第十二讲 隐马尔可夫模型

牟克典

2021年5月28日

概要

隐马尔可夫模型的基础是马尔可夫链.

- 马尔可夫链是刻画随机变量序列的概率分布的模型.
- 设$\{X_t|t=1,2,\cdots\}$ (简记为$\{X_t\}$)是随机序列，若X_t 都在S中取值，则称S是$\{X_t\}$的状态空间，S中的元素称为状态.
- 在S中只有有限个状态或可列个状态时，我们将其进行编号，记为$S=\{s_1,s_2,\cdots\}$.
- 如果对任何正整数$t\geq 2$和S中的状态 $s_i,s_j,s_{i_1},s_{i_2},\cdots,s_{i_{t-1}}$，随机序列$\{X_t\}$满足

$$\begin{aligned} & P(X_{t+1}=s_j|X_t=s_i,X_{t-1}=s_{i_{t-1}},\cdots,X_1=s_{i_1}) \\ = \quad & P(X_{t+1}=s_j|X_t=s_i) \\ = \quad & P(X_2=s_j|X_1=s_i), \end{aligned}$$

则称$\{X_t\}$为时齐的马尔可夫链，简称马氏链.

- 我们称

$$a_{ij} = P(X_2 = s_j | X_1 = s_i),\ s_i, s_j \in S$$

为马氏链$\{X_t\}$的转移概率.

- 称矩阵

$$A = [a_{ij}]$$

为马氏链$\{X_t\}$的一步转移概率矩阵，简称为转移矩阵.

- 设$|S| = N$，则转移矩阵为$N \times N$矩阵，且

$$\sum_{j=1}^{N} a_{ij} = 1,$$

即转移矩阵的各行之和为1.

- 如果X_1有概率分布

$$\pi_j = P(X_1 = s_j),\ s_j \in S.$$

则称X_1的分布列

$$\pi = (\pi_1, \pi_2, \cdots, \pi_N)$$

为$\{X_t\}$的初始分布.显然

$$\sum_{j=1}^{N} \pi_j = 1.$$

- 马氏链所满足的性质（通常称为马氏性）直观来说就是已知现在$X_t = s_i$,将来$X_{t+1} = s_j$与过去$X_{t-1} = s_{i_{t-1}}, \cdots, X_1 = s_{i_1}$独立.

- 对任何状态观测序列

$$X_t = s_{i_t}, X_{t-1} = s_{i_{t-1}}, \cdots, X_1 = s_{i_1}$$

来说，其似然

$$\begin{aligned}
& P(X_t = s_{i_t}, X_{t-1} = s_{i_{t-1}}, \cdots, X_1 = s_{i_1}) \\
= \; & P(X_t = s_{i_t} | X_{t-1} = s_{i_{t-1}}, \cdots, X_1 = s_{i_1}) \\
& \times P(X_{t-1} = s_{i_{t-1}}, \cdots, X_1 = s_{i_1}) \\
= \; & P(X_t = s_{i_t} | X_{t-1} = s_{i_{t-1}}) \\
& \times P(X_{t-1} = s_{i_{t-1}} | X_{t-2} = s_{i_{t-2}}) \\
& \times \cdots \times P(X_2 = s_{i_2} | X_1 = s_{i_1}) P(X_1 = s_{i_1}) \\
= \; & \pi_{i_1} \times a_{i_1 i_2} \times \cdots \times a_{i_{t-1} i_t}.
\end{aligned}$$

概要

- 在实际应用中并不一定能观测到马尔可夫链对应的状态序列.
- 隐马尔可夫模型(Hidden Markov Model，简称HMM)刻画
  - 由一个马尔可夫链随机生成不可观测的状态随机序列$\{X_t\}$,
  - 再由每个状态X_t生成一个观测O_t而生成观测随机序列$\{O_t\}$的过程.
- 假定O_t都在V中取值，且$V = \{\nu_1, \nu_2, \cdots, \nu_M\}$.
- 观测生成遵循观测独立性假设:

$$\begin{aligned} & P(O_t = \nu_{i_t} | X_t = s_{i_t}, O_{t-1} = \nu_{i_{t-1}}, \cdots, X_1 = s_{i_1}, O_1 = \nu_{i_1}) \\ = \quad & P(O_t = \nu_{i_t} | X_t = s_{i_t}) \end{aligned}$$

- 在转移矩阵$A = [a_{ij}]$和初始分布$\pi = (\pi_1, \pi_2, \cdots, \pi_N)$之外，再引进如下观测概率矩阵

$$B = \left[b_j(k)\right]_{N\times M},$$

其中

$$b_j(k) = P(O_t = \nu_k | X_t = s_j)$$

是马尔可夫链在时刻t状态为s_j的条件下生成观测ν_k的概率，这里$j = 1,2,\cdots, N; k = 1, 2, \cdots, M.$

- 利用转移矩阵A、初始分布π和观测概率矩阵B，就可以确定一个隐马尔可夫模型：
  - 被转移矩阵A和初始分布π刻画的马尔可夫链生成不可观测的状态序列，
  - 再依观测概率矩阵B从状态序列生成观测序列.
- 因此用转移矩阵A、初始分布π和观测概率矩阵B构成的三元组来表示一个隐马尔可夫模型λ，即$\lambda = (A, B, \pi)$.
- 给定$\lambda = (A, B, \pi)$，则按如下过程生成一个长度为T的观测序列$\{O_1, O_2, \cdots, O_T\}$：

(1) 令$t = 1$，并根据初始分布π产生状态X_1；
(2) 根据状态X_t和观测概率矩阵B生成观测O_t；
(3) 根据状态X_t和转移矩阵A生成状态X_{t+1}；
(4) 若$t < T$，令$t = t + 1$，并转到第(2)步，否则停止.

