

# Variational Bayes

## Lecture 11

Konstantin Yakovlev <sup>1</sup>

<sup>1</sup>MIPT  
Moscow, Russia

MIPT 2023

- Latent Variable Model and Variational Autoencoders
- Improving the representation power of the variational posterior
- Discrete latent variables and Concrete distribution
- Black-box gradient estimation
- Vector-Quantized Variational Autoencoder

# Discriminative vs Generative modeling<sup>1</sup>

**Discriminative:**  $p(y|\mathbf{x})$

Advantages:

- ① Solve the problem you are evaluating on
- ② Very accurate given a sufficiently large amount of data
- ③ Effective training procedure

**Generative:**  $p(y, \mathbf{x})$

Advantages:

- ① Injection of expert knowledge
- ② Could be turned to a discriminator with Bayes rule
- ③ Facilitates semi/un-supervised learning

---

<sup>1</sup>Kingma D. et. al, An Introduction to Variational Autoencoders, 2019

# Latent Variable Model

## Probabilistic model:

Given an observed  $\mathbf{x} \sim \pi(\mathbf{x})$ , where  $\pi(\mathbf{x})$  is unknown. We attempt to approximate  $\pi(\cdot)$  with a parametric model  $p_\theta(\cdot)$ .

## Latent variable model:

Introduce a latent variable  $\mathbf{z}$ :

$$p_\theta(\mathbf{x}) = \int p_\theta(\mathbf{x}, \mathbf{z}) d\mathbf{z}.$$

The most common approach:

$$p_\theta(\mathbf{x}, \mathbf{z}) = p_\theta(\mathbf{z})p_\theta(\mathbf{x}|\mathbf{z}),$$

where  $p_\theta(\mathbf{z})$  is the *prior* distribution.

## Example:

$$p(\mathbf{z}) = \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{I}), \quad p_\theta(\mathbf{x}|\mathbf{z}) = \prod_{j=1}^D \text{Bern}(x_j | f_\theta^j(\mathbf{z})),$$
$$f_\theta^j(\mathbf{z}) = \sigma(\text{MLP}(\mathbf{z})_j).$$

## Maximul Likelihood Learning:

Given a set of  $N$  i.i.d. datapoints  $\mathcal{D} = \{\mathbf{x}_i\}_{i=1}^N$

$$\log p_\theta(\mathcal{D}) = \sum_{\mathbf{x} \in \mathcal{D}} \log p_\theta(\mathbf{x}) \rightarrow \max_{\theta}.$$

**Intractabilities:** the marginal likelihood  $p_\theta(\mathbf{x})$  is intractable due to the integral.

**Note:** while  $\hat{\nabla}_{\theta} \log p_\theta(\mathbf{x}) = (p_\theta(\mathbf{x}))^{-1} \hat{\nabla}_{\theta} p_\theta(\mathbf{x}|\hat{\mathbf{z}})$  is intractable ( $p_\theta(\mathbf{z}) = p(\mathbf{z})$  for simplicity)

# Evidence Lower Bound (ELBO)

**Challenge:** the intractability of  $\log p_{\theta}(\mathbf{x})$ .

**Solution:** variational lower bound. First, note that  $p_{\theta}(\mathbf{x})$  is tractable  $\Leftrightarrow p_{\theta}(\mathbf{z}|\mathbf{x}) = p_{\theta}(\mathbf{x}, \mathbf{z})/p_{\theta}(\mathbf{x})$  is tractable.

Second, introduce a parametric *variational distribution*  $q_{\phi}(\mathbf{z}|\mathbf{x})$ , where  $\phi$  is the vector of *variational parameters*. So, derive the lower bound

$$\begin{aligned}\log p_{\theta}(\mathbf{x}) &= \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} \log p_{\theta}(\mathbf{x}) \\ &= \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} \log \frac{p_{\theta}(\mathbf{x}, \mathbf{z}) q_{\phi}(\mathbf{z}|\mathbf{x})}{q_{\phi}(\mathbf{z}|\mathbf{x}) p_{\theta}(\mathbf{z}|\mathbf{x})} \\ &= \underbrace{\mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} \log \frac{p_{\theta}(\mathbf{x}, \mathbf{z})}{q_{\phi}(\mathbf{z}|\mathbf{x})}}_{\mathcal{L}_{\theta, \phi}(\mathbf{x})} + \underbrace{\mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} \log \frac{q_{\phi}(\mathbf{z}|\mathbf{x})}{p_{\theta}(\mathbf{z}|\mathbf{x})}}_{\text{KL}(q_{\phi}(\mathbf{z}|\mathbf{x}) || p_{\theta}(\mathbf{z}|\mathbf{x})) \geq 0}\end{aligned}$$

Here  $\mathcal{L}_{\theta, \phi}(\mathbf{x})$  is the *variational lower bound*.

