Laboratorio 3

Autor: Arturo Lazcano **Profesor:** Felipe Tobar
Auxiliares: Cristóbal Alcázar, Camilo Carvajal Reyes
Ayudante: Joaquín Barceló

P1

(a) Por la definición de divergencia de Kullback-Leibler:

$$D_{KL}(q(z|x)||p(z|x)) = \mathbb{E}_{z\sim q(\cdot|x)}\left[\log\left(\frac{q(z|x)}{p(z|x)}\right)\right]$$

Luego, por Bayes se cumple que

$$p(x) = \frac{p(x,z)}{p(z|x)} \iff p(z|x) = \frac{p(x,z)}{p(x)} \tag{1}$$

Por lo tanto, reemplazando $p(z|x)$ en la primera ecuación,

$$\begin{aligned}\mathbb{E}_{z\sim q(\cdot|x)}\left[\log\left(\frac{q(z|x)}{p(z|x)}\right)\right] &= \mathbb{E}_{z\sim q(\cdot|x)}\left[\log\left(\frac{q(z|x)p(x)}{p(x,z)}\right)\right] = \mathbb{E}_{z\sim q(\cdot|x)}\left[\log\left(\frac{q(z|x)}{p(x,z)}\right) + \log(p(x))\right] \\ &= \mathbb{E}_{z\sim q(\cdot|x)}\left[\log\left(\frac{q(z|x)}{p(x,z)}\right)\right] + \mathbb{E}_{z\sim q(\cdot|x)}[\log(p(x))] \\ &= \mathbb{E}_{z\sim q(\cdot|x)}\left[\log\left(\frac{q(z|x)}{p(x,z)}\right)\right] + \log(p(x))\end{aligned}$$

Donde se usaron las propiedades del logaritmo, en particular que separa multiplicación en suma de logaritmos junto con la linealidad de la esperanza. Por último, notar que $\mathbb{E}_{z\sim q(\cdot|x)}[\log(p(x))] = \int \log p(x) q(z|x) dz = \log p(x) \int q(z|x) = \log p(x)$ ya que $q(z|x)$ es una distribución y por ende integra 1.

(b) Partiendo del lado derecho:

$$\begin{aligned}&D_{KL}(q(z|x)||p(z)) - \mathbb{E}_{z\sim q(\cdot|x)}(\log p(x|z)) \\ &= \mathbb{E}_{z\sim q(\cdot|x)}\left[\log\left(\frac{q(z|x)}{p(z)}\right)\right] - \mathbb{E}_{z\sim q(\cdot|x)}(\log p(x|z)) && \text{(definición)} \\ &= \mathbb{E}_{z\sim q(\cdot|x)}\left[\log\left(\frac{q(z|x)p(x|z)}{p(x,z)}\right)\right] - \mathbb{E}_{z\sim q(\cdot|x)}(\log p(x|z)) && \text{(por ecuación (1))} \\ &= \mathbb{E}_{z\sim q(\cdot|x)}\left[\log\left(\frac{q(z|x)}{p(x,z)}\right) + \log(p(x|z))\right] - \mathbb{E}_{z\sim q(\cdot|x)}(\log p(x|z)) && \text{(prop. logaritmos)} \\ &= \mathbb{E}_{z\sim q(\cdot|x)}\left[\log\left(\frac{q(z|x)}{p(x,z)}\right)\right] + \mathbb{E}_{z\sim q(\cdot|x)}(\log(p(x|z))) - \mathbb{E}_{z\sim q(\cdot|x)}(\log p(x|z)) && \text{(prop. esperanza)} \\ &= \mathbb{E}_{z\sim q(\cdot|x)}\left[\log\left(\frac{q(z|x)}{p(x,z)}\right)\right]\end{aligned}$$

(c) Partiendo desde el hint obtenemos:

$$\begin{aligned}0 \leq D_{KL}(q(z|x)||p(z|x)) &= \log p(x) + \mathbb{E}_{z\sim q(\cdot|x)}\left[\log\left(\frac{q(z|x)}{p(x,z)}\right)\right] \\ &= \log p(x) + D_{KL}(q(z|x)||p(z)) - \mathbb{E}_{z\sim q(\cdot|x)}(\log p(x|z))\end{aligned}$$

