Lecture notes on variational auto-encoders (VAEs)

1.1 Prerequisites

Divergence minimisation

Given a set of probability distributions $\mathcal{P}$ on a random variable X, a divergence is defined as a function $\mathrm{D}[\cdot||\cdot] : \mathcal{P} \times \mathcal{P} \to \mathbb{R}$ such that $\mathrm{D}[P||Q] \geq 0$ for all $P, Q \in \mathcal{P}$, and $\mathrm{D}[P||Q] = 0$ iff. $P = Q$.

The definition of divergence is much weaker than that for a *distance* such as the ℓ_2-norm, since it does not need to satisfy either symmetry in arguments or the triangle inequality. There exist many available divergences to use, some of them will be introduced throughout this course.

In this course we assume the probability distributions/measures in $\mathcal{P}$ are dominated by the Lebesgue measure of the underlying Eucledean space, so that we can work with *probability density functions* (PDFs)

$$p(\boldsymbol{x}) = \frac{dP}{d\boldsymbol{x}}, \forall P \in \mathcal{P}. \tag{1}$$

In the following we will also write $\mathcal{P}$ as the set of PDFs w.l.o.g., and use the terms probability distribution and probability density functions interchangeably (unless specifically mentioned).

Jensen's inequality

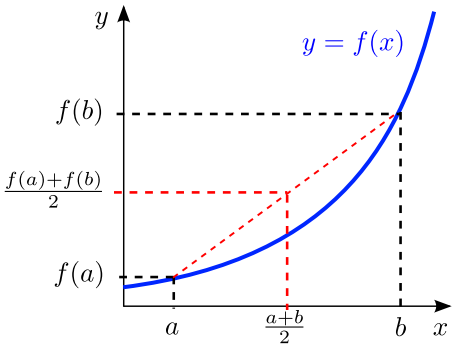
Below we introduce Jensen's inequality as a prerequisite for later discussions on divergences.

Proposition 1. *(Jensen's inequality) If $f : \mathbb{R} \to \mathbb{R}$ is a convex function, then for any distribution $p(x)$,*

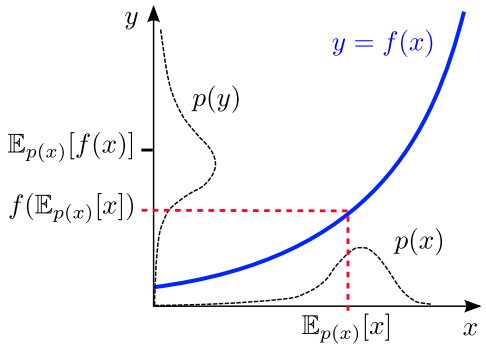
$$\mathbb{E}_{p(x)}[f(x)] \geq f(\mathbb{E}_{p(x)}[x]),$$

with equality holds iff. f is linear or $p(x)$ is a delta measure.

A visual proof is provided in the below figures.



(a) for discrete distributions



(b) for continuous distributions

Jensen's inequality can be generalised to functions formed by compositions of functions. To see this, we first introduce the *law of the unconscious statisticians* (LOTUS) rule:

Proposition 2. *(LOTUS) Given a distribution $p_X(\boldsymbol{x})$ and a function $\mathbf{y} = g(\boldsymbol{x})$ such that $\mathbb{E}_{p_X(\boldsymbol{x})}[g(\boldsymbol{x})] < +\infty$, the random variable $Y = g(X)$ has its distribution $p_Y(\mathbf{y})$ satisfying $\mathbb{E}_{p_Y(\mathbf{y})}[\mathbf{y}] = \mathbb{E}_{p_X(\boldsymbol{x})}[g(\boldsymbol{x})]$.*

Then a generalised version of Jensen's inequality reads as follows.

Proposition 3. *(Generalised Jensen's inequality) If a function $g(\boldsymbol{x})$ maps inputs to scalar outputs in $\mathbb{R}$ and $f : \mathbb{R} \to \mathbb{R}$ is a convex function, then for any distribution $p_X(\boldsymbol{x})$,*

$$\mathbb{E}_{p_X(\boldsymbol{x})}[f(g(\boldsymbol{x}))] \geq f(\mathbb{E}_{p_X(\boldsymbol{x})}[g(\boldsymbol{x})]),$$

with equality holds iff. f is linear or $p_X(\boldsymbol{x})$ is a delta measure.

