

# 模式识别

HMM: Hidden Markov Model - part 2  
隐马尔科夫模型第二部分

吴建鑫  
南京大学计算机系, 2018

# Evaluation

---

# 假设隐状态已知

- ✓ 已知 $\lambda, o_{1:T}$ , 求 $P(o_{1:T}|\lambda)$
- ✓ 若假设oracle已告知所有的隐变量的值 $q_{1:T}$ 
  - $P(o_{1:T}|\lambda, q_{1:T}) = \prod_{i=1}^T P(o_t|q_t, \lambda) = \prod_{i=1}^T b_{q_i}(o_i)$
  - 证明? 含义?
  - $\lambda$ 的存在只是表明概率的大小是基于该模型参数计算的, 可以去除而不影响计算
- ✓ 关于各随机变量之间的独立性的判断, 进一步参阅PRML第八章

# 一种naïve的计算方法

✓ 那么隐变量序列 $q_{1:T}$ 的可能性多大呢？

- $P(q_{1:T}|\lambda) = \pi_{q_1} A_{q_1 q_2} A_{q_2 q_3} \cdots A_{q_{T-1} q_T}$

- 含义？

✓ 用全概率公式对**所有可能的** $q_{1:T}$ 求和可以得到 $P(o_{1:T}|\lambda)$

- $P(o_{1:T}|\lambda) = \sum_{\text{all } q} P(o_{1:T}|\lambda, q_{1:T})P(q_{1:T}|\lambda)$ , 复杂度？

- $O(T \times N^T)$

✓ 虽然不实用，但可以从中学到一种思考问题的方法

- 后面EM学习算法用相似的思路

# 那么，如何快速计算？

✓ 动态规划！

✓ 只看最后一步 ( $t = T$ )，该如何计算？

1. 最后一步 ( $t = T$ ) 时一共可能有  $N$  种状态： $q_T = S_1, \dots, S_N$ ，其概率  $P(o_{1:T-1}, q_T = S_i | \lambda) = ?$
2. 若最后一步状态为  $S_i$ ，那么观察到输出  $o_T$  的概率是多少？
3. 所求的值是多少？（全概率公式）

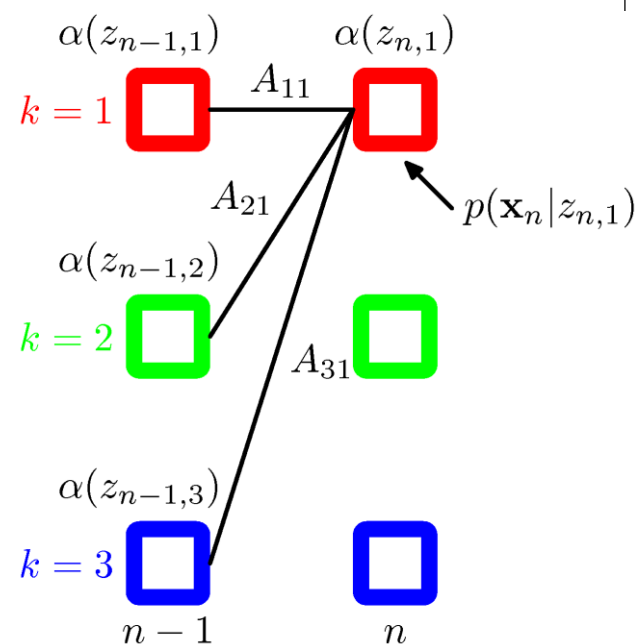
$$P(o_{1:T} | \lambda) = \sum_{i=1}^N P(o_{1:T-1}, q_T = S_i | \lambda) b_{S_i}(o_T)$$

- 只限于最后一步吗？

## 快速计算 (2)

✓ 如何计算  $P(o_{1:T-1}, q_T = S_i | \lambda)$ ?

- PRML Fig. 13.12
- 有  $N$  种可能, 即  $T-1$  时刻状态为  $q_{T-1} = S_j$ ,  $j = 1, 2, \dots, N$ , 然后通过概率  $A_{ji}$  转移
- 全概率公式, again!



$$P(o_{1:T-1}, q_T = S_i | \lambda) = \sum_{j=1}^N P(o_{1:T-1}, q_{T-1} = S_j | \lambda) A_{ji}$$

# 快速计算小结

✓ 
$$P(o_{1:T}|\lambda) = \sum_{i=1}^N P(o_{1:T-1}, q_T = S_i|\lambda) b_{S_i}(o_T) = \sum_{i=1}^N \left( b_{S_i}(o_T) \sum_{j=1}^N P(o_{1:T-1}, q_{T-1} = S_j|\lambda) A_{ji} \right)$$

