This document covers mathematical and algorithmic techniques underlying the four most popular families of deep generative models. Thus, we explore autoregressive models (Section 1), reparameterization trick (Section 2), variational autoencoders (VAEs, Section 3-4), normalizing flows (Question 5), and generative adversarial networks (GANs, Section 6).

Section 1 (**Autoregressive**)**.** One way to enforce autoregressive conditioning is via masking the weight parameters.[1] We consider a two-hidden-layer convolutional neural network without kernel flipping, with kernel size 3×3 and padding size 1 on each border (so that an input feature map of size 5×5 is convolved into a 5×5 output). We define mask of type A and mask of type B as

$$(\boldsymbol{M}^A)_{::ij} := \begin{cases} 1 & \text{if } i = 2 \text{ and } j < 2 \\ 1 & \text{if } i = 3 \\ 0 & \text{elsewhere} \end{cases} \qquad (\boldsymbol{M}^B)_{::ij} := \begin{cases} 1 & \text{if } i = 2 \text{ and } j \leq 2 \\ 1 & \text{if } i = 3 \\ 0 & \text{elsewhere} \end{cases}$$

where the index starts from 1. Masking is achieved by multiplying the kernel with the binary mask (elementwise). We show the receptive field of the output pixel that corresponds to the third row and the fourth column (index 34 of Figure 1) in each of the following 4 cases:

11	12	13	14	15
21	22	23	24	25
31	32	33	34	35
41	42	43	44	45
51	52	53	54	55

FIGURE 1 – 5×5 convolutional feature map.

1. If we use $\boldsymbol{M}^A$ for the first layer and $\boldsymbol{M}^A$ for the second layer.
2. If we use $\boldsymbol{M}^A$ for the first layer and $\boldsymbol{M}^B$ for the second layer.
3. If we use $\boldsymbol{M}^B$ for the first layer and $\boldsymbol{M}^A$ for the second layer.
4. If we use $\boldsymbol{M}^B$ for the first layer and $\boldsymbol{M}^B$ for the second layer.

1. An example of this is the use of masking in the Transformer architecture. Here, it's inpired by PixelCNN.

Steps 1.

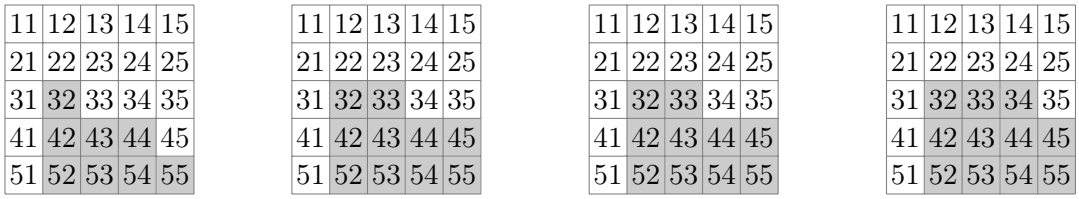

FIGURE 2 – From left to right, the case 1, 2, 3, and 4.

Steps for case 2. The other cases follow the exact same procedure except that the masks applied are a different combinations.

At the first layer, the mask $\boldsymbol{M}^A$ is applied on the important pixel for our case according to Figure 3.

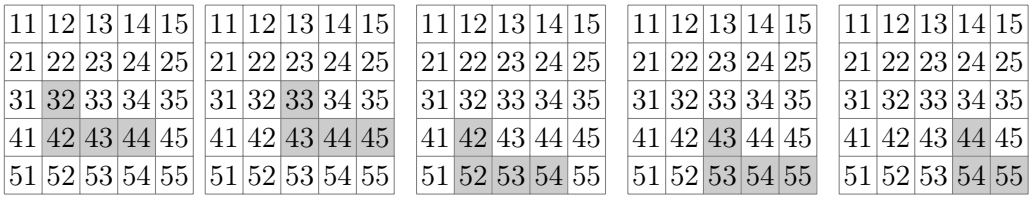

FIGURE 3 – From left to right, receptive field of the pixel 33, 34, 43, 44, and 45 at the first layer when using $\boldsymbol{M}^A$

At the next layer, when using mask $\boldsymbol{M}^B$, the receptive field of the output pixel 34 is depending on pixel 33, 34, 43, 44, and 45 of the previous layer. Therefore, by merging the receptive field of Figure 3, we obtain our final receptive field corresponding to the second one at Figure 2.

Section 2 (**Reparameterization trick**)**.** Reparameterization trick is a standard technique that makes the samples of a random variable differentiable. The trick represents the random variable as a simple mapping from another random variable drawn from some simple distribution [2]. If the reparameterization is a bijective function, the induced density of the resulting random variable can be computed using the change-of-variable density formula, whose computation requires evaluating the determinant of the Jacobian of the mapping.

We consider a random vector $Z \in \mathbb{R}^K$ with a density function $q(\boldsymbol{z}; \phi)$ and a random variable $Z_0 \in \mathbb{R}^K$ having a ϕ-independent density function $q(\boldsymbol{z}_0)$. We want to find a deterministic function $\boldsymbol{g} : \mathbb{R}^K \to \mathbb{R}^K$ that depends on ϕ, to transform Z_0, such that the induced distribution of the transformation has the same density as Z. The change of density for a bijective, differentiable $\boldsymbol{g}$ is:

$$q(\boldsymbol{g}(\boldsymbol{z}_0)) = q(\boldsymbol{z}_0) \left|\det \boldsymbol{J}_{\boldsymbol{z}_0} \boldsymbol{g}(\boldsymbol{z}_0)\right|^{-1} = q(\boldsymbol{z}_0) \left|\det \left(\frac{\partial \boldsymbol{g}(\boldsymbol{z}_0)}{\partial \boldsymbol{z}_0}\right)\right|^{-1} \tag{1}$$

