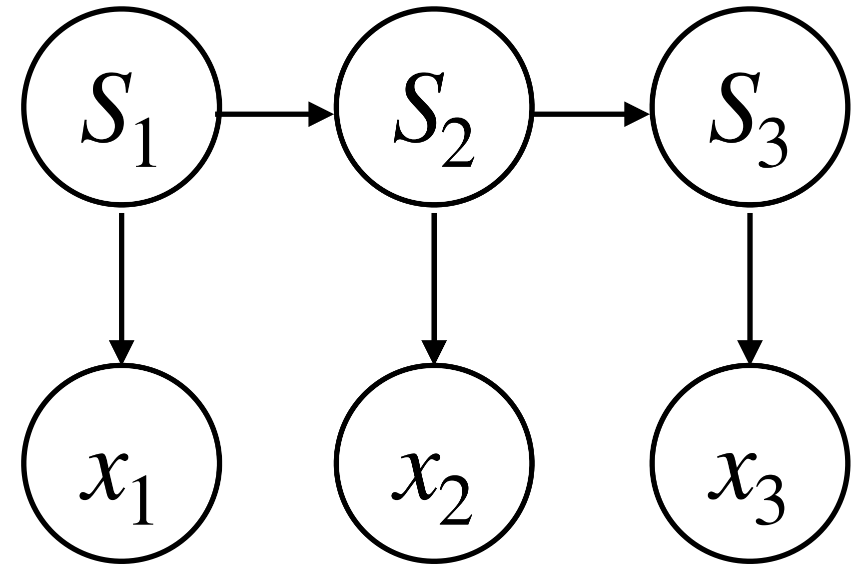
em.hmm

Lili Mou
lmou@ualberta.ca
lili-mou.github.io

UNIVERSITY OF ALBERTA

Unsupervised Learning



- Suppose an HMM model is given
- Training

$$\mathscr{D} = \left\{ \left(x_1^{(i)}, x_2^{(1)}, \cdots, x_{T^{(i)}}^{(i)} \right) \right\}_{i=1}^{n}$$

- Inference
  - Given an unseen sample $x_1, x_2, \cdots, x_T$
  - Predict their states $s_1, s_2, \cdots, s_T$

General Criteria for Latent Variables

- Training
  - Marginalization
    - Something of $\mathbb{E}$
    - $\mathbb{E}$ of something
    - All sorts of variants

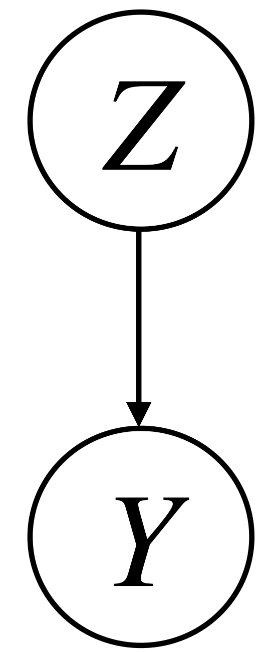
- Inference (depending on applications)
  - Target prediction: Marginalization
  - Latent variable prediction
    - Max *a posteriori*
    - Sampling