✓ 红色部分是什么？

- 一个规模小一点的相同问题 ( $T - 1$ )
- 但是需要对所有  $j$  的可能取值计算
- 正如DTW中一样，可以通过动态规划解决，但是需要解决比原问题更多数目的小规模子问题
- 但是，复杂的是，目前牵涉两个数值而不是一个： $P(o_{1:T-1}, q_T = S_i|\lambda)$  和  $P(o_{1:T}|\lambda)$
- 计算的方向应该是什么？

# 动态规划算法（前向forward算法）

✓  $P(o_{1:T}|\lambda) = \sum_{i=1}^N P(o_{1:T-1}, q_T = S_i|\lambda) b_{S_i}(o_T) =$   
 $\sum_{i=1}^N (b_{S_i}(o_T) \sum_{j=1}^N P(o_{1:T-1}, q_{T-1} = S_j|\lambda) A_{ji})$

✓ 定义

- $\alpha_t(i) = P(o_{1:t}, q_t = S_i|\lambda)$  - 含义是?
- Initialization:  $\alpha_1(i) = \pi_i b_{S_i}(o_1), \quad 1 \leq i \leq N$
- Induction: For  $1 \leq t \leq T-1$

$$\alpha_{t+1}(i) = \left[ \sum_{j=1}^N \alpha_t(j) A_{ji} \right] b_{S_i}(o_{t+1}), \quad 1 \leq i \leq N$$

- Termination (output):  $P(o_{1:T}|\lambda) = \sum_{i=1}^N \alpha_T(i)$



# 后向算法backward algorithm

- ✓ 定义  $\beta_t(i) = P(o_{t+1:T} | q_t = S_i, \lambda)$ 
  - 若在时刻  $t$  状态为  $S_i$ ，将来观测到  $o_{t+1:T}$  的概率
- ✓ 初始化：  $\beta_T(i) = 1, 1 \leq i \leq N$
- ✓ 反向更新：  $t = T - 1, T - 2, \dots, 2, 1$

$$\beta_t(i) = \sum_{j=1}^N A_{ij} b_{S_j}(o_{t+1}) \beta_{t+1}(j), \quad 1 \leq i \leq N$$

- ✓ 输出：  $\beta_1(i) = P(o_{2:T} | q_1 = S_i, \lambda)$

$$P(o_{1:T} | \lambda) = \sum_{i=1}^N \pi_i b_{S_i}(o_1) \beta_1(i)$$

# Decoding

---

# 发现“最好”的隐变量值

✓ 标准1：对于每个时刻，发现其后验概率最大的状态

- 定义  $\gamma_t(i) = P(q_t = S_i | o_{1:T}, \lambda)$ ，当观测到输出为  $o_{1:T}$  时，时刻  $t$  时隐变量为第  $i$  个状态的后验概率
- 那么，对于一个输出序列  $o_{1:T}$ ，选择

$$q_t = \operatorname{argmax}_{1 \leq i \leq N} \gamma_t(i), \quad t = 1, 2, \dots, T$$

- 可能出现什么问题？
- 不存在这样的路径  $q_{1:T}$

# 怎样计算 $\gamma$

✓  $\alpha_t(i)\beta_t(i) = P(o_{1:T}, q_t = S_i | \lambda)$

- 为什么?

✓ 贝叶斯定理

$$\gamma_t(i) = P(q_t = S_i | o_{1:T}, \lambda) = \frac{P(o_{1:T}, q_t = S_i | \lambda)}{P(o_{1:T} | \lambda)} = \frac{\alpha_t(i)\beta_t(i)}{P(o_{1:T} | \lambda)}$$

- $P(o_{1:T} | \lambda) = \sum_{i=1}^N \alpha_t(i)\beta_t(i)$  for any  $t$ !

- 三种计算方法计算 $P(o_{1:T} | \lambda)$ 了

✓ 或者 1)  $\gamma_i = \alpha_t(i)\beta_t(i)$  2) L1 normalize:  $\gamma_i \leftarrow \frac{\gamma_i}{\sum_i \gamma_i}$

# 寻找最大概率的路径

- ✓ 一共有  $N^T$  种可能的路径，有些的概率可能为0
  - 比如通过准则1得到的路径
  - 那么，如果寻找所有可能路径里面概率最大的那个呢？
$$q_{1:T} = \underset{Q_{1:T}}{\operatorname{argmax}} P(Q_{1:T} | o_{1:T}, \lambda) = \underset{Q_{1:T}}{\operatorname{argmax}} P(Q_{1:T}, o_{1:T} | \lambda)$$
- ✓ Naïve的方法复杂性是  $N^T$ ，有没有更好的方法？
  - Viterbi方法
  - 猜猜这是一种什么类型的方法？
  - Andrew J. Viterbi, USC的工程学院以其命名

# Viterbi decoding

✓  $q_{1:T} = \underset{Q_{1:T}}{\operatorname{argmax}} P(Q_{1:T}, o_{1:T} | \lambda)$

✓ 定义更多的子问题

$$\delta_t(i) = \max_{q_{1:t-1}} P(q_{1:t-1}, q_t = S_i, o_{1:t} | \lambda)$$

- 含义：当限定两个条件1) 前 $t$ 个时刻的输出为 $o_{1:t}$ ，2) 第 $t$ 个时刻的隐状态为第 $i$ 个状态的时候，最佳路径所能取得的最大概率
- 怎么取得 $q_t$ ?
  - 用另外一个变量 $\psi_t(i)$ 做记录
- 怎么从 $t$ 进展到 $t + 1$ ?