2. More specifically, these mapping should be differentiable wrt the density function's parameters.

1. We assume $q(\boldsymbol{z}_0) = \mathcal{N}(\mathbf{0}, \boldsymbol{I}_K)$ and $\boldsymbol{g}(\boldsymbol{z}_0) = \mu + \sigma \odot \boldsymbol{z}_0$, where $\mu \in \mathbb{R}^K$ and $\sigma \in \mathbb{R}^K_{>0}$. Note that $\odot$ is element-wise product. We show that $\boldsymbol{g}(\boldsymbol{z}_0)$ is distributed by $\mathcal{N}(\mu, \text{diag}(\sigma^2))$ using Equation (1).
2. We compute the time complexity of evaluating $|\det \boldsymbol{J}_{\boldsymbol{z}_0} \boldsymbol{g}(\boldsymbol{z}_0)|$ when $\boldsymbol{g}(\boldsymbol{z}_0) = \mu + \sigma \odot \boldsymbol{z}_0$. Using the big $\mathcal{O}$ notation and expressing the time complexity as a function of K.
3. We assume $\boldsymbol{g}(\boldsymbol{z}_0) = \mu + \boldsymbol{S}\boldsymbol{z}_0$, where $\boldsymbol{S}$ is a non-singular $K \times K$ matrix. We derive the density of $\boldsymbol{g}(\boldsymbol{z}_0)$ using Equation (1).
4. The time complexity of the general Jacobian determinant is at least $\mathcal{O}(K^{2.373})$ [3]. We assume instead $\boldsymbol{g}(\boldsymbol{z}_0) = \mu + \boldsymbol{S}\boldsymbol{z}_0$ with $\boldsymbol{S}$ being a $K \times K$ lower triangular matrix; i.e. $\boldsymbol{S}_{ij} = 0$ for $j > i$, and $\boldsymbol{S}_{ii} > 0$. We compute the time complexity of evaluating $|\det \boldsymbol{J}_{\boldsymbol{z}_0} \boldsymbol{g}(\boldsymbol{z}_0)|$.

Steps 2.

1. We know that

   $X \sim \mathcal{N}(\mu, diag(\sigma^2))$ where $X \in \mathbb{R}^K$ has the density function $p(x) = \prod_{i=1}^{K} \frac{1}{\sqrt{2\pi\sigma_i^2}} exp(\frac{-(x_i - \mu_i)^2}{2\sigma_i^2})$

   and we also know that, if $\boldsymbol{A}$ is a diagonal matrix $n \times n$, then $det(\boldsymbol{A}) = \prod_{i=1}^{n} a_{ii}$
   We have

   $$\boldsymbol{z}' = g(\boldsymbol{z}_0) = \mu + \sigma \odot \boldsymbol{z}_0 \implies \boldsymbol{z}_0 = \frac{\boldsymbol{z}' - \mu}{\sigma} \tag{2}$$

   With (2),

   $$\mathcal{N}(\mathbf{0}, \boldsymbol{I}_K) = q(\boldsymbol{z}_0) = q(\frac{\boldsymbol{z}' - \mu}{\sigma}) \implies q(\boldsymbol{z}_0) = q(\frac{\boldsymbol{z}' - \mu}{\sigma}) = \prod_{i=1}^{K} \frac{1}{\sqrt{2\pi}} exp(\frac{-(\boldsymbol{z}_i' - \mu_i)^2}{2\sigma_i^2}) \tag{3}$$

   Additionally,

   $$\begin{aligned}
   \left|\det\left(\frac{\partial \boldsymbol{g}(\boldsymbol{z}_0)}{\partial \boldsymbol{z}_0}\right)\right|^{-1} &= \left|\det\left(\begin{bmatrix} \frac{\partial(\mu_1 + \sigma_1 \boldsymbol{z}_{0_1})}{\partial \boldsymbol{z}_{0_1}} & \cdots & \frac{\partial(\mu_1 + \sigma_1 \boldsymbol{z}_{0_1})}{\partial \boldsymbol{z}_{0_K}} \\ \vdots & \ddots & \vdots \\ \frac{\partial(\mu_K + \sigma_K \boldsymbol{z}_{0_K})}{\partial \boldsymbol{z}_{0_1}} & \cdots & \frac{\partial(\mu_K + \sigma_K \boldsymbol{z}_{0_K})}{\partial \boldsymbol{z}_{0_K}} \end{bmatrix}\right)\right|^{-1} \\
   &= \left|\det\left(\begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & \sigma_K \end{bmatrix}\right)\right|^{-1} = |\det(diag(\sigma))|^{-1} \\
   &= \prod_{i=1}^{K} \sigma_i^{-1}
   \end{aligned} \tag{4}$$

   Therefore, using (1), (3), and (4),

   $$q(\boldsymbol{g}(\boldsymbol{z}_0)) = q(\boldsymbol{z}_0)\left|\det\left(\frac{\partial \boldsymbol{g}(\boldsymbol{z}_0)}{\partial \boldsymbol{z}_0}\right)\right|^{-1} = \prod_{i=1}^{K} \frac{1}{\sqrt{2\pi\sigma_i^2}} exp(\frac{-(\boldsymbol{z}_i' - \mu_i)^2}{2\sigma_i^2}) = \mathcal{N}(\mu, diag(\sigma^2))$$

3. https://en.wikipedia.org/wiki/Computational_complexity_of_mathematical_operations

2. Because $\boldsymbol{J}_{\boldsymbol{z}_0}\boldsymbol{g}(\boldsymbol{z}_0)$ in $|\det \boldsymbol{J}_{\boldsymbol{z}_0}\boldsymbol{g}(\boldsymbol{z}_0)|$ is a diagonal matrix, there is only K σ_i's to evaluate and, then, there is K elements to multiply, i.e. $K-1$ multiplication. Therefore, we have $\mathcal{O}(eK+m(K-1))$ where e corresponds to the time complexity of evaluating σ_i and m corresponds to the time complexity of doing a multiplication. Because e and m are not in function of K, the time complexity as a function of K simplifies to $\mathcal{O}(K)$.
3. We know that

$$X \sim \mathcal{N}(\mu, \Sigma) \text{ where } X \in \mathbb{R}^K \text{ has the density function } p(x) = \frac{1}{(2\pi)^{\frac{K}{2}}\sqrt{det(\Sigma)}} e^{\frac{-1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu)}$$

We also know that, if $\boldsymbol{A}$ and $\boldsymbol{B}$ are non-singular $K \times K$ matrix, then

— $(\boldsymbol{A}^{-1})^\top = (\boldsymbol{A}^\top)^{-1}$
— $(\boldsymbol{AB})^{-1} = \boldsymbol{B}^{-1}\boldsymbol{A}^{-1}$
— $(\boldsymbol{AB})^\top = \boldsymbol{B}^\top\boldsymbol{A}^\top$
— $det(\boldsymbol{A}) = det(\boldsymbol{A}^\top)$
— $det(\boldsymbol{AB}) = det(\boldsymbol{A})det(\boldsymbol{B})$