UNIVERSITY OF ALBERTA

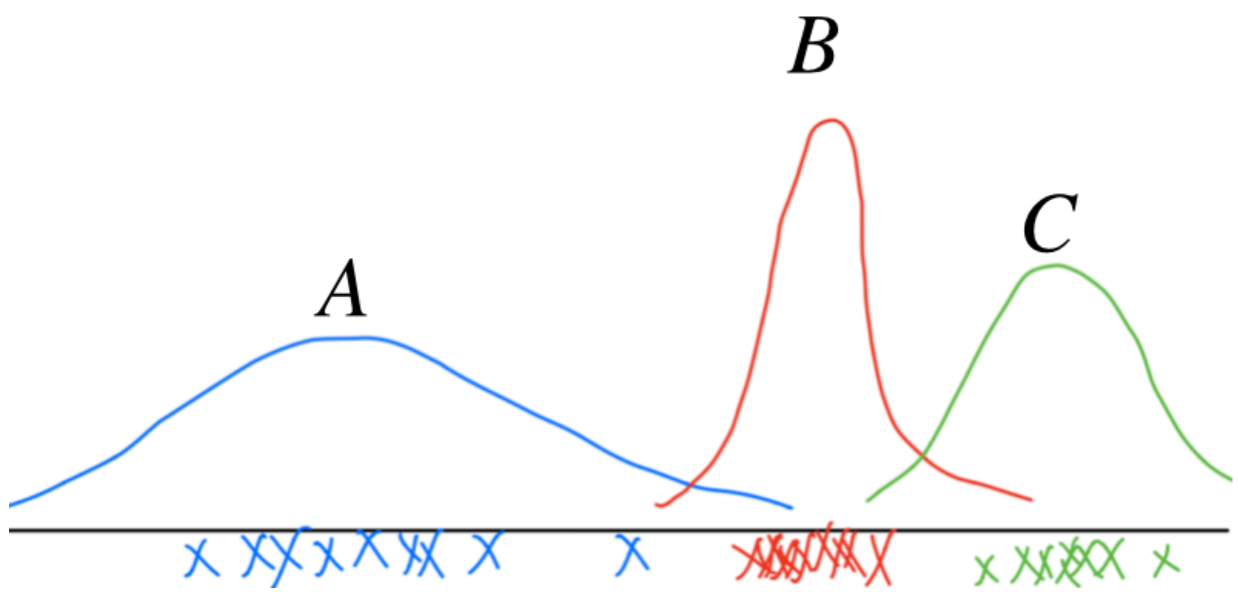
Gaussian Mixture Model

- **Gaussian mixture model:** $z^{(n)} \to \boldsymbol{y}^{(n)}$

  $$z^{(n)} \in \{1, \cdots, K\}, \boldsymbol{y}^{(n)} \in \mathbb{R}^d$$



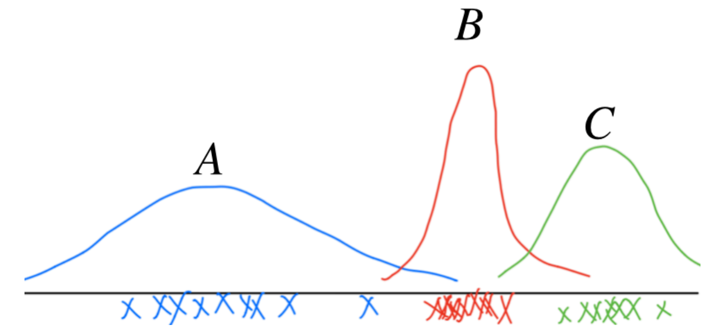
- Generative process:
  - Generate $z^{(n)} \sim \text{cat}(\pi_1, \pi_2, \cdots, \pi_k)$
  - Given $z^{(n)} = k$, generate $\boldsymbol{y}^{(n)} \sim \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$



Bishop CM. *Pattern Recognition and Machine Learning*. Springer, 2006.

UNIVERSITY OF ALBERTA

Expectation Maximization



- **Gaussian mixture model:** $z^{(n)} \to \boldsymbol{y}^{(n)}$

$$z^{(n)} \in \{1, \cdots, K\}, \boldsymbol{y}^{(n)} \in \mathbb{R}^d$$

- Expectation maximization
  - **E-step:** Evaluate posterior of each latent category

$$w_k^{(i)} = \frac{\pi_k \mathcal{N}(\boldsymbol{y}^{(n)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_k \mathcal{N}(\boldsymbol{y}^{(n)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}$$

  - **M-step:** Estimate model parameter

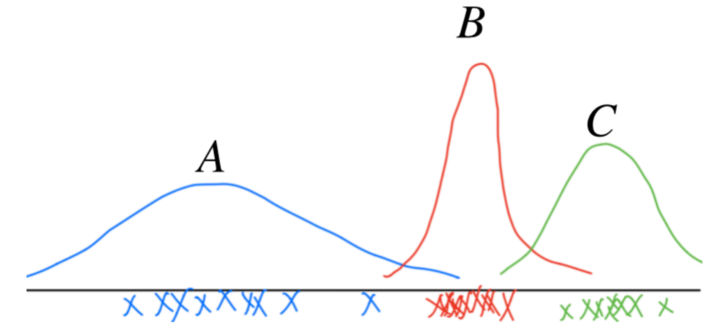
$$\boldsymbol{\mu}_k^{(new)} = \frac{1}{N_k} \sum_{n=1}^{N} w_k^{(i)} \boldsymbol{y}^{(n)}$$

$$\boldsymbol{\Sigma}_k^{(new)} = \frac{1}{N_k} \sum_{n=1}^{N} w_k^{(i)} (\boldsymbol{y}^{(n)} - \boldsymbol{\mu}_k)(\boldsymbol{y}^{(n)} - \boldsymbol{\mu}_k)^\top$$

$$\pi_k^{new} = \frac{N_k}{N} \qquad \text{where} \qquad N_k = \sum_{i=1}^{N} w_k^{(i)}$$

