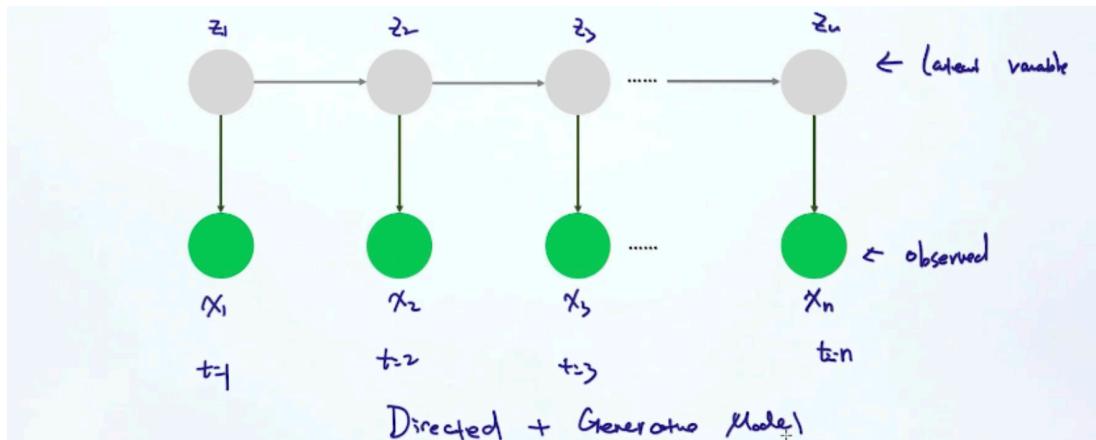


时序模型

HMM/CRF 传统模型

RNN/LSTM 深度学习模型

1. HMM 介绍



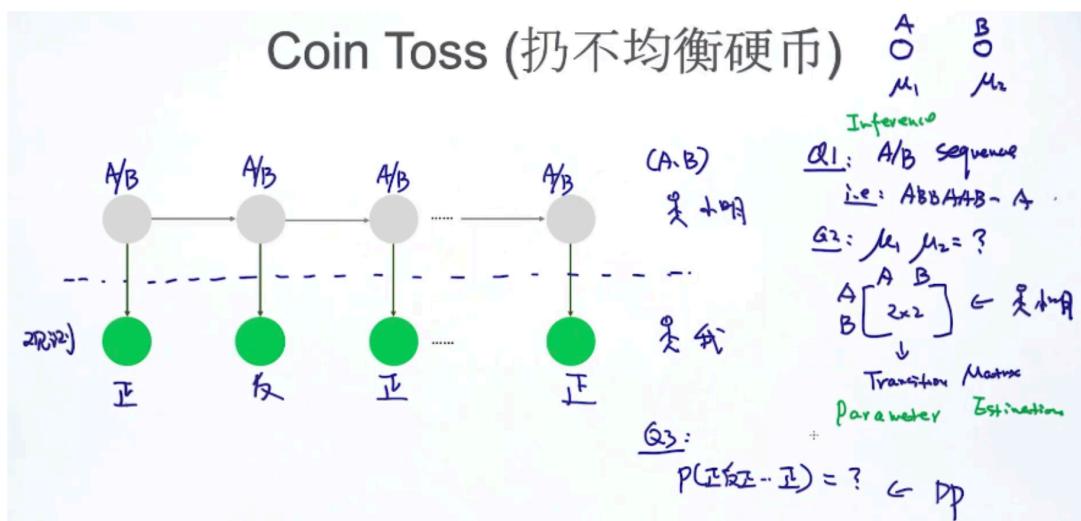
有方向的

生成的，可以生成新的数据

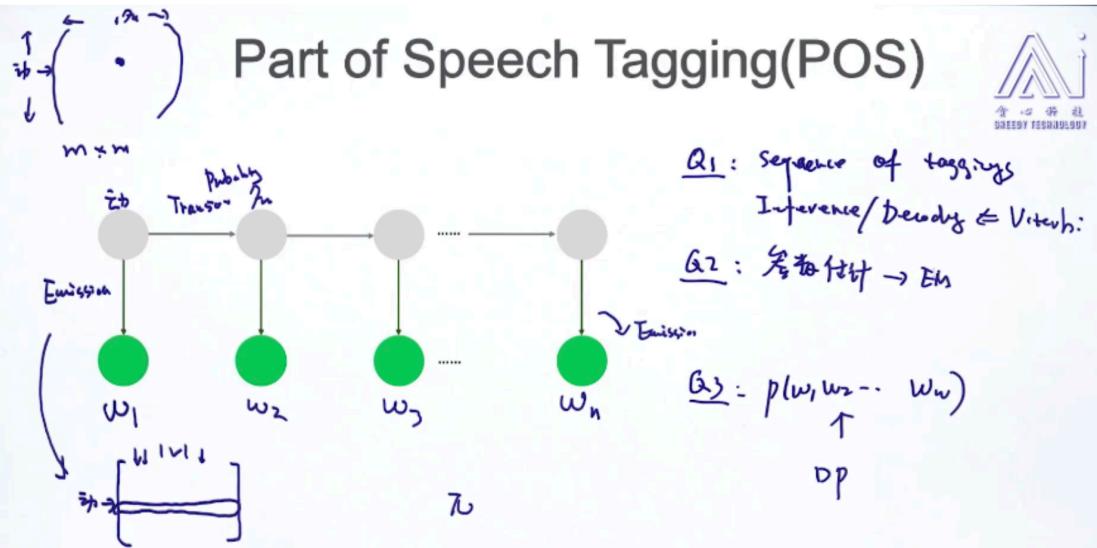
也可以做成判别模型

1.1 应用例子

1) Coin toss

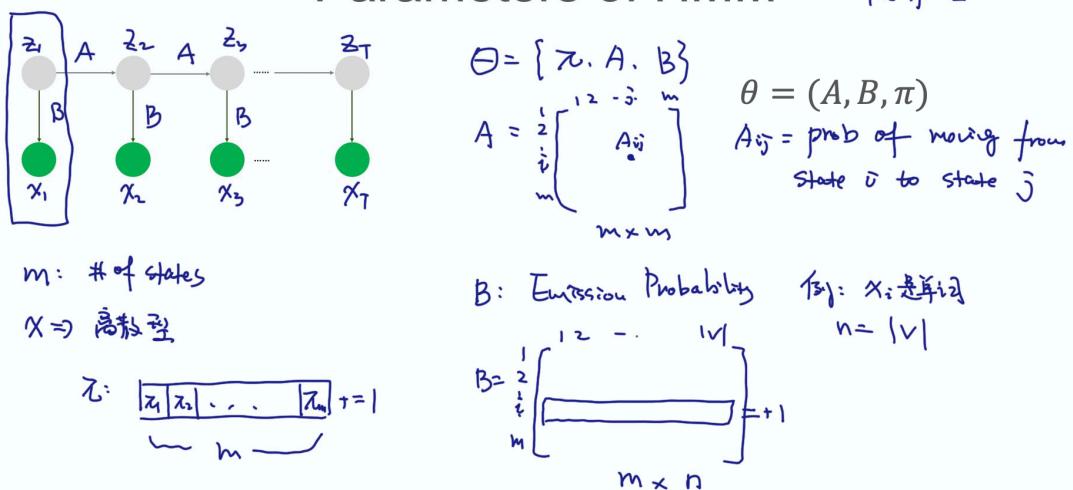


2) speech tagging



HMM 两个较重要的参数：转移概率、初始概率

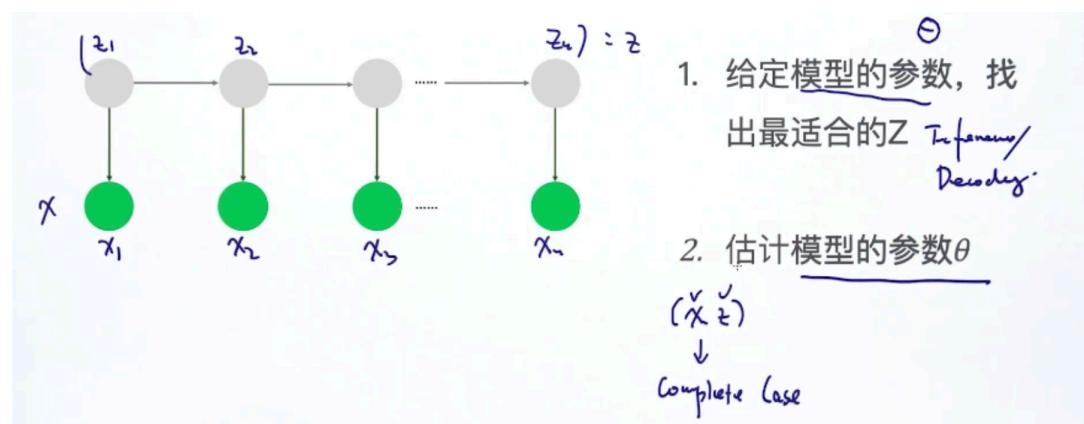
1.2 参数



状态转移的概率不是随机的

B 相当于生成矩阵 (因此, HMM 也可以看成生成模型)