First, we have

$$\boldsymbol{z}' = g(\boldsymbol{z}_0) = \mu + \boldsymbol{S}\boldsymbol{z}_0 \implies \boldsymbol{z}_0 = \boldsymbol{S}^{-1}(\boldsymbol{z}' - \mu) \tag{5}$$

With (5),

$$\begin{aligned} q(\boldsymbol{z}_0) &= q(\boldsymbol{S}^{-1}(\boldsymbol{z}' - \mu)) \\ &= \frac{1}{(2\pi)^{\frac{K}{2}}} e^{\frac{-1}{2}(\boldsymbol{S}^{-1}(\boldsymbol{z}'-\mu))^\top \boldsymbol{S}^{-1}(\boldsymbol{z}'-\mu)} \\ &= \frac{1}{(2\pi)^{\frac{K}{2}}} e^{\frac{-1}{2}(\boldsymbol{z}'-\mu)^\top (\boldsymbol{S}^\top)^{-1}\boldsymbol{S}^{-1}(\boldsymbol{z}'-\mu)} \\ &= \frac{1}{(2\pi)^{\frac{K}{2}}} e^{\frac{-1}{2}(\boldsymbol{z}'-\mu)^\top (\boldsymbol{S}\boldsymbol{S}^\top)^{-1}(\boldsymbol{z}'-\mu)} \end{aligned} \tag{6}$$

Additionally, we have

$$\begin{aligned} \frac{\partial \boldsymbol{g}(\boldsymbol{z}_0)}{\partial \boldsymbol{z}_0} &= \begin{bmatrix} \frac{\partial(\mu_1 + \sum_{i=1}^K \boldsymbol{S}_{1i}\boldsymbol{z}_{0_i})}{\partial \boldsymbol{z}_{0_1}} & \cdots & \frac{\partial(\mu_1 + \sum_{i=1}^K \boldsymbol{S}_{1i}\boldsymbol{z}_{0_i})}{\partial \boldsymbol{z}_{0_K}} \\ \vdots & \ddots & \vdots \\ \frac{\partial(\mu_K + \sum_{i=1}^K \boldsymbol{S}_{Ki}\boldsymbol{z}_{0_i})}{\partial \boldsymbol{z}_{0_1}} & \cdots & \frac{\partial(\mu_K + \sum_{i=1}^K \boldsymbol{S}_{1K}\boldsymbol{z}_{0_K})}{\partial \boldsymbol{z}_{0_K}} \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{S}_{11} & \cdots & \boldsymbol{S}_{1K} \\ \vdots & \ddots & \vdots \\ \boldsymbol{S}_{K1} & \cdots & \boldsymbol{S}_{KK} \end{bmatrix} \\ &= \boldsymbol{S} \end{aligned} \tag{7}$$

Also,

$$det(\boldsymbol{S})^{-2/2} = \frac{1}{det(\boldsymbol{S})^{2/2}} = \frac{1}{det(\boldsymbol{S})^{1/2}det(\boldsymbol{S})^{1/2}} = \frac{1}{det(\boldsymbol{S})^{1/2}det(\boldsymbol{S}^\top)^{1/2}} = \frac{1}{det(\boldsymbol{S}\boldsymbol{S}^\top)^{1/2}} \tag{8}$$

Therefore, with (6), (7), and (8),

$$q(\boldsymbol{g}(\boldsymbol{z}_0)) = q(\boldsymbol{z}_0)\left|\det\left(\frac{\partial \boldsymbol{g}(\boldsymbol{z}_0)}{\partial \boldsymbol{z}_0}\right)\right|^{-1} = \frac{1}{(2\pi)^{\frac{K}{2}}\sqrt{det(\boldsymbol{S}\boldsymbol{S}^\top)}} e^{\frac{-1}{2}(\boldsymbol{z}'-\mu)^\top(\boldsymbol{S}\boldsymbol{S}^\top)^{-1}(\boldsymbol{z}'-\mu)} = \mathcal{N}(\mu, \boldsymbol{S}\boldsymbol{S}^\top)$$

4. Because $\boldsymbol{J}_{\boldsymbol{z}_0}\boldsymbol{g}(\boldsymbol{z}_0) = \frac{\partial(\mu+\boldsymbol{S}\boldsymbol{z}_0)}{\partial \boldsymbol{z}_0} = \boldsymbol{S}$, to compute $|\det \boldsymbol{J}_{\boldsymbol{z}_0}\boldsymbol{g}(\boldsymbol{z}_0)|$, it is only necessary to compute $|\det \boldsymbol{S}|$. The determinant of a triangular matrix is equal to the product of the diagonal elements. Therefore, it is exactly like the previous question and it simplifies to $\mathcal{O}(K)$.

Section 3 (VAE). We consider a latent variable model $p_\theta(\boldsymbol{x}) = \int p_\theta(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})dz$, where $p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_K)$ and $\boldsymbol{z} \in \mathbb{R}^K$.The encoder network (aka "recognition model") of variational autoencoder, $q_\phi(\boldsymbol{z}|\boldsymbol{x})$, is used to produce an approximate (variational) posterior distribution over latent variables $\boldsymbol{z}$ for any input datapoint $\boldsymbol{x}$. [4] This distribution is trained to match the true posterior by maximizing the evidence lower bound (ELBO):

$$\mathcal{L}(\theta, \phi; \boldsymbol{x}) = \mathbb{E}_{q_\phi}[\log p_\theta(\boldsymbol{x} \mid \boldsymbol{z})] - D_{\mathrm{KL}}(q_\phi(\boldsymbol{z} \mid \boldsymbol{x})||p(\boldsymbol{z}))$$

Let $\mathcal{Q}$ be the family of variational distributions with a feasible set of parameters $\mathcal{P}$; i.e. $\mathcal{Q} = \{q(\boldsymbol{z}; \pi) : \pi \in \mathcal{P}\}$; for example π can be mean and standard deviation of a normal distribution. We assume q_ϕ is parameterized by a neural network (with parameters ϕ) that outputs the parameters, $\pi_\phi(\boldsymbol{x})$, of the distribution $q \in \mathcal{Q}$, i.e. $q_\phi(\boldsymbol{z}|\boldsymbol{x}) := q(\boldsymbol{z}; \pi_\phi(\boldsymbol{x}))$.

