Conditional random field (条件随机场)

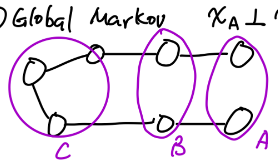
无向图 Review

三. 无向图 — 马尔科夫网络 Markov Network (Markov Random Field)

一. 条件独立性 　　MRF ⟺ Gibbs Distribution

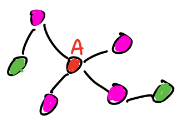
条件独立性可以体现在以下三个方面:

① Global Markov 　$x_A \perp x_C \mid x_B$. 即 $P(x_A, x_C \mid x_B) = P(x_A \mid x_B) \cdot P(x_C \mid x_B)$



② Local Markov

$x_A \perp x_D \mid x_B$, 其中A是变量, B是其邻居, D为其余变量.



→ $x_{红} \perp x_{绿} \mid x_{紫}$

③ Pairwise Markov

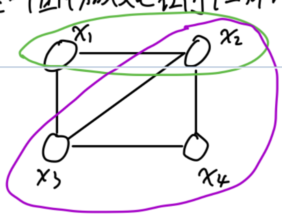
$x_A \perp x_B \mid x_{-\{A,B\}}$, 即给定所有其他变量, 两个非邻接变量条件独立.

二. 因子分解

① 基本概念. 团

团: 一个关于结点的集合. 集合内的结点之间相互连通

最大团: 若在一个团中加入其他任何节点都不再成团, 则称该团为最大团.



绿圈为一个团, 紫圈为一个最大团.

针对相关结点之间定义势函数, 在 Markov Random Field 中, 多个变量的联合概率分布能基于团分解为多个势函数的乘积, 每个团对应一个势函数.