## Proposition

$$p_{\theta}(\mathbf{x}) \geq \mathcal{L}_{\theta, \phi}(\mathbf{x}) = \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} \log \frac{p_{\theta}(\mathbf{x}, \mathbf{z})}{q_{\phi}(\mathbf{z}|\mathbf{x})}$$

**Note 1:** The gap between ELBO and the marginal likelihood is called the *tightness bound*. The better  $q_{\phi}(\mathbf{z}|\mathbf{x})$  approximates the true posterior  $p_{\theta}(\mathbf{z}|\mathbf{x})$  (in terms of KL), the closer the gap.

**Note 2:** We will show that there are tractable gradients of ELBO w.r.t.  $\theta, \phi$ .

# Stochastic Gradient-Based optimization of the ELBO

## Unbiased gradient of ELBO w.r.t the generative model parameters

$$\begin{aligned}\nabla_{\theta} \mathcal{L}_{\theta, \phi}(\mathbf{x}) &= \nabla_{\theta} \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} [\log p_{\theta}(\mathbf{x}, \mathbf{z}) - \log q_{\phi}(\mathbf{z}|\mathbf{x})] \\ &= \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} \nabla_{\theta} \log p_{\theta}(\mathbf{x}, \mathbf{z}) \\ &\approx \nabla_{\theta} \log p_{\theta}(\mathbf{x}, \hat{\mathbf{z}}), \quad \hat{\mathbf{z}} \sim q_{\phi}(\mathbf{z}|\mathbf{x}).\end{aligned}$$

**Reparametrization trick:** First, assume that we can express  $\mathbf{z} \sim q_{\phi}(\mathbf{z}|\mathbf{x})$  as  $\mathbf{z} = \mathbf{g}(\epsilon, \phi, \mathbf{x})$ ,  $\epsilon \sim p(\epsilon)$ , where  $\mathbf{g}$  is differentiable w.r.t.  $\phi$ , i.e.  $q_{\phi}(\mathbf{z}|\mathbf{x})$  is *reparametrizable*. Also assume that  $f(\cdot)$  is differentiable. Let  $\hat{\epsilon} \sim p(\epsilon)$ .

$$\begin{aligned}\nabla_{\phi} \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} f(\mathbf{z}) &= \nabla_{\phi} \mathbb{E}_{p(\epsilon)} f(\mathbf{g}(\epsilon, \phi, \mathbf{x})) \\ &= \mathbb{E}_{p(\epsilon)} \nabla_{\phi} f(\mathbf{g}(\epsilon, \phi, \mathbf{x})) \approx \nabla_{\phi} f(\mathbf{g}(\hat{\epsilon}, \phi, \mathbf{x}))\end{aligned}$$

## Unbiased gradient of ELBO w.r.t. the variational parameters

$$\begin{aligned}\nabla_{\phi} \mathcal{L}_{\theta, \phi}(\mathbf{x}) &= \nabla_{\phi} \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})} [\log p_{\theta}(\mathbf{x}, \mathbf{z}) - \log q_{\phi}(\mathbf{z}|\mathbf{x})] \\ &= \mathbb{E}_{p(\epsilon)} [\nabla_{\phi} \log p_{\theta}(\mathbf{x}, \mathbf{z}) - \nabla_{\phi} \log q_{\phi}(\mathbf{z}|\mathbf{x})] \Big|_{\mathbf{z}=\mathbf{g}(\epsilon, \phi, \mathbf{x})} \\ &\approx [\nabla_{\phi} \log p_{\theta}(\mathbf{x}, \mathbf{z}) - \nabla_{\phi} \log q_{\phi}(\mathbf{z}|\mathbf{x})] \Big|_{\mathbf{z}=\mathbf{g}(\hat{\epsilon}, \phi, \mathbf{x})}, \quad \hat{\epsilon} \sim p(\epsilon)\end{aligned}$$

**Note:** also assume that we have access to  $q_{\phi}(\mathbf{z}|\mathbf{x})$ ,  $\nabla_{\mathbf{z}} \log q_{\phi}(\mathbf{z}|\mathbf{x})$ , and  $\nabla_{\phi} \log q_{\phi}(\mathbf{z}|\mathbf{x})$ .

# Optimization of the ELBO

**Factorized Gaussian posterior:** a common choice of  $q_\phi(\mathbf{z}|\mathbf{x})$  is a factorized Gaussian

$$(\boldsymbol{\mu}, \log \boldsymbol{\sigma}) = \text{MLP}_\phi(\mathbf{x}), \quad q_\phi(\mathbf{z}|\mathbf{x}) = \prod_{i=1}^{\dim \mathbf{z}} \mathcal{N}(z_i|\mu_i, \sigma_i^2),$$

$$\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}.$$

**Limitation:** factorized distributions are not flexible. By increasing the flexibility of  $q_\phi(\mathbf{z}|\mathbf{x})$ , we improve the tightness of the ELBO ( $\text{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x}))$ ).