Por lo tanto, tenemos que

$$\begin{aligned}\log p(x) &\geq \mathbb{E}_{z\sim q(\cdot|x)}(\log p(x|z)) - D_{KL}(q(z|x)||p(z)) \\ &= \mathbb{E}_{z\sim q(\cdot|x)}(\log p(x|z)) - \mathbb{E}_{z\sim q(\cdot|x)}\left[\log\left(\frac{q(z|x)}{p(z)}\right)\right] \\ &= \mathbb{E}_{z\sim q(\cdot|x)}\left[\log\left(\frac{p(x|z)p(z)}{q(z|x)}\right)\right] \\ &= \mathbb{E}_{z\sim q(\cdot|x)}\left[\log\left(\frac{p(x,z)}{q(z|x)}\right)\right]\end{aligned}$$

Donde la última igualdad es por Bayes. Así, encontramos una cota inferior a la log-verosimilitud (ELBO).

P2 **(a)** Como se vió en la parte anterior, tenemos una cota inferior de la log-verosimilitud, sin embargo, esta la podemos reescribir como sigue

$$\begin{aligned}\log p(x) &\geq \mathbb{E}_{z\sim q(\cdot|x)}\left[\log\left(\frac{p(x,z)}{q(z|x)}\right)\right] \\ &= \mathbb{E}_{z\sim q(\cdot|x)}\left[\log\left(\frac{p(x|z)p(z)}{q(z|x)}\right)\right] \\ &= \mathbb{E}_{z\sim q(\cdot|x)}[\log p(x|z)] - D_{KL}(q(z|x)||p(z))\end{aligned}$$

Luego, como queremos los parámetros que maximicen esta cota (para acercarnos al objetivo), tendremos el siguiente problema de optimización

$$\arg\max_{\theta,\phi} \mathbb{E}_{z\sim q(\cdot|x)}[\log p(x|z)] - D_{KL}(q(z|x)||p(z)) \approx \arg\max_{\theta,\phi} \frac{1}{L}\sum_{l=1}^{L} \log p(x|z_l) - D_{KL}(q(z|x)||p(z))$$

Donde esta aproximación viene dada por Monte Carlo con z_l es la variable latente sampleada de $q_\phi(z|x)$ para cada observación x. Luego, con el truco de la reparametrización, podemos escribir

$$z = \mu + \sigma \odot \epsilon_l \quad \text{con } \epsilon_l \sim \mathcal{N}(0,1)\ \forall l = 1, ..., L.$$

Así, podemos aproximar $\mathcal{L}_{\phi,\theta}(x)$ como

$$\mathcal{L}_{\phi,\theta}(x) \approx \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x|z_l) + \frac{1}{2}\sum_{j=1}^{J}\left(1 + \log\sigma_j^2 - \mu_j^2 - \sigma_j^2\right)$$

(b) Notar que el encoder recibe un input x para samplear un objeto en el espacio latente mientras que el decoder recibe un objeto z del espacio latente y reconstruye el input x' (idealmente esta reconstrucción debiese ser el mismo input x pero esto no sucede en la práctica). Con esto y la formulación de VAE, podemos ver que el encoder $q_\phi(z|x)$ modela la variable z dado el input x con la distribución $q_\phi(z|x) \sim \mathcal{N}(\mu, \sigma^2)$ mientras que el decoder $p_\theta(x|z)$ modela x dado el valor del espacio latente z, es decir, parámetros de una red neuronal.

(c) Para generar nuevos puntos de datos, debemos generar un elemento en el espacio latente con la distribución prior (en este caso una normal estándar). Luego, con el dato obtenido $z \sim \mathcal{N}(0, 1)$ hacemos que pase por el decoder y así generar un dato del espacio de los inputs.
La expresividad del modelo viene dada por la aleatoriedad de las distribuciones, el entrenamiento de las redes neuronales involucradas, que los datos puedan ser condicionados para obtener generaciones similares o que incluso puedan parecerse a un dato x_i con probabilidad p y a uno x_j con probabilidad $1 - p$.

(d) El problema de usar descenso de gradiente estocástico para optimizar ELBO antes del truco de reparametrización es que al derivar $\mathcal{L}_{\phi,\theta}$ con respecto a ϕ y estimarlo usando Monte Carlo entrega como resultado algo con mucha varianza. Por otro lado, sin este truco, el cálculo del gradiente no puede fluir entre nodos de la red neuronal (backpropagation). Esto pues, antes de la reparametrización, la variable que entrega la arista aleatoria es la variable latente z, sin embargo, después de la reparametrización, quien entrega aleatoriedad al modelo es la variable ϵ.