1. We show that maximizing the expected complete data log likelihood (ECLL)

   $$\mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})]$$

   for a fixed $q(\boldsymbol{z}|\boldsymbol{x})$, wrt the model parameter θ, is equivalent to maximizing

   $$\log p_\theta(\boldsymbol{x}) - D_{\mathrm{KL}}(q(\boldsymbol{z}|\boldsymbol{x})||p_\theta(\boldsymbol{z}|\boldsymbol{x}))$$

   This means the maximizer of the ECLL coincides with that of the marginal likelihood only if $q(\boldsymbol{z}|\boldsymbol{x})$ perfectly matches $p(\boldsymbol{z}|\boldsymbol{x})$.

2. We consider a finite training set $\{\boldsymbol{x}_i : i \in \{1, ..., n\}\}$, n being the size the training data. Let ϕ^* be the maximizer $\arg\max_\phi \sum_{i=1}^n \mathcal{L}(\theta, \phi; \boldsymbol{x}_i)$ with θ fixed. In addition, for each $\boldsymbol{x}_i$ let $q_i \in \mathcal{Q}$ be an "instance-dependent" variational distribution, and denote by q_i^* the maximizer of the corresponding ELBO. We compare $D_{\mathrm{KL}}(q_{\phi^*}(\boldsymbol{z}|\boldsymbol{x}_i)||p_\theta(\boldsymbol{z}|\boldsymbol{x}_i))$ and $D_{\mathrm{KL}}(q_i^*(\boldsymbol{z})||p_\theta(\boldsymbol{z}|\boldsymbol{x}_i))$, and show which one is bigger.

3. Following the previous subsection, we compare the two approaches at Section 3.2
   (a) in terms of bias of estimating the marginal likelihood via the ELBO, in the best case scenario (i.e. when both approaches are optimal within the respective families)
   (b) from the computational point of view (efficiency)
   (c) in terms of memory (storage of parameters)

4. Using a recognition model in this way is known as "amortized inference"; this can be contrasted with traditional variational inference approaches (see, e.g., Chapter 10 of Bishop's *Pattern Recognition an Machine Learning*), which fit a variational posterior independently for each new datapoint.

Steps 3.

1.

$$\mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})] = \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}[\log \frac{p_\theta(\boldsymbol{z}|\boldsymbol{x})p_\theta(\boldsymbol{x})}{p(\boldsymbol{z})}p(\boldsymbol{z})] = \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{z}|\boldsymbol{x})p_\theta(\boldsymbol{x})] \tag{9}$$

Using (9) and the fact that $q(\boldsymbol{z}|\boldsymbol{x})$ is fixed and doesn't depend on θ, we obtain

$$\begin{aligned}
\underset{\theta}{\operatorname{argmax}} \left(\mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})]\right) &= \underset{\theta}{\operatorname{argmax}} \left(\mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}[\log \frac{p_\theta(\boldsymbol{z}|\boldsymbol{x})p_\theta(\boldsymbol{x})}{q(\boldsymbol{z}|\boldsymbol{x})}]\right) \\
&= \underset{\theta}{\operatorname{argmax}} \left(\mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}[\log(p_\theta(\boldsymbol{x})) - \log(\frac{q(\boldsymbol{z}|\boldsymbol{x})}{p_\theta(\boldsymbol{z}|\boldsymbol{x})})]\right) \\
&= \underset{\theta}{\operatorname{argmax}} \left(\mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x})] - \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})}[\log \frac{q(\boldsymbol{z}|\boldsymbol{x})}{p_\theta(\boldsymbol{z}|\boldsymbol{x})}]\right) \\
&= \underset{\theta}{\operatorname{argmax}} \Big(\log p_\theta(\boldsymbol{x}) - D_{\mathrm{KL}}(q(\boldsymbol{z}|\boldsymbol{x})||p_\theta(\boldsymbol{z}|\boldsymbol{x}))\Big)
\end{aligned}$$

2.

$$\mathcal{L}(\theta, \phi^*; \boldsymbol{x}_i) = \log p_\theta(\boldsymbol{x}_i) - D_{\mathrm{KL}}(q_{\phi^*}(\boldsymbol{z}|\boldsymbol{x}_i)||p_\theta(\boldsymbol{z}|\boldsymbol{x}_i))$$

We denote by $\mathcal{L}_{q_i^*}$ the ELBO corresponding to q_i^*

$$\begin{aligned}
\mathcal{L}_{q_i^*} &= \mathbb{E}_q[\log p_\theta(\boldsymbol{x}_i, \boldsymbol{z}) - \log q_i^*(\boldsymbol{z})] \\
&= \mathbb{E}_q[\log \frac{p_\theta(\boldsymbol{x}_i, \boldsymbol{z})}{q_i^*(\boldsymbol{z})}] \\
&= \mathbb{E}_q[\log \frac{p_\theta(\boldsymbol{z}|\boldsymbol{x}_i)p_\theta(\boldsymbol{x}_i)}{q_i^*(\boldsymbol{z})}] \\
&= \log p_\theta(\boldsymbol{x}_i) - D_{\mathrm{KL}}(q_i^*(\boldsymbol{z})||p_\theta(\boldsymbol{z}|\boldsymbol{x}_i))
\end{aligned}$$

Therefore,

$$\begin{aligned}
&\log p_\theta(\boldsymbol{x}_i) = \log p_\theta(\boldsymbol{x}_i) \\
\implies &\mathcal{L}_{q_i^*} + D_{\mathrm{KL}}(q_i^*(\boldsymbol{z})||p_\theta(\boldsymbol{z}|\boldsymbol{x}_i)) = \mathcal{L}(\theta, \phi^*; \boldsymbol{x}_i) + D_{\mathrm{KL}}(q_{\phi^*}(\boldsymbol{z}|\boldsymbol{x}_i)||p_\theta(\boldsymbol{z}|\boldsymbol{x}_i)) \\
\implies &\mathcal{L}_{q_i^*} - \mathcal{L}(\theta, \phi^*; \boldsymbol{x}_i) = D_{\mathrm{KL}}(q_{\phi^*}(\boldsymbol{z}|\boldsymbol{x}_i)||p_\theta(\boldsymbol{z}|\boldsymbol{x}_i)) - D_{\mathrm{KL}}(q_i^*(\boldsymbol{z})||p_\theta(\boldsymbol{z}|\boldsymbol{x}_i))
\end{aligned}$$