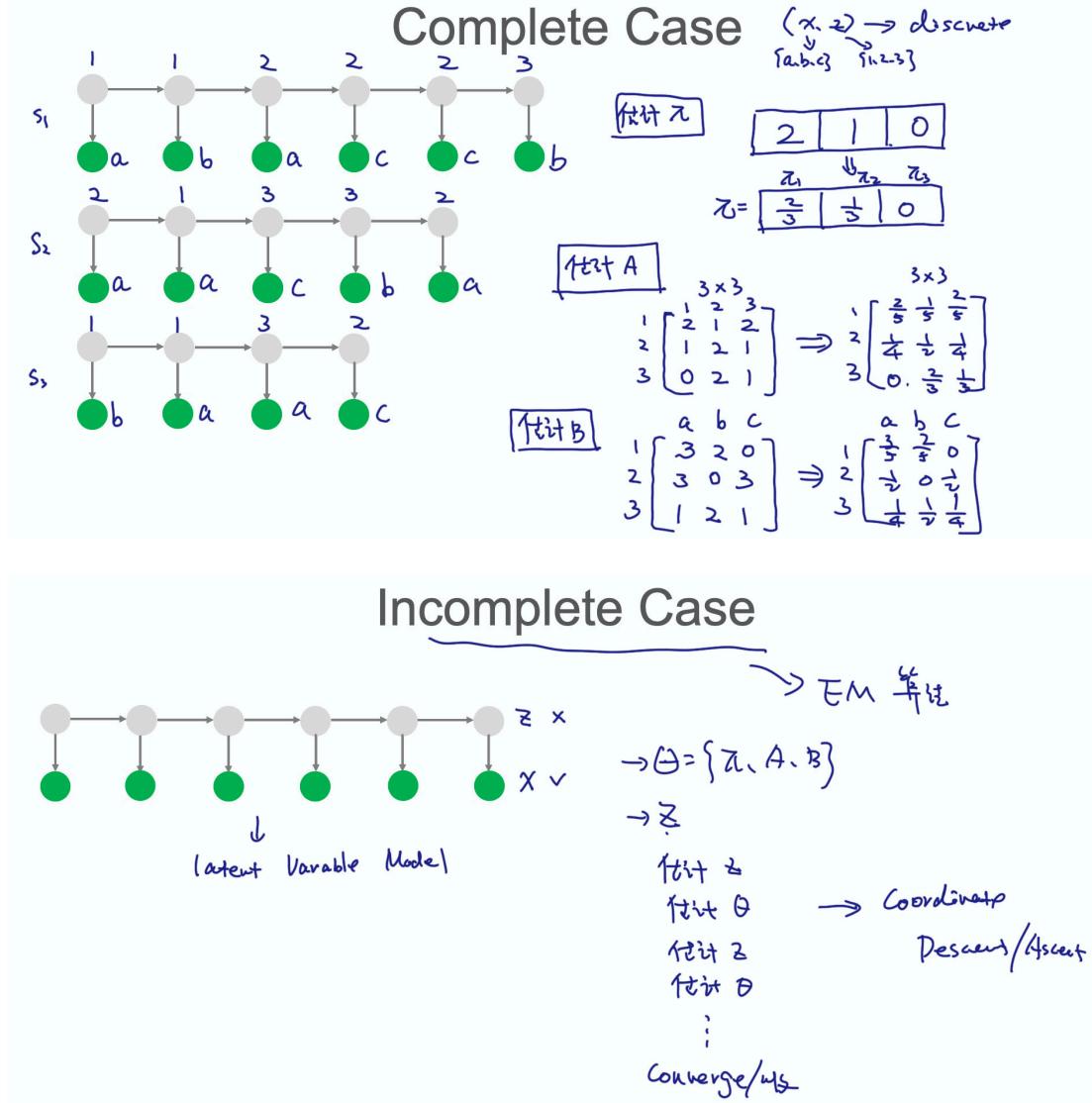
1.3 major tasks



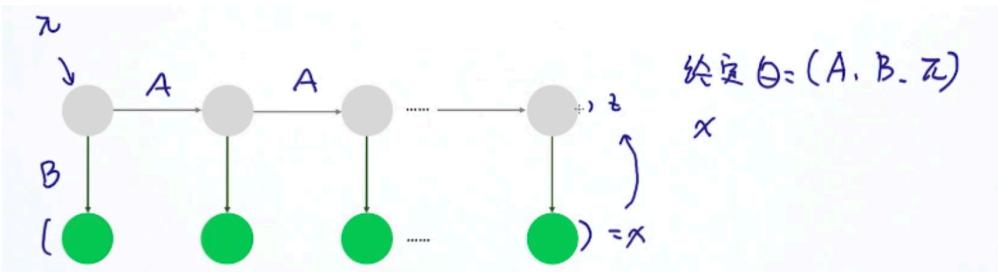
第一种，给定魔性的参数，找出最合适的 Z

第二种，估计魔性的参数 θ

1. complete case, x 和 z 都已知
2. incomplete case, x 已知, z 未知。 (需要用到 EM) (3.3)



