T_k ，其中数据集中 $\mathbf{O}_1, \dots, \mathbf{O}_N$ 之间的序列长度可以不同。

2. 可以利用极大似然估计来估计隐马尔可夫模型的参数。

- 转移概率 $a_{i,j}$ 的估计：设样本中前一时刻处于状态 i 、且后一时刻处于状态 j 的频数为 $A_{i,j}$ ，则状态转移概率 $a_{i,j}$ 的估计是：

$$\hat{a}_{i,j} = \frac{A_{i,j}}{\sum_{u=1}^Q A_{i,u}}, \quad i = 1, 2, \dots, Q; j = 1, 2, \dots, Q$$

- 观测概率 $b_j(k)$ 的估计：设样本中状态为 j 并且观测为 k 的频数为 $B_{j,k}$ ，则状态为 j 并且观测为 k 的概率 $b_j(k)$ 的估计为：

$$\hat{b}_j(k) = \frac{B_{j,k}}{\sum_{v=1}^V B_{j,v}}, \quad j = 1, 2, \dots, Q; k = 1, 2, \dots, V$$

- 初始状态概率的估计：设样本中初始时刻（即： $t = 1$ ）处于状态 i 的频数为 C_i ，则初始状态概率 π_i 的估计为： $\hat{\pi}_i = \frac{C_i}{\sum_{j=1}^Q C_j}$ ， $i = 1, 2, \dots, Q$ 。

2.2.2 无监督学习

1. 监督学习需要使用人工标注的训练数据。由于人工标注往往代价很高，所以经常会利用无监督学习的方法。

隐马尔可夫模型的无监督学习通常使用 `Baum-Welch` 算法求解。

2. 在隐马尔可夫模型的无监督学习中，数据集为 $\mathbb{D} = \{\mathbf{O}_1, \mathbf{O}_2, \dots, \mathbf{O}_N\}$ 。其中：

- $\mathbf{O}_1, \dots, \mathbf{O}_N$ 为 N 个观测序列。
- 序列 \mathbf{O}_k 的长度为 T_k ，其中数据集中 $\mathbf{O}_1, \dots, \mathbf{O}_N$ 之间的序列长度可以不同。

3. 将观测序列数据看作观测变量 \mathbf{O} ，状态序列数据看作不可观测的隐变量 \mathbf{I} ，则隐马尔可夫模型事实上是一个含有隐变量的概率模型： $P(\mathbf{O}; \lambda) = \sum_{\mathbf{I}} P(\mathbf{O} | \mathbf{I}; \lambda) P(\mathbf{I}; \lambda)$ 。其参数学习可以由 EM 算法实现。

- E 步：求 Q 函数（其中 $\bar{\lambda}$ 是参数的当前估计值）

$$Q(\lambda, \bar{\lambda}) = \sum_{j=1}^N \left(\sum_{\mathbf{I}} P(\mathbf{I} | \mathbf{O} = \mathbf{O}_j; \bar{\lambda}) \log P(\mathbf{O} = \mathbf{O}_j, \mathbf{I}; \lambda) \right)$$

将 $P(\mathbf{I} | \mathbf{O} = \mathbf{O}_j; \bar{\lambda}) = \frac{P(\mathbf{I}, \mathbf{O} = \mathbf{O}_j; \bar{\lambda})}{P(\mathbf{O}_j; \bar{\lambda})}$ 代入上式，有：

$$Q(\lambda, \bar{\lambda}) = \sum_{j=1}^N \frac{1}{P(\mathbf{O}_j; \bar{\lambda})} \left(\sum_{\mathbf{I}} P(\mathbf{I}, \mathbf{O} = \mathbf{O}_j; \bar{\lambda}) \log P(\mathbf{I}, \mathbf{O} = \mathbf{O}_j; \lambda) \right)$$

- 在给定参数 $\bar{\lambda}$ 时， $P(\mathbf{O}_j; \bar{\lambda})$ 是已知的常数，记做 \tilde{P}_j 。
- 在给定参数 $\bar{\lambda}$ 时， $P(\mathbf{I}, \mathbf{O} = \mathbf{O}_j; \bar{\lambda})$ 是 \mathbf{I} 的函数，记做 $\tilde{P}_j(\mathbf{I})$ 。