**Optimization problem**

$$\log p_\theta(\mathcal{D}) \geq \sum_{\mathbf{x} \in \mathcal{D}} \mathcal{L}_{\theta, \phi}(\mathbf{x}) \rightarrow \max_{\theta, \phi}.$$

**Data:**  $\mathcal{D} = \{\mathbf{x}_i\}_{i=1}^N$

**Result:** learned  $\theta, \phi$

$\theta, \phi \leftarrow$  initialization;

**while** *not converged* **do**

$\mathbf{x}_{1:M} \leftarrow$  random minibatch of  $M$   
    datapoints;

$\boldsymbol{\epsilon} \leftarrow$  sample random noise from  
     $p(\boldsymbol{\epsilon})$ ;

$\hat{\mathbf{g}} \leftarrow$  stochastic gradients of  
     $\mathcal{L}_{\theta, \phi}(\mathbf{x})$  w.r.t  $\theta, \phi$ ;

$\theta, \phi \leftarrow$  update parameters using  $\hat{\mathbf{g}}$

**end**

**Algorithm 1:** Minibatch version of the Auto-Encoding VB

# Estimation of the Marginal Likelihood and Sampling<sup>2</sup>

## Estimation of the Marginal Likelihood

### Theorem

For all  $k \geq 1$  the following is true:

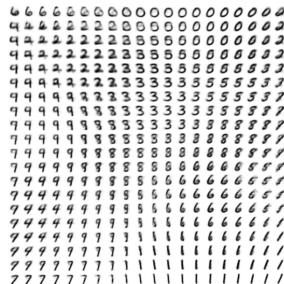
$$\log p(\mathbf{x}) \geq \mathcal{L}_{k+1}(\mathbf{x}) \geq \mathcal{L}_k(\mathbf{x}),$$

$$\mathcal{L}_k(\mathbf{x}) := \mathbb{E}_{\mathbf{z}_{1:k} \sim q_\phi(\mathbf{z}|\mathbf{x})} \log \left( \frac{1}{k} \sum_{i=1}^k \frac{p_\theta(\mathbf{x}, \mathbf{z}_i)}{q_\phi(\mathbf{z}_i|\mathbf{x})} \right).$$

Moreover, if  $p_\theta(\mathbf{x}, \mathbf{z})/q_\phi(\mathbf{z}|\mathbf{x})$  is bounded, then  $\mathcal{L}_k(\mathbf{x})$  approaches  $\log p(\mathbf{x})$  as  $k$  goes to infinity.

## Sampling<sup>a</sup>

$$\mathbf{x} \sim p_\theta(\mathbf{x}) \Leftrightarrow \mathbf{z} \sim p_\theta(\mathbf{z}), \mathbf{x} \sim p_\theta(\mathbf{x}|\mathbf{z}).$$



<sup>a</sup>Kingma D. et. al, Auto-Encoding Variational Bayes, 2014

<sup>2</sup>Burda Y. et. al, Importance Weighted Autoencoders, 2016



# Semi-Implicit Variational Inference<sup>3</sup>

**Challenge:** the representation power of the variational family is limited by the assumption that  $q(\mathbf{z}|\mathbf{x})$  is factorizable.

**Solution:** introduce a mixing distribution on the parameters on the original  $q(\mathbf{z}|\mathbf{x})$ .

$$\mathcal{H} := \{h_\phi(\mathbf{z}) : h_\phi(\mathbf{z}) = \mathbb{E}_{q_\phi(\psi)} q(\mathbf{z}|\psi)\},$$

$q(\mathbf{z}|\psi)$  explicit and reparametrizable,  
 $q_\phi(\psi)$  implicit and reparametrizable,  
 $\Rightarrow h_\phi(\mathbf{z})$  implicit in the general case

**Lower bound of ELBO:**

$$\text{ELBO} = \mathbb{E}_{\mathbf{z} \sim h_\phi(\mathbf{z})} \log \frac{p(\mathbf{x}, \mathbf{z})}{h_\phi(\mathbf{z})} =$$

$$\begin{aligned} \log p(\mathbf{x}) - \text{KL}(\mathbb{E}_{\psi \sim q_\phi(\psi)} q(\mathbf{z}|\psi) || p(\mathbf{z}|\mathbf{x})) &\geq \\ - \mathbb{E}_{\psi \sim q_\phi(\psi)} \text{KL}(q(\mathbf{z}|\psi) || p(\mathbf{z}|\mathbf{x})) + \log p(\mathbf{x}) &= \\ \mathbb{E}_{\psi \sim q_\phi(\psi)} \mathbb{E}_{\mathbf{z} \sim q(\mathbf{z}|\psi)} \log \frac{p(\mathbf{x}, \mathbf{z})}{q(\mathbf{z}|\psi)} &=: \underline{\mathcal{L}}(q(\mathbf{z}|\psi), q_\phi(\psi)) \end{aligned}$$