UNIVERSITY OF ALBERTA

EM as MLE



- Likelihood involves marginalization

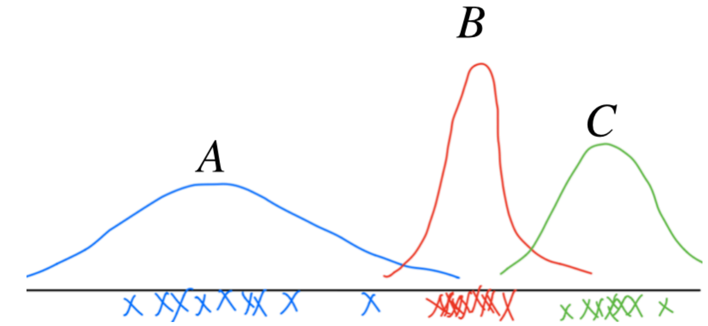
$$\log p(\boldsymbol{Y};\boldsymbol{\theta}) = \log\left(\sum\nolimits_z p(\boldsymbol{Y},z;\boldsymbol{\theta})\right)$$

$$= \underbrace{\sum_z q(z\,|\,\boldsymbol{Y}) \log \frac{p(\boldsymbol{Y},z;\boldsymbol{\theta})}{q(z\,|\,\boldsymbol{Y})}}_{\substack{L(q,\boldsymbol{\theta}) \\ \text{Lower bound}}} + \underbrace{\sum_z q(z\,|\,\boldsymbol{Y}) \log \frac{q(z\,|\,\boldsymbol{Y})}{p(z\,|\,\boldsymbol{y};\boldsymbol{\theta})}}_{\mathrm{KL}(q(Z\,|\,\boldsymbol{Y})\|p(Z\,|\,\boldsymbol{Y}))}$$

For those only/over-familiar with VAE:

KL here is different from KL within the lower bound

EM as MLE



- Likelihood involves marginalization

$$\log p(\boldsymbol{Y};\boldsymbol{\theta}) = \log\left(\sum\nolimits_z p(\boldsymbol{Y}, z;\boldsymbol{\theta})\right)$$

$$= \underbrace{\sum_z q(z\,|\,\boldsymbol{Y})\,\log\frac{p(\boldsymbol{Y}, z;\boldsymbol{\theta})}{q(z\,|\,\boldsymbol{Y})}}_{L(q,\boldsymbol{\theta})} + \underbrace{\sum_z q(z\,|\,\boldsymbol{Y})\,\log\frac{q(z\,|\,\boldsymbol{Y})}{p(z\,|\,\boldsymbol{y};\boldsymbol{\theta})}}_{\mathrm{KL}(q(Z\,|\,\boldsymbol{Y})\|p(Z\,|\,\boldsymbol{Y}))}$$

- **E-step**: Fix $\boldsymbol{\theta}$, maximize $L(q,\boldsymbol{\theta})$ wrt $q(Z\,|\,\boldsymbol{Y})$
  - Equivalent to minimize KL($\cdot\|\cdot$), as $\log p(\boldsymbol{Y}\,|\,\boldsymbol{\theta})$ is constant
  - $q(Z\,|\,\boldsymbol{Y}) \stackrel{set}{=} p(Z\,|\,\boldsymbol{Y})$
- **M-step**: Fix $q(\,\cdot\,|\,\cdot\,)$, maximize $L(q,\boldsymbol{\theta})$ wrt $\boldsymbol{\theta}$

EM as MLE

$$\ell(\boldsymbol{\theta}_{t+1}) = \sum_i \log p(\boldsymbol{y}_i; \boldsymbol{\theta}_{t+1})$$

$$= \sum_i \log \left(\sum_z p(\boldsymbol{y}_i, z; \boldsymbol{\theta}_{t+1}) \right)$$