根据 $P(\mathbf{O}, \mathbf{I}; \lambda) = \pi_{i_1} b_{i_1}(o_1) a_{i_1, i_2} b_{i_2}(o_2) \cdots a_{i_{T-1}, i_T} b_{i_T}(o_T)$ 得到：

$$Q(\lambda, \bar{\lambda}) = \sum_{j=1}^N \frac{1}{\tilde{P}_j} \left(\sum_{\mathbf{I}} (\log \pi_{i_1}) \tilde{P}_j(\mathbf{I}) + \sum_{\mathbf{I}} \left(\sum_{t=1}^{T_j-1} \log a_{i_t, i_{t+1}} \right) \tilde{P}_j(\mathbf{I}) \right. \\ \left. + \sum_{\mathbf{I}} \left(\sum_{t=1}^{T_j} \log b_{i_t}(o_t^{(j)}) \right) \tilde{P}_j(\mathbf{I}) \right)$$

其中： T_j 表示第 j 个序列的长度， $o_t^{(j)}$ 表示第 j 个观测序列的第 t 个位置。

- M 步：求 Q 函数的极大值：

$$\bar{\lambda}^{<new>} \leftarrow \arg \max_{\lambda} Q(\lambda, \bar{\lambda})$$

极大化参数在 Q 函数中单独的出现在3个项中，所以只需要对各项分别极大化。

- $\frac{\partial Q(\lambda, \bar{\lambda})}{\partial \pi_i} = 0$ ：

$$\frac{\partial Q(\lambda, \bar{\lambda})}{\partial \pi_i} = \frac{\partial (\sum_{j=1}^N \frac{1}{\tilde{P}_j} \sum_{\mathbf{I}} (\log \pi_{i_1}) \tilde{P}_j(\mathbf{I}))}{\partial \pi_i} \\ = \sum_{j=1}^N \frac{1}{\tilde{P}_j} \sum_{i_1=1}^Q P(i_1, \mathbf{O} = \mathbf{O}_j; \bar{\lambda}) \frac{\partial \log \pi_{i_1}}{\partial \pi_i}$$

将 $\pi_Q = 1 - \pi_1 - \cdots - \pi_{Q-1}$ 代入，有：

$$\frac{\partial Q(\lambda, \bar{\lambda})}{\partial \pi_i} = \sum_{j=1}^N \frac{1}{\tilde{P}_j} \left(\frac{P(i_1 = i, \mathbf{O} = \mathbf{O}_j; \bar{\lambda})}{\pi_i} - \frac{P(i_1 = Q, \mathbf{O} = \mathbf{O}_j; \bar{\lambda})}{\pi_Q} \right) = 0$$

将 $\tilde{P}_j = P(\mathbf{O} = \mathbf{O}_j; \bar{\lambda})$ 代入，即有：

$$\pi_i \propto \sum_{j=1}^N P(i_1 = i | \mathbf{O} = \mathbf{O}_j; \bar{\lambda})$$

考虑到 $\sum_{i=1}^Q \pi_i = 1$ ，以及 $\sum_{i=1}^Q \sum_{j=1}^N P(i_1 = i | \mathbf{O} = \mathbf{O}_j; \bar{\lambda}) = N$ ，则有：

$$\pi_i = \frac{\sum_{j=1}^N P(i_1 = i | \mathbf{O} = \mathbf{O}_j; \bar{\lambda})}{N}$$

其物理意义为：统计在给定参数 $\bar{\lambda}$ ，已知 $\mathbf{O} = \mathbf{O}_j$ 的条件下， $i_1 = i$ 的出现的频率。它就是 $i_1 = i$ 的后验概率的估计值。

- $\frac{\partial Q(\lambda, \bar{\lambda})}{\partial a_{i,j}} = 0$ ：同样的处理有：

$$\frac{\partial Q(\lambda, \bar{\lambda})}{\partial a_{i,j}} = \sum_{k=1}^N \frac{1}{\bar{P}_k} \sum_{t=1}^{T_k-1} \left(\frac{P(i_t = i, i_{t+1} = j, \mathbf{O} = \mathbf{O}_k; \bar{\lambda})}{a_{i,j}} - \frac{P(i_t = i, i_{t+1} = Q, \mathbf{O} = \mathbf{O}_k; \bar{\lambda})}{a_{i,Q}} \right)$$