- 隐马尔可夫模型中状态和观测变量之间的相关关系：

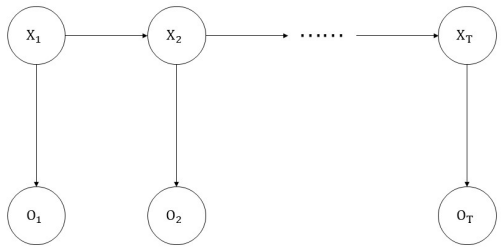

Figure: 隐马尔可夫模型

- 状态序列$\{X_1, X_2, \cdots, X_T\}$和观测序列$\{O_1, O_2, \cdots, O_T\}$的联合概率分布可以表示成如下形式：

$$P(X_1, O_1, \cdots, X_T, O_T) = P(X_1)P(O_1|X_1)\prod_{i=2}^{T} P(X_i|X_{i-1})P(O_i|X_i).$$

- 像隐马尔可夫模型这样把概率表示和图结构相结合的模型我们称之为概率图模型.

隐马尔可夫模型的三个基本问题：

(1) 概率(似然)计算问题： 给定隐马尔可夫模型$\lambda = (A, B, \pi)$和观测序列$\mathbf{O} = \{O_1, O_2, \cdots, O_T\}$， 计算该模型生成观测序列$\mathbf{O}$的概率$P(\mathbf{O}|\lambda)$。

(2) 解码问题：给定隐马尔可夫模型 $\lambda = (A, B, \pi)$ 和观测序列 $\mathbf{O} = \{O_1, O_2, \cdots, O_T\}$， 求与观测序列 $\mathbf{O}$ 最匹配的状态序列 $\mathbf{X} = \{X_1, X_2, \cdots, X_T\}$。

(3) 学习问题：给定观测序列$\mathbf{O} = \{O_1, O_2, \cdots, O_T\}$， 估计隐马尔可夫模型$\lambda = (A, B, \pi)$的参数使得似然$P(\mathbf{O}|\lambda)$最大。

概要

- 给定隐马尔可夫模型$\lambda = (A, B, \pi)$和观测序列$\mathbf{O} = \{O_1, O_2, \cdots, O_T\}$，则对任意的状态序列$\mathbf{X} = \{X_1, X_2, \cdots, X_T\}$，理论上我们可以计算出$P(\mathbf{X}, \mathbf{O}|\lambda)$，进而利用

$$P(\mathbf{O}|\lambda) = \sum_{\mathbf{X}} P(\mathbf{X}, \mathbf{O}|\lambda)$$

  来计算$P(\mathbf{O}|\lambda)$.
- 但是长度为T的不同状态序列一共有N^T个，因此在实际中当N和T比较大的时候这种方法的计算开销难以承受.
- 前向算法和后向算法:基于动态规划思想的概率计算方法.

- 如果考虑用观测序列与时刻T状态的联合概率

$$P(\mathbf{O}, X_T = s_i|\lambda),$$

来替代$P(\mathbf{X}, \mathbf{O}|\lambda)$，即考虑通过

$$P(\mathbf{O}|\lambda) = \sum_{i=1}^{N} P(\mathbf{O}, X_T = s_i|\lambda),$$

来计算$P(\mathbf{O}|\lambda)$，则只需要计算出N个联合概率$P(\mathbf{O}, X_T = s_i|\lambda)$，$i = 1,2,\cdots, N$.

- 更一般，定义前向概率$\alpha_t(i)$如下：

$$\alpha_t(i) = P(O_1, O_2, \cdots, O_t, X_t = s_i|\lambda),$$

即到时刻t观测序列为$O_1, O_2, \cdots, O_t$，且状态为s_i的概率.

- 显然

$$\alpha_T(i) = P(\mathbf{O}, X_T = s_i|\lambda),$$

并且

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

- 特别地，

$$\begin{aligned} & \alpha_1(i) = P(O_1, X_1 = s_i|\lambda) \\ = \quad & P(X_1 = s_i|\lambda)P(O_1|X_1 = s_i, \lambda) = \pi_i b_i(O_1). \end{aligned}$$

- 前向概率的递推公式

$$\begin{aligned}
\alpha_{t+1}(i) &= P(O_1, O_2, \cdots, O_t, O_{t+1}, X_{t+1} = s_i|\lambda) \\
&= \sum_{j=1}^{N} P(O_1, O_2, \cdots, O_t, O_{t+1}, X_t = s_j, X_{t+1} = s_i|\lambda) \\
&= \sum_{j=1}^{N} \alpha_t(j) P(O_{t+1}|X_{t+1} = s_i, \lambda) P(X_{t+1} = s_i|X_t = s_j, \lambda) \\
&= \sum_{j=1}^{N} \alpha_t(j) b_i(O_{t+1}) a_{ji} = \left(\sum_{j=1}^{N} a_{ji} \alpha_t(j) \right) b_i(O_{t+1}).
\end{aligned}$$

