CS461 Homework 5

Due: Optional (Due Dec. 16 11:59pm)

1. [EM Algorithm] Using the four data points provided below, perform a single iteration of the Expectation-Maximization (EM) algorithm for a Gaussian Mixture Model (GMM): $f_X(x) = \pi_0 \cdot \mathcal{N}(\mu_0, \sigma_0^2) + \pi_1 \cdot \mathcal{N}(\mu_1, \sigma_1^2)$.

data num	x
d_1	2
d_2	1
d_3	-1
d_4	-2

1.1 Compute the log-likelihood for the parameters provided below. Assume that the four points are independent and identically distributed (i.i.d).

- $\pi_0(t) = \pi_1(t) = 1/2$
- $\mu_0(t) = -1, \mu_1(t) = 1$
- $\sigma_0^2(t) = \sigma_1^2(t) = 1$.

sol)

$$\log p(d_1, d_2, d_3, d_4) = \sum_{n=1}^{4} \log p(d_n) = \sum_{n=1}^{4} \log \{1/2 \cdot \frac{1}{\sqrt{2\pi}} \exp \frac{-(d_i+1)^2}{2} + 1/2 \cdot \frac{1}{\sqrt{2\pi}} \exp \frac{-(d_i-1)^2}{2}\}$$
$$= -7.16$$

1.2 E-step: compute γ_{n0} and γ_{n1} for all $n = 1, 2, 3, 4$.

$$\gamma_{n0} = P(C_0|d_n) = \frac{P(d_n|C_0)P(C_0)}{P(d_n)}$$
$$= \frac{P(d_n|C_0)P(C_0)}{P(d_n|C_0)P(C_0) + P(d_n|C_1)P(C_1)}$$
$$\gamma_{n0} = 1 - \gamma_{n1}$$

n	γ_{n0}	γ_{n1}
1	0.018	0.98
2	0.12	0.88
3	0.88	0.12
4	0.98	0.018

1.3 M-step: update the parameters $\pi_0(t+1), \pi_1(t+1), \mu_0(t+1), \mu_1(t+1), \sigma_0^2(t+1), \sigma_1^2(t+1)$.

$$\pi_0(t+1) = \frac{\sum_{n=1}^{4} \gamma_{n0}}{N} = 0.5$$
$$\pi_1(t+1) = 1 - \pi_0(t+1) = 0.5$$
$$\mu_0(t+1) = \frac{\sum_{n=1}^{4} \gamma_{n0} d_n}{\sum_{n=1}^{4} \gamma_{n0}} = -1.3448$$
$$\mu_1(t+1) = \frac{\sum_{n=1}^{4} \gamma_{n1} d_n}{\sum_{n=1}^{4} \gamma_{n1}} = 1.3448$$
$$\sigma_0^2(t+1) = \frac{\sum_{n=1}^{4} \gamma_{n0} (d_n - \mu_0(t+1))^2}{\sum_{n=1}^{4} \gamma_{n0}} = 0.7$$
$$\sigma_1^2(t+1) = \frac{\sum_{n=1}^{4} \gamma_{n1} (d_n - \mu_1(t+1))^2}{\sum_{n=1}^{4} \gamma_{n1}} = 0.7$$

1.4 Compute log-likelihood using the updated parameters and check the log-likelihood value increases.
sol) updated LL = -6.47. The log-likelihood value has increased.

2. [Exact vs. Approximate Inference] Given the Bayesian network below, compute P[Cloudy | Sprinkler = T, WetGrass = T] by using Variable Elimination and Gibbs Sampling.