得到：

$$a_{i,j} \propto \sum_{k=1}^N \sum_{t=1}^{T_k-1} P(i_t = i, i_{t+1} = j \mid \mathbf{O} = \mathbf{O}_k; \bar{\lambda})$$

考虑到 $\sum_{j=1}^Q a_{i,j} = 1$ ，则有：

$$\begin{aligned} a_{i,j} &= \frac{\sum_{k=1}^N \sum_{t=1}^{T_k-1} P(i_t = i, i_{t+1} = j \mid \mathbf{O} = \mathbf{O}_k; \bar{\lambda})}{\sum_{j'=1}^Q \sum_{k=1}^N \sum_{t=1}^{T_k-1} P(i_t = i, i_{t+1} = j' \mid \mathbf{O} = \mathbf{O}_k; \bar{\lambda})} \\ &= \frac{\sum_{k=1}^N \sum_{t=1}^{T_k-1} P(i_t = i, i_{t+1} = j \mid \mathbf{O} = \mathbf{O}_k; \bar{\lambda})}{\sum_{k=1}^N \sum_{t=1}^{T_k-1} P(i_t = i \mid \mathbf{O} = \mathbf{O}_k; \bar{\lambda})} \end{aligned}$$

其物理意义为：统计在给定参数 $\bar{\lambda}$ ，已知 $\mathbf{O} = \mathbf{O}_j$ 的条件下，统计当 $i_t = i$ 的情况下 $i_{t+1} = j$ 的出现的频率。它就是 $i_{t+1} = j \mid i_t = i$ 的后验概率的估计值。

- $\frac{\partial Q(\lambda, \bar{\lambda})}{\partial b_j(k)} = 0$ ：同样的处理有：

$$\frac{\partial Q(\lambda, \bar{\lambda})}{\partial b_j(k)} = \sum_{i=1}^N \frac{1}{\bar{P}_i} \sum_{t=1}^{T_i} \left(\frac{P(i_t = j, o_t = k, \mathbf{O} = \mathbf{O}_i; \bar{\lambda})}{b_j(k)} - \frac{P(i_t = j, o_t = V, \mathbf{O} = \mathbf{O}_i; \bar{\lambda})}{b_j(V)} \right)$$

得到：

$$b_j(k) \propto \sum_{i=1}^N \sum_{t=1}^{T_i} P(i_t = j, o_t = k \mid \mathbf{O} = \mathbf{O}_i; \bar{\lambda})$$

其中如果第 i 个序列 \mathbf{O}_i 的第 t 个位置 $o_t^{(i)} \neq k$ ，则 $P(i_t = j, o_t = k \mid \mathbf{O} = \mathbf{O}_i; \bar{\lambda}) = 0$ 。

考虑到 $\sum_{k=1}^V b_j(k) = 1$ ，则有：

$$\begin{aligned} b_j(k) &= \frac{\sum_{i=1}^N \sum_{t=1}^{T_i} P(i_t = j, o_t = k \mid \mathbf{O} = \mathbf{O}_i; \bar{\lambda})}{\sum_{k'=1}^V \sum_{i=1}^N \sum_{t=1}^{T_i} P(i_t = j, o_t = k' \mid \mathbf{O} = \mathbf{O}_i; \bar{\lambda})} \\ &= \frac{\sum_{i=1}^N \sum_{t=1}^{T_i} P(i_t = j, o_t = k \mid \mathbf{O} = \mathbf{O}_i; \bar{\lambda})}{\sum_{i=1}^N \sum_{t=1}^{T_i} P(i_t = j \mid \mathbf{O} = \mathbf{O}_i; \bar{\lambda})} \end{aligned}$$

其物理意义为：统计在给定参数 $\bar{\lambda}$ ，已知 $\mathbf{O} = \mathbf{O}_j$ 的条件下，统计当 $i_t = j$ 的情况下 $o_t = k$ 的出现的频率。它就是 $o_t = k \mid i_t = j$ 的后验概率的估计值。

4. 令 $\gamma_t^{(s)}(i) = P(i_t = i \mid \mathbf{O} = \mathbf{O}_s; \bar{\lambda})$ ，其物理意义为：在序列 \mathbf{O}_s 中，第 t 时刻的隐状态为 i 的后验概率。