[Lower bound holds for any q_t]

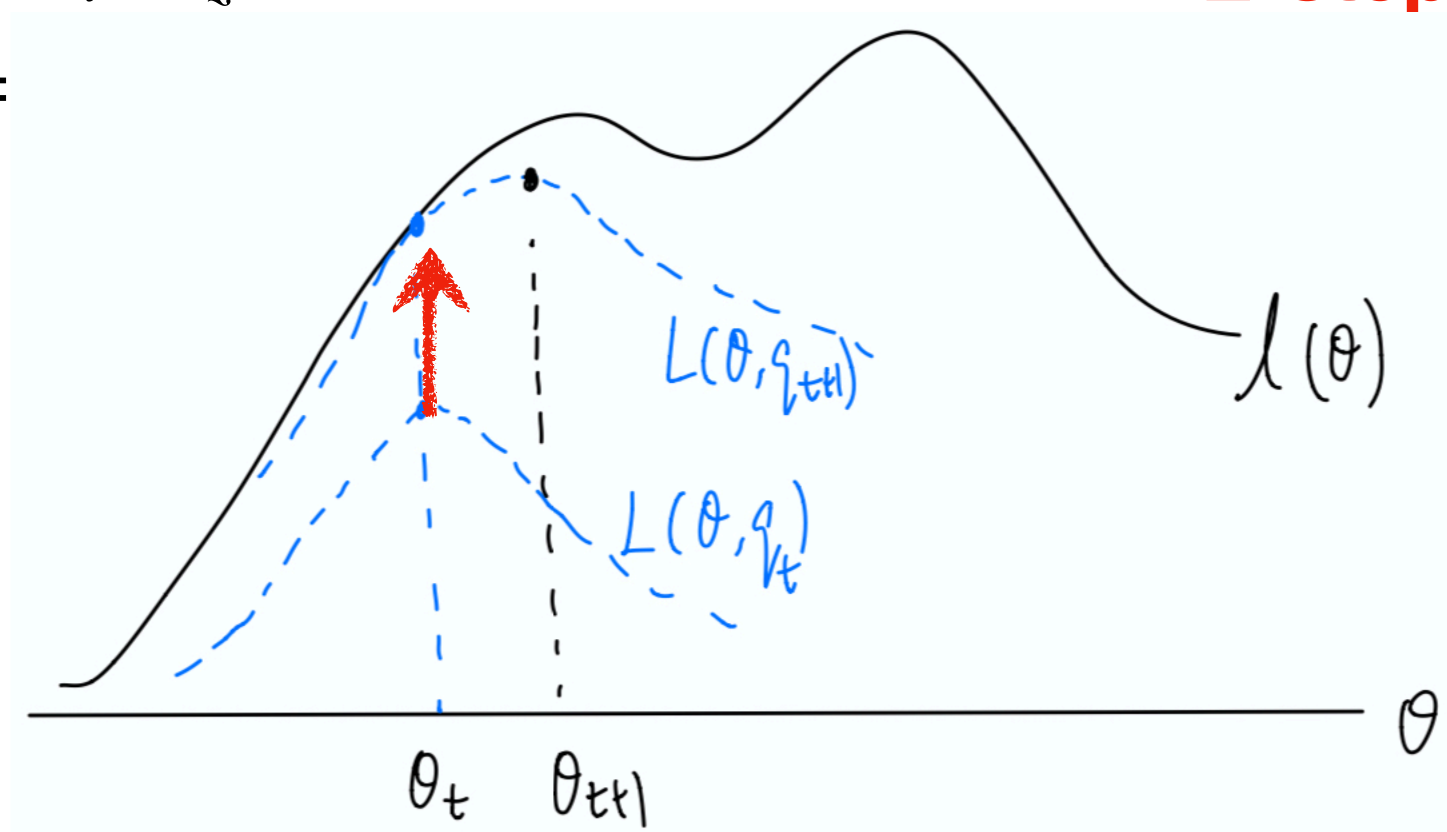
$$\geq \sum_i \sum_z q_t(z \,|\, \boldsymbol{y}_i) \log \frac{p(\boldsymbol{y}_i, z; \boldsymbol{\theta}_{t+1})}{q_t(z \,|\, \boldsymbol{y}_i)}$$