Because q_i^* maximizes the ELBO of the instance i

$$\begin{aligned}
&\mathcal{L}_{q_i^*} \geq \mathcal{L}(\theta, \phi^*; \boldsymbol{x}_i) \\
\implies &\mathcal{L}_{q_i^*} - \mathcal{L}(\theta, \phi^*; \boldsymbol{x}_i) \geq 0 \\
\implies &D_{\mathrm{KL}}(q_{\phi^*}(\boldsymbol{z}|\boldsymbol{x}_i)||p_\theta(\boldsymbol{z}|\boldsymbol{x}_i)) - D_{\mathrm{KL}}(q_i^*(\boldsymbol{z})||p_\theta(\boldsymbol{z}|\boldsymbol{x}_i) \geq 0 \\
\implies &D_{\mathrm{KL}}(q_{\phi^*}(\boldsymbol{z}|\boldsymbol{x}_i)||p_\theta(\boldsymbol{z}|\boldsymbol{x}_i)) \geq D_{\mathrm{KL}}(q_i^*(\boldsymbol{z})||p_\theta(\boldsymbol{z}|\boldsymbol{x}_i)
\end{aligned}$$

3.

(a) When both approaches are optimal, the bias of the version with q_i^* is better or equal than with ϕ^* because, according to the previous question, the Kullback–Leibler divergence (i.e. the bias) of the approach with ϕ^* cannot be smaller than the approach with q_i^*.

(b) The q_i^* approach optimizes for each of the n samples in the dataset. Therefore, the amount of computation increases linearly with the dataset. The advantage of using q_{ϕ^*} is that it amortizes the amount of computation and it doesn't increases with the dataset's size.

(c) The approach with ϕ^* has only one q_{ϕ^*}. The other approaches has one q_i^* for each of the n samples in the training set.

Section 4 (VAE). Let $p(x, z)$ be the joint probability of a latent variable model where x and z denote the observed and unobserved variables, respectively. Let $q(z|x)$ be an auxiliary distribution which we call the *proposal*, and define

$$\mathcal{L}_K[q(z|x)] = \int \cdots \int \left(q(z_1|x)...q(z_K|x) \log \frac{1}{K} \sum_{j=1}^{K} \frac{p(x, z_j)}{q(z_j|x)} \right) dz_1 dz_2 ... dz_K$$

This objective is a tighter lower bound on $\log p(x)$ than the evidence lower bound (ELBO), which is equal to $\mathcal{L}_1$; that is $\mathcal{L}_1[q(z|x)] \leq \mathcal{L}_K[q(z|x)] \leq \log p(x)$.

In fact, $\mathcal{L}_K[q(z|x)]$ can be interpreted as the ELBO with a refined proposal distribution. For z_j drawn i.i.d. from $q(z|x)$ with $2 \leq j \leq K$, we define the *unnormalized* density

$$\tilde{q}(z|x, z_2, ..., z_K) := \frac{p(x, z)}{\frac{1}{K}\left(\frac{p(x,z)}{q(z|x)} + \sum_{j=2}^{K} \frac{p(x,z_j)}{q(z_j|x)}\right)}$$

(Note: in what follows, we use the fact that if $w_1, ..., w_K$ are random variables that have the same distribution, then $K\mathbb{E}[w_1] = \sum_i \mathbb{E}[w_i] = \mathbb{E}[\sum_i w_i]$.)

1. We show that $\mathcal{L}_K[q(z|x)] = \mathbb{E}_{z_{2:K}}[\mathcal{L}_1[\tilde{q}(z|x, z_2, ..., z_K)]]$; that is, the importance-weighted lower bound with K samples is equal to the average ELBO with the unnormalized density as a refined proposal.
2. We show that $q_K(z|x) := \mathbb{E}_{z_{2:K}}[\tilde{q}(z|x, z_2, ..., z_K)]$ is in fact a probability density function. Also, we show that $\mathcal{L}_1[q_K(z|x)]$ is an even tighter lower bound than $\mathcal{L}_K[q(z|x)]$. This implies $q_K(z|x)$ is closer to the true posterior $p(z|x)$ than $q(z|x)$ due to resampling, since $\mathcal{L}_K[q(z|x)] \geq \mathcal{L}_1[q(z|x)]$. (Note: $f(x) := -x \log x$ is concave.)

Steps 4.
Using Blei 2002 and Cremer et al. 2017

1.

$$
\begin{aligned}
\mathbb{E}_{z_{2:K}}[\mathcal{L}_1[\tilde{q}(z|x,z_2,...,z_K)]] &= \int \tilde{q}(z|x,z_2,..,z_K)\mathbb{E}_{z_{2:K}}[\log\left(\frac{p(x,z)}{\tilde{q}(z|x,z_2,..,z_K)}\right)]dz \\
&= \mathbb{E}_{z_{2:K}}[\int \tilde{q}(z|x,z_2,..,z_K)\log\left(\frac{p(x,z)}{\tilde{q}(z|x,z_2,..,z_K)}\right)dz] \\
&= \mathbb{E}_{z_{2:K}}[\int \tilde{q}(z_1|x,z_2,..,z_K)\log\left(\frac{p(x,z_1)}{\tilde{q}(z_1|x,z_2,..,z_K)}\right)dz_1] \\
&= \mathbb{E}_{z_{2:K}}[\int \frac{p(x,z_1)}{\frac{1}{K}\left(\frac{p(x,z_1)}{q(z_1|x)}+\sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}\log\left(\frac{\frac{p(x,z_1)}{p(x,z_1)}}{\frac{1}{K}\left(\frac{p(x,z_1)}{q(z_1|x)}+\sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}\right)dz_1] \\
&= \mathbb{E}_{z_{2:K}}[\int \frac{\frac{p(x,z_1)}{q(z_1|x)}q(z_1|x)}{\frac{1}{K}\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}}\log\left(\frac{p(x,z_1)}{\frac{p(x,z_1)}{\frac{1}{K}\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}}}\right)dz_1] \\
&= \mathbb{E}_{z_{1:K}}[\frac{K\frac{p(x,z_1)}{q(z_1|x)}}{\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}}\log\left(\frac{1}{K}\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)] \\
&\text{Using the note where } w_1 = \frac{p(x,z_1)}{q(z_1|x)} \\
&= K\mathbb{E}_{z_{1:K}}\left[\frac{w_1}{\left(\sum_{j=1}^{K}w_j\right)}\log\left(\frac{1}{K}\sum_{j=1}^{K}w_j\right)\right] \\
&= K\mathbb{E}_{z_{1:K}}[\tilde{w_1}] \\
&= \mathbb{E}_{z_{1:K}}\left[\sum_{i=1}^{K}\frac{w_i}{\left(\sum_{j=1}^{K}w_j\right)}\log\left(\frac{1}{K}\sum_{j=1}^{K}w_j\right)\right] \\
&= \mathbb{E}_{z_{1:K}}[\frac{\sum_{i=1}^{K}\frac{p(x,z_i)}{q(z_i|x)}}{\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}}\log\left(\frac{1}{K}\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)] \\
&= \mathbb{E}_{z_{1:K}}[\log\left(\frac{1}{K}\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)] \\
&= \int\cdots\int\left(q(z_1|x)...q(z_K|x)\log\frac{1}{K}\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)dz_1dz_2...dz_K \\
&= \mathcal{L}_K[q(z|x)]
\end{aligned}
$$

2.