令 $\xi_t^{(s)}(i, j) = P(i_t = i, i_{t+1} = j \mid \mathbf{O} = \mathbf{O}_s; \bar{\lambda})$ ，其物理意义为：在序列 \mathbf{O}_s 中，第 t 时刻的隐状态为 i 、且第 $t+1$ 时刻的隐状态为 j 的后验概率。

则 **M** 步的估计值改写为：

$$\begin{aligned}\pi_i &= \frac{\sum_{s=1}^N P(i_1 = i \mid \mathbf{O} = \mathbf{O}_s; \bar{\lambda})}{N} = \frac{\sum_{s=1}^N \gamma_1^{(s)}(i)}{N} \\ a_{i,j} &= \frac{\sum_{s=1}^N \sum_{t=1}^{T_s-1} P(i_t = i, i_{t+1} = j \mid \mathbf{O} = \mathbf{O}_s; \bar{\lambda})}{\sum_{s=1}^N \sum_{t=1}^{T_s-1} P(i_t = i \mid \mathbf{O} = \mathbf{O}_s; \bar{\lambda})} = \frac{\sum_{s=1}^N \sum_{t=1}^{T_s-1} \xi_t^{(s)}(i, j)}{\sum_{s=1}^N \sum_{t=1}^{T_s-1} \gamma_t^{(s)}(i)} \\ b_j(k) &= \frac{\sum_{s=1}^N \sum_{t=1}^{T_s} P(i_t = j, o_t = k \mid \mathbf{O} = \mathbf{O}_s; \bar{\lambda})}{\sum_{s=1}^N \sum_{t=1}^{T_s} P(i_t = j \mid \mathbf{O} = \mathbf{O}_s; \bar{\lambda})} = \frac{\sum_{s=1}^N \sum_{t=1}^{T_s} \gamma_t^{(s)}(j) \mathbb{I}(o_t^{(s)} = k)}{\sum_{s=1}^N \sum_{t=1}^{T_s} \gamma_t^{(s)}(j)}\end{aligned}$$

其中 $\mathbb{I}(o_t^{(s)} = k)$ 为示性函数，其意义为：当 \mathbf{O}_s 的第 t 时刻为 k 时，取值为 1；否则取值为 0。

5. Baum-Welch 算法：

- 输入：观测数据 $\mathbb{D} = \{\mathbf{O}_1, \mathbf{O}_2, \dots, \mathbf{O}_N\}$
- 输出：隐马尔可夫模型参数
- 算法步骤：
 - 初始化： $n = 0$ ，选取 $a_{i,j}^{<0>}, b_j(k)^{<0>}, \pi_i^{<0>}$ ，得到模型 $\lambda^{<0>} = (\mathbf{A}^{<0>}, \mathbf{B}^{<0>}, \bar{\pi}^{<0>})$
 - 迭代，迭代停止条件为：模型参数收敛。迭代过程为：
 - 求使得 **Q** 函数取极大值的参数：

$$\begin{aligned}\pi_i^{<n+1>} &= \frac{\sum_{s=1}^N P(i_1 = i \mid \mathbf{O} = \mathbf{O}_s; \bar{\lambda})}{N} = \frac{\sum_{s=1}^N \gamma_1^{(s)}(i)}{N} \\ a_{i,j}^{<n+1>} &= \frac{\sum_{s=1}^N \sum_{t=1}^{T_s-1} P(i_t = i, i_{t+1} = j \mid \mathbf{O} = \mathbf{O}_s; \bar{\lambda})}{\sum_{s=1}^N \sum_{t=1}^{T_s-1} P(i_t = i \mid \mathbf{O} = \mathbf{O}_s; \bar{\lambda})} = \frac{\sum_{s=1}^N \sum_{t=1}^{T_s-1} \xi_t^{(s)}(i, j)}{\sum_{s=1}^N \sum_{t=1}^{T_s-1} \gamma_t^{(s)}(i)} \\ b_j(k)^{<n+1>} &= \frac{\sum_{s=1}^N \sum_{t=1}^{T_s} P(i_t = j, o_t = k \mid \mathbf{O} = \mathbf{O}_s; \bar{\lambda})}{\sum_{s=1}^N \sum_{t=1}^{T_s} P(i_t = j \mid \mathbf{O} = \mathbf{O}_s; \bar{\lambda})} = \frac{\sum_{s=1}^N \sum_{t=1}^{T_s} \gamma_t^{(s)}(j) \mathbb{I}(o_t^{(s)} = k)}{\sum_{s=1}^N \sum_{t=1}^{T_s} \gamma_t^{(s)}(j)}\end{aligned}$$