1.4 find best Z



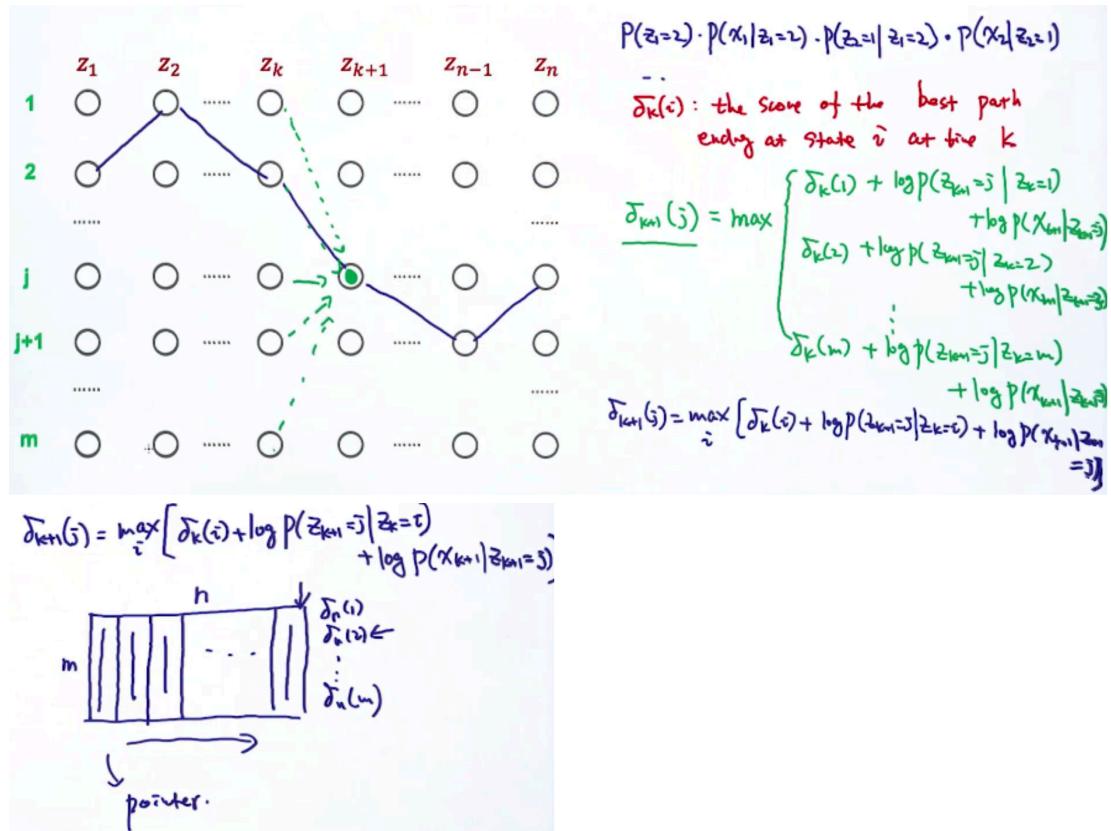
1) 遍历算法

把所有可能的 z 序列罗列出来，可以很容易的求得每个概率。

选出最大概率的。

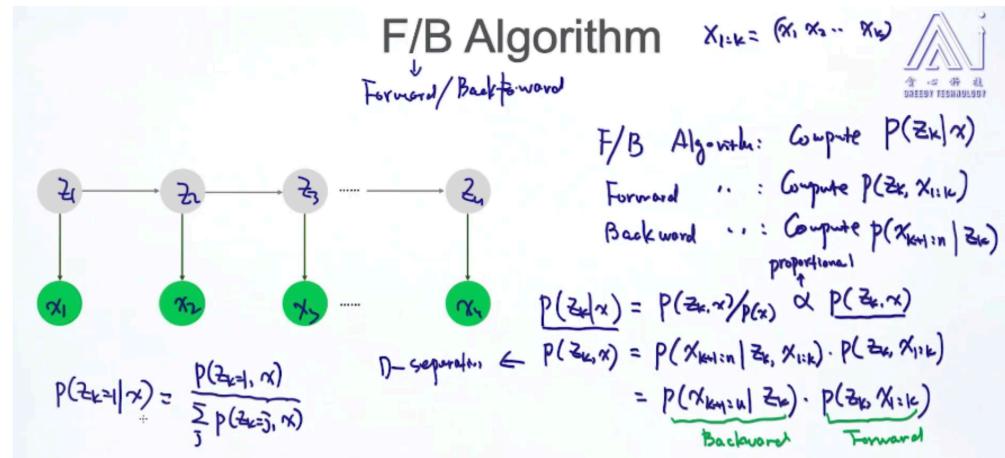
2) viterbi 动态规划

HMM 中有一个条件非常重要---隐式变量 z 只与前后两个变量相关联---这样可以大大减轻计算量。



2. HMM 中的 F/B 算法

F/B : forward/backward



用处：求出模型参数；change detection

2.1 forward

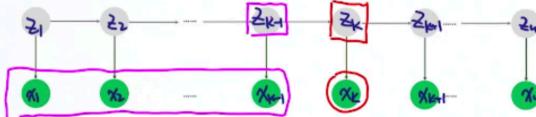
目标: $p(z_k, x_{1:k})$

Forward Algorithm



D-separation

$$\alpha_1(z_1) = p(z_1, x) = \underbrace{p(z_1)}_A \cdot \underbrace{p(x_1|z_1)}_B$$



$$p(z_k, x_{1:k}) = \boxed{\quad} \cdot p(z_{k+1}, x_{1:k+1})$$

$$p(z_k, x_{1:k}) = \sum_{z_{k+1}} p(z_{k+1}, z_k, x_{1:k})$$

$$\alpha_k(z_k) = \sum_{z_{k+1}} p(z_{k+1}, x_{1:k+1}) \cdot p(z_k | z_{k+1}, x_{1:k+1}) \cdot p(x_k | z_k, z_{k+1}, x_{1:k+1})$$

$$= \sum_{z_{k+1}} \underbrace{p(z_{k+1}, x_{1:k+1})}_A \cdot \underbrace{p(z_k | z_{k+1})}_B \cdot \underbrace{p(x_k | z_k)}_C$$

$$\alpha_k(z_k) = \sum_{z_{k+1}} \alpha_{k+1}(z_{k+1}) \cdot D(z_k | z_{k+1}) \cdot P(x_k | z_k)$$

2.2 backward

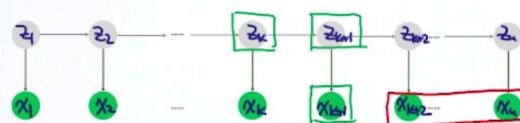
目标: $p(x_{k+1:n} | z_k)$

