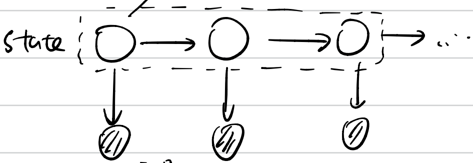
HMM

①

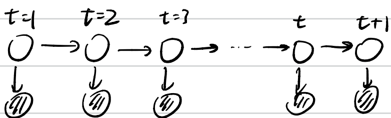
隐变量



(观测变量)

State { 离散 → HMM
连续 { 线性 → Kalman Filter
非线性 → Particle Filter

② HMM(一个模型)



状态转移矩阵

$\lambda = (\pi, A, B)$ → 发射矩阵

π ↓ 初始 prob dist

- 观测变量 : o $\quad o_1, o_2, \cdots o_t \cdots \rightarrow V = \{v_1, v_2, \cdots v_m\}$

  状态变量 : i $\quad i_1, i_2, \cdots i_t \cdots \rightarrow Q = \{q_1, q_2, \cdots q_n\}$

- $A = [a_{ij}] \quad a_{ij} = p(i_{t+1} = q_j \mid i_t = q_i)$

  $B = [b_{jk}] \quad b_{jk} = p(o_t = v_k \mid i_t = q_j)$ 在状态为 q_j 时，观测到 v_k 的概率

③ 两个假设 : (i) 齐次 Markov 假设 (ii) 观察独立假设.

(i) $p(i_{t+1} \mid i_t, i_{t-1}, \cdots i_1, o_t, o_{t-1}, \cdots o_1) = p(i_{t+1} \mid i_t)$

(ii) $p(o_t \mid i_t, i_{t-1}, \cdots i_1, o_{t-1}, o_{t-2}, \cdots o_1) = p(o_t \mid i_t)$

④ 三个问题 {

Evaluation : $p(O \mid \lambda)$ → 前向后向

Learning : λ 如何求 → EM

$\lambda = \arg\max p(O \mid \lambda)$

Decoding (i) 找到一个状态序列 $I = i_1, i_2 \cdots i_t$，使得 $p(I \mid O)$ 可以达到最大

(ii) { 滤波 : → $p(i_t \mid o_1, o_2, \cdots o_t)$

预测 → $p(i_{t+1} \mid o_1, o_2, \cdots o_t)$

⑤ evaluation 问题: Given λ 求 $p(O|\lambda)$

联合概率乘法

$$p(O|\lambda) = \sum_I p(I,O|\lambda) = \sum_I p(O|\lambda,I)\,p(I|\lambda)$$

$$p(I|\lambda) = p(i_1, i_2, \cdots i_t|\lambda) = \underline{p(i_t|i_1, i_2, \cdots i_{t-1}, \lambda)} \cdot \underbrace{p(i_1, i_2, \cdots i_{t-1}|\lambda)} = a_{i_{t-1},i_t} \cdot a_{i_{t-2},i_{t-1}} \cdots a_{i_2,i_1} \cdot \pi(i_1)$$

假设 $= p(i_t|i_{t-1}) \downarrow a_{i_{t-1},i_t}$

$$= \pi(i_1) \prod_{t=2}^{T} a_{i_{t-1},i_t}$$

运用假设2

$$p(O|\lambda, I) = \prod_{i=2}^{T} b_{i_t}(O_t)$$

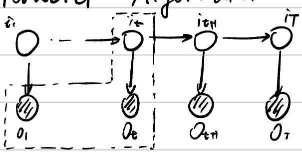
Δ

$$\therefore\ p(O|\lambda) = \sum_I \pi(i_1) \prod_{t=2}^{T} a_{i_{t-1},i_t} \cdot \prod_{i=2}^{T} b_{i_t}(O_t) = \sum_{i_1} \underbrace{\sum_{i_2} \sum_{i_3} \cdots \sum_{i_t}} \pi(i_1) \prod_{t=2}^{T} a_{i_{t-1},i_t} \cdot \prod_{i=2}^{T} b_{i_t}(O_t)$$

N 复杂度 $O(N^T)$

计算量非常大

Δ Forward Algorithm $O(TN^2)$



记 $\alpha_t(i) = p(O_1 \cdots O_t, i_i = q_t|\lambda)$

$\alpha_T(i) = p(O, i_t = q_T|\lambda)$

$$p(O|\lambda) = \sum_{i=1}^{N} p(O, i_t = q_T|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

$$\boxed{\alpha_{t+1}(j)} = p(O_1, \cdots O_t, O_{t+1}, i_{t+1} = q_j|\lambda)$$

$$= \sum_{i=1}^{N} p(O_1, \cdots O_t, O_{t+1}, i_{t+1} = q_j, i_t = q_i|\lambda)$$

$$= \sum_{i=1}^{N} p(O_{t+1}|O_1, \cdots O_t, i_t = q_i, i_{t+1} = q_j, \lambda) \cdot p(O_1, \cdots O_t, i_t = q_i, i_{t+1} = q_j|\lambda)$$