- 判断模型是否收敛。如果不收敛，则 $n \leftarrow n + 1$ ，继续迭代。
- 最终得到模型 $\lambda^{<n>} = (\mathbf{A}^{<n>}, \mathbf{B}^{<n>}, \bar{\pi}^{<n>})$ 。

2.3 预测问题

2.3.1 近似算法

1. 近似算法思想：在每个时刻 t 选择在该时刻最有可能出现的状态 i_t^* ，从而得到一个状态序列 $\mathbf{I}^* = (i_1^*, i_2^*, \dots, i_T^*)$ ，然后将它作为预测的结果。
2. 近似算法：给定隐马尔可夫模型 $\lambda = (\mathbf{A}, \mathbf{B}, \bar{\pi})$ ，观测序列 $\mathbf{O} = (o_1, o_2, \dots, o_T)$ ，在时刻 t 它处于状态 \mathbf{q}_i 的概率为：

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{P(\mathbf{O}; \lambda)} = \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^Q \alpha_t(j)\beta_t(j)}$$

在时刻 t 最可能的状态： $i_t^* = \arg \max_{1 \leq i \leq Q} \gamma_t(i)$ 。

3. 近似算法的优点是：计算简单。

近似算法的缺点是：不能保证预测的状态序列整体是最有可能的状态序列，因为预测的状态序列可能有实际上不发生的部分。

- 近似算法是局部最优（每个点最优），但是不是整体最优的。

- 近似算法无法处理这种情况：转移概率为 0。因为近似算法没有考虑到状态之间的迁移。

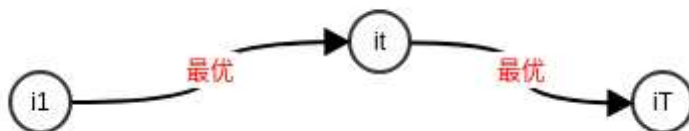
2.3.2 维特比算法

1. 维特比算法用动态规划来求解隐马尔可夫模型预测问题。

它用动态规划求解概率最大路径（最优路径），这时一条路径对应着一个状态序列。

2. 维特比算法思想：

- 根据动态规划原理，最优路径具有这样的特性：如果最优路径在时刻 t 通过结点 i_t^* ，则这一路径从结点 i_t^* 到终点 i_T^* 的部分路径，对于从 i_t^* 到 i_T^* 的所有可能路径来说，也必须是最优的。



- 只需要从时刻 $t = 1$ 开始，递推地计算从时刻 1 到时刻 t 且时刻 t 状态为 $i, i = 1, 2, \dots, N$ 的各条部分路径的最大概率（以及取最大概率的状态）。于是在时刻 $t = T$ 的最大概率即为最优路径的概率 P^* ，最优路径的终点 i_T^* 也同时得到。
- 之后为了找出最优路径的各个结点，从终点 i_T^* 开始，由后向前逐步求得结点 i_{T-1}^*, \dots, i_1^* ，得到最优路径 $\mathbf{I}^* = (i_1^*, i_2^*, \dots, i_T^*)$ 。

3. 定义在时刻 t 状态为 i 的所有单个路径 (i_1, i_2, \dots, i_t) 中概率最大值为：

$$\delta_t(i) = \max_{i_1, i_2, \dots, i_{t-1}} P(i_t = i, i_{t-1}, \dots, i_1, o_t, \dots, o_1; \lambda), \quad i = 1, 2, \dots, Q$$

它就是算法导论中《动态规划》一章提到的“最优子结构”