Backward Algorithm

$$\boxed{p(z_k|x)}$$



D-separation



$$p(x_{k+1:n} | z_k) = \boxed{\quad} \cdot p(x_{k+2:n} | z_{k+1:n})$$

$$p(x_{k+1:n} | z_k) = \sum_{z_{k+1}} p(x_{k+2:n} | z_{k+1:n}, z_{k+1})$$

$$\beta_k(z_k) = \sum_{z_{k+1}} p(x_{k+2:n} | z_{k+1}, z_k, z_{k+1:n}) \cdot p(x_{k+1} | z_{k+1}, z_k) \cdot p(z_{k+1} | z_k)$$

$O(n \cdot m^2)$

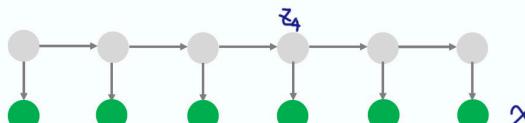
$$= \sum_{z_{k+1}} p(x_{k+2:n} | z_{k+1:n}) \cdot \underbrace{p(x_{k+1} | z_{k+1})}_B \cdot \underbrace{p(z_{k+1} | z_k)}_A$$

$$\beta_k(z_k) = \sum_{z_{k+1}} \beta_{k+1}(z_{k+1}) \cdot \underbrace{P(x_{k+1} | z_{k+1})}_B \cdot \underbrace{P(z_{k+1} | z_k)}_A$$

Forward

2.3 review

Review of F/B Algorithm



\rightarrow F/B: Compute $p(z_k | x) \rightarrow p(z_4=1|x) \rightarrow p(z_4=2|x) \rightarrow p(z_4=3|x)$

F: Compute $p(z_k, x_{1:k}) \leftarrow \alpha_k(z_k) \quad \alpha_1(z_1) \quad \alpha_2(z_2) \dots \alpha_T(z_T)$

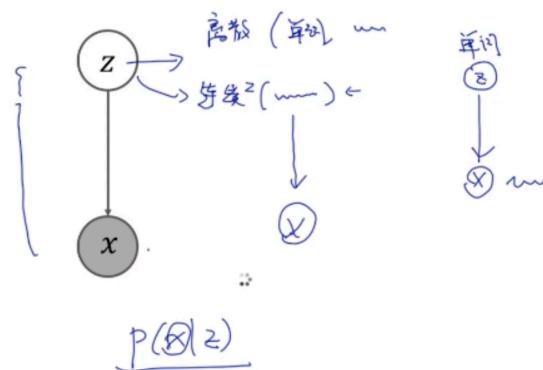
B: Compute $p(x_{k+1:n} | z_k) \leftarrow \beta_k(z_k) \quad \beta_1(z_1) \quad \dots \quad \beta_T(z_T)$