(a) We show that $q_K(z|x)$ is a probability density function:

$$q_K(z|x) = \mathbb{E}_{z_{2:K}}[\tilde{q}(z|x, z_2, ..., z_K)] = \mathbb{E}_{z_{2:K}}\left[\frac{p(x,z)}{\frac{1}{K}\left(\frac{p(x,z)}{q(z|x)} + \sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}\right] \geq 0$$

and

$$\begin{aligned}
\int q_K(z|x)dz &= \int \mathbb{E}_{z_{2:K}}[\tilde{q}(z|x, z_2, ..., z_K)]dz \\
&= \int \mathbb{E}_{z_{2:K}}\left[\frac{p(x,z_1)}{\frac{1}{K}\left(\frac{p(x,z_1)}{q(z_1|x)} + \sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}\right]dz_1 \\
&= \int \frac{q(z_1|x)}{q(z_1|x)}\mathbb{E}_{z_{2:K}}\left[\frac{p(x,z_1)}{\frac{1}{K}\left(\frac{p(x,z_1)}{q(z_1|x)} + \sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}\right]dz_1 \\
&= \int q(z_1|x)\mathbb{E}_{z_{2:K}}\left[\frac{\frac{p(x,z_1)}{q(z_1|x)}}{\frac{1}{K}\left(\frac{p(x,z_1)}{q(z_1|x)} + \sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}\right]dz_1 \\
&= \mathbb{E}_{z_1}\mathbb{E}_{z_{2:K}}\left[\frac{\frac{p(x,z_1)}{q(z_1|x)}}{\frac{1}{K}\left(\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}\right] \\
&= \mathbb{E}_{z_{1:K}}\left[\frac{K\frac{p(x,z_1)}{q(z_1|x)}}{\left(\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}\right] \\
&\text{Using the note where } w_1 = \frac{p(x,z_1)}{q(z_1|x)} \\
&= K\mathbb{E}_{z_{1:K}}\left[\frac{w_1}{\left(\sum_{j=1}^{K} w_j\right)}\right] \\
&= K\mathbb{E}_{z_{1:K}}\left[\tilde{w}_1\right] \\
&= \mathbb{E}_{z_{1:K}}\left[\sum_{i=1}^{K}\frac{\frac{p(x,z_i)}{q(z_i|x)}}{\left(\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}\right] \\
&= \mathbb{E}_{z_{1:K}}\left[\frac{\sum_{i=1}^{K}\frac{p(x,z_i)}{q(z_i|x)}}{\left(\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}\right] \\
&= \mathbb{E}_{z_{1:K}}\left[\frac{\sum_{i=1}^{K}\frac{p(x,z_i)}{q(z_i|x)}}{\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}}\right] \\
&= \mathbb{E}_{z_{1:K}}\left[1\right] \\
&= 1
\end{aligned}$$

(b) We show that $\mathcal{L}_1[q_K(z|x)] \geq \mathcal{L}_K[q(z|x)]$:

$$\begin{aligned}
\mathcal{L}_1[q_K(z|x)] &= \mathbb{E}_z[\log\left(\frac{p(x,z)}{q_K(z|x)}\right)] \\
&= \mathbb{E}_z[\log\left(\frac{p(x,z)}{\mathbb{E}_{z_{2:K}}[\tilde{q}(z|x,z_2,...,z_K)]}\right)] \\
&= \mathbb{E}_z[\log\left(\frac{p(x,z)}{\mathbb{E}_{z_{2:K}}[\frac{p(x,z)}{\frac{1}{K}\left(\frac{p(x,z)}{q(z|x)}+\sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}]}\right)] \\
&= \mathbb{E}_z[\log\left(\frac{1}{\mathbb{E}_{z_{2:K}}[\frac{1}{\frac{1}{K}\left(\frac{p(x,z)}{q(z|x)}+\sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}]}\right)] \\
&= \mathbb{E}_z\left[0-\log\left(\mathbb{E}_{z_{2:K}}[\frac{1}{\frac{1}{K}\left(\frac{p(x,z)}{q(z|x)}+\sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}]\right)\right] \\
&= -\int q_K(z|x)\log\left(\mathbb{E}_{z_{2:K}}[\frac{1}{\frac{1}{K}\left(\frac{p(x,z)}{q(z|x)}+\sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}]\right)dz \\
&= -\int \mathbb{E}_{z_{2:K}}[\frac{p(x,z)}{\frac{1}{K}\left(\frac{p(x,z)}{q(z|x)}+\sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}]\log\left(\mathbb{E}_{z_{2:K}}[\frac{1}{\frac{1}{K}\left(\frac{p(x,z)}{q(z|x)}+\sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}]\right)dz \\
&= -\int p(x,z)\mathbb{E}_{z_{2:K}}[\frac{1}{\frac{1}{K}\left(\frac{p(x,z)}{q(z|x)}+\sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}]\log\left(\mathbb{E}_{z_{2:K}}[\frac{1}{\frac{1}{K}\left(\frac{p(x,z)}{q(z|x)}+\sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}]\right)dz
\end{aligned}$$

With the note $f(w) = -w\log w$ is concave and with Jensen's Inequality, we have $-f(\mathbb{E}_{z_{2:K}}[w]) \leq -\mathbb{E}_{z_{2:K}}[f(w)]$ where $w = \dfrac{1}{\frac{1}{K}\left(\frac{p(x,z)}{q(z|x)}+\sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}$