- 这意味着我们可以从$\{\alpha_1(i)\}_{i=1}^N$出发向前递推得到

$$\alpha_T(i) = P(\mathbf{O}, X_T = s_i|\lambda).$$

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

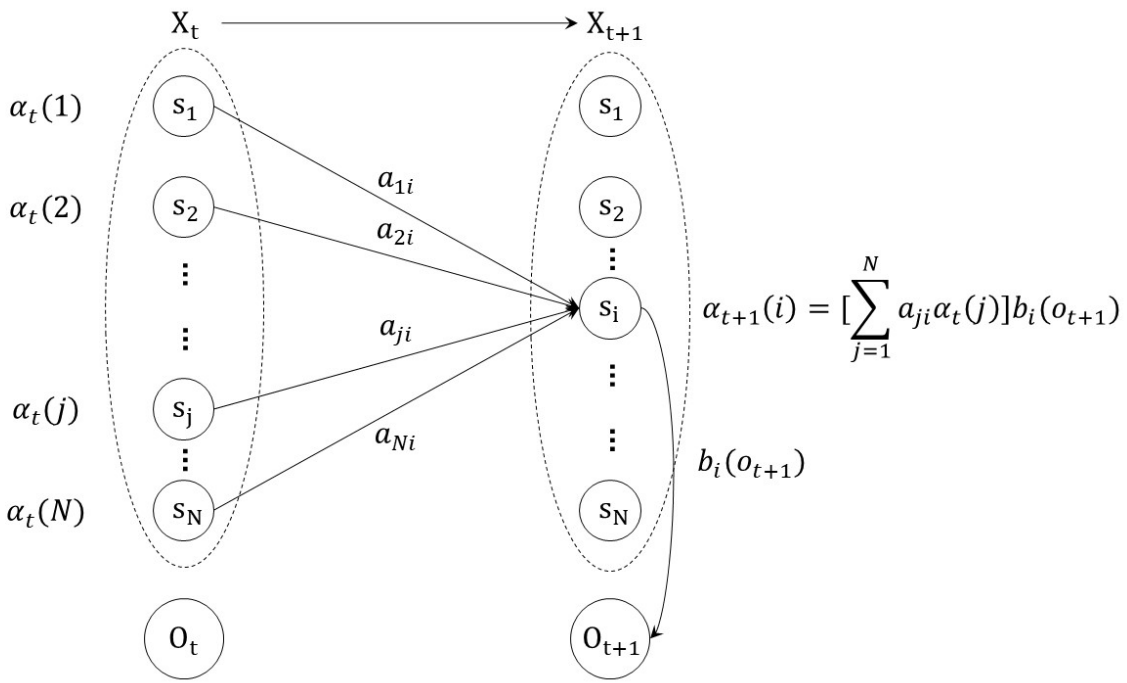

Figure: 前向算法

前向算法

输入: $\lambda = (A, B, \pi)$，观测序列 $\mathbf{O} = \{O_1, O_2, \cdots, O_T\}$;

输出: 观测序列概率$P(\mathbf{O}|\lambda)$.

1. **for** $i = 1, 2, \cdots, N$ **do**
2. $\quad \alpha_1(i) = \pi_i b_i(O_1)$;
3. **end for**
4. **for** $t = 1, 2, \cdots, T-1$ **do**
5. $\quad$ **for** $i = 1, 2, \cdots, N$ **do**
6. $\quad\quad \alpha_{t+1}(i) = \left(\sum_{j=1}^{N} \alpha_t(j) a_{ji} \right) b_i(O_{t+1})$;
7. $\quad$ **end for**
8. **end for**
9. $P(\mathbf{O}|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$;
10. **return** $P(\mathbf{O}|\lambda)$

- 如果考虑观测序列与初始时刻状态的联合概率

$$P(\mathbf{O}, X_1 = s_i|\lambda),$$

来替代$P(\mathbf{X}, \mathbf{O}|\lambda)$，即考虑通过

$$P(\mathbf{O}|\lambda) = \sum_{i=1}^{N} P(\mathbf{O}, X_1 = s_i|\lambda),$$

来计算$P(\mathbf{O}|\lambda)$.

- 注意到

$$\begin{aligned} & P(\mathbf{O}, X_1 = s_i|\lambda) \\ = \; & P(O_1|X_1 = s_i, \lambda)P(O_2, \cdots, O_T|X_1 = s_i, \lambda)P(X_1 = s_i|\lambda) \\ = \; & \pi_i b_i(O_1)P(O_2, \cdots, O_T|X_1 = s_i, \lambda), \end{aligned}$$

$$P(\mathbf{O}|\lambda) = \sum_{i=1}^{N} \pi_i b_i(O_1) P(O_2, \cdots, O_T | X_1 = s_i, \lambda).$$

- 如果能有效计算出N个条件概率

$$P(O_2, \cdots, O_T | X_1 = s_i, \lambda), \quad i = 1, 2, \cdots, N,$$

我们就可以得到$P(\mathbf{O}|\lambda)$.

- 对任意时刻$t < T$定义后向概率$\beta_t(i)$如下：

$$\beta_t(i) = P(O_{t+1}, \cdots, O_T | X_t = s_i, \lambda),$$

即在时刻T状态为S_i的条件下，从时刻$t+1$到T的部分观测序列为$O_{t+1}, \cdots, O_T$的概率.

- 特别地，

$$\beta_1(i) = P(O_2, \cdots, O_T | X_1 = s_i, \lambda), \ \text{ı} = 1, 2, \cdots, N.$$

- 规定对任意$1 \leq i \leq N$,

$$\beta_T(i) = 1,$$

- 对$1 \leq t \leq T-1$，后向概率有如下递推公式：

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j), \quad i = 1, 2, \cdots, N.$$

- $\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)$.