3. Latent Variable Model

3.1 data representation

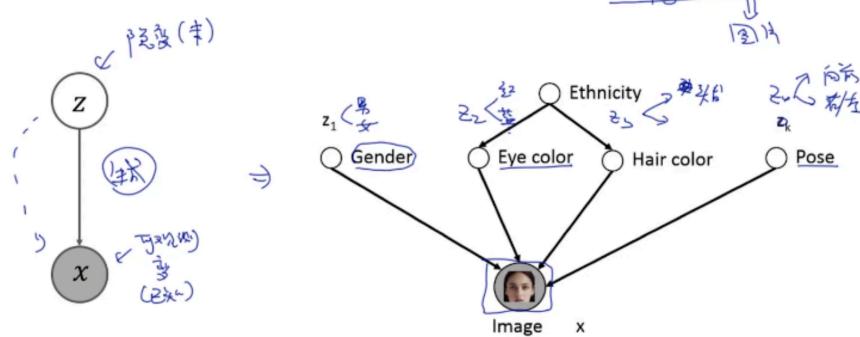
更倾向于用低维空间表示信息。---》去掉冗余信息。

3.2 latent variable models

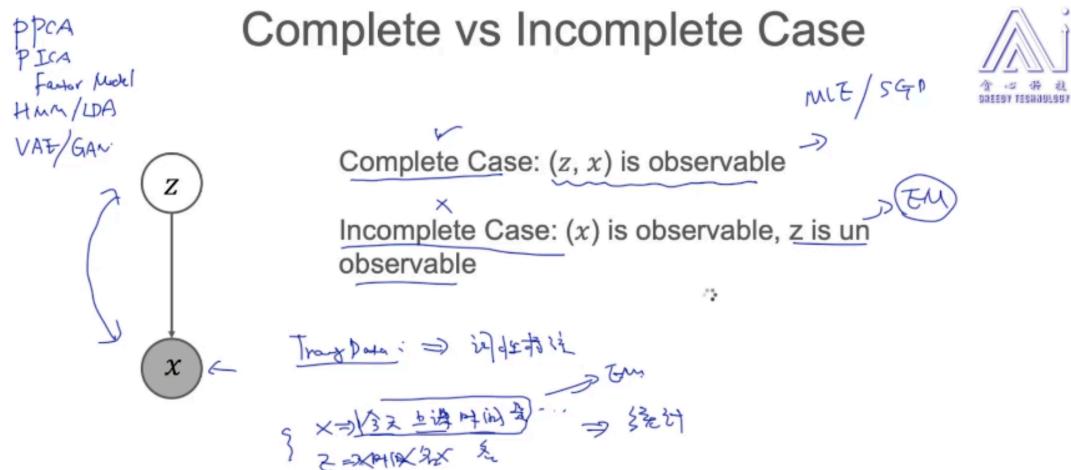


给定隐变量，生成新的数据（生成模型）

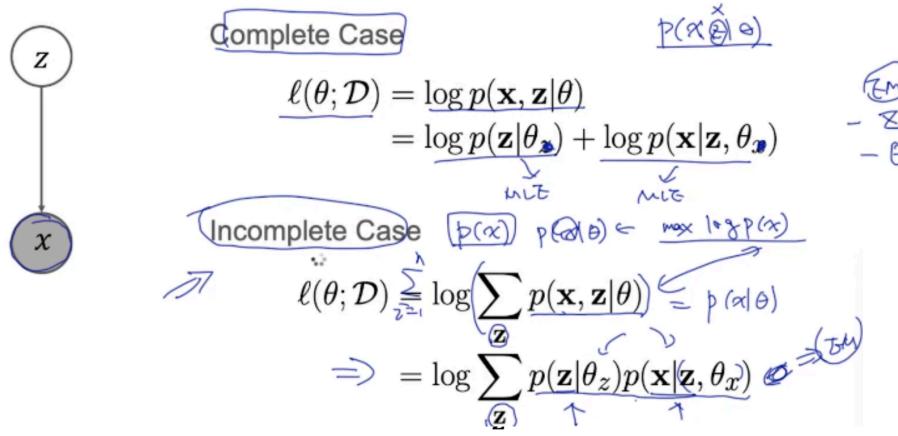
$z \rightarrow$ i.e., images / (GAN/VAE/LDA/PCA/ICA/...)
Configuration of Hidden Variable



3.3 Complete vs Incomplete Case



3.4 MLE



4. EM 算法

4.1 derivation

EM Derivation

θ : Model parameters
 x : observed
 z : latent variable

MLE

 $L(\theta) = \ln P(x|\theta)$
 $\text{argmax}_{\theta} L(\theta) = \text{argmax} \ln P(x|\theta)$
 $L(\theta) \text{ is after } n \text{ iterations } \theta_n$
 $\text{argmax}_{\theta} L(\theta) - L(\theta_n) = \ln P(x|\theta) - \ln P(x|\theta_n)$
 $= \ln \sum_z P(x|z, \theta) - \ln P(x|\theta_n)$
 $= \ln \sum_z P(x|z, \theta) \cdot P(z|\theta) - \ln P(x|\theta_n)$
 $= \ln \sum_z P(x|z, \theta) \cdot \frac{P(z|x, \theta)}{P(z|x, \theta_n)} - \ln P(x|\theta_n)$
 $= \ln \sum_z P(x|z, \theta) \cdot \left(\frac{P(x|z, \theta) \cdot P(z|\theta)}{P(z|x, \theta_n)} \right) - \ln P(x|\theta_n)$

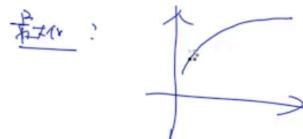
$\log \left(\sum_z f(z) \right) \geq \sum_z \lambda_i f(z)$
 Jensen's inequality $\Leftrightarrow \ln \sum_{i=1}^n \lambda_i x_i \geq \sum_{i=1}^n \lambda_i \ln x_i$
 $\sum_{i=1}^n \lambda_i = 1$