Proof.

$$\begin{aligned}
\mathbb{E}_{p_X(\boldsymbol{x})}[f(g(\boldsymbol{x}))] &= \mathbb{E}_{p_Y(y)}[f(y)] && \text{(LOTUS applied to } y = g(\boldsymbol{x})) \\
&\geq f[\mathbb{E}_{p_Y(y)}[y]] && \text{(Jensen's inequality)} \\
&= f(\mathbb{E}_{p_X(\boldsymbol{x})}[g(\boldsymbol{x})]). && \text{(LOTUS applied to } y = g(\boldsymbol{x}))
\end{aligned}$$

□

Kullback-Leibler (KL) divergence

Kullback-Leibler divergence [Kullback and Leibler, 1951; Kullback, 1959], or *KL divergence*, is arguably one of the most widely used divergence measures in machine learning, statistics, and information theory.

Definition 1. *(Kullback-Leibler Divergence) The Kullback-Leibler (KL) divergence on $\mathcal{P}$ is defined as a function $\mathrm{KL}[\cdot||\cdot] : \mathcal{P} \times \mathcal{P} \to \mathbb{R}$ with the following form*

$$\mathrm{KL}[p||q] = \int p(\boldsymbol{x}) \log \frac{p(\boldsymbol{x})}{q(\boldsymbol{x})} d\boldsymbol{x}, \quad p, q \in \mathcal{P}, \tag{2}$$

where log *is the natural logarithm (to base e).*

One can easily check that indeed the above definition is a valid divergence: define $f(x) = -\log x$ (which is convex) and $g(\boldsymbol{x}) = q(\boldsymbol{x})/p(\boldsymbol{x})$, we have

$$\begin{aligned}
\mathrm{KL}[p||q] &= \mathbb{E}_{p(\boldsymbol{x})}[-\log g(\boldsymbol{x})] \\
&\geq -\log \mathbb{E}_{p(\boldsymbol{x})}[g(\boldsymbol{x})] && \text{(Jensen's inequality)} \\
&= -\log \int p(\boldsymbol{x}) \frac{q(\boldsymbol{x})}{p(\boldsymbol{x})} d\boldsymbol{x} = -\log 1 = 0,
\end{aligned}$$

and the equality holds iff. $p(\boldsymbol{x}) = q(\boldsymbol{x})$.[1] This means one can minimise the KL divergence in order to fit a distribution to a target one. Also notice that the KL divergence is asymmetric, i.e. $\mathrm{KL}[p||q] \neq \mathrm{KL}[q||p]$ in general.

Maximum likelihood estimation (MLE)

Given a dataset $\{(\boldsymbol{x}_n)\}_{n=1}^N \sim p_{\text{data}}(\boldsymbol{x})$, we would like to fit to it a generative model $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ with parameter $\boldsymbol{\theta}$. Since the KL divergence can be used to measure the closeness of the model to the underlying data distribution, it makes sense to find the optimal parameters by minimising the KL divergence:

$$\boldsymbol{\theta}^* = \arg\min \mathrm{KL}[p_{\text{data}}(\boldsymbol{x})||p_{\boldsymbol{\theta}}(\boldsymbol{x})]. \tag{3}$$

Expanding the above objective and re-arranging terms, we have

$$\mathrm{KL}[p_{\text{data}}(\boldsymbol{x})||p_{\boldsymbol{\theta}}(\boldsymbol{x})] = \underbrace{\mathbb{E}_{p_{\text{data}}(\boldsymbol{x})}[\log p_{\text{data}}(\boldsymbol{x})]}_{\text{constant w.r.t. } \boldsymbol{\theta}} - \underbrace{\mathbb{E}_{p_{\text{data}}(\boldsymbol{x})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x})]}_{\text{dependent on } \boldsymbol{\theta}}.$$

[1] Technically speaking: $p(\boldsymbol{x}) = q(\boldsymbol{x})$ almost everywhere.