M-step: $\boldsymbol{\theta}_{t+1} = \arg\max\{\ \cdot\ \}$

$$\geq \sum_i \sum_z q_t(z \,|\, \boldsymbol{y}_i) \log \frac{p(\boldsymbol{y}_i, z; \boldsymbol{\theta}_t)}{q_t(z \,|\, \boldsymbol{y}_i)}$$

E-step: make lower bound tight

$=$



EM as MLE

$$\ell(\boldsymbol{\theta}_{t+1}) = \sum_i \log p(\boldsymbol{y}_i; \boldsymbol{\theta}_{t+1})$$

$$= \sum_i \log \left(\sum_z p(\boldsymbol{y}_i, z; \boldsymbol{\theta}_{t+1}) \right)$$

$$\geq \sum_i \sum_z q_t(z \,|\, \boldsymbol{y}_i) \log \frac{p(\boldsymbol{y}_i, z; \boldsymbol{\theta}_{t+1})}{q_t(z \,|\, \boldsymbol{y}_i)}$$

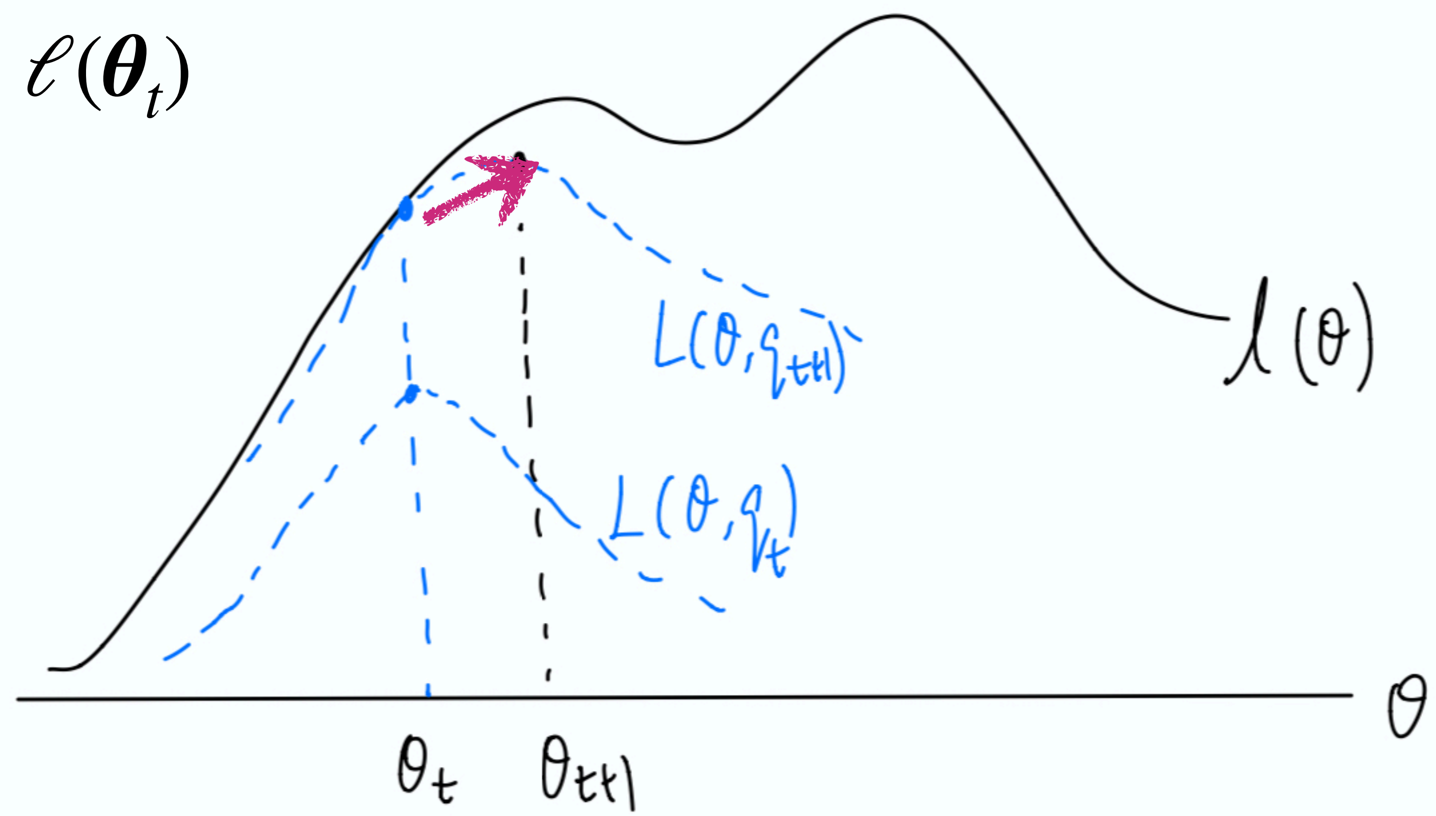
$$\geq \sum_i \sum_z q_t(z \,|\, \boldsymbol{y}_i) \log \frac{p(\boldsymbol{y}_i, z; \boldsymbol{\theta}_t)}{q_t(z \,|\, \boldsymbol{y}_i)}$$