$$P(x) = \frac{1}{Z} \prod_{i=1}^{K} \psi_{C_i}(x_{C_i})$$

C_i: 最大团

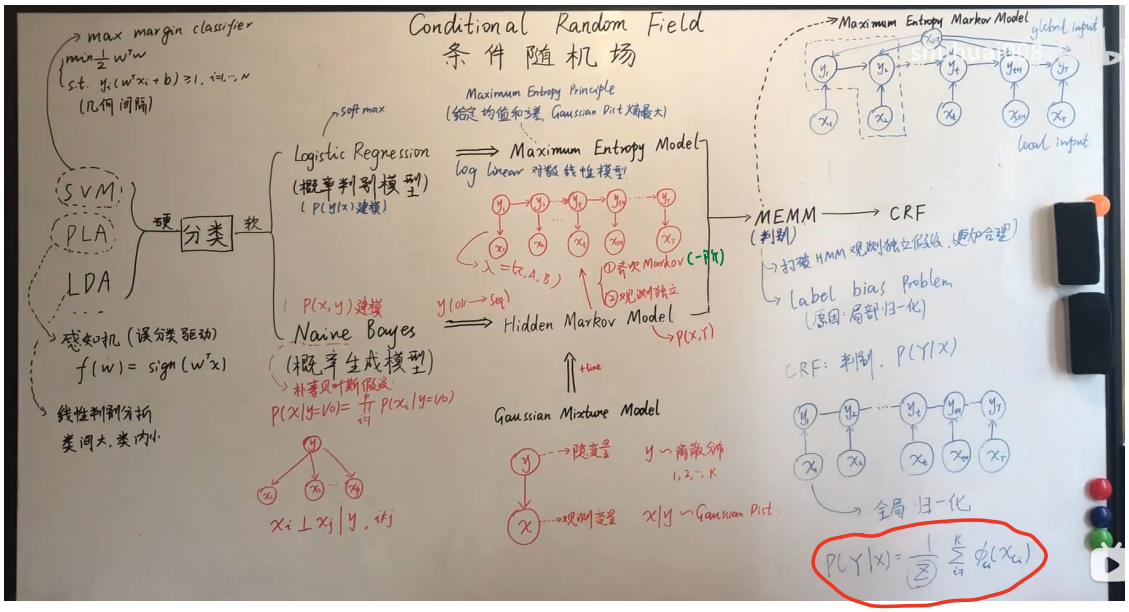
x_{C_i}: 最大团随机变量集合

$\psi(x_{C_i})$: 势函数, 必须为正

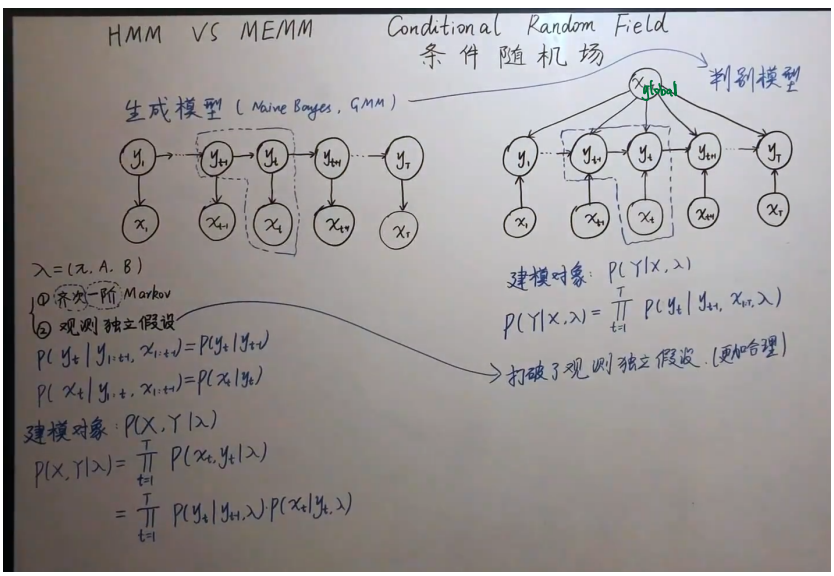
Z: 归一化因子. $Z = \sum_{x} \prod_{i=1}^{K} \psi(x_{C_i})$