This means we can ignore the constant terms w.r.t. $\boldsymbol{\theta}$ and instead work with the following *maximum likelihood* objective:

$$\boldsymbol{\theta}^* = \arg\max \mathbb{E}_{p_{\text{data}}(\boldsymbol{x})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x})]. \tag{4}$$

The obtained optimal parameters $\boldsymbol{\theta}^*$ is called the *maximum likelihood estimate* (MLE) of the parameters.In practice the data distribution is approximated by the empirical distribution on the dataset $\{(\boldsymbol{x}_n)\}_{n=1}^N \sim p_{\text{data}}(\boldsymbol{x})$, leading to

$$\boldsymbol{\theta}^* = \arg\max \frac{1}{N}\sum_{n=1}^{N} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_n). \tag{5}$$

1.2 Variational inference

We are interested in fitting the following latent variable model (LVM) to the data:

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}) = \int p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})d\boldsymbol{z}. \tag{6}$$

In deep generative modelling context, this LVM is often constructed as (for continuous data)

$$p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}; \mathbf{0}, \mathbf{I}), \quad p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{x}; G_{\boldsymbol{\theta}}(\boldsymbol{z}), \sigma^2\mathbf{I}), \tag{7}$$

with $G_{\boldsymbol{\theta}}(\cdot)$ define as a neural network transform that is parameterised by weights $\boldsymbol{\theta}$. For discrete variables $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$ is usually defined as a categorical distribution with a neural network generator in use accordingly. Now to fit $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ to $p_{\text{data}}(\boldsymbol{x})$ we optimise the MLE objective (4) w.r.t. $\boldsymbol{\theta}$, which involves computing the integral (6). This is intractable as it involves computing the non-linear transformation $G_{\boldsymbol{\theta}}(\boldsymbol{z})$ for every single configuration of $\boldsymbol{z}$ within the support of the Gaussian prior $p(\boldsymbol{z})$, which is the full space $\boldsymbol{z} \in \mathbb{R}^d$.

Variational inference provides a variational lower-bound of $\log p_{\boldsymbol{\theta}}(\boldsymbol{x})$ as an approximation to it. For any distribution $q(\boldsymbol{z})$ satisfying $q(\boldsymbol{z}) > 0$ whenever $p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}) > 0$, we have

$$\begin{aligned}
\log p_{\boldsymbol{\theta}}(\boldsymbol{x}) &= \int p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})d\boldsymbol{z} \\
&= \log \int q(\boldsymbol{z})\frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})}{q(\boldsymbol{z})}d\boldsymbol{z} \\
&\geq \int q(\boldsymbol{z}) \log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})}{q(\boldsymbol{z})}d\boldsymbol{z} \qquad \text{(Jensen's inequality)} \\
&= \mathbb{E}_{q(\boldsymbol{z})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})] - \text{KL}[q(\boldsymbol{z})||p(\boldsymbol{z})] := \mathcal{L}(\boldsymbol{x}, q, \boldsymbol{\theta}).
\end{aligned} \tag{8}$$

With suitable choice of $q(\boldsymbol{z})$ and tricks that will be introduced later, this varitational lower-bound can be used as a tractable approximation to the marginal log-likelihood $\log p_{\boldsymbol{\theta}}(\boldsymbol{x})$.

The choice of the $q(\boldsymbol{z})$ distribution is crucial to the quality of the approximation (or the tightness of the lower-bound). To see this, note that

$$p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}) = \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})}{p_{\boldsymbol{\theta}}(\boldsymbol{x})}, \qquad \text{(Bayes' rule)} \tag{9}$$

$$\begin{aligned}
\log p_{\boldsymbol{\theta}}(\boldsymbol{x}) - \text{KL}[q(\boldsymbol{z})||p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})] &= \log p_{\boldsymbol{\theta}}(\boldsymbol{x}) - \mathbb{E}_{q(\boldsymbol{z})}\left[\log \frac{q(\boldsymbol{z})}{p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})}\right] \\
&= \log p_{\boldsymbol{\theta}}(\boldsymbol{x}) + \mathbb{E}_{q(\boldsymbol{z})}\left[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})}{q(\boldsymbol{z})p_{\boldsymbol{\theta}}(\boldsymbol{x})}\right] \qquad \text{(Bayes' rule)} \\
&= \mathbb{E}_{q(\boldsymbol{z})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})] - \text{KL}[q(\boldsymbol{z})||p(\boldsymbol{z})] = \mathcal{L}(\boldsymbol{x}, q, \boldsymbol{\theta}).
\end{aligned} \tag{10}$$

This means the gap (or the approximation error) between the variational lower-bound $\mathcal{L}(\boldsymbol{x}, q, \boldsymbol{\theta})$ and the marginal log-likelihood $\log p_{\boldsymbol{\theta}}(\boldsymbol{x})$ is the KL divergence $\mathrm{KL}[q(\boldsymbol{z})||p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})]$. Therefore the lower-bound improves as $q(\boldsymbol{z})$ approaches to the exact posterior $p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})$. It also motivates the optimisation of the variational lower-bound w.r.t. the q distribution to obtain an approxiate posterior: since $\log p_{\boldsymbol{\theta}}(\boldsymbol{x})$ is constant w.r.t. q, maximising $\mathcal{L}(\boldsymbol{x}, q, \boldsymbol{\theta})$ is equivalent to minimising $\mathrm{KL}[q(\boldsymbol{z})||p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})]$.