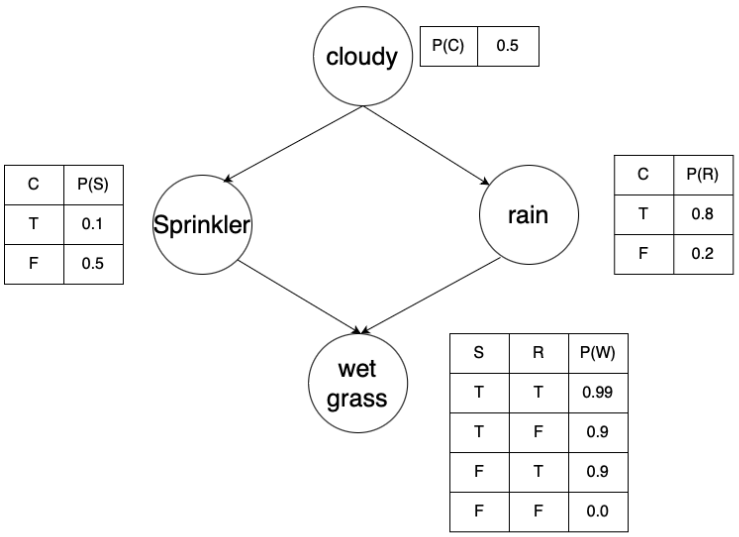


Figure 1: Bayesian Network

2.1 Compute the posterior by using Variable Elimination.
sol)

$$
\begin{aligned}
P[C \mid S:+, W:+] &= \alpha P[C, S:+, W:+] = \alpha \sum_R P[C, S:+, W:+, R] \\
&= \alpha \sum_R P[C] P[S+|C] P[R|C] P[W|S+, R] \\
&= \alpha P[C] \cdot P[S+|C] \sum_R P[R|C] P[W|S+, R] \\
&= \alpha P[C] \cdot P[S+|C] \cdot (P[R+|C] P[W+|S+, R+] + P[R-|C] P[W+|S+, R-]) \\
&= \alpha P[C] \cdot P[S+|C] \cdot (0.99 \cdot \begin{bmatrix} 0.8 \\ 0.2 \end{bmatrix} + 0.9 \cdot \begin{bmatrix} 0.2 \\ 0.8 \end{bmatrix}) \\
&= \alpha \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} \cdot \begin{bmatrix} 0.1 \\ 0.5 \end{bmatrix} \cdot (0.99 \cdot \begin{bmatrix} 0.8 \\ 0.2 \end{bmatrix} + 0.9 \cdot \begin{bmatrix} 0.2 \\ 0.8 \end{bmatrix}) = \begin{bmatrix} 0.05 \\ 0.23 \end{bmatrix}
\end{aligned}
$$

$$
P[C \mid S:+, W:+] = \begin{bmatrix} 0.175 \\ 0.825 \end{bmatrix}
$$

2.2 Estimate the posterior by using Gibbs Sampling.
sol)

- given S+ W+, C and R are initialized randomly. For example, we start with sample D(1): S:+, W:+, C:+, R: -
- compute $P[C|S+R-] = \alpha P[C] P[S+|C] P[R-|C]$ and sample C based on the computed probability.

$$
\alpha \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} \cdot \begin{bmatrix} 0.1 \\ 0.5 \end{bmatrix} \cdot \begin{bmatrix} 0.2 \\ 0.8 \end{bmatrix} = \alpha \begin{bmatrix} 0.01 \\ 0.2 \end{bmatrix} = \begin{bmatrix} 0.048 \\ 0.952 \end{bmatrix}
$$

- Suppose we got C- then the next sample is D(2): S:+, W:+, C:-, R: -
- compute $P[R|S+C-W+]$ and sample R based on the probability.

- suppose we got R+, then the next sample is D(3): S:+, W:+, C:-, R: +
- repeat the process multiple times.
- estimate $P[C + | S + W +] \approx$ (the number of samples containing (C+ S+ W+) / (the number of samples containing S+ W+ : the number of all data samples).

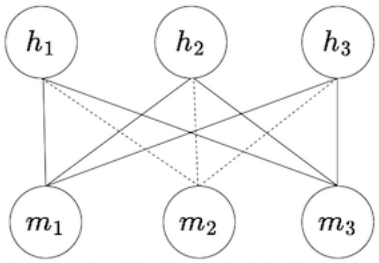
3. [VAE Evidence Lower Bound (ELBO)] Derive the following inequality below.

$$\begin{aligned}
\log p_\theta(x_i) &= \log \sum_z p_\theta(x_i, z) \\
&= \log \sum_z \frac{p_\theta(x_i, z) q_\phi(z|x_i)}{q_\phi(z|x_i)} \\
&\geq E_{q_\phi(z|x_i)}[\log p_\theta(x_i, z) - \log q_\phi(z|x_i)] \\
&= -D_{KL} q_\phi(z|x_i)||p_\theta(z) + E_{q_\phi(z|x_i)}[\log p_\theta(x_i|z)]
\end{aligned}$$

sol)

$$\begin{aligned}
\log p_\theta(x_i) &= \log \sum_z p_\theta(x_i, z) \\
&= \log \sum_z \frac{p_\theta(x_i, z) q_\phi(z|x_i)}{q_\phi(z|x_i)} \\
&\geq E_{q_\phi(z|x_i)}[\log p_\theta(x_i, z) - \log q_\phi(z|x_i)] \\
&= E_{q_\phi(z|x_i)}[\log p_\theta(z) + \log p_\theta(x_i|z) - \log q_\phi(z|x_i)] \\
&= E_{q_\phi(z|x_i)}[\log p_\theta(z) - \log q_\phi(z|x_i)] + E_{q_\phi(z|x_i)}[\log p_\theta(x_i|z)] \\
&= -D_{KL} q_\phi(z|x_i)||p_\theta(z) + E_{q_\phi(z|x_i)}[\log p_\theta(x_i|z)]
\end{aligned}$$