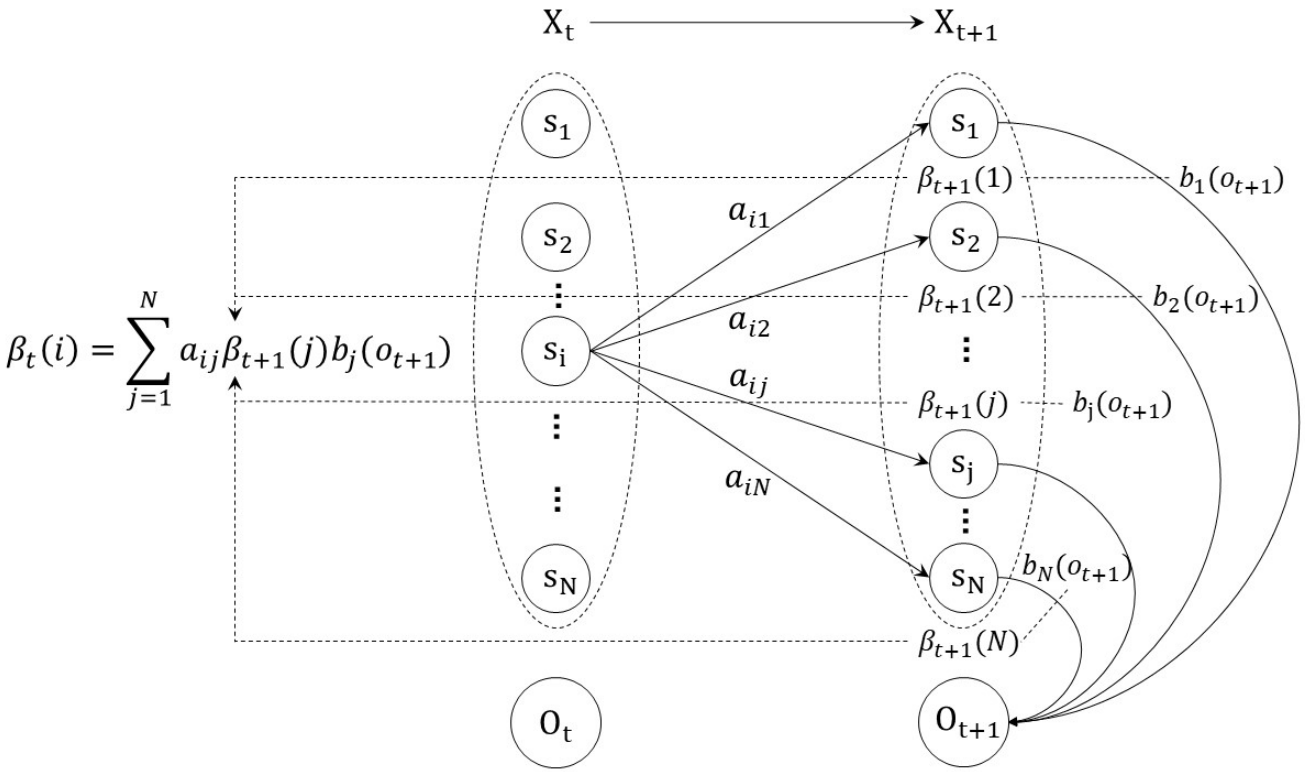

Figure: 后向算法

后向算法

输入: $\lambda = (A, B, \pi)$，观测序列$\mathbf{O} = \{O_1, O_2, \cdots, O_T\}$;

输出: 观测序列概率$P(\mathbf{O}|\lambda)$.

1. **for** $i = 1, 2, \cdots, N$ **do**
2. $\quad \beta_T(i) = 1;$
3. **end for**
4. **for** $t = T-1, T-2, \cdots, 1$ **do**
5. $\quad$ **for** $i = 1, 2, \cdots, N$ **do**
6. $\quad\quad \beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j);$
7. $\quad$ **end for**
8. **end for**
9. $P(\mathbf{O}|\lambda) = \sum_{i=1}^{N} \pi_i b_i(O_1) \beta_1(i);$
10. **return** $P(\mathbf{O}|\lambda)$

概要

解码问题

- 给定$\lambda = (A, B, \pi)$和长度为T的观测序列

$$\mathbf{O} = \{O_1, O_2, \cdots, O_T\},$$

求与观测序列$\mathbf{O}$最匹配的状态序列

$$\mathbf{X} = \{X_1, X_2, \cdots, X_T\}.$$

- 相当于找到使得$P(\mathbf{X}|\mathbf{O}, \lambda)$最大的状态序列$\mathbf{X}^*$，即

$$\mathbf{X}^* = \underset{\mathbf{X}}{\operatorname{argmax}} P(\mathbf{X}|\mathbf{O}, \lambda).$$

- 所求的最优状态序列$\mathbf{X}^*$也可以定义为

$$\mathbf{X}^* = \underset{\mathbf{X}}{\operatorname{argmax}} P(\mathbf{X}, \mathbf{O}|\lambda).$$

- 所求的最优状态序列$\mathbf{X}^*$也可以定义为

$$\mathbf{X}^* = \underset{\mathbf{X}}{\operatorname{argmax}} P(\mathbf{X}, \mathbf{O}|\lambda).$$

- 状态序列$\{X_1, X_2, \cdots, X_T\}$和观测序列$\{O_1, O_2, \cdots, O_T\}$的联合概率分布可以表示成如下形式：

$$P(\mathbf{X}, \mathbf{O}|\lambda) = P(X_1)P(O_1|X_1)\prod_{i=2}^{T} P(X_i|X_{i-1})P(O_i|X_i).$$

维特比算法:

- 如果把一个长度为T的状态序列按照状态序列的生成结构看做是一条从初始状态到时刻T状态的路径，则维特比算法实质上是用动态规划的思想来求最优路径.
- 考虑时刻 T 状态为 s_i （ $1 \le i \le N$ ）的所有单个路径

$$(X_1, X_2, \cdots, X_{T-1}, X_T = s_i)$$

所对应的概率的最大值为

$$\delta_T(i) = \max_{X_1, X_2, \cdots, X_{T-1}} P(X_1, X_2, \cdots, X_{T-1}, O_1, O_2, \cdots, O_T, X_T = s_i|\lambda),$$

显然对最优路径$\mathbf{X}^*$而言， $P(\mathbf{X}^*, \mathbf{O}|\lambda) = \max_{1 \le i \le N} \delta_T(i)$. 而且

$$X_T^* = \operatorname*{argmax}_{s_i,\ 1 \le i \le N} \delta_T(i).$$

- 基于动态规划思想，维特比算法定义 时刻t状态为s_i（$1 \leq i \leq N$）的所有单个路径$(X_1, X_2, \cdots, X_{t-1}, X_t = s_i)$所对应的概率的最大值为

$$\begin{aligned} & \delta_t(i) \\ = & \max_{X_1, X_2, \cdots, X_{t-1}} P(X_1, X_2, \cdots, X_{t-1}, O_1, O_2, \cdots, O_t, X_t = s_i | \lambda) \end{aligned}$$

特别地，

$$\delta_1(i) = \pi_i b_i(O_1).$$

- 路径最大概率$\delta_t(i)(2 \leq t \leq T)$的递推公式如下：

$$\delta_t(i) = \max_{1 \leq j \leq N} [\delta_{t-1}(j) a_{ji}] b_i(O_t), \ i = 1, 2, \cdots, N.$$

由此我们可以递推计算出$\delta_T(i)$，$i = 1, 2, \cdots, N$.

- 对$t \geq 2$，我们以$\Psi_t(s_i)$来记录时刻t状态为s_i的所有单个路径

$$(X_1, X_2, \cdots, X_{t-1}, X_t = s_i)$$

中概率最大的路径的第$t-1$个结点，即

$$\Psi_t(s_i) = \underset{s_j, 1 \leq j \leq N}{\operatorname{argmax}} \delta_{t-1}(j) a_{ji}, \ i = 1, 2, \cdots, N.$$

- 在找到X_T^*后，我们可以通过利用$\Psi_t(s_i)$从后向前依次得到最优状态序列.

维特比算法示意，其中红色表示到达当前时刻所关注状态的最优路径:

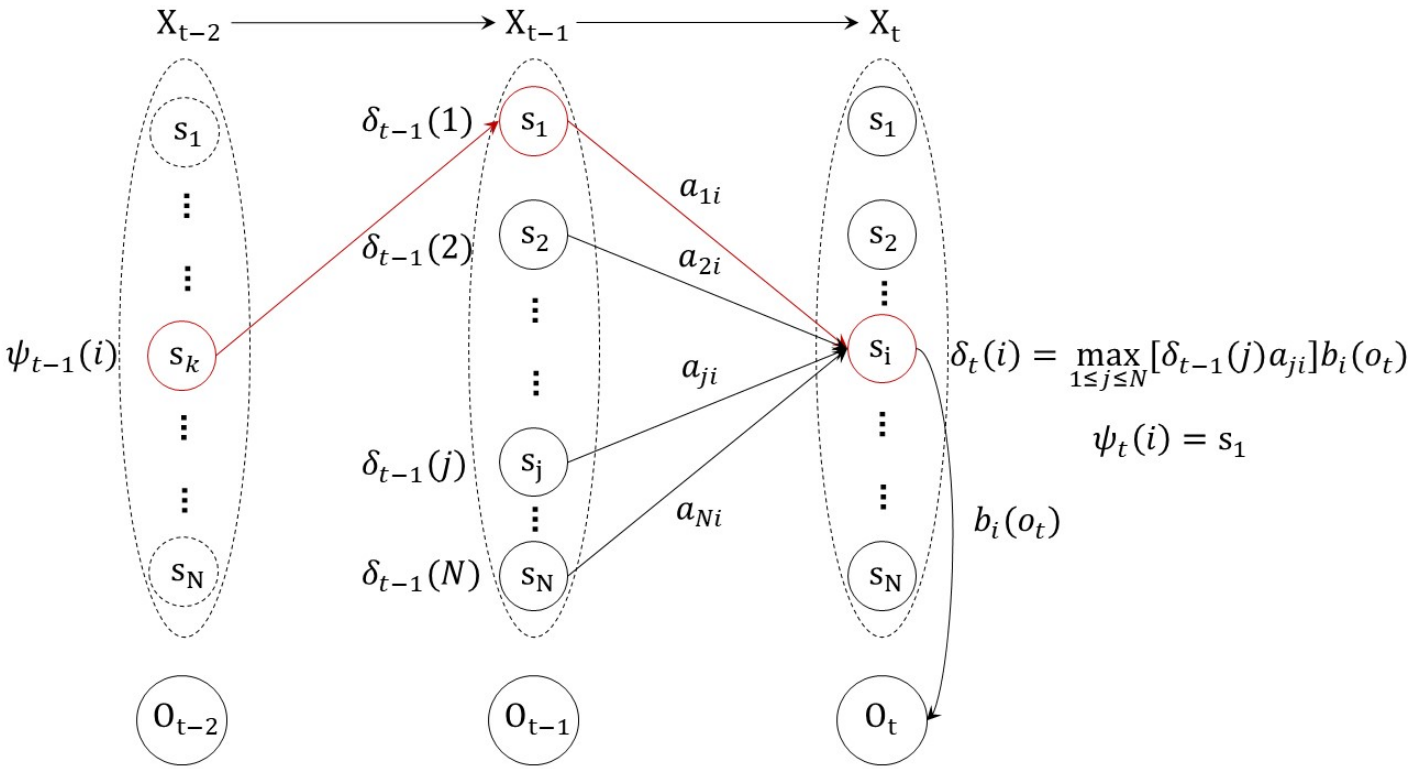

Figure: 维特比算法

维特比算法

输入: $\lambda = (A, B, \pi)$，观测序列$\mathbf{O} = \{O_1, O_2, \cdots, O_T\}$;