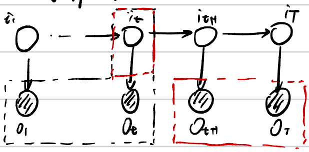
$$= \sum_{i=1}^{N} p(O_{t+1}|i_{t+1})\,p(O_1, \cdots O_t, i_t = q_i, i_{t+1} = q_j|\lambda)$$

$$= \sum_{i=1}^{N} p(O_{t+1}|i_{t+1})\,\underline{p(i_{t+1} = q_j|O_1, \cdots O_t, i_t = q_i, \lambda)} \cdot \boxed{p(O_1, \cdots O_t, i_t = q_i|\lambda)}$$

$\hookrightarrow p(i_{t+1} = q_j|i_t = q_i, \lambda)$

$$= \sum_{i=1}^{N} \underbrace{p(O_{t+1}|i_{t+1})}_{b_{q_j}(O_{t+1})}\,\underbrace{p(i_{t+1} = q_j|i_t = q_i, \lambda)}_{a_{ij}} \cdot \boxed{\alpha_t(i)}$$

4 后向算法 $O(TN^2)$

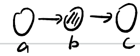


记 $\beta_t(i) = p(o_{t+1},\dots,o_T \mid i_t = q_i, \lambda)$

$\beta_1(i) = p(o_2,\dots,o_T \mid i_1 = q_i, \lambda)$

$$
\begin{aligned}
p(O|\lambda) &= p(o_1,\dots,o_T|\lambda) \\
&= \sum_{i=1}^{N} p(o_1,\dots,o_T, i_1 = q_i) \\
&= \sum_{i=1}^{N} p(o_1,\dots,o_T \mid i_1 = q_i)\, \overbrace{\boxed{p(i_1 = q_i)}}^{\pi_i} \\
&= \sum_{i=1}^{N} p(o_1 \mid o_2 \cdots o_T, i_1 = q_i)\, p(o_2,\dots,o_T \mid i_1 = q_i)\, \pi_i \\
&= \sum_{i=1}^{N} p(o_1 \mid i_1 = q_i)\, \beta_1(i)\, \pi_i \\
&= \sum_{i=1}^{N} b_i(o_1)\, \pi_i\, \beta_1(i)
\end{aligned}
$$

$$
\begin{aligned}
\beta_t(i) &= p(o_{t+1},\dots,o_T \mid i_t = q_i) \\
&= \sum_{j=1}^{N} p(o_{t+1},\dots,o_T, i_{t+1} = q_j \mid i_t = q_i) \\
&= \sum_{j=1}^{N} p(o_{t+1},\dots,o_T \mid i_{t+1} = q_j, i_t = q_i)\, \underbrace{p(i_{t+1} = q_j \mid i_t = q_i)}_{a_{ij}} \\
&= \sum_{j=1}^{N} p(o_{t+1},\dots,o_T \mid i_{t+1} = q_j)
\end{aligned}
$$

利用贝叶斯网络 

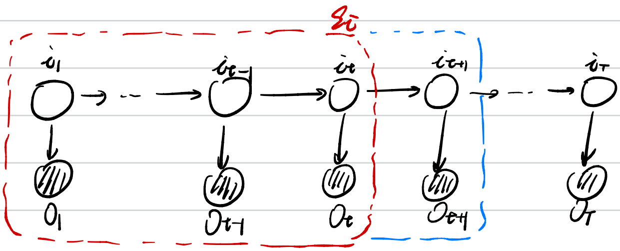
一旦给定 b，a、c 相互独立

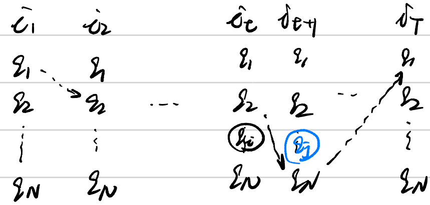
$$
\begin{aligned}
&= \sum_{j=1}^{N} \overbrace{p(o_{t+1} \mid o_{t+2},\dots,o_T, i_{t+1} = q_j)}^{b_j(o_{t+1})} \cdot \underbrace{p(o_{t+2},\dots,o_T \mid i_{t+1} = q_j)}_{\beta_{t+1}(j)}\, a_{ij} \\
&= \sum_{j=1}^{N} b_j(o_{t+1})\, a_{ij}\, \beta_{t+1}(j)
\end{aligned}
$$

⑥ Learning 问题

$$\lambda_{MLE} = \arg\max_{\lambda} p(O|\lambda)$$

⑦ Decoding 问题





N^T 种组合方式

从这些组合中挑出一种，使得后验概率最大

$$\delta_t(i) = \max_{i_1, i_2, \cdots, i_{t-1}} p(o_1, o_2, \cdots, o_t, i_1, i_2, \cdots, i_{t-1}, i_t = q_i)$$

$$\delta_{t+1}(j) = \max_{i_1, i_2, \cdots, i_t} P(o_1, o_2, \cdots, o_t, o_{t+1}, i_1, i_2, \cdots, i_t, i_{t+1} = q_j)$$ ⟹ 动态规划问题

$$= \max_{1 \le i \le N} \delta_t(i)\, a_{ij}\, b_j(o_{t+1})$$

直接根据图像理解

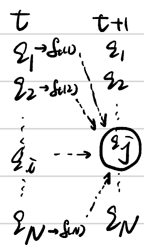
$q_1 \to q_2 = p(q_2 | q_1)$

概率类比为距离 $= \frac{1}{p}$

概率越大 距离越短

$$\psi_{t+1}(j) = \boxed{\arg}\max_{1 \le i \le N} \delta_t(i) \cdot a_{ij}$$ 记录上一时刻的状态