 $L(\theta) - L(\theta_n) \geq \sum_z P(z|x, \theta_n) \ln \frac{P(x|z, \theta) \cdot P(z|\theta)}{P(z|x, \theta_n)} - \ln P(x|\theta_n)$
 $= \sum_z P(z|x, \theta_n) \cdot \ln \frac{P(x|z, \theta) \cdot P(z|\theta)}{P(z|x, \theta_n) \cdot P(x|\theta_n)}$
 $= \Delta(\theta|\theta_n)$
 $L(\theta) - L(\theta_n) \geq \Delta(\theta|\theta_n)$
 $\Rightarrow L(\theta) \geq L(\theta_n) + \Delta(\theta|\theta_n)$
 $\theta_{n+1} = \text{argmax}_{\theta} [L(\theta_n) + \Delta(\theta|\theta_n)]$
 $= \text{argmax}_{\theta} \left[L(\theta_n) + \sum_z P(z|x, \theta_n) \ln \frac{P(x|z, \theta) \cdot P(z|\theta)}{P(z|x, \theta_n) \cdot P(x|\theta_n)} \right]$
 $= \text{argmax}_{\theta} \left\{ \sum_z P(z|x, \theta_n) \ln P(x|z, \theta) \cdot P(z|\theta) \right\}$

E-step: $E_{z|x, \theta_n} \{ \ln P(x|z, \theta) \}$
 M-step:

4.2 remarks

- E step (\hat{z})
 - M step (Θ)
 - Global Optimal (Local Optimal)
 - EM (严格增大)
- 直至几次(使用不同的
初始结果)



不能保证全局最优，每次迭代都能找到比当前更优秀的解

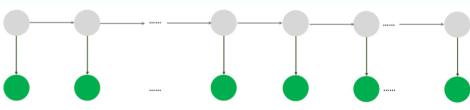
$$\arg \max_{\Theta} \left[\mathbb{E}_{z|x} \left\{ \ln p(x, z|\Theta) \right\} \right]$$

$\uparrow \quad \uparrow$

E-step : 求出 z 的期望
 M-step : 极大化 $\ln p(x, z|\Theta)$
 Converge.

$z = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix}$
 $z_1 = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$

EM for HMM Parameter Estimation

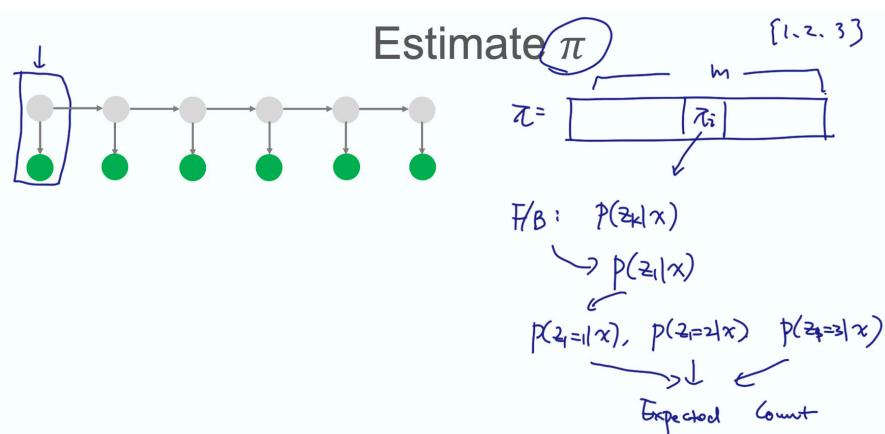


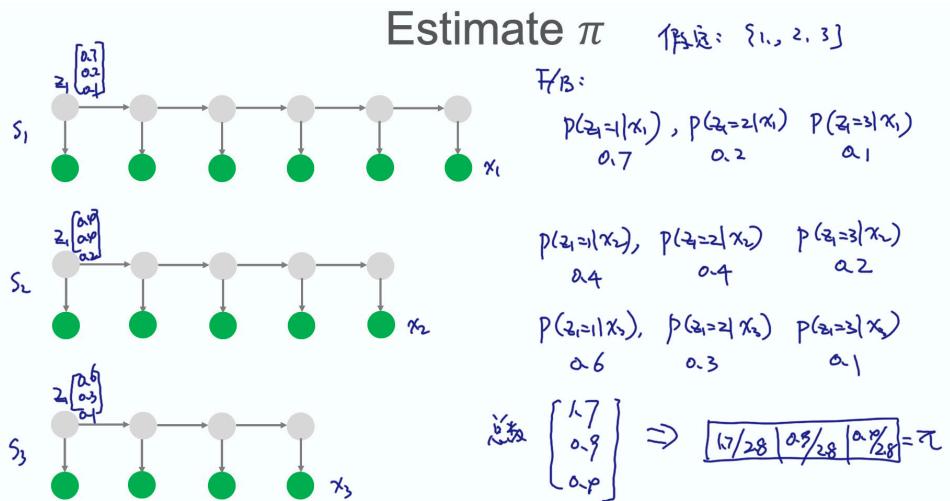
```
while (not converged):
    ① Compute  $\hat{z}$  (expectation)
    ② update  $\pi, A, B$ 
```

