Advanced Machine Learning – Deep Generative Models Exercise Sheet 4

Generative Models: Denoising Diffusion

Problem 1: To train a diffusion model, we want to maximize the evidence lower bound of the data

$$\mathcal{L} = \mathop{\mathbb{E}}_{q_{\phi(\boldsymbol{x}_0)}} \left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_0, \boldsymbol{z}_{1:N}) - \log q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_{1:N})\right] \leq \log p(\boldsymbol{x}_0).$$

where $q_{\phi(\boldsymbol{x}_0)}$ and $p_{\boldsymbol{\theta}}$ are the forward and reverse distributions as defined in the lecture. Show that the ELBO is equal to

$$\mathcal{L} = -\mathbb{KL}[q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_N) \mid p(\boldsymbol{z}_N)] - \sum_{n>1} \mathbb{KL}[q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n) \mid p_{\boldsymbol{\theta}}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n)] + \mathop{\mathbb{E}}_{q_{\phi(\boldsymbol{x}_0)}}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_0 \mid \boldsymbol{z}_1)].$$

Hint: Make use of the Markov property / definition of $p_{\boldsymbol{\theta}}$ and $q_{\phi(\boldsymbol{x}_0)}$.

The KL-divergence is defined as

$$\mathbb{KL}[q \mid p] = \mathop{\mathbb{E}}_{q} \left[\log \frac{q}{p}\right],$$

so to get the ELBO into the desired form, we have to match parts of $p_{\boldsymbol{\theta}}$ and $q_{\phi(\boldsymbol{x}_0)}$.

$$\mathcal{L} = \mathop{\mathbb{E}}_{q_{\phi(\boldsymbol{x}_0)}} \left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_0, \boldsymbol{z}_{1:N}) - \log q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_{1:N})\right]$$

Plug in the definitions of $p_{\boldsymbol{\theta}}$ and $q_{\phi(\boldsymbol{x}_0)}$ as Markov chains

$$= \mathop{\mathbb{E}}_{q_{\phi(\boldsymbol{x}_0)}} \left[\log p(\boldsymbol{z}_N) + \sum_{n>1} \log p_{\boldsymbol{\theta}}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n) + \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_0 \mid \boldsymbol{z}_1) - \sum_{n>1} \log q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_n \mid \boldsymbol{z}_{n-1}) - \log q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_1)\right]$$

Match the parts as we want them to go together into the KL-divergences

$$= \mathop{\mathbb{E}}_{q_{\phi(\boldsymbol{x}_0)}} \left[\log p(\boldsymbol{z}_N) + \sum_{n>1} \log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n)}{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_n \mid \boldsymbol{z}_{n-1})} + \log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}_0 \mid \boldsymbol{z}_1)}{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_1)}\right]$$

Apply Bayes rule to make the order match between $p_{\boldsymbol{\theta}}$ and $q_{\phi(\boldsymbol{x}_0)}$ in the middle term

$$= \mathop{\mathbb{E}}_{q_{\phi(\boldsymbol{x}_0)}} \left[\log p(\boldsymbol{z}_N) + \sum_{n>1} \log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n)}{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n)} \frac{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_{n-1})}{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_n)} + \log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}_0 \mid \boldsymbol{z}_1)}{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_1)}\right]$$

Distribute the log

$$= \mathop{\mathbb{E}}_{q_{\phi(\boldsymbol{x}_0)}} \left[\log p(\boldsymbol{z}_N) + \sum_{n>1} \log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n)}{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n)} + \sum_{n>1} \log \frac{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_{n-1})}{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_n)} + \log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}_0 \mid \boldsymbol{z}_1)}{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_1)}\right]$$

The new sum is a telescope sum, so every term except the first and last vanish

$$= \mathop{\mathbb{E}}_{q_{\phi(\boldsymbol{x}_0)}} \left[\log p(\boldsymbol{z}_N) + \sum_{n>1} \log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n)}{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n)} + \log q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_1) - \log q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_N) + \log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}_0 \mid \boldsymbol{z}_1)}{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_1)}\right]$$

Now match terms again and the $q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_1)$ cancel

$$= \mathop{\mathbb{E}}_{q_{\phi(\boldsymbol{x}_0)}}\left[\log \frac{p(\boldsymbol{z}_N)}{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_N)} + \sum_{n>1} \log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n)}{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n)} + \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_0 \mid \boldsymbol{z}_1)\right]$$