Explanation:
$\sum_{x} P(x) = 1$

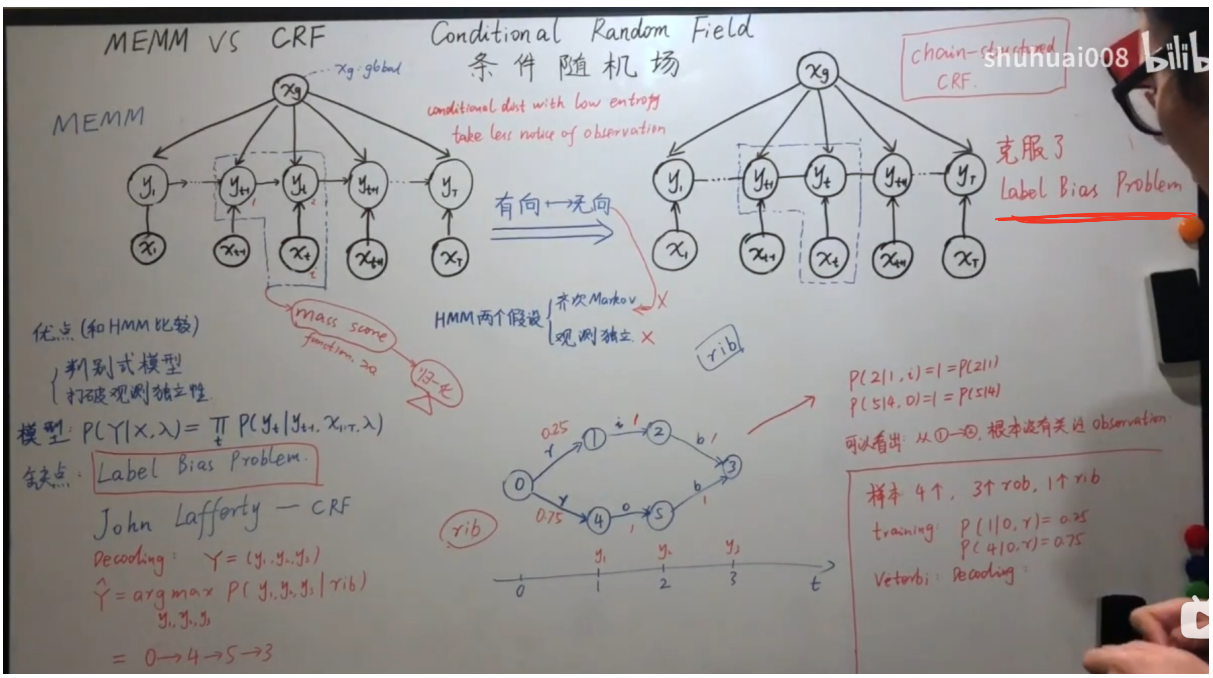
Background



HMM VS MEMM (maximum-entropy Markov model)



MEMM vs. CRF



HMM → MEMM → CRF (条件随机场)

判别 → 无向图

$$p(Y|X) = \frac{1}{Z} \exp \sum_{i=1}^{K} F_i(Y_{ci})$$

$$= \frac{1}{Z} \exp \sum_{i=1}^{T} F(y_{t-1}, y_t, X_{1:T})$$

最大团

Potential function:

$$p(x) = \frac{1}{Z} \prod_{i=1}^{K} \psi_i(x_{ci})$$

能量函数

$$= \frac{1}{Z} \exp \sum_{i=1}^{K} [-E(x_{ci})]$$

$$= \frac{1}{Z} \exp \sum_{i=1}^{K} F_i(x_{ci})$$

C_i: 最大团

K: K个最大团

$$F(y_{t-1}, y_t, X_{1:T}) = \Delta_{y_{t-1}, X_{1:T}} + \Delta_{y_t, X_{1:T}} + \Delta_{y_{t-1}, y_t, X_{1:T}}$$

状态函数 转移函数

$$= \Delta_{y_t, X_{1:T}} + \Delta_{y_{t-1}, y_t, X_{1:T}}$$

$$\Delta_{y_{t-1}, y_t, X_{1:T}} = \sum_{k=1}^{K} \lambda_k f_k(y_{t-1}, y_t, X_{1:T})$$

$$\text{同理 } \Delta_{y_t, X_{1:T}} = \sum_{l=1}^{L} \eta_l g_l(y_t, X_{1:T})$$

f_k, g_l: 特定的特征函数

λ_k, η_l: 系数

$$\Rightarrow p(y|x) = \frac{1}{Z} \exp \sum_{t=1}^{T} \left[\sum_{k=1}^{K} \lambda_k \underbrace{f_k}_{\text{转移}}(y_{t-1}, y_t, x_{1:T}) + \sum_{l=1}^{L} \eta_l \underbrace{g_l}_{\text{状态}}(y_t, x_{1:T}) \right]$$

CRF 的 PDF ★

试图化简上式 PDF，用向量形式表示：

首先想去掉内部的 $\sum_{k=1}^{K}$ 和 $\sum_{l=1}^{L}$

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_T \end{bmatrix} \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_T \end{bmatrix} \quad \lambda = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_K \end{bmatrix} \quad \eta = \begin{bmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_L \end{bmatrix}$$

$$f = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_K \end{bmatrix} = f(y_{t-1}, y_t, x_{1:T}) \quad g = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_L \end{bmatrix} = g(y_t, x_{1:T})$$

★ $\Rightarrow p(y|x) = \frac{1}{Z} \exp \sum_{t=1}^{T} \left[\lambda^T f(y_{t-1}, y_t, x_{1:T}) + \eta^T g(y_t, x_{1:T}) \right]$

然后进一步想去掉 $\sum_{t=1}^{T}$，t 只与 f, g 项有关：

$$p(y|x) = \frac{1}{Z} \exp \left[\lambda^T \sum_{t=1}^{T} f(y_{t-1}, y_t, x_{1:T}) + \eta^T \sum_{t=1}^{T} g(y_t, x_{1:T}) \right]$$

$$\theta = \begin{bmatrix} \lambda \\ \eta \end{bmatrix}_{K \times L} \qquad H = \begin{bmatrix} \sum_{t=1}^{T} f(y_{t-1}, y_t, x_{1:T} \\ \sum_{t=1}^{T} g(y_t, x_{1:T}) \end{bmatrix}_{K \times L}$$