5. HMM 参数估计

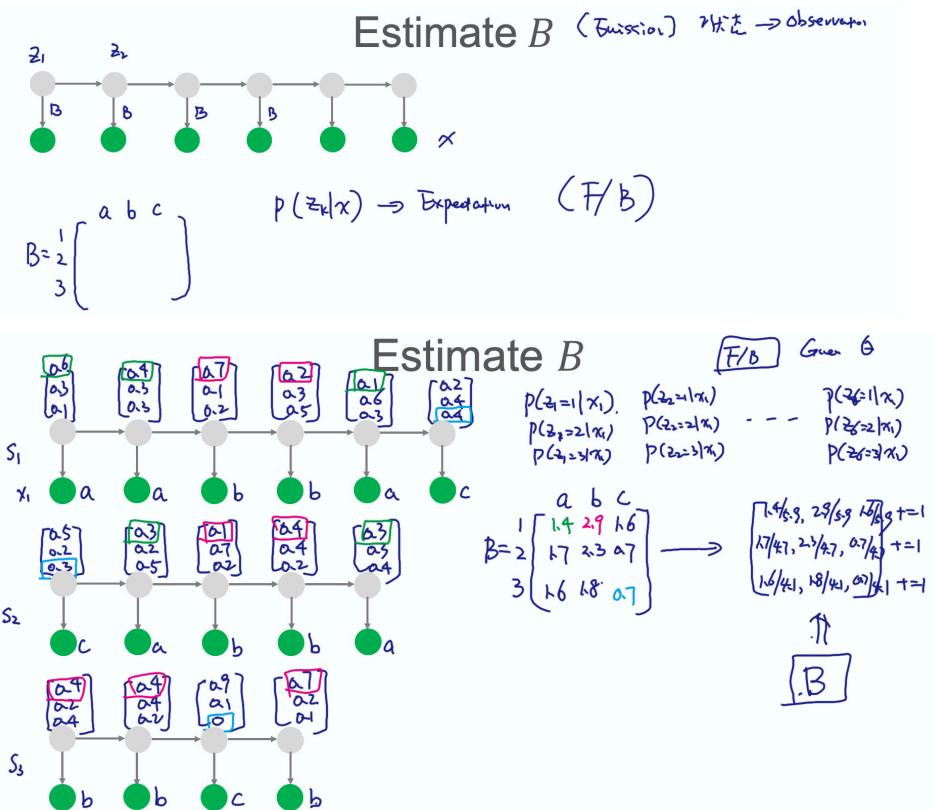
5.1 估计 π

某一个时刻出现某个状态的概率





5.2 估计 B



5.3 估计 A

Estimate A: Review of Language Model

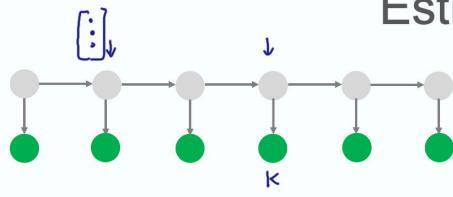
Bi-gram

$$w_i \rightarrow w_j \\ p(w_j | w_i) = \frac{c(w_i, w_j)}{c(w_i)}$$

$$p(\text{今日} | \text{今}) \\ = \frac{3}{8}$$

今	日
今	日
今	日
今	日

Estimate A



$$p(z_k=i, z_{k+1}=j | x) = ?$$

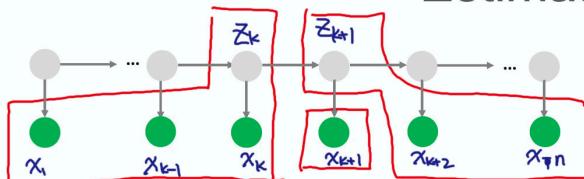
Forward + Backward

$$A = \begin{bmatrix} & & & \\ & \ddots & & \\ & & A_{ij} & \\ & & & \ddots \\ & & & & A_{mn} \end{bmatrix}$$

$$p(z_{k+1}=j | z_k=i) = \frac{p(z_{k+1}=j, z_k=i)}{p(z_k=i) \in T/B} = \frac{C(z_k=i, z_{k+1}=j)}{C(z_k=i) \in}.$$

Estimate A

$$[p(z_k=i, z_{k+1}=j | x) = ?]$$



$$| = \left\{ \begin{array}{l} p(z_k=1, z_{k+1}=1 | x) \propto \beta_k(1, 1) \\ p(z_k=1, z_{k+1}=2 | x) \propto \beta_k(1, 2) \\ p(z_k=1, z_{k+1}=3 | x) \propto \beta_k(1, 3) \\ p(z_k=2, z_{k+1}=1 | x) \propto \beta_k(2, 1) \\ \vdots \\ p(z_k=3, z_{k+1}=3 | x) \propto \beta_k(3, 3) \end{array} \right.$$

$$p(z_k=i, z_{k+1}=j | x) \propto p(z_k=i, z_{k+1}=j, x)$$

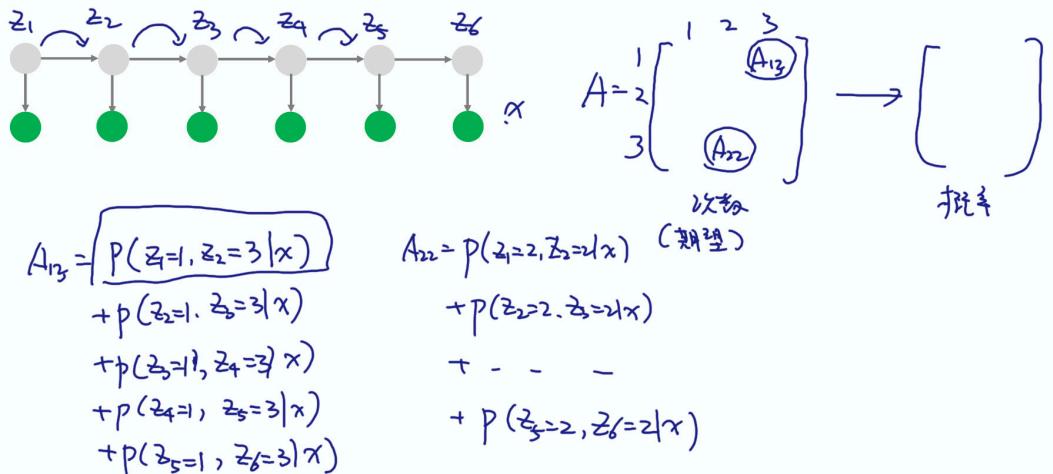
$$= p(z_k=i, z_{k+1}=j, X_{1:k}, X_{k+1}, X_{k+2:n})$$

$$= p(z_k, X_{1:k}) \cdot p(X_{k+1:n} | z_{k+1}) \cdot p(z_{k+1}, z_k) \cdot p(X_{k+1} | z_{k+1}) = \frac{\beta_k(1, 1)}{\beta_k(1, 1) + \beta_k(1, 2) + \dots + \beta_k(3, 3)} \quad \text{Normalize}$$