输出: 最优状态序列$\mathbf{X}^*$.

for $i = 1, 2, \cdots, N$ **do**
  $\delta_1(i) = \pi_i b_i(O_1)$;
  $\Psi_1(i) = 0$;
end for
for $t = 2, , 3, \cdots, T$ **do**
  for $i = 1, 2, \cdots, N$ **do**
    $\delta_t(i) = \max\limits_{1 \le j \le N} [\delta_{t-1}(j) a_{ji}] b_i(O_t)$;
    $\Psi_t(i) = \operatorname{argmax}\limits_{s_j, 1 \le j \le N} [\delta_{t-1}(j) a_{ji}]$;
  end for
end for

11 $P^* = \max_{1 \le i \le N} \delta_T(i)$;

12 $X_T^* = \underset{s_i, 1 \le i \le N}{\operatorname{argmax}} \delta_T(i)$;

13 **for** $t = T-1, T-2, \cdots, 1$ **do**

14 $\quad X_t^* = \Psi_{t+1}(X_{t+1}^*)$;

15 **end for**

16 **return** $\mathbf{X}^* = (X_1^*, X_2^*, \cdots, X_T^*)$

概要

如何从观测数据来学习隐马尔可夫模型的参数？

- 如果马尔可夫链是可见的，则可以利用极大似然法来估计隐马尔可夫模型的参数.
- 如果马尔可夫链是隐藏的，观测序列对应的状态序列不可观测，可以通过EM 算法来迭代求解.
- Baum-Welch算法是估计隐马尔可夫模型参数的EM算法.
- 假定状态空间S和观测可能的取值的集合V都是已知的.

Baum-Welch算法

- 给定长度为T的观测序列$\mathbf{O}=\{O_1,O_2,\cdots,O_T\}$，我们以$\mathbf{X}=\{X_1=s_{i_1},X_2=s_{i_2},\cdots,X_T=s_{i_T}\}$表示相应的状态序列（隐数据）.
- 我们用$\bar{\lambda}$表示隐马尔可夫模型参数的当前估计，则EM算法E步的Q函数为

$$\begin{aligned} Q(\lambda,\bar{\lambda}) &= \sum_{\mathbf{X}} P(\mathbf{X}|\mathbf{O},\bar{\lambda})\log P(\mathbf{O},\mathbf{X}|\lambda) \\ &= \sum_{\mathbf{X}} \frac{P(\mathbf{O},\mathbf{X}|\bar{\lambda})}{P(\mathbf{O}|\bar{\lambda})}\log P(\mathbf{O},\mathbf{X}|\lambda) \\ &= \frac{1}{P(\mathbf{O}|\bar{\lambda})}\sum_{\mathbf{X}} P(\mathbf{O},\mathbf{X}|\bar{\lambda})\log P(\mathbf{O},\mathbf{X}|\lambda). \end{aligned}$$

- 在Baum-Welch算法中直接将Q函数定义为

$$Q(\lambda, \bar{\lambda}) = \sum_{\mathbf{X}} P(\mathbf{O}, \mathbf{X}|\bar{\lambda}) \log P(\mathbf{O}, \mathbf{X}|\lambda).$$

- 我们前面已经提到过观测序列和状态序列的联合概率的如下表示:

$$P(\mathbf{O}, \mathbf{X}|\lambda) = \pi_{i_1} b_{i_1}(O_1) \prod_{t=1}^{T-1} a_{i_t i_{t+1}} b_{i_{t+1}}(O_{t+1}),$$

则Q函数可以进一步写成

$$Q(\lambda, \bar{\lambda}) = \sum_{\mathbf{X}} P(\mathbf{O}, \mathbf{X}|\bar{\lambda}) \log \left[\pi_{i_1} b_{i_1}(O_1) \prod_{t=1}^{T-1} a_{i_t i_{t+1}} b_{i_{t+1}}(O_{t+1}) \right].$$

- Q函数可以进一步写成

$$
\begin{aligned}
Q(\lambda, \bar{\lambda}) &= \sum_{\mathbf{X}} P(\mathbf{O}, \mathbf{X}|\bar{\lambda}) \log \left[\pi_{i_1} b_{i_1}(O_1) \prod_{t=1}^{T-1} a_{i_t i_{t+1}} b_{i_{t+1}}(O_{t+1}) \right] \\
&= \sum_{\mathbf{X}} P(\mathbf{O}, \mathbf{X}|\bar{\lambda}) [\log \pi_{i_1} + \log b_{i_1}(O_1) \\
&\quad + \sum_{t=1}^{T-1} (\log a_{i_t i_{t+1}} + \log b_{i_{t+1}}(O_{t+1}))] \\
&= \sum_{\mathbf{X}} P(\mathbf{O}, \mathbf{X}|\bar{\lambda}) \log \pi_{i_1} + \sum_{\mathbf{X}} P(\mathbf{O}, \mathbf{X}|\bar{\lambda}) \left[\sum_{t=1}^{T-1} \log a_{i_t i_{t+1}} \right] \\
&\quad + \sum_{\mathbf{X}} P(\mathbf{O}, \mathbf{X}|\bar{\lambda}) \left[\sum_{t=1}^{T} \log b_{i_t}(O_t) \right].
\end{aligned}
$$