(上一步从哪个状态过来的)



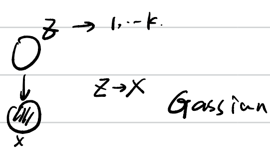
$(\psi_1\ \psi_2\ \psi_3 \cdots \psi_T) \Longrightarrow$ 路径

Viterbi 算法

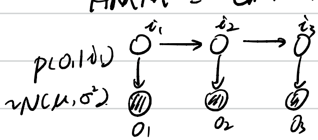
⑧ 总结

(i) Dynamic Model → State space Model GMM

(ii) HMM (mixture + time)



HMM = GMM + time



HMM ↗ i_1, i_2 离散

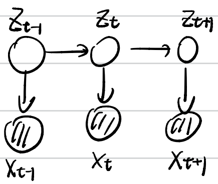
(11) 解决的问题

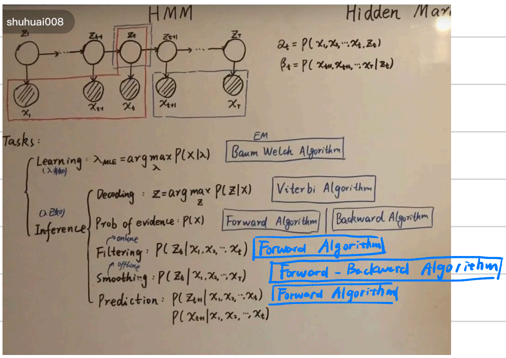
$\begin{cases} \text{learning (根据数据学习模型参数)} \\ \text{inference } p(Z|X) \text{ 后验分布} \end{cases}$

$Z_1, Z_2 \cdots Z_n, \quad X_1, X_2 \cdots X_N$

数据与数据之间不是独立同分布

- decoding $\max p(Z_1, Z_2, \cdots Z_t | X_1, X_2, \cdots X_t)$
- prob of evidence $= p(X|\theta) = p(X_1, \cdots X_t | \theta)$
- filtering: $P(Z_t | X_1, X_2, \cdots X_t)$
- smoothing: $P(Z_t | X_1, X_2, \cdots X_T)$ 拿到所有数据
- prediction $= \begin{cases} p(Z_{t+1} | X_1, X_2, \cdots X_t) \\ p(X_{t+1}, X_{t+2} | X_1, X_2, \cdots X_t) \end{cases}$





⑨ Filtering 问题

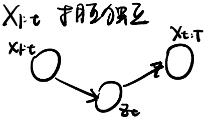
$$p(Z_t | X_{1:t}) = \frac{P(Z_t, X_{1:t})}{p(X_{1:t})} = \frac{p(X_{1:t}, Z_t)}{\sum_{Z_t} p(X_{1:t}, Z_t)} \propto p(X_{1:t}, Z_t) = \alpha_t$$

⑩ Smoothing 问题

$$p(Z_t | X_{1:T}) = \frac{p(Z_t, X_{1:T})}{p(X_{1:T})} = \frac{p(X_{1:T}, Z_t)}{\sum_{Z_t} p(X_{1:T}, Z_T)} \propto \alpha_t \beta_t$$

$$p(X_{1:T}, Z_T) = p(X_{1:t}, X_{t+1:T}, Z_t)$$
$$= p(X_{t+1:T} | X_{1:t}, Z_t) \underbrace{p(X_{1:t}, Z_t)}_{\alpha_t}$$

在给定 Z_t 的情况下, $X_{t+1:T}$ 和 $X_{1:t}$ 相互独立



$\Rightarrow p(X_{t+1:T} | Z_t) = \beta_t$

① prediction 问题

$$p(z_{t+1} \mid x_{1:t}) = \sum_{z_t} p(z_{t+1}, z_t \mid x_{1:t}) = \sum_{z_t} \underbrace{p(z_{t+1} \mid z_t, x_{1:t})}_{p(z_{t+1} \mid z_t)\ dt} \underbrace{p(z_t \mid x_{1:t})}_{\text{filtering}}$$

$$p(x_{t+1} \mid x_{1:t}) = \sum_{z_{t+1}} p(x_{t+1}, z_{t+1} \mid x_{1:t})$$

$$= \sum_{z_{t+1}} \underbrace{p(x_{t+1} \mid z_{t+1}, x_{1:t})}_{p(x_{t+1} \mid z_{t+1})} p(z_{t+1} \mid x_{1:t})$$