1.3 Variational auto-encoders

As discussed so far, we wish to fit the generative model (6) to the data by maximum likelihood (4), and variational inference provides a useful approximation $\mathcal{L}(\boldsymbol{x}, q, \boldsymbol{\theta}) \leq \log p_{\boldsymbol{\theta}}(x)$ for a given datum $\boldsymbol{x}$. Since this approximation is required for every datapoint in $\{\boldsymbol{x}_n\}_{n=1}^N$, having N separated q distributions $q_1(\boldsymbol{z}_1), ..., q_N(\boldsymbol{z}_n)$ to pair with $\boldsymbol{x}_1, ..., \boldsymbol{x}_N$ can be memory inefficient. However, notice that the exact posterior $p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})$ depends on the input $\boldsymbol{x}$, and the variational lower-bound is tight when $q_n(\boldsymbol{z}_n) \approx p_{\boldsymbol{\theta}}(\boldsymbol{z}_n|\boldsymbol{x}_n)$. This motivates the *variational auto-encoder* (VAE) approach [Kingma and Welling, 2014; Rezende et al., 2014] which defines the q distribution as $q(\boldsymbol{z}) := q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$, with the distribution often defined by a neural network:

$$q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{z}; \boldsymbol{\mu}_{\boldsymbol{\phi}}(\boldsymbol{x}), diag(\sigma^2_{\boldsymbol{\phi}}(\boldsymbol{x}))), \quad \boldsymbol{\mu}_{\boldsymbol{\phi}}(\boldsymbol{x}), \log \sigma_{\boldsymbol{\phi}}(\boldsymbol{x}) = \mathrm{NN}_{\boldsymbol{\phi}}(\boldsymbol{x}). \tag{11}$$

This allows us to define the VAE optimisation objective:

$$\boldsymbol{\phi}^*, \boldsymbol{\theta}^* = \arg\max \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta}), \quad \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta}) = \mathbb{E}_{p_{\mathrm{data}}(\boldsymbol{x})}[\underbrace{\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})] - \mathrm{KL}[q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})||p(\boldsymbol{z})]}_{:=\mathcal{L}(\boldsymbol{x}, \boldsymbol{\phi}, \boldsymbol{\theta})}]. \tag{12}$$

Given that both $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ and $p(\boldsymbol{z})$ are all factorised Gaussian distributions, the KL divergence term in (12) has an analytic form (assuming $\boldsymbol{z} \in \mathbb{R}^d$):

$$\mathrm{KL}[q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})||p(\boldsymbol{z})] = \frac{1}{2}\left(||\boldsymbol{\mu}_{\boldsymbol{\phi}}(\boldsymbol{x})||_2^2 + ||\boldsymbol{\sigma}_{\boldsymbol{\phi}}(\boldsymbol{x})||_2^2 - 2\langle \log \sigma_{\boldsymbol{\phi}}(\boldsymbol{x}), \mathbf{1}\rangle - d\right). \tag{13}$$

Monte Carlo estimation

The VAE objective $\mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta})$ in (12) is still intractable since the expectation computation $\mathbb{E}_{q_{\boldsymbol{\phi}}}[\cdot]$ requires evaluating neural network transformations for all possible $\boldsymbol{z}$. Monte Carlo (MC) estimation comes into rescue, as we can replace the expectation with MC approximations:

$$\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})] \approx \log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}), \quad \boldsymbol{z} \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}). \tag{14}$$

By doing so, the gradient of the objective w.r.t. $\boldsymbol{\theta}$ can be estimated as

$$\nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\phi}, \boldsymbol{\theta}) \approx \nabla_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}), \quad \boldsymbol{z} \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}). \tag{15}$$

It remains to compute the gradient of the objective w.r.t. $\boldsymbol{\phi}$

$$\nabla_{\boldsymbol{\phi}} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\phi}, \boldsymbol{\theta}) \approx \nabla_{\boldsymbol{\phi}} \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})] - \nabla_{\boldsymbol{\phi}} \mathrm{KL}[q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})||p(\boldsymbol{z})]. \tag{16}$$

While the gradient w.r.t. the KL term tractable (by differentiate eq. (13) w.r.t. $\boldsymbol{\phi}$), MC approximation is still required for the first term in (16).