$\therefore p(y|x) = \frac{1}{Z} \exp \left(\theta^T H(y_{t-1}, y_t, x_{1:T}) \right)$

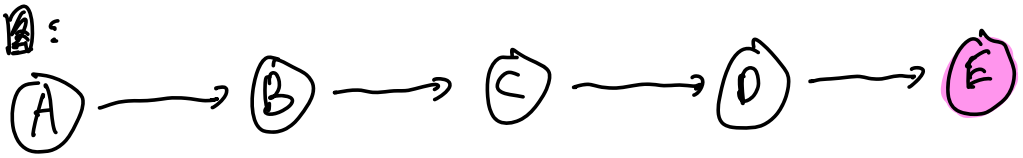
模型要解决的问题

- Learning：参数学习出来 (parameter estimation) ⇒ $\hat{\theta} = \arg\max_{\theta} \prod_{i=1}^{N} P(y_i|x_i)$
- Inference：
  - marginal prob: $P(y_t|x) \rightarrow P(y_t = i|x, \theta)$
  - conditional prob: 生成模型的后验，这里没有
  - MAP inference; decoding: $\hat{y} = \arg\max_{y=y_1 y_2 \cdots y_T} P(y|x)$

Review:

1. Variable inference

有向图：



$$P(E) = \sum_{A,B,C,D} P(A,B,C,D,E)$$

因式分解：

$$P(E) = \sum_{A,B,C,D} P(A) \cdot P(B|A) \cdot P(C|B) \cdot P(D|C) \cdot P(E|D)$$

先消A：

$$= \sum_{B,C,D} P(C|B) P(D|C) P(E|D) \sum_{A} P(A) P(B|A)$$

$$= \sum_{B,C,D} P(C|B) P(D|C) P(E|D) P(B)$$

消B：

$$= \sum_{C,D} P(D|C) P(E|D) \sum_{B} P(B) P(C|B)$$

$$= \sum_{C,D} P(D|C) P(E|D) P(C)$$

消C：

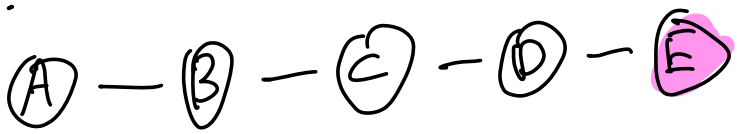
$$= \sum_{D} P(E|D) \sum_{C} P(C) P(D|C)$$