- 令

$$
\begin{aligned}
Q_1 &= \sum_{\mathbf{X}} P(\mathbf{O}, \mathbf{X}|\bar{\lambda}) \log \pi_{i_1}, \\
Q_2 &= \sum_{\mathbf{X}} P(\mathbf{O}, \mathbf{X}|\bar{\lambda}) \left[\sum_{t=1}^{T-1} \log a_{i_t i_{t+1}} \right], \\
Q_3 &= \sum_{\mathbf{X}} P(\mathbf{O}, \mathbf{X}|\bar{\lambda}) \left[\sum_{t=1}^{T} \log b_{i_t}(O_t) \right],
\end{aligned}
$$

则

$$
Q(\lambda, \bar{\lambda}) = Q_1 + Q_2 + Q_3.
$$

- Baum-Welch算法的M步要通过极大化$Q(\lambda, \bar{\lambda})$来估计参数$\lambda = (A, B, \pi)$.
- 注意到λ、A和B分别出现在Q_1、Q_2和Q_3 中，因此我们分别通过极大化它们各自对应的部分来估计这些参数.
- 先来看$Q(\lambda, \bar{\lambda})$只含初始分布参数的第一部分

$$\begin{aligned} Q_1 &= \sum_{i=1}^{N} \left(\sum_{X_1 = s_i, X_2, \cdots, X_T} P(\mathbf{O}, \mathbf{X} | \bar{\lambda}) \log \pi_{i_1} \right) \\ &= \sum_{i=1}^{N} P(\mathbf{O}, X_1 = s_i | \bar{\lambda}) \log \pi_i. \end{aligned}$$

考虑初始分布参数满足的约束$\sum_{i=1}^{N} \pi_i = 1$，我们采用拉格朗日乘子法.

- 引进拉格朗日乘子γ来构造拉格朗日函数如下：

$$L_1(\gamma,\pi)=\sum_{i=1}^{N}P(\mathbf{O},X_1=s_i|\bar{\lambda})\log\pi_i+\gamma(1-\sum_{i=1}^{N}\pi_i).$$

- 令$\frac{\partial L_1(\gamma,\pi)}{\partial\pi_i}=0$得：

$$P(\mathbf{O},X_1=s_i|\bar{\lambda})=\gamma\pi_i.$$

两边对i求和得

$$\gamma=P(\mathbf{O}|\bar{\lambda}).$$

故此

$$\pi_i=\frac{P(\mathbf{O},X_1=s_i|\bar{\lambda})}{P(\mathbf{O}|\bar{\lambda})}=P(X_1=s_i|\mathbf{O},\bar{\lambda}).$$

- $Q(\lambda, \bar{\lambda})$只含转移概率参数的第二部分可以表示成

$$Q_2 = \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{t=1}^{T-1} P(\mathbf{O}, X_t = s_i, X_{t+1} = s_j | \bar{\lambda}) \log a_{ij},$$

- 对应的M步为

$$\max_{[a_{ij}]} \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{t=1}^{T-1} P(\mathbf{O}, X_t = s_i, X_{t+1} = s_j | \bar{\lambda}) \log a_{ij}$$

$$s.t. \quad \sum_{j=1}^{N} a_{ij} = 1, \quad i = 1, 2, \cdots, N.$$

- 应用拉格朗日乘子法可得

$$
\begin{aligned}
a_{ij} &= \frac{\sum_{t=1}^{T-1} P(\mathbf{O}, X_t = s_i, X_{t+1} = s_j|\bar{\lambda})}{\sum_{t=1}^{T-1} P(\mathbf{O}, X_t = s_i|\bar{\lambda})} \\
&= \frac{\sum_{t=1}^{T-1} \frac{P(\mathbf{O}, X_t=s_i, X_{t+1}=s_j|\bar{\lambda})}{P(\mathbf{O}|\bar{\lambda})}}{\sum_{t=1}^{T-1} \frac{P(\mathbf{O}, X_t=s_i|\bar{\lambda})}{P(\mathbf{O}|\bar{\lambda})}} = \frac{\sum_{t=1}^{T-1} P(X_t = s_i, X_{t+1} = s_j|\mathbf{O}, \bar{\lambda})}{\sum_{t=1}^{T-1} P(X_t = s_i|\mathbf{O}, \bar{\lambda})}.
\end{aligned}
$$

- $Q(\lambda, \bar{\lambda})$只含观测概率参数的第三部分可以表示成

$$Q_3 = \sum_{j=1}^{N} \sum_{t=1}^{T} P(\mathbf{O}, X_t = s_j | \bar{\lambda}) \log b_j(O_t),$$