4. [RBM Movie Recommendation System] It is known that a part of Netflix's recommendation system utilizes a Restricted Boltzmann Machine (RBM). Suppose you are given an RBM and energy function as below and the preference of a user for movie1 (m_1) and movie3 (m_3), but no information is given for movie2 (m_2). Predict if the user likes movie2 or not. For the prediction, you will need a sampling process. Assume the sampling process is deterministic; $x = 1$ if $P(x) > 1/2$ and $x = -1$ (bipolar case) otherwise.



$$\mathcal{E}(v,h) = -v^t W h - b^t v - c^t h$$

W			
	-1	1	0
	0	1	2
	-1	-2	3

b	
	1
	-1
	1

c	
	-2
	0
	1

Figure 2: RBM and Energy Function of Recommendation System

4.1 In this recommendation system, the RBM needs to operate on bipolar coding $(+1, -1)$ instead of using binary coding 1 and 0. Why is this necessary?
sol) we need the bipolar coding to handle the missing visible information. With binary coding, disliked movie would be encoded as "0", which would be indistinguishable from missing information.

4.2 In the class, we studied the conditional probability of an RBM using binary coding 1 and 0. The formulation is shown below. How would you modify the formulation for the RBM using bipolar coding?

$$P(h_1 = 1|m_1, m_2, m_3) = \frac{1}{1 + \exp^{-m^t W[:,1] - c_1}} = \sigma(m^t W[:,1] + c_1)$$
$$P(m_1 = 1|h_1, h_2, h_3) = \frac{1}{1 + \exp^{-W[1,:]h - d_1}} = \sigma(W[1,:]h + b_1)$$

$$P(h_1 = 1|m_1, m_2, m_3) = \alpha \exp^{m^t W[:,1] + c_1}$$
$$P(h_1 = -1|m_1, m_2, m_3) = \alpha \exp^{-m^t W[:,1] - c_1}$$
$$P(h_1 = 1|m_1, m_2, m_3) = \frac{\exp^{m^t W[:,1] + c_1}}{\exp^{m^t W[:,1] + c_1} + \exp^{-m^t W[:,1] - c_1}}$$
$$= \frac{1}{1 + \exp^{-2m^t W[:,1] - 2c_1}} = \sigma(2m^t W[:,1] + 2c_1)$$

$$P(m_1 = 1|h_1, h_2, h_3) = \alpha \exp^{W[1,:]h + b_1}$$
$$P(m_1 = -1|h_1, h_2, h_3) = \alpha \exp^{-W[1,:]h - b_1}$$
$$P(h1 = 1|m_1, m_2, m_3) = \frac{\exp^{W[1,:]h + b_1}}{\exp^{W[1,:]h + b_1} + \exp^{-W[1,:]h - b_1}}$$
$$= \frac{1}{1 + \exp^{-2W[1,:]h - 2b_1}} = \sigma(2W[1,:]h + 2b_1)$$

4.3 Suppose the user liked m_1 but disliked m_3. Based on the user's preference, predict the preference for m_2.
sol)

$$P[v1, v2, v3|m_1 = +, m_3 = -] = \sigma(2[+1, 0, -1]^t W + 2[-2, 0, 1]))$$

$$2[+1, 0, -1]^t W + 2 \begin{bmatrix} -2 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -2 \\ 3 \\ -2 \end{bmatrix}$$

Hence, the hidden layer prediction will be $\begin{bmatrix} -1 \\ 1 \\ -1 \end{bmatrix}$

$$P[m_2|h_1 = -1, h_2 = 1, h_3 = -1] = \sigma(2W[-1, 1, -1] + 2[1, -1, 1]))$$

$$2W[-1, 1, -1] + 2 \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 \\ -2 \\ 3 \end{bmatrix}$$

Hence, the preference prediction for movie2 is "disliked".