$$= \ell(\boldsymbol{\theta}_t)$$

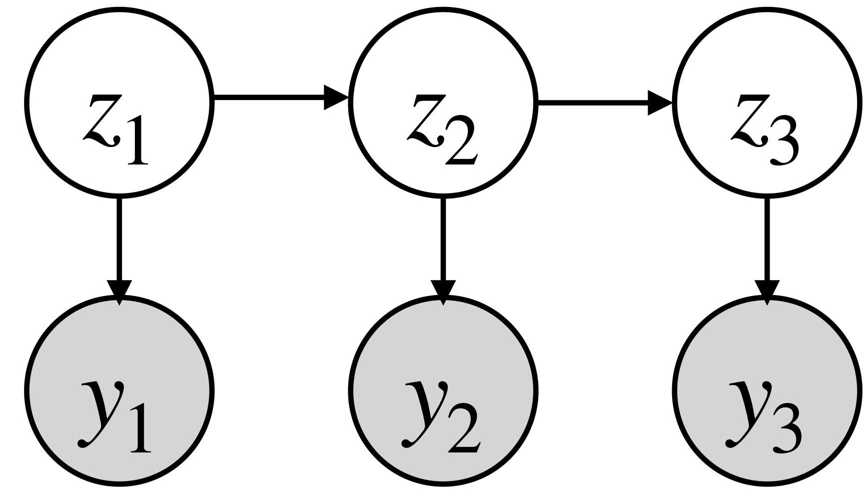
[Lower bound holds for any q_t]

M-step: $\boldsymbol{\theta}_{t+1} = \arg\max\{\,\cdot\,\}$

E-step: make lower bound tight



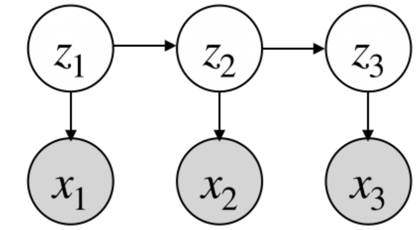
Hidden Markov Models



- Observed tokens: $y_1, y_2, \cdots, y_T$
- Latent states: $z_1, \cdots, z_T$
- Generative procedure
  - Choose z_1 (omitted here)
  - For every step t:
    - Pick $z_t \sim p(z_t \mid z_{t-1})$
    - Emit $y_t \sim p(y_t \mid z_t)$
  - Suppose both parametrized by probability tables
- Example
  - $y_1, y_2, \cdots, y_T$: a sequence of words
  - $z_1, z_2, \cdots, z_T$: POS tags

UNIVERSITY OF ALBERTA

Hidden Markov Models



- **E-step** (expectation for sufficient statistics)
  - Expectation of a state, that is, $\gamma_t(i) \triangleq \mathbb{E}[z_t = i \mid \cdot\ ]$
  - Expectation of two consecutive states, that is, $\xi_t(i, j) \triangleq \mathbb{E}[z_t = i, z_{t+1} = j \mid \cdot\ ]$
  - Computed by