则根据定义，得到变量 δ 的递推公式：

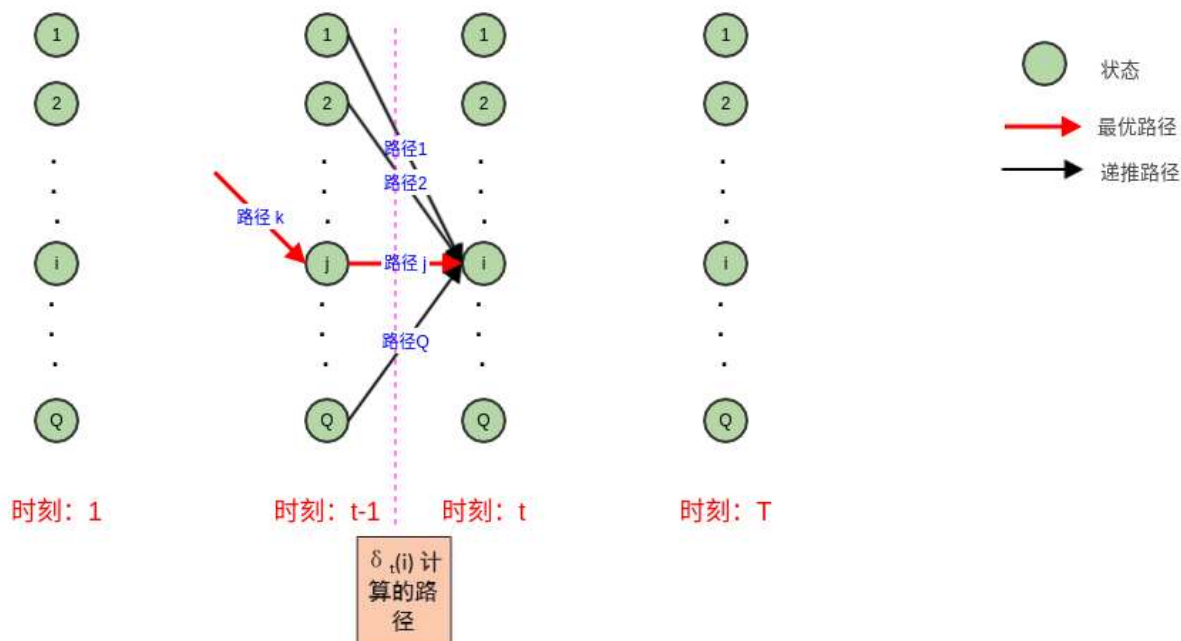
$$\delta_{t+1}(i) = \max_{i_1, i_2, \dots, i_t} P(i_{t+1} = i, i_t, \dots, i_1, o_{t+1}, \dots, o_1; \lambda) = \max_{1 \leq j \leq Q} \delta_t(j) \times a_{j,i} \times b_i(o_{t+1})$$

$$i = 1, 2, \dots, Q; t = 1, 2, \dots, T-1$$

4. 定义在时刻 t 状态为 i 的所有单个路径中概率最大的路径的第 $t-1$ 个结点为：

$$\Psi_t(i) = \arg \max_{1 \leq j \leq Q} \delta_{t-1}(j) a_{j,i}, \quad i = 1, 2, \dots, Q$$

它就是最优路径中，最后一个结点（其实就是时刻 t 的 \mathbf{q}_i 结点）的前一个结点。



5. 维特比算法:

◦ 输入:

- 隐马尔可夫模型 $\lambda = (\mathbf{A}, \mathbf{B}, \vec{\pi})$
- 观测序列 $\mathbf{O} = (o_1, o_2, \dots, o_T)$

◦ 输出: 最优路径 $\mathbf{I}^* = (i_1^*, i_2^*, \dots, i_T^*)$

◦ 算法步骤:

- 初始化: 因为第一个结点的之前没有结点, 所以有:

$$\delta_1(i) = \pi_i b_i(o_1), \Psi_1(i) = 0, \quad i = 1, 2, \dots, Q$$

- 递推: 对 $t = 2, 3, \dots, T$

$$\delta_t(i) = \max_{1 \leq j \leq Q} \delta_{t-1}(j) a_{j,i} b_i(o_t), \quad i = 1, 2, \dots, Q; t = 2, 3, \dots, T$$

$$\Psi_t(i) = \arg \max_{1 \leq j \leq Q} \delta_{t-1}(j) a_{j,i}, \quad i = 1, 2, \dots, Q$$