Next, we distribute the expected value

$$= \mathop{\mathbb{E}}_{q_{\phi(\boldsymbol{x}_0)}}\left[\log \frac{p(\boldsymbol{z}_N)}{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_N)}\right] + \sum_{n>1} \mathop{\mathbb{E}}_{q_{\phi(\boldsymbol{x}_0)}}\left[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n)}{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n)}\right] + \mathop{\mathbb{E}}_{q_{\phi(\boldsymbol{x}_0)}}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_0 \mid \boldsymbol{z}_1)\right]$$

and pull a minus out of the logarithms by flipping the fractions

$$= -\mathop{\mathbb{E}}_{q_{\phi(\boldsymbol{x}_0)}}\left[\log \frac{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_N)}{p(\boldsymbol{z}_N)}\right] - \sum_{n>1} \mathop{\mathbb{E}}_{q_{\phi(\boldsymbol{x}_0)}}\left[\log \frac{q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n)}{p_{\boldsymbol{\theta}}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n)}\right] + \mathop{\mathbb{E}}_{q_{\phi(\boldsymbol{x}_0)}}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_0 \mid \boldsymbol{z}_1)\right]$$

which gives us the desired result by pattern matching against the definition of the KL-divergence

$$= -\mathbb{KL}[q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_N) \mid p(\boldsymbol{z}_N)] - \sum_{n>1} \mathbb{KL}[q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n) \mid p_{\boldsymbol{\theta}}(\boldsymbol{z}_{n-1} \mid \boldsymbol{z}_n)] + \mathop{\mathbb{E}}_{q_{\phi(\boldsymbol{x}_0)}}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_0 \mid \boldsymbol{z}_1)\right]$$

Problem 2: Given the variational distribution (diffusion process) from the lecture

$$\begin{aligned} q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_1) &= \mathcal{N}(\sqrt{1-\beta_1}\boldsymbol{x}_0, \beta_1 \boldsymbol{I}), \quad 0 < \beta_1 < 1, \\ q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_n \mid \boldsymbol{z}_{n-1}) &= \mathcal{N}(\sqrt{1-\beta_n}\boldsymbol{z}_{n-1}, \beta_n \boldsymbol{I}), \quad 0 < \beta_n < 1, \end{aligned}$$

show that $q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_n)$ has the closed form

$$q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_n) = \mathcal{N}(\sqrt{\bar{\alpha}_n}\boldsymbol{x}_0, (1-\bar{\alpha}_n)\boldsymbol{I}), \quad \text{where } \alpha_n = 1 - \beta_n \text{ and } \bar{\alpha}_n = \prod_i^n \alpha_n.$$

Hint: Construct a sample of $q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_n)$ from a sample $\boldsymbol{z}_{n-1} \sim q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_{n-1})$.

We show it by induction. The base case for $q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_1)$ is already given by our definition of the variational distribution.

Now we assume it to be true for $q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_{n-1})$ and examine $q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_n)$. Let $\boldsymbol{z}_{n-1}$ be a sample from $q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_{n-1})$. Then with the reparameterization trick, we can write

$$\boldsymbol{z}_{n-1} = \sqrt{\bar{\alpha}_{n-1}}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_{n-1}}\boldsymbol{\varepsilon}_{n-1} \quad \text{and} \quad \boldsymbol{z}_n = \sqrt{\alpha_n}\boldsymbol{z}_{n-1} + \sqrt{\beta_n}\boldsymbol{\varepsilon}_n$$

where $\boldsymbol{\varepsilon}_{n-1}, \boldsymbol{\varepsilon}_n \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. Now we plug $\boldsymbol{z}_{n-1}$ into $\boldsymbol{z}_n$ to receive

$$\begin{aligned} \boldsymbol{z}_n &= \sqrt{\alpha_n}\left(\sqrt{\bar{\alpha}_{n-1}}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_{n-1}}\boldsymbol{\varepsilon}_{n-1}\right) + \sqrt{\beta_n}\boldsymbol{\varepsilon}_n \\ &= \sqrt{\bar{\alpha}_n}\boldsymbol{x}_0 + \sqrt{\alpha_n(1-\bar{\alpha}_{n-1})}\boldsymbol{\varepsilon}_{n-1} + \sqrt{\beta_n}\boldsymbol{\varepsilon}_n \end{aligned}$$

Because $\sqrt{\alpha_n(1-\bar{\alpha}_{n-1})}\boldsymbol{\varepsilon}_{n-1} \sim \mathcal{N}(\mathbf{0}, \alpha_n(1-\bar{\alpha}_{n-1})\boldsymbol{I})$ and $\sqrt{\beta_n}\boldsymbol{\varepsilon}_n \sim \mathcal{N}(\mathbf{0}, \beta_n\boldsymbol{I})$, their sum is

$$\sqrt{\alpha_n(1-\bar{\alpha}_{n-1})}\boldsymbol{\varepsilon}_{n-1} + \sqrt{\beta_n}\boldsymbol{\varepsilon}_n \sim \mathcal{N}(\mathbf{0}, (\alpha_n(1-\bar{\alpha}_{n-1}) + \beta_n)\boldsymbol{I})$$