$\Big($i.e. $f(\mathbb{E}_{z_{2:K}}[w]) \geq \mathbb{E}_{z_{2:K}}[f(w)]\Big)$

$$\begin{aligned}
&\geq -\int p(x,z)\mathbb{E}_{z_{2:K}}\left[\frac{1}{\frac{1}{K}\left(\frac{p(x,z)}{q(z|x)}+\sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}\log\left(\frac{1}{\frac{1}{K}\left(\frac{p(x,z)}{q(z|x)}+\sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)}\right)\right]dz \\
&= -\int p(x,z)\mathbb{E}_{z_{2:K}}\left[w\log(w)\right]dz
\end{aligned}$$

$$= -\int p(x,z)\int ...\int q(z_2|x)...q(z_K|x)w\log(w)\,dzdz_2...dz_K$$

$$= -\int p(x,z_1)\int ...\int q(z_2|x)...q(z_K|x)w\log(w)\,dz_1dz_2...dz_K$$

$$\left(\text{Changing } z \text{ by } z_1 \implies w = \frac{K}{\left(\frac{p(x,z_1)}{q(z_1|x)} + \sum_{j=2}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)} = \frac{K}{\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}}\right)$$

$$= -\int ...\int p(x,z_1)q(z_2|x)...q(z_K|x)w\log(w)\,dz_1...dz_K$$

$$= -\int ...\int \frac{p(x,z_1)}{q(z_1|x)}q(z_1|x)q(z_2|x)...q(z_K|x)w\log(w)\,dz_1...dz_K$$

$$= \int ...\int \frac{p(x,z_1)}{q(z_1|x)}q(z_1|x)...q(z_K|x)w\log\left(w^{-1}\right)dz_1...dz_K$$

$$= k\int ...\int \frac{\frac{p(x,z_1)}{q(z_1|x)}}{\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}}q(z_1|x)...q(z_K|x)\log\left(w^{-1}\right)dz_1...dz_K$$

$$= k\mathbb{E}_{z_{1:K}}\left[\frac{\frac{p(x,z_1)}{q(z_1|x)}}{\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}}\log\left(\frac{1}{K}\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)\right]$$

Using the hint $K\mathbb{E}[w_1] = \mathbb{E}[\sum_i w_i]$ where $w_1 = \frac{p(x,z_1)}{q(z_1|x)}$

$$= K\mathbb{E}_{z_{1:K}}\left[\frac{w_1}{\left(\sum_{j=1}^{K}w_j\right)}\log\left(\frac{1}{K}\sum_{j=1}^{K}w_j\right)\right]$$

$$= K\mathbb{E}_{z_{1:K}}\left[\tilde{w}_1\right]$$

$$= \mathbb{E}_{z_{1:K}}\left[\sum_{i=1}^{K}\frac{w_i}{\left(\sum_{j=1}^{K}w_j\right)}\log\left(\frac{1}{K}\sum_{j=1}^{K}w_j\right)\right]$$

$$= \mathbb{E}_{z_{1:K}}[\frac{\sum_{i=1}^{K}\frac{p(x,z_i)}{q(z_i|x)}}{\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}}\log\left(\frac{1}{K}\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)]$$

$$= \mathbb{E}_{z_{1:K}}[\log\left(\frac{1}{K}\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)]$$

$$= \int\cdots\int q(z_1|x)...q(z_K|x)\log\left(\frac{1}{K}\sum_{j=1}^{K}\frac{p(x,z_j)}{q(z_j|x)}\right)dz_1...dz_K$$

$$= \mathcal{L}_K[q(z|x)]$$

Section 5 (**Normalizing flows**)**.** Normalizing flows are expressive invertible transformations of probability distributions. In this section, we will see how to satisfy the invertibility constraint of some family of parameterizations. For the first 3 subsections, we assume the function $g : \mathbb{R} \to \mathbb{R}$ maps from real space to real space.

1. Let $g(z) = af\,(bz + c)$ where f is the ReLU activation function $f(x) = \max(0, x)$. We show that g is non-invertible.
2. Let $g(z) = \sigma^{-1}(\sum_{i=1}^{N} w_i \sigma(a_i z + b_i))$, $0 < w_i < 1$, where $\sum_i w_i = 1$, $a_i > 0$, and $\sigma(x) = 1/(1 + \exp(-x))$ is the logistic sigmoid activation function and σ^{-1} is its inverse. We show that g is *strictly monotonically increasing* on its domain $(-\infty, \infty)$, which implies invertiblity.
3. We consider a residual function of the form $g(z) = z + f(z)$ and we show that $df/dz > -1$ implies g is invertible.
4. We consider the following transformation:

$$g(\boldsymbol{z}) = \boldsymbol{z} + \beta h(\alpha, r)(\boldsymbol{z} - \boldsymbol{z}_0) \tag{10}$$

where $\boldsymbol{z}_0 \in \mathbb{R}^D$, $\alpha \in \mathbb{R}^+$, $\beta \in \mathbb{R}$, and $r = ||\boldsymbol{z} - \boldsymbol{z}_0||_2$, $h(\alpha, r) = 1/(\alpha + r)$. We also consider the following decomposition of $\boldsymbol{z} = \boldsymbol{z}_0 + r\tilde{\boldsymbol{z}}$. (i) Given $\boldsymbol{y} = g(\boldsymbol{z})$, we show that $\beta \geq -\alpha$ is a sufficient condition to derive the unique r from equation (10). (ii) Given r and $\boldsymbol{y}$, we show that equation (10) has a unique solution $\tilde{\boldsymbol{z}}$.

Steps 5.

1.
$g(z) = x$ is invertible $\iff$ $g(z)$ is bijective
However, $g(\frac{-(c+1)}{b}) = a \times max(0, b\frac{-(c+1)}{b} + c) = a \times max(0, -1) = 0$ and $g(\frac{-(c+2)}{b}) = a \times max(0, b\frac{-(c+2)}{b} + c) = a \times max(0, -2) = 0$. Therefore, $g(z)$ is not bijective and non-invertible.