$= \sum_D P(E|D)\,P(D)$

$O(kn^2)$ k is the number of variables

$P(x) = \frac{1}{Z}\prod_{i=1}^{K}\psi_i(x_{ci})$

因为图：

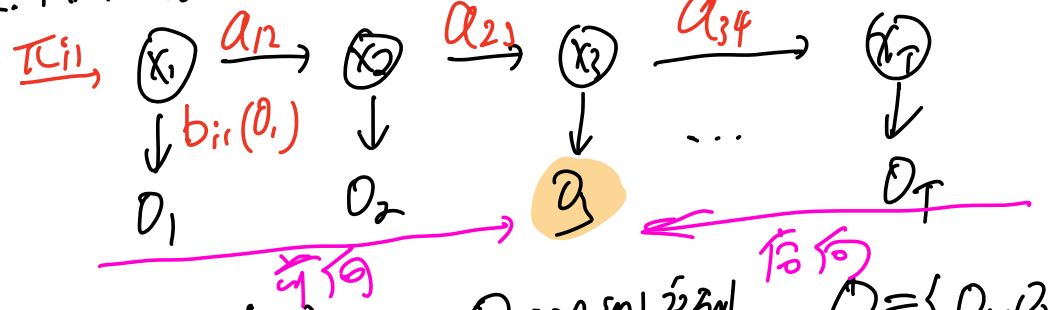


$$
\begin{aligned}
P(e) &\propto \sum_{a,b,c,d}\phi(a,b)\phi(b,c)\phi(c,d)\phi(d,e)\\
&= \sum_{b,c,d}\phi(b,c)\phi(c,d)\phi(d,e)\sum_{a}\phi(a,b)\\
&= \sum_{b,c,d}\phi(b,c)\phi(c,d)\phi(d,e)m_a(b)\\
&= \sum_{c,d}\phi(c,d)\phi(d,e)\sum_{b}\phi(b,c)m_a(b)\\
&= \sum_{c,d}\phi(c,d)\phi(d,e)m_b(c)\\
&= \sum_{d}\phi(d,e)\sum_{c}\phi(c,d)m_b(c)\\
&= \sum_{d}\phi(d,e)m_c(d)\\
&= \sum_{d}\phi(d,e)m_c(d)\\
&= m_d(e)
\end{aligned}
$$

Finally we normalize to obtain a proper probability:

$$P(e) = \frac{m_d(e)}{\sum_e m_d(e)}$$

2. HMM的 forward-backward 算法



Target: $P(O|\lambda)$ O：观测序列 $O=\{O_1, O_2, O_3, \cdots O_T\}$

I：状态序列 $I=\{i_1, i_2, i_3, \cdots i_T\}$

$$P(I|\lambda) = \pi_{i1}a_{i_1 i_2}a_{i_2 i_3}\cdots a_{i_{T-1} i_T}$$

$$P(O|I,\lambda) = b_{i1}(O_1)\, b_{i2}(O_2)\, b_{i3}(O_3) \cdots b_{iT}(O_T)$$

$$P(O,I|\lambda) = P(I|\lambda) \cdot P(I|O,\lambda)$$

$$= \pi_{i1} a_{i1 i2} b_{i1}(O_1) \cdots \pi_{iT} a_{i_{T-1} i_T} b_{i_T}(O_T)$$

$$P(O|\lambda) = \sum_{I} \pi_{i1} a_{i1 i2} b_{i1}(O_1) \cdots \pi_{iT} a_{i_{T-1} i_T} b_{i_T}(O_T)$$

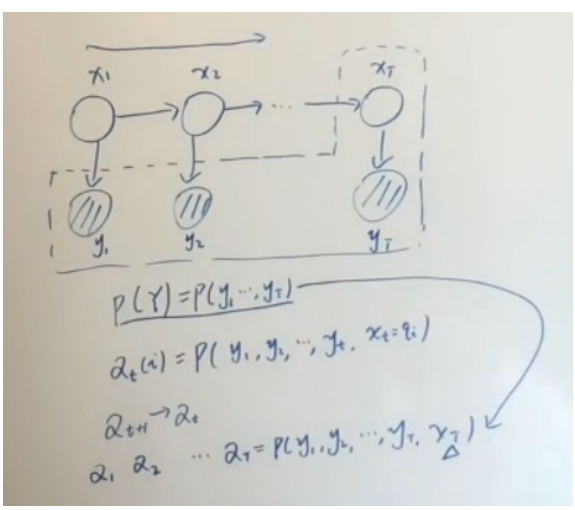
$$P(O|\lambda) = \sum_{i_1 i_2 i_3 \cdots i_T} \pi_{i1} a_{i1 i2} b_{i1}(O_1) \cdots \pi_{iT} a_{i_{T-1} i_T} b_{i_T}(O_T)$$

定义: $\alpha_t(i) = P(O_1, O_2, \cdots O_T, i_t = q_i|\lambda)$

VE法:

$$P(O|\lambda) = \sum_{i_2 \cdots i_T} \cdots \sum_{i_1} \pi_{i_1} a_{i_1} a_{i_2} b_{i_1}(O_1)$$

$$\alpha_{t+1}(i) = \sum_{j=1}^{N} \alpha_t(j) a_{ji}\, b_i(O_{t+1}) \quad i = 1, 2, \cdots N$$