- 对应的M步为

$$\max_{[b_j(k)]} \sum_{j=1}^{N} \sum_{t=1}^{T} P(\mathbf{O}, X_t = s_j | \bar{\lambda}) \log b_j(O_t)$$

$$s.t. \quad \sum_{k=1}^{M} b_j(k) = 1, \quad j = 1, 2, \cdots, N.$$

- 注意到在观测给定的情况下，目标函数中出现的参数集$\{b_j(O_t)\}$不一定能够覆盖我们所要估计的参数集$\{b_j(k)\}$，因此我们将目标函数中的$\log b_j(O_t)$ 改写成

$$\sum_{k=1}^{M} I(O_t = \nu_k) \log b_j(k),$$

引入所有我们需要估计的参数，

- 并 应用拉格朗日乘子法可得

$$b_j(k) = \frac{\sum_{t=1}^{T} P(\mathbf{O}, X_t = s_j|\bar{\lambda}) I(O_t = \nu_k)}{\sum_{t=1}^{T} P(\mathbf{O}, X_t = s_j|\bar{\lambda})}.$$

$$
\begin{aligned}
b_j(k) &= \frac{\sum_{t=1}^{T} P(\mathbf{O}, X_t = s_j|\bar{\lambda}) I(O_t = \nu_k)}{\sum_{t=1}^{T} P(\mathbf{O}, X_t = s_j|\bar{\lambda})} \\
&= \frac{\sum_{t=1}^{T} \frac{P(\mathbf{O}, X_t = s_j|\bar{\lambda}) I(O_t = \nu_k)}{P(\mathbf{O}|\bar{\lambda})}}{\sum_{t=1}^{T} \frac{P(\mathbf{O}, X_t = s_j|\bar{\lambda})}{P(\mathbf{O}|\bar{\lambda})}} \\
&= \frac{\sum_{t=1}^{T} P(X_t = s_j|\mathbf{O}, \bar{\lambda}) I(O_t = \nu_k)}{\sum_{t=1}^{T} P(X_t = s_j|\mathbf{O}, \bar{\lambda})}.
\end{aligned}
$$

- Baum-Welch算法的M步关于三类参数的更新依赖于如下两组概率计算：

$$P(X_t = s_i|\mathbf{O}, \bar{\lambda}),\ 1 \le t \le T; 1 \le i \le N.$$

和

$$P(X_t = s_i, X_{t+1} = s_j|\mathbf{O}, \bar{\lambda}),\ 1 \le t \le T-1; 1 \le i, j \le N.$$

- 令

$$\gamma_t(i|\lambda) = P(X_t = s_i|\mathbf{O}, \lambda),$$

则

$$\gamma_t(i|\lambda) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)}.$$

- 其次，我们令

$$\xi_t(i,j|\lambda) = P(X_t = s_i, X_{t+1} = s_j|\mathbf{O}, \lambda),$$

则

$$\xi_t(i,j|\lambda) = \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{\sum_{l=1}^{N}\sum_{k=1}^{N} \alpha_t(l)a_{lk}b_k(O_{t+1})\beta_{t+1}(k)}$$

Baum-Welch算法参数更新可以表示为：

$$
\begin{aligned}
\pi_i &= \gamma_1(i|\bar{\lambda}), \\
a_{ij} &= \frac{\sum_{t=1}^{T-1} \xi(i,j|\bar{\lambda})}{\sum_{t=1}^{T-1} \gamma_t(i|\bar{\lambda})} \\
b_j(k) &= \frac{\sum_{t=1,O_t=\nu_k}^{T} \gamma_t(i|\bar{\lambda})}{\sum_{t=1}^{T} \gamma_t(i|\bar{\lambda})}
\end{aligned}
$$

这里$i,j = 1,2,\cdots,N; K = 1,2,\cdots,M.$

Baum-Welch算法

输入： 观测序列$\mathbf{O}=\{O_1,O_2,\cdots,O_T\}$

输出： 隐马尔可夫模型$\lambda=(A,B,\pi)$

(1) 初始化：$\lambda^{(0)}=(A^{(0)},B^{(0)},\pi^{(0)}),\ n=0$

(2) 重复如下迭代直到收敛

(a) 计算 $\gamma_t(i|\lambda^{(n)}),\ t=1,2,\cdots,T;\ i=1,2,\cdots,N$

(b) 计算 $\xi_t(i,j|\lambda^{(n)})$,
$t=1,2,\cdots,T;\ i=1,2,\cdots,N;\ j=1,2,\cdots,N$

(c) 计算 $\pi_i^{(n+1)}=\gamma_1(i|\lambda^{(n)}),\ i=1,2,\cdots,N$

(2) 重复如下迭代直到收敛

(d) 计算 $a_{ij}^{(n+1)} = \dfrac{\sum_{t=1}^{T-1} \xi(i,j|\lambda^{(n)})}{\sum_{t=1}^{T-1} \gamma_t(i|\lambda^{(n)})}, i = 1,2,\cdots,N; j = 1,2,\cdots,N$

(e) 计算 $b_j^{(n+1)}(k) = \dfrac{\sum_{t=1, O_t=\nu_k}^{T} \gamma_t(i|\lambda^{(n)})}{\sum_{t=1}^{T} \gamma_t(i|\lambda^{(n)})}$,

$j = 1,2,\cdots,N; k = 1,2,\cdots,M$

(f) $n = n+1$

(3) 得到模型参数 $\lambda^{(n)} = (A^{(n)}, B^{(n)}, \pi^{(n)})$.

小结

- 隐马尔可夫模型
  - 概率计算问题
  - 解码问题
  - 学习问题
- 前向算法
- 后向算法
- 维特比算法
- Baum-Welch算法

概要

1. 机器学习概论 √
2. 监督学习 √
   - 决策树 √
   - 支持向量机与核方法 √
   - 基于近邻的方法 √
   - 朴素贝叶斯法与Logistic 回归模型 √
   - 神经网络初步 √
   - 回归模型 √
   - 集成学习 √
3. 非监督学习 √
   - 聚类 √
4. 概率图模型 √
   - 隐马尔可夫模型 √
5. 计算学习理论 √?