2.
We know that:
— log and exp are strictly monotonically increasing
— if f is strictly monotonically increasing then f^{-1} is strictly monotonically increasing

For $h(z) = a_i z + b_i$ where $a_i > 0$
$y > x \implies a_i y > a_i x \implies a_i y + b_i > a_i x + b_i$
Therefore, $h(z)$ is strictly monotonically increasing

$$y > x \implies -y < -x \implies \exp(-y) < \exp(-x) \implies 1 + \exp(-y) < 1 + \exp(-x)$$
$$\implies \frac{1}{1 + \exp(-x)} < \frac{1}{1 + \exp(-y)} \implies \sigma(x) < \sigma(y)$$

Therefore, $\sigma(z)$ is strictly monotonically increasing
Therefore, $\sigma^{-1}(z)$ is strictly monotonically increasing

If $y > x$ and $0 < w_i < 1$, we have:

$$\sigma(a_i y + b_i) > \sigma(a_i x + b_i) \implies w_i\sigma(a_i y + b_i) > w_i\sigma(a_i x + b_i) \implies \sum_i^N w_i\sigma(a_i y + b_i) > \sum_i^N w_i\sigma(a_i x + b_i)$$

$$\implies \sigma^{-1}(\sum_i^N w_i\sigma(a_i y + b_i)) > \sigma^{-1}(\sum_i^N w_i\sigma(a_i x + b_i)) \implies g(y) > g(x)$$

Therefore, $g(z)$ is strictly monotonically increasing

3.

if $y > x$ and $z \in (x, y)$, then

$$\frac{g(y) - g(x)}{y - x} = \frac{d(g(z))}{dz} = \frac{d(z + f(z))}{dz} > 1 - 1 = 0$$

and

$$\frac{g(y) - g(x)}{y - x} > 0 \implies g(y) - g(x) > 0 \implies g(y) > g(x)$$
$$\implies g(z) \text{ is strictly monotonically increasing}$$
$$\implies g(z) \text{ is invertible}$$

4.

(a) We know that
— for $c \in \mathbb{R}$ and $\boldsymbol{v} \in \mathbb{R}^D$, $||cv||_2 = |c| \times ||v||_2$
— $r = ||\boldsymbol{z} - \boldsymbol{z}_0||_2 \geq 0$

$$y = g(\boldsymbol{z}) = \boldsymbol{z} + \beta h(\alpha, r)(\boldsymbol{z} - \boldsymbol{z}_0) = \boldsymbol{z}_0 + r\tilde{\boldsymbol{z}} + \beta\frac{r\tilde{\boldsymbol{z}}}{\alpha + r} = \boldsymbol{z}_0 + r\tilde{\boldsymbol{z}}(1 + \frac{\beta}{\alpha + r})$$
$$\implies y - \boldsymbol{z}_0 = r\tilde{\boldsymbol{z}}(1 + \frac{\beta}{\alpha + r})$$
$$\implies ||y - \boldsymbol{z}_0||_2 = ||r\tilde{\boldsymbol{z}}(1 + \frac{\beta}{\alpha + r})||_2 = |r(1 + \frac{\beta}{\alpha + r})|\ ||\tilde{z}||_2 = |r(1 + \frac{\beta}{\alpha + r})|\ ||\frac{\boldsymbol{z} - \boldsymbol{z}_0}{r}||_2$$
$$= r|(1 + \frac{\beta}{\alpha + r})|\ \frac{r}{r} = r|(1 + \frac{\beta}{\alpha + r})|$$

To have an unique value we want

$$(1 + \frac{\beta}{\alpha + r}) \geq 0 \implies \beta \geq -\alpha - r$$

Because $r > 0$, $-\alpha \geq -\alpha - r$. Therefore, it is sufficient that $\beta \geq -\alpha$

(b)

$$y = \boldsymbol{z}_0 + r\tilde{\boldsymbol{z}} + \beta\frac{r\tilde{\boldsymbol{z}}}{\alpha + r}$$
$$\implies y - \boldsymbol{z}_0 = r\tilde{\boldsymbol{z}} + \beta\frac{r\tilde{\boldsymbol{z}}}{\alpha + r} = \tilde{\boldsymbol{z}}(r + \frac{r\beta}{\alpha + r})$$
$$\implies \tilde{\boldsymbol{z}} = \frac{y - \boldsymbol{z}_0}{r + \frac{r\beta}{\alpha + r}}$$

Section 6 (GAN). In this section, we are concerned with analyzing the training dynamics of GANs. We consider the following value function

$$V(d, g) = dg \tag{11}$$

with $g \in \mathbb{R}$ and $d \in \mathbb{R}$. We will use this simple example to study the training dynamics of GANs.

1. We consider gradient descent/ascent with learning rate α as the optimization procedure to iteratively minimize $V(d, g)$ w.r.t. g and maximize $V(d, g)$ w.r.t. d. We want to apply the gradient descent/ascent to update g and d simultaneously. We show the update rule of g and d in the following form
$$[d_{k+1}, g_{k+1}]^\top = A[d_k, g_k]^\top$$
where A is a 2×2 matrix.
2. The optimization procedure found in 6.1 characterizes a map which has a stationary point [5], we show the coordinates of the stationary points.
3. We analyze the eigenvalues of A and predict what will happen to d and g as we update them jointly. In other word, we predict the behaviour of d_k and g_k as $k \to \infty$.

Steps 6.

1.
$$d_{k+1} = d_k + \alpha \frac{\partial(d_k g_k)}{\partial d_k} = d_k + \alpha g_k$$
$$g_{k+1} = d_k - \alpha \frac{\partial(d_k g_k)}{\partial g_k} = g_k - \alpha d_k$$
Therefore,
$$A = \begin{bmatrix} 1 & \alpha \\ -\alpha & 1 \end{bmatrix}$$

2.
$$\nabla V(d, g) = \begin{bmatrix} g \\ -d \end{bmatrix} = 0 \implies g = 0 \text{ and } d = 0$$
Therefore, the coordinates are (0,0)

3.
$$\left| \begin{bmatrix} 1 & \alpha \\ -\alpha & 1 \end{bmatrix} \right| = (1 - \lambda)^2 + \alpha^2$$
$$(1 - \lambda)^2 + \alpha^2 = 0 \implies \lambda = 1 \pm \sqrt{-\alpha^2} = 1 \pm i\alpha$$
According to Mescheder (2018), because the real-part is positive, the eigenvalues mean that it is not locally convergent.

5. A stationary point is a point on the surface of the graph (of the function) where all its partial derivatives are zero (equivalently, the gradient is zero). Source: https://en.wikipedia.org/wiki/Stationary_point

Références

David M. Blei. Variational inference, 2002.

Chris Cremer, Quaid Morris, and David Duvenaud. Reinterpreting importance-weighted autoencoders, 2017.

Lars M. Mescheder. On the convergence properties of GAN training. *CoRR*, abs/1801.04406, 2018. URL `http://arxiv.org/abs/1801.04406`.