PAGE NO.
DATE: / /

Question 1

Describe steps needed to sample from such a model

=>

We first obtain set of sample $z^{(l)}$ where $i = 1, \ldots L$

=> We allow expectation

$$E[f] = \int f(z)\, p(z)\, dz$$

to be approximated by

$$\hat{f} = \frac{1}{L} \sum_{i=1}^{L} f(z)^{(l)}$$

=> We let directed graph with some nodes instantiated with values given by

$$P(L = l^{\circ}, S = s^{\circ}) = \sum_{D, I, G} P(D)\, P(I)\, P(G)$$

=> Next we input the factor graph

=> We order all given variables from top to bottom so that the parent variables come before child variables

=> Next, for each variable in the parent variable, we retrieve their sampled values which exist already after the ordering

=> However, with the parent factor being deterministic, this will simply compute the child value

=> Lastly, we output the sampled values within each variable in the graph

PAGE NO.:
DATE: / /

Question 2

=> This is because we cannot compute any given new point likelihood say y under the probability distribution efficiently, since it involves integrating over the hidden variable

=> Also the naive spherical Gaussian noise model produces noisy sample or rather overlly smooth ones instead if noise is not added to it.

=> Thus the system may converge into a local minimum for which any latent variable is completely ignored with the encoder predicting prior resulting into posterior collapse

Question 3

=> Example resulting very small KL-divergence

$$\Rightarrow KL(p,q) = \log \frac{\sigma_1}{\sigma_2} + \frac{\sigma_2^2 - (\mu_1 + \mu_2)^2}{2\sigma_1^2} - \frac{1}{2}$$

=> Example resulting very large KL-divergence

$$KL(p,q) = \log \frac{\sigma_2}{\sigma_1} + \frac{\sigma_1^2 + (\mu_1 - \mu_2)^2}{2\sigma_1^2} + \frac{1}{2}$$

DATE: / /

Question 4

=> This will be because the log-probability will be an expectation under the complicated posterior distribution which is what we are to approximate, and we cannot usually evaluate it

=> However, with the lower-bound this encourages the fit to concentrate on the plausible parameters

Question 5

=> ① This basically optimizes the likelihood using the approximate distribution thus maximizing the log-likelihood

=> ② This forces the approximations to match the true posterior meaning that when Variational approximate posterior is perfect then KL-Vanishes

Question 6

=> So with the optimization of the log-likelihood this results into ignoring the precision which only can be fine-tuned through hyper-parameter tuning hence will result into balancing problems for the relevant parameters

Question 7

=> Given that $P_\theta(z_n|x_n)$ we have that

$$P_\theta(z|x) = \frac{P_\theta(x|z)P_\theta(z)}{P_\theta(x)}$$

$$= \frac{P_\theta(x|z)P_\theta(z)}{\int P_\theta(x|z)P_\theta(z)dz}$$

=> This is equivalent of $q_\phi(z,x)$

=> Thus, the log-marginal likelihood will be given as

$$\Rightarrow \log P_\theta(x) = D_{KL}\left[q_\phi(z|x) \| P_\theta(z|x)\right] + L(\theta,\phi|x)$$

$$\Rightarrow L(\theta,\phi|x) = E_{q_\phi(z,x)}\left(\log P_\theta(x,z) - \log q_\phi(z|x)\right)$$

$$\Rightarrow \int\left[\log P_\theta(x|z) - \log q_\phi(z|x) + \log P_\theta(z)\right] q_\phi(z|x)\,dz$$

=> This simplifies into

$$\int \frac{P_\theta(z)}{q_\phi(z|x)} q_\phi(z|x)dz + \int \log P_\theta(x|z) q_\phi(z|x)dz$$

$$\Rightarrow E_{q_\phi(z|x)}\left[\log P_\theta(x|z) - D_{KL}\left[q_\phi(z|x) \| P_\theta(z)\right]\right]$$

=> Optimizing the ELBO, we have the mean lower bound over sample given as

DATE:

$\Rightarrow L(\theta,\phi|x) = -\frac{1}{N}\sum_{n=1}^{N} -D_{KL}\left(q_\phi(z|x_n)\,||\,P_\theta(z)\right) + E_{q\phi}\left[\log p_\theta(x_n|z)\right]$

$\Rightarrow$ let the reconstruction term be given as

$$\Rightarrow L_n^{recon} = -E_{q_\phi(z|x_n)}\left[\log p_\theta(x_n|z)\right]$$

with

$$\Rightarrow L_n^{reg} = D_{KL}\left(q_\theta(z|x_n)\,||\,P_\theta(z)\right)$$

$\Rightarrow$ Thus the loss minimization function becomes

$$L(\theta,\phi|x) = \frac{1}{N}\sum_{n=1}^{N} L_n^{recon} + l_n^{reg}$$

$\boxtimes$

<u>Question 8</u>

$\Rightarrow$ This normally happens because the network simply involves a sampling step and there is literally zero way to differentiate through this, implying that it will be impossible to make updates to the parameters that will occur earlier within the network