Reparameterisation trick

The MC approximation to $\nabla_{\boldsymbol{\phi}} \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})]$ is further assisted by the reparameterisation trick [Kingma and Welling, 2014; Rezende et al., 2014]. Note that the sampling procedure of a Gaussian variable is the following:

$$\boldsymbol{z} \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) \quad \Leftrightarrow \quad \boldsymbol{z} = \boldsymbol{\mu}_{\boldsymbol{\phi}} + \boldsymbol{\sigma}_{\boldsymbol{\phi}} \odot \boldsymbol{\epsilon}, \ \boldsymbol{\epsilon} \sim \mathcal{N}(\epsilon; \mathbf{0}, \mathbf{I}), \tag{17}$$

with $\odot$ denoting element-wise product. Writing $\pi(\boldsymbol{\epsilon}) := \mathcal{N}(\boldsymbol{\epsilon}; \mathbf{0}, \mathbf{I})$ and $T_{\boldsymbol{\phi}}(\boldsymbol{x}, \boldsymbol{\epsilon}) := \boldsymbol{\mu}_{\boldsymbol{\phi}} + \boldsymbol{\sigma}_{\boldsymbol{\phi}} \odot \boldsymbol{\epsilon}$, we have, by LOTUS & MC estimation,

$$\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})] = \mathbb{E}_{\pi(\boldsymbol{\epsilon})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|T_{\boldsymbol{\phi}}(\boldsymbol{x}, \boldsymbol{\epsilon}))], \tag{18}$$

$$\nabla_{\boldsymbol{\phi}} \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})] = \mathbb{E}_{\pi(\boldsymbol{\epsilon})}[\nabla_{\boldsymbol{\phi}} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}|T_{\boldsymbol{\phi}}(\boldsymbol{x}, \boldsymbol{\epsilon}))] = \mathbb{E}_{\pi(\boldsymbol{\epsilon})}[\nabla_{\boldsymbol{\phi}} \boldsymbol{z} \nabla_{\boldsymbol{z}} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})|_{\boldsymbol{z}=T_{\boldsymbol{\phi}}(\boldsymbol{x},\boldsymbol{\epsilon})}]. \tag{19}$$

Then with MC estimation:

$$\mathbb{E}_{\pi(\boldsymbol{\epsilon})}[\nabla_{\boldsymbol{\phi}} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}|T_{\boldsymbol{\phi}}(\boldsymbol{x}, \boldsymbol{\epsilon}))] \approx \nabla_{\boldsymbol{\phi}} \boldsymbol{z} \nabla_{\boldsymbol{z}} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})|_{\boldsymbol{z}=T_{\boldsymbol{\phi}}(\boldsymbol{x},\boldsymbol{\epsilon})}, \ \boldsymbol{\epsilon} \sim \pi(\boldsymbol{\epsilon}) \tag{20}$$

Combined with eq. (15) and mini-batch training, one can compute an MC estimation of the VAE objective (12) as

$$\begin{aligned} \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta}) \approx \frac{1}{M} \sum_{m=1}^{M} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_m|T_{\boldsymbol{\phi}}(\boldsymbol{x}_m, \boldsymbol{\epsilon}_m)) - \mathrm{KL}[q_{\boldsymbol{\phi}}(\boldsymbol{z}_m|\boldsymbol{x}_m)||p(\boldsymbol{z}_m)], \\ \boldsymbol{x}_1, ..., \boldsymbol{x}_m \sim \{\boldsymbol{x}_n\}^M, \ \boldsymbol{\epsilon}_1, ..., \boldsymbol{\epsilon}_M \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \end{aligned} \tag{21}$$

and apply e.g. automatic differentiation to obtain the (MC estimation of) gradient of the VAE objective w.r.t. parameters $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$.

References

Kingma, D. P. and Welling, M. (2014). Auto-encoding variational Bayes. In *International Conference on Learning Representations*.

Kullback, S. (1959). *Information theory and statistics*. John Wiley & Sons.

Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86.

Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on Machine Learning*, pages 1278–1286.