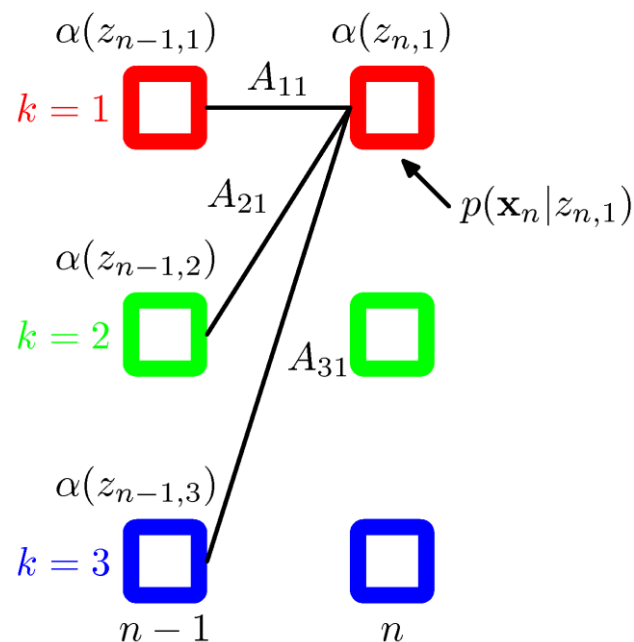
# 两个步骤

✓ 从 $t$ 进展到 $t + 1$

- $\delta_{t+1}(i) = \max_j ([\delta_t(j)A_{ji}] b_{S_i}(o_{t+1}))$
- $\delta_{t+1}(i)$ 是概率，如果只需要发现概率最大那个状态， $b_i(o_{t+1})$ ？

✓ 所以在时刻 $t + 1$ ，需要用另外一个变量 $\psi_t(i)$ 记录最大概率的路径在时刻 $t$ 是哪一个状态

- $\psi_{t+1}(i) = \operatorname{argmax}_{1 \leq j \leq N} ([\delta_t(j)A_{ji}])$



# Viterbi算法

✓ 初始化:  $\delta_1(i) = \pi_i b_{S_i}(o_1)$ ,  $\psi_1(i) = 0$ ,  $1 \leq i \leq N$

✓ 递归:  $2 \leq t \leq T$ ,  $1 \leq i \leq N$

$$\delta_t(i) = \max_{1 \leq j \leq N} ([\delta_{t-1}(j) A_{ji}] b_{S_i}(o_t))$$

$$\psi_t(i) = \operatorname{argmax}_{1 \leq j \leq N} ([\delta_{t-1}(j) A_{ji}])$$

✓ 输出:

• 最大概率:  $P^* = \max_{1 \leq i \leq n} \delta_T(i)$

• 时刻 $T$ 的最佳路径变量:  $q_T^* = \operatorname{argmax}_{1 \leq i \leq N} (\delta_T(i))$

• 时刻 $T-1, T-2, \dots, 2, 1$ 的最佳路径变量:  $q_{\textcolor{red}{t}}^* = \psi_{\textcolor{red}{t}+1}(q_{t+1}^*)$



# 分析

- ✓ 问题1的动态规划  $\alpha_{t+1}(i) = \sum_{j=1}^N \alpha_t(j) A_{ji}$
- ✓ 问题2的动态规划  $\delta_t(i) = \max_j ([\delta_{t-1}(j) A_{ji}] b_i(o_t))$
- ✓ 最重要的操作分别是sum-product和max-product
  - 其复杂性均为  $N^2T$
  - 和naïve方法的  $TN^T$  比较，极其巨大的速度提高
- ✓ 进一步阅读：sum-product和max-product是更为通用的算法，在图模型graphical model中有极为广泛的应用。

# 问题3：学习系统的参数

- ✓ 发现  $\lambda = (A, B, \pi)$ ，使得对于固定的  $N$ ， $T$ ，和观察值  $\mathbf{O}$ ，**似然** (likelihood)  $P(\mathbf{O}|\lambda)$  最大
- 目前没有方法能发现全局最优的解
- 常用的方法是 Baum-Welch 算法，发现一个局部最优的解
- **进一步** 阅读，Baum-Welch 是 EM 方法的一种具体体现，更多内容可参考上个 PPT 的进一步阅读部分，EM 算法的一个 tutorial
  - 可参考我的 EM Note