- 终止: $P^* = \max_{1 \leq i \leq Q} \delta_T(i), \quad i_T^* = \arg \max_{1 \leq i \leq Q} \delta_T(i)$ 。
- 最优路径回溯: 对 $t = T-1, T-2, \dots, 1: i_t^* = \Psi_{t+1}(i_{t+1}^*)$ 。
- 最优路径 $\mathbf{I}^* = (i_1^*, i_2^*, \dots, i_T^*)$ 。

三、最大熵马尔科夫模型MEMM

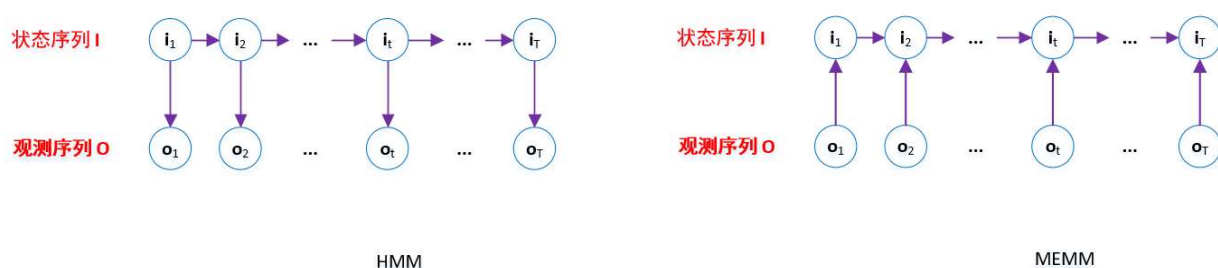
1. HMM 存在两个基本假设:

- 观察值之间严格独立。
- 状态转移过程中, 当前状态仅依赖于前一个状态 (一阶马尔科夫模型)。

如果放松第一个基本假设, 则得到最大熵马尔科夫模型 MEMM。

2. 最大熵马尔科夫模型并不通过联合概率建模, 而是学习条件概率 $P(i_t | i_{t-1}, o_t)$ 。

它刻画的是：在当前观察值 o_t 和前一个状态 i_{t-1} 的条件下，当前状态 i_t 的概率。



3. MEMM 通过最大熵算法来学习。

根据最大熵推导的结论：

$$P_{\vec{w}}(y | \vec{x}) = \frac{1}{Z_{\vec{w}}(\vec{x})} \exp \left(\sum_{i=1}^n w_i f_i(\vec{x}, y) \right)$$

$$Z_{\vec{w}}(\vec{x}) = \sum_y \exp \left(\sum_{i=1}^n w_i f_i(\vec{x}, y) \right)$$

这里 \vec{x} 就是当前观测 o_t 和前一个状态 i_{t-1} ，因此： $\vec{x} = (i_{t-1}, o_t)$ 。这里 y 就是当前状态 i_t ，因此： $y = i_t$ 。因此得到：

$$P_{\vec{w}}(i_t | i_{t-1}, o_t) = \frac{1}{Z_{\vec{w}}(i_{t-1}, o_t)} \exp \left(\sum_{i=1}^n w_i f_i(i_t, i_{t-1}, o_t) \right)$$

$$Z_{\vec{w}}(i_{t-1}, o_t) = \sum_{i_t} \exp \left(\sum_{i=1}^n w_i f_i(i_t, i_{t-1}, o_t) \right)$$