and we can write

$$\boldsymbol{z}_n = \sqrt{\bar{\alpha}_n}\boldsymbol{x}_0 + \sqrt{\alpha_n(1-\bar{\alpha}_{n-1}) + \beta_n}\boldsymbol{\varepsilon}'_n$$

with $\boldsymbol{\varepsilon}'_n \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$. Rewriting

$$\alpha_n(1-\bar{\alpha}_{n-1}) + \beta_n = \alpha_n - \bar{\alpha}_n + 1 - \alpha_n = 1 - \bar{\alpha}_n$$

gets us

$$\boldsymbol{z}_n = \sqrt{\bar{\alpha}_n}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_n}\boldsymbol{\varepsilon}'_n$$

which shows by reverse application of the reparameterization trick that

$$\boldsymbol{z}_n \sim \mathcal{N}(\sqrt{\bar{\alpha}_n}\boldsymbol{x}_0, (1-\bar{\alpha}_n)\boldsymbol{I}) = q_{\phi(\boldsymbol{x}_0)}(\boldsymbol{z}_n).$$

Problem 3: Let $\boldsymbol{x}_n = \boldsymbol{x}_0 + \sigma_n\boldsymbol{\varepsilon}$ be the noising function with $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ and $\sigma_n > 0$ that we use in score matching to learn the data distribution. $p(\boldsymbol{x}_n \mid x_0)$ denotes the distribution of the noisy version $\boldsymbol{x}_n$ of $\boldsymbol{x}_0$.

a) Derive the conditional score $\nabla_{\boldsymbol{x}_n} \log p(\boldsymbol{x}_n \mid \boldsymbol{x}_0)$.

b) In the lecture, the model predicts the score $s_{\boldsymbol{\theta}}(\boldsymbol{x}_n, n) \approx \nabla_{\boldsymbol{x}_n} \log p(\boldsymbol{x}_n \mid \boldsymbol{x}_0)$. Now we change (reparameterize) the model so that, instead of predicting the score, it predicts the noise $\boldsymbol{\varepsilon} \approx \boldsymbol{\varepsilon}_{\boldsymbol{\theta}}(\boldsymbol{x}_n, n)$ added to $\boldsymbol{x}_0$. Derive how the score function $\nabla_{\boldsymbol{x}_n} \log p(\boldsymbol{x}_n \mid \boldsymbol{x}_0)$ can be expressed as a function $s(\boldsymbol{\varepsilon}_\theta(\boldsymbol{x}_n, n))$ of the noise estimate $\boldsymbol{\varepsilon}_\theta(\boldsymbol{x}_n, n)$.

a) We first have to write the conditional distribution

$$p(\boldsymbol{x}_n \mid \boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{x}_0, \sigma_n\boldsymbol{I}) = \frac{1}{\sigma_n\sqrt{2\pi}} \exp\left(-\frac{(\boldsymbol{x}_n - \boldsymbol{x}_0)^2}{2\sigma_n^2}\right).$$

The log-likelihood is given by

$$\log p(\boldsymbol{x}_n \mid \boldsymbol{x}_0) = \log \frac{1}{\sigma_n\sqrt{2\pi}} - \frac{(\boldsymbol{x}_n - \boldsymbol{x}_0)^2}{2\sigma_n^2},$$

which finally gives us the score function as

$$\nabla_{\boldsymbol{x}_n} \log p(\boldsymbol{x}_n \mid \boldsymbol{x}_0) = -\frac{2(\boldsymbol{x}_n - \boldsymbol{x}_0)}{2\sigma_n^2} = -\frac{(\boldsymbol{x}_n - \boldsymbol{x}_0)}{\sigma_n^2}.$$

b) We start by expressing the conditional score function in terms of the noise $\boldsymbol{\varepsilon}$ by plugging in the definition of the noisy data $\boldsymbol{x}_n$.

$$\nabla_{\boldsymbol{x}_n} \log p(\boldsymbol{x}_n \mid \boldsymbol{x}_0) = -\frac{(\boldsymbol{x}_n - \boldsymbol{x}_0)}{\sigma_n^2} = -\frac{(\boldsymbol{x}_0 + \sigma_n\boldsymbol{\varepsilon} - \boldsymbol{x}_0)}{\sigma_n^2} = -\frac{\boldsymbol{\varepsilon}}{\sigma_n}$$

Therefore, if the model predicts $\boldsymbol{\varepsilon}$ with a learnable function $\boldsymbol{\varepsilon}_{\boldsymbol{\theta}}(\boldsymbol{x}_n, n)$, we should predict the score as

$$s(\boldsymbol{\varepsilon}_{\theta}(\boldsymbol{x}_n, n)) = -\frac{\boldsymbol{\varepsilon}_{\boldsymbol{\theta}}(\boldsymbol{x}_n, n)}{\sigma_n}.$$