Learning: Given $P(Y=y|X=x)$, 求 $P(y_t=i|x)$

$$P(y_t=i|x) = \sum_{y_1 y_2 \cdots y_{t-1} y_{t+1} y_T} P(y|x)$$

$$= \sum_{y_1 \cdots y_{t-1}} \sum_{y_{t+1} y_T} \frac{1}{Z} \prod_{t'=1}^{T} \psi_{t'}(y_{t'-1}, y_{t'}, x)$$

$$= \frac{1}{Z} \Delta_{tf} \Delta_{tb}$$

VE

算 Δ_{tf}: $\sum_{t-1} \psi_t(y_t, y_{t-1}, x) \cdots \cdots \sum_{y_1} \psi_2(y_2, y_1, x) \sum_{y_0} \psi_1(y_0, y_1, x)$

y_1, y_0

记 $\Delta_{tf} = \alpha_t(i) = \sum_{j \in S} \psi_t(y_{t-1}=j, y_t=i, x)\, \alpha_{t-1}(j)$

Similarly, 得到

$$\Delta_{tb} = \beta_t(i) = \sum_{j \in S} \psi_{t+1}(y_t=i, y_{t+1}=j, x)\, \beta_{t+1}(j)$$

本质: VE算法

HMM中的 forward - backward 算法!

learning

target: $\hat{\theta} = \arg\max \prod_{i=1}^{N} p(y^{(i)}|x^{(i)})$ $\theta = (\lambda, \eta)$

在进行各种类型的推断之前，还需要对参数进行学习：

$$\begin{aligned}\hat{\theta} &= \underset{\theta}{argmax} \prod_{i=1}^{N} p(y^i|x^i) \\ &= \underset{\theta}{argmax} \sum_{i=1}^{N} \log p(y^i|x^i) \\ &= \underset{\theta}{argmax} \sum_{i=1}^{N} [-\log Z(x^i, \lambda, \eta) + \sum_{t=1}^{T} [\lambda^T f(y_{t-1}, y_t, x) + \eta^T g(y_t, x)]]\end{aligned}$$

上面的式子中，第一项是对数配分函数，根据指数族分布的结论：

$$\nabla_\lambda (\log Z(x^i, \lambda, \eta)) = \mathbb{E}_{p(y^i|x^i)} [\sum_{t=1}^{T} f(y_{t-1}, y_t, x^i)]$$

其中，和 η 相关的项相当于一个常数。求解这个期望值：

$$\mathbb{E}_{p(y^i|x^i)} [\sum_{t=1}^{T} f(y_{t-1}, y_t, x^i)] = \sum_{y} p(y|x^i) \sum_{t=1}^{T} f(y_{t-1}, y_t, x^i)$$

第一个求和号的复杂度为 $O(S^T)$，重新排列求和符号：

$$\begin{aligned}\mathbb{E}_{p(y^i|x^i)} [\sum_{t=1}^{T} f(y_{t-1}, y_t, x^i)] &= \sum_{t=1}^{T} \sum_{y_{1:t-2}} \sum_{y_{t-1}} \sum_{y_t} \sum_{y_{t+1:T}} p(y|x^i) f(y_{t-1}, y_t, x^i) \\ &= \sum_{t=1}^{T} \sum_{y_{t-1}} \sum_{y_t} p(y_{t-1}, y_t|x^i) f(y_{t-1}, y_t, x^i)\end{aligned}$$

$= \sum_{t=1}^{T} [\sum_{y_{1:t-2}} \sum_{y_{t+1:T}} p(y|x^{(i)})] \sum_{y_{t-1}} \sum_{y_t} f(y_{t-1}, y_t, x^i)$

和上面的边缘概率类似，也可以通过前向后向算法得到上面式子中的边缘概率。

于是：

$$\nabla_\lambda L = \sum_{i=1}^{N} \sum_{t=1}^{T} [f(y_{t-1}, y_t, x^i) - \sum_{y_{t-1}} \sum_{y_t} p(y_{t-1}, y_t|x^i) f(y_{t-1}, y_t, x^i)]$$

利用梯度上升算法可以求解。对于 η 也是类似的过程。

$\lambda^{(t+1)} = \lambda^{(t)} + step \cdot \nabla_\lambda L(\lambda^{(t)}, \eta^{(t)})$

$\eta^{(t+1)} = \eta^{(t)} + step \cdot \nabla_\eta L(\lambda^{(t)}, \eta^{(t)})$