## Theorem

Let  $\psi^* = \arg \max_{\psi} \mathbb{E}_{q(\mathbf{z}|\psi)} \frac{p(\mathbf{x}, \mathbf{z})}{q(\mathbf{z}|\psi)}$ . Then

$$\begin{aligned} \max_{q_\phi(\psi)} \underline{\mathcal{L}}(q(\mathbf{z}|\psi), q_\phi(\psi)) &= \mathbb{E}_{\mathbf{z} \sim q(\mathbf{z}|\psi^*)} \log \frac{p(\mathbf{x}, \mathbf{z})}{q(\mathbf{z}|\psi^*)}, \\ \arg \max_{q_\phi(\psi)} \underline{\mathcal{L}}(q(\mathbf{z}|\psi), q_\phi(\psi)) &= \delta(\psi - \psi^*) \end{aligned}$$

Therefore, SIVI degenerates to vanilla VI.

<sup>3</sup>Yin M. et. al, Semi-Implicit Variational Inference, 2018

# Semi-Implicit Variational Inference: preventing degeneracy

Introduce a regularizer:

$$B_K := \mathbb{E}_{\psi, \psi^{(1:K)} \sim q_\phi(\psi)} \text{KL}(q(\mathbf{z}|\psi) || \tilde{h}_K(\mathbf{z})),$$
$$\tilde{h}_K(\mathbf{z}) := \frac{1}{K+1} \left( q(\mathbf{z}|\psi) + \sum_{k=1}^K q(\mathbf{z}|\psi^{(k)}) \right)$$

Note that  $B_K = 0 \Leftrightarrow K = 0$  or  $q_\phi(\psi) = \delta(\psi - \mathbf{a})$ .

## Theorem

$$\lim_{K \rightarrow \infty} (\underline{\mathcal{L}} + B_K) = \text{ELBO} = \mathbb{E}_{\mathbf{z} \sim h_\phi(\mathbf{z})} \log \frac{p(\mathbf{x}, \mathbf{z})}{h_\phi(\mathbf{z})}.$$

**Informal intuition:** use the strong law of large numbers:

$$\begin{aligned} \lim_{K \rightarrow \infty} B_K &= \mathbb{E}_{\psi \sim q_\phi(\psi)} \text{KL}(q(\mathbf{z}|\psi) || h_\phi(\mathbf{z})). \\ \Rightarrow \underline{\mathcal{L}} + B_\infty &= \mathbb{E}_{\psi \sim q_\phi(\psi)} \mathbb{E}_{\mathbf{z} \sim q(\mathbf{z}|\psi)} \log \frac{p(\mathbf{x}, \mathbf{z})}{q(\mathbf{z}|\psi)} + \\ &\mathbb{E}_{\psi \sim q_\phi(\psi)} \text{KL}(q(\mathbf{z}|\psi) || h_\phi(\mathbf{z})) = \\ &\mathbb{E}_{\psi \sim q_\phi(\psi)} \mathbb{E}_{\mathbf{z} \sim q(\mathbf{z}|\psi)} \left( \log \frac{p(\mathbf{x}, \mathbf{z})}{q(\mathbf{z}|\psi)} + \log \frac{q(\mathbf{z}|\psi)}{h_\phi(\mathbf{z})} \right) = \\ &\mathbb{E}_{\mathbf{z} \sim h_\phi(\mathbf{z})} \log \frac{p(\mathbf{x}, \mathbf{z})}{h_\phi(\mathbf{z})} = \text{ELBO} \end{aligned}$$

## Negative binomial model

$$x_i \sim \text{NB}(r, p), \quad r \sim \text{Gamma}(a, 1/b), \\ p \sim \text{Beta}(\alpha, \beta), \quad a = b = \alpha = \beta = 0.01.$$

Therefore, the posterior is intractable.

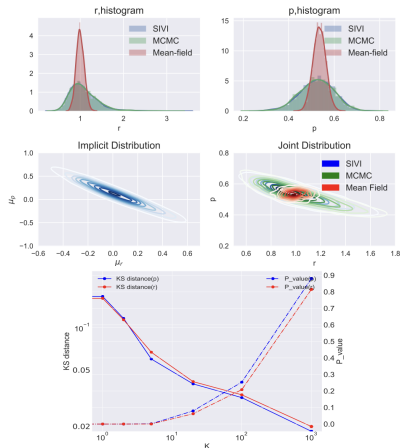
## Mean-Field VI

$$q(r, p) = \text{Gamma}(r|\tilde{a}, 1/\tilde{b})\