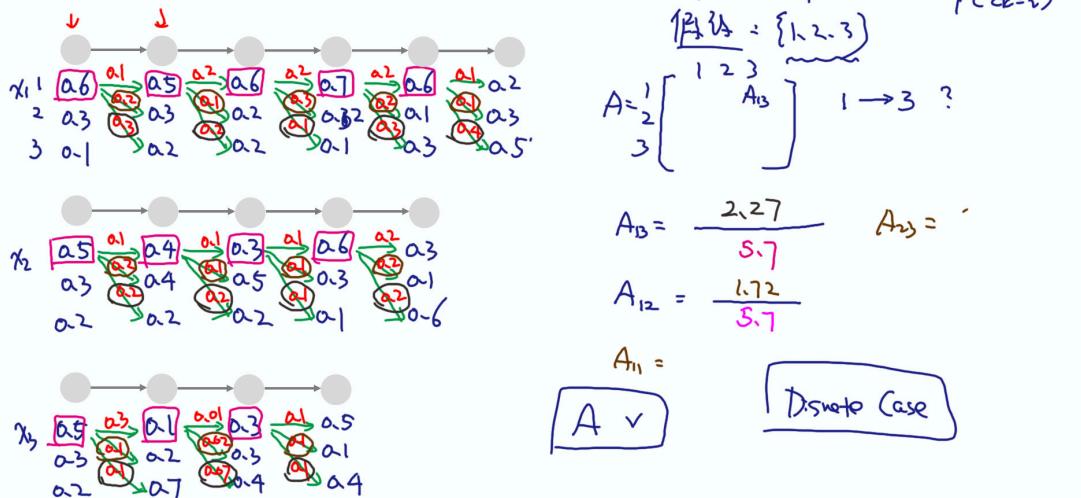
$$= \alpha_k(z_k) - \beta_{km}(z_{km}) - p(z_{km} | z_k) - p(X_{km} | z_{km})$$

$$\underbrace{\alpha_k(z_k)}_{BA} - \underbrace{\beta_{km}(z_{km})}_{B}$$

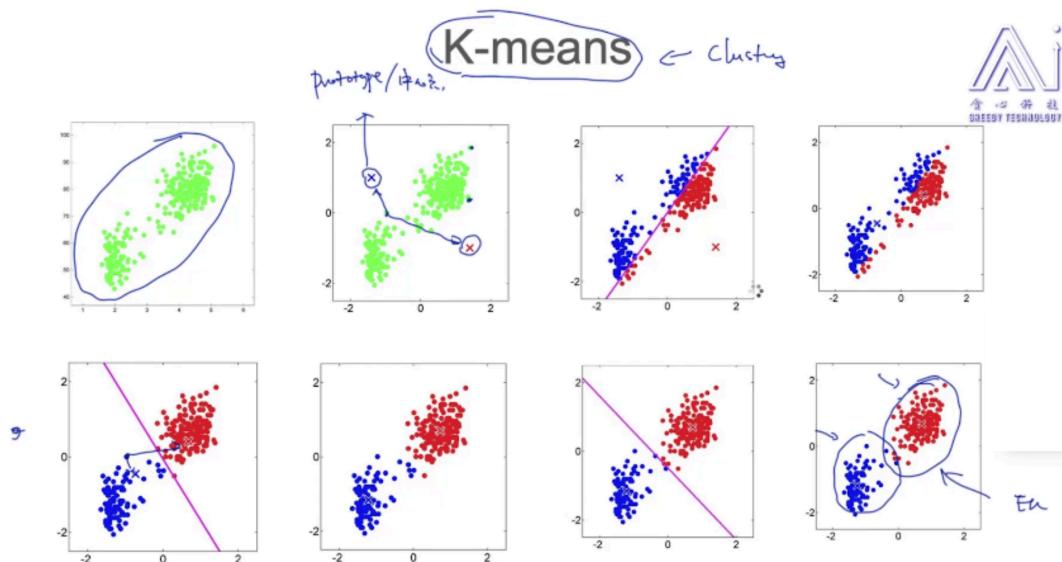
Estimate A



Estimate A $P(z_{k+n}=j | z_k=i) = \frac{P(z_k=i, z_{k+n}=j)}{P(z_k=i)}$



6. K-means

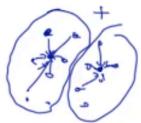


K-means Cost Function

目标函数:

$$Y_{nk} = \begin{cases} 1 & \text{if } X_n \text{ belongs to } k\text{th cluster} \\ 0 & \text{otherwise} \end{cases}$$

一个点属于其中某一个 cluster



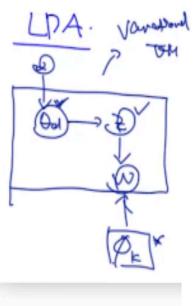
minimize

$$X_n \Rightarrow Y_{n1}, Y_{n2}, Y_{n3}, \dots, Y_{nk}$$

$$(1, 0, 0, \dots, 0)$$

$$\sum_{n=1}^N \sum_{k=1}^K (Y_{nk} \| X_n - \mu_k \|^2) \rightarrow \text{EM}$$

Latent(Z) 参数



① 需要: 一个点属于哪个 cluster?

② 中心点?

k clusters: μ_k
中心点. 和坐标.

E-step: $\min_{Y_{nk}} \sum_{k=1}^K Y_{nk} \| X_n - \mu_k \|^2$

M-step: $\text{Center } \mu_k (\text{已知 } Y_{nk}):$