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{p(\boldsymbol{Y})} \qquad \xi_t(i, j) = \frac{\alpha_t(i)p_{\boldsymbol{\theta}}(x_t \mid z_n = i)p_{\boldsymbol{\theta}}(z_t = j \mid z_{t-1} = i)\beta_t(j)}{p(\boldsymbol{Y})}$$

where

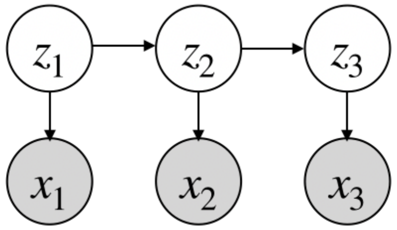
$$\alpha_t(i) \triangleq p(\boldsymbol{y}_{1:t}, z_t = i)$$

and

$$\beta_t(i) \triangleq p(\boldsymbol{y}_{t+1:T} \mid z_t = i)$$

are given by dynamic programming

Dynamic Programming



$$\alpha_t(i) \triangleq p(\boldsymbol{y}_{1:t}, z_t)$$

- Initialization

$$\alpha_1(i) \triangleq p(x_1, z_1 = i) = \pi_i \cdot p(x_1 \mid z_1 = i)$$

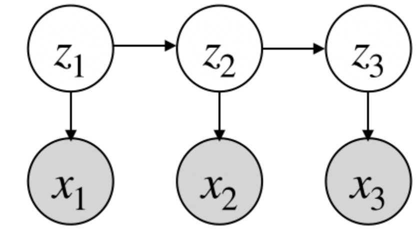
- Recursion

$$\alpha_t(i) = \sum_j \alpha_{t-1}(i) p(s_t = i \mid s_{t-1} = j) p(x_t \mid s_t = j)$$

- Termination

When $t = T$

Dynamic Programming



$$\beta_t(i) \triangleq p(\boldsymbol{y}_{t+1:T} \mid z_t)$$

- Initialization

$$\beta_T(i) = 1$$

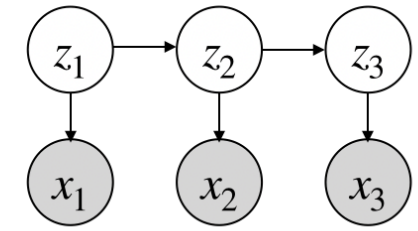
- Recursion

$$\beta_t(i) = \sum_j \beta_{t+1}(j) p(s_{t+1} = j \mid s_t = i) p(x_{t+1} \mid s_{t+1} = j)$$

- Termination

When $t = 1$

Hidden Markov Models



- **E-step** (expectation for sufficient statistics)
  - Expectation of a state, that is, $\gamma_t(i) \triangleq \mathbb{E}[z_t = i \mid \cdot\,]$
  - Expectation of two consecutive states, that is, $\xi_t(i,j) \triangleq \mathbb{E}[z_t = i, z_{t+1} = j \mid \cdot\,]$
- **M-step** (MLE by soft counting)

$$p(z_t = j \mid z_{t-1} = i) = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$$

$$p(x \mid z_t = j) = \frac{\sum_{t=1}^{T} \gamma_t(j) \mathbb{1}\{X_t = x\}}{\sum_{t=1}^{T} \gamma_t(j)}$$

Other Treatments

$$\log p(\boldsymbol{Y} \mid \boldsymbol{\theta}) = \log\left(\boxed{\sum_z} p(\boldsymbol{Y}, z \mid \boldsymbol{\theta})\right)$$

- Exact marginalization (enumeration as in GMM, DP as in HMM)
- Choose the single best z
  - E.g., K-means clustering
- Choose top-N latent variables
  - Beam search
- Sampling
- Back propagation
  - If Y continuous, be careful of the degenerated distribution
  - If $p(Y \mid z)$ is by CPT, be aware of the constraint $\sum_y p(y \mid z) = 1$

Assignment

- Consider a Bayesian network: $X \to Z \to Y$
- All variables are discrete, taking N_x, N_y, N_z values, resp.
- Observation: $\{(x_i, y_i)\}_{i=1}^{M}$
- Goal:
  - Figure out parameters as in conditional probability tables
  - Give an EM algorithm to estimate the parameters. Note that z is unobserved.

Suggested Reading

- CS229
  - Note: http://cs229.stanford.edu/notes/cs229-notes8.pdf
  - Video: https://www.youtube.com/watch?v=ZZGTuAkF-Hw&list=PLEBC422EC5973B4D8&index=12
- Chap 9, Bishop, *Pattern Recognition and Machine Learning*.
- Rabiner, L.R., 1989. A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE, 77(2), pp.257-286.

Thank you!

Q&A

UNIVERSITY OF ALBERTA