$\Rightarrow$ However, via reparametrization trick we can simply move the stochastic part into numerous branches of the network which inturn draws a sample from the given norm to draw from intended Gaussian function

$\Rightarrow$ Under this approach we can compute the derivatives since there is no need for the back propagation algorithm to pass down the stochastic branch

Question 9 & 10

See attached code files

Question 11

$\Rightarrow$ From $L(x; \theta_d, \phi_g) = \frac{1}{n} \sum_{i=1}^{n} [\log D_\phi(x^{(i)}) + \log(1 - D_\phi(G_\theta(z^{(i)})))]$

$\Rightarrow \nabla_{\theta_d} L(x; \theta_d, \phi_g) = \frac{1}{n} \sum_{i=1}^{n} \nabla_{\theta_d} \log D_\phi(x^{(i)}) + \nabla_{\theta_d} \log(1 - D_\phi(G_\theta(z^{(i)})))]$

$= \frac{\nabla_{\theta_d}}{n} \sum_{i=1}^{n} \log D_\phi(x^{(i)}) + \log(1 - D_\phi(G_\theta(z^{(i)})))]$

Question 12

$\Rightarrow$ From

$$L(\theta_d, \theta_g) = \frac{1}{n} \sum_{i=1}^{n} \log D(x^{(i)}) + \log(1 - D(G(z^{(i)})))$$

$$\Rightarrow \frac{\partial L(\theta_d, \theta_g)}{\partial z_d} = \frac{1}{n} \sum_{i=1}^{n} 0 + \frac{(0 - D(G(z^{(i-1)})))}{1 - D(G(z^{(i)}))}$$

$$= \frac{1}{n} \sum_{i=1}^{n} \frac{-D(G(z^{(i-1)}))}{1 - D(G(z^{(i)}))}$$

Question 13

$$\Rightarrow \frac{\partial L(\theta_d, \theta_g)}{\partial z_d^L} = \frac{1}{n} \sum_{i=1}^{n} \frac{-D(G(z^{(i+1)}))}{1 - D(G(z^{(i)}))}$$

for next value $i+1$ we will have

$$= \frac{1}{n} \sum_{i=1}^{n} \frac{-D(G(z^{(i+1-1)}))}{1 - D(G(z^{(i+1)}))}$$

$$= \frac{1}{n} \sum_{i=1}^{n} \frac{-D(G(z^{(i)}))}{1 - D(G(z^{(i+1)}))}$$

with $w_d^{L+1} = D(G(z^{i+1}))$ and

$$g_d^L = D(G(z^{(i)}))$$

$$= \frac{1}{n} \sum_{i=1}^{n} \frac{-g_d^L}{1 - w_d^{L+1}}$$

Question 14

$\Rightarrow$ with the output generator function

$$L(\theta_d, \theta_g) = \frac{1}{n} \sum_{i=1}^{n} \log(1 - D(G(z^{(i)}))$$

$$\Rightarrow \frac{\partial L(\theta_d, \theta_g)}{\partial g(z_i, \theta_g)} = \frac{1}{n} \sum_{i=1}^{n} \frac{D(z^{(i)})}{1 - D(G(z^{(i)}))} \quad \text{with}$$

$$w_g^i = D(z^{(i)})$$

and

$$\frac{\partial L(\theta_d, \theta_g)}{\partial z_d^L} = \frac{1}{1 - D(G(z^{(i)}))}$$

$$\Rightarrow \frac{1}{n} \sum_{i=1}^{n} \frac{D(z^{(i)})}{1 - D(G(z^{(i)}))}$$

Question 15

$$\Rightarrow L(\theta_d, \theta_g) = \frac{1}{n} \sum_{i=1}^{n} \log(1 - D(G(z^{(i)}))$$

$$\Rightarrow \nabla_{\theta_g} L(\theta_d, \theta_g) = \frac{1}{n} \sum_{i=1}^{n} \nabla_{\theta_g} \log(1 - D(G(z^{(i)}))$$

$$\Rightarrow \frac{\nabla_{\theta_g}}{n} \sum_{i=1}^{n} \frac{D(z^{(i)})}{1 - D(G(z^{(i)}))}$$