4. MEMM 的参数学习使用最大熵中介绍的 IIS 算法或者拟牛顿法，解码任务使用维特比算法。

5. 标注偏置问题：

如下图所示，通过维特比算法解码得到：

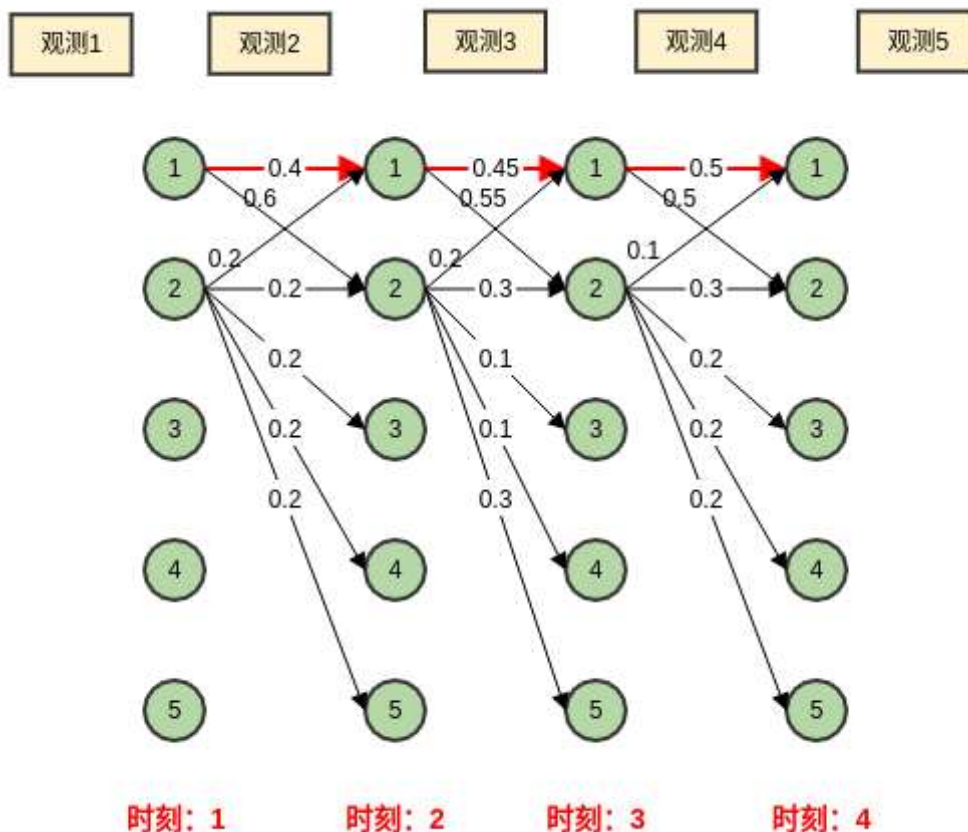
$$P(1 \rightarrow 1 \rightarrow 1 \rightarrow 1) = 0.4 \times 0.45 \times 0.5 = 0.09$$

$$P(2 \rightarrow 2 \rightarrow 2 \rightarrow 2) = 0.2 \times 0.3 \times 0.3 = 0.018$$

$$P(1 \rightarrow 2 \rightarrow 1 \rightarrow 2) = 0.6 \times 0.2 \times 0.5 = 0.06$$

$$P(1 \rightarrow 1 \rightarrow 2 \rightarrow 2) = 0.4 \times 0.55 \times 0.3 = 0.066$$

可以看到：维特比算法得到的最优路径为 $1 \rightarrow 1 \rightarrow 1 \rightarrow 1$ 。



- 实际上，状态 1 倾向于转换到状态 2；同时状态 2 也倾向于留在状态 2。但是由于状态 2 可以转化出去的状态较多，从而使得转移概率均比较小。

而维特比算法得到的最优路径全部停留在状态 1，这样与实际不符。

- MEMM 倾向于选择拥有更少转移的状态，这就是标记偏置问题。

6. 标记偏置问题的原因是：计算 $P_{\tilde{\mathbf{w}}}(i_t | i_{t-1}, o_t)$ 仅考虑局部归一化，它仅仅考虑指定位置的所有特征函数。

- 如上图， $P_{\tilde{\mathbf{w}}}(i_t | i_{t-1} = 2, o_t)$ 只考虑在 $(i_{t-1} = 2, o_t)$ 这个结点的归一化。

- 对于 $(i_{t-1} = 2, o_t)$ ，其转出状态较多，因此每个转出概率都较小。
- 对于 $(i_{t-1} = 1, o_t)$ ，其转出状态较少，因此每个转出概率都较大。

- CRF 解决了标记偏置问题，因为 CRF 是全局归一化的：

$$P(\mathbf{Y} | \mathbf{X}) = \frac{1}{Z} \exp \left(\sum_{j=1}^{K_1} \sum_{i=1}^{n-1} \lambda_j t_j(Y_i, Y_{i+1}, \mathbf{X}, i) + \sum_{k=1}^{K_2} \sum_{i=1}^n \mu_k s_k(Y_i, \mathbf{X}, i) \right)$$

它考虑了所有位置、所有特征函数。