Question 16

$$\nabla_{\theta_d} L(\theta_d, \theta_g) = \frac{1}{n} \sum_{i=1}^{n} \nabla_{\theta_d} \log(1 - D(G(z^{(i)}))$$

$$= \frac{1}{n} \sum_{i=1}^{n} \nabla_{\theta_d} \frac{(0 - G(z^{(i)})}{1 - D(G(z^{(i)}))}$$

$$= \frac{1}{n} \sum_{i=1}^{n} \nabla_{\theta_d} \frac{(-G(z^{(i)}))}{1 - D(G(z^{(i)}))}$$

$$\Rightarrow \frac{\nabla_{\theta_d}}{n} \sum_{i=1}^{n} \frac{-G(z^{(i)})}{1 - D(G(z^{(i)}))}$$

PAGE NO.
DATE: / /

Question 17

$$\Rightarrow \nabla_{\theta g} L(\theta_d, \theta_g) = \frac{\nabla_{\theta g}}{n} \sum_{i=1}^{n} \log(1 - D(G(z^{(i)}))$$

with fixed learning rate

$$\Rightarrow \frac{1}{n} \sum_{i=1}^{n} \nabla_{\theta g} \frac{D(G(z^{(i)}))}{1 - D(G(z^{(i)}))}$$

$$\Rightarrow \frac{\nabla_{\theta g}}{n} \sum_{i=1}^{n} \frac{D(G(z^{(i)}))}{1 - D(G(z^{(i)}))}$$

Question 18

$\Rightarrow$ Deriving $\frac{\partial x_{L+1}}{\partial x_L}$

$$\Rightarrow \text{From} \quad \frac{\partial L}{\partial x^{L-1}} = [W^{(L)}]^T \cdot \frac{\partial L}{\partial x^L} \cdot g^{(L-1)} x^{(L-1)}$$

$$\Rightarrow \frac{\partial L}{\partial W^{(L)}} = \frac{\partial L}{\partial x^L} [a^{(L-1)}]^T$$

$$\Rightarrow \frac{\partial L}{\partial a^{(L)}} = \frac{\partial L}{\partial x^L}$$

$\Rightarrow$ For a mini-batch extension we have

$$a^{(L-1)} = \begin{pmatrix} a_1^{(L-1)} \\ a_2^{(L-1)} \\ \vdots \\ a_n^{(L-1)} \end{pmatrix}$$

PAGE NO:
DATE: / /

=> This error can be propagated for layer L back to its previous layer L-1 by

$$\Rightarrow \frac{\partial L}{\partial x^{(L-1)}} = \left[W^{(L)}\right]^T \cdot \frac{\partial L}{\partial x^{(L)}} \cdot g^{(L-1)'} x^{(L-1)}$$

=> Implying that for all mini-batch samples

$$\frac{\partial L}{\partial w^{(L)}} = \frac{1}{n} \sum_{i=1}^{n} \frac{\partial L^{(i)}}{\partial w^{(L)}}$$ and

$$\frac{\partial L}{\partial a^{(L)}} = \frac{1}{n} \sum_{i=1}^{n} \frac{\partial L^{(i)}}{\partial a^{(L)}}$$

=> Applying chain rule, we obtain that

$$\frac{\partial x_{L+1}}{\partial x_L} = \left.\frac{\partial x_{L+1}}{\partial z}\right|_{z=f(x)} \cdot \frac{\partial f(x)}{\partial x_L}$$

$$\Rightarrow \frac{\partial x_{L+1}(x_1, x_2, \ldots x_n)}{\partial x_L} = \left.\frac{\partial x_{L+1}(z_1, \ldots z_L)}{\partial z_L}\right|_{\substack{z_1 = f_1(x_1, x_2) \\ z_2 = f_2(x_1, x_2) \\ \vdots \\ z_L = f_L(x_1, x_2)}} \cdot p$$

where $p = \frac{\partial f(x_1, x_2, \ldots x_n)}{\partial x_L}$

=> For the one-pass and x_L for the top most layer we have.

DATE: / /

$$\frac{\partial x_{L+1}(x_1,x_2)}{\partial x_1} = \frac{\partial x_{L+1}(z_1,z_2)}{\partial z_1}\Bigg|_{\substack{z_1=f_1(x_1,x_2)\\ z_2=f_2(x_1,x_2)}} \cdot \frac{\partial f(x_1,x_2)}{\partial x_1}$$

which simplifies into

$$\frac{\partial x_{L+1}(x_1,\dots x_n)}{\partial x_L'} = \sum_{i=1}^{n} \frac{\partial x_{L+1}(z_1,\dots z_n)}{\partial z_i} \cdot \frac{\partial f_i(x_1,\dots x_n)}{\partial x_L}$$

Question 19

=) The Consequences is that when training deep models the backpropagation learning rule tends to get stuck in the local minima
=) This error will be a function of all weights in a given multidimensional space, However, there is no proof that this global minimum error surface has been reached already
=) Thus, the derivative of the given transfer function will vanish to zero for very small summed inputs and similarly for very large summed inputs as well

Question 20

See attached code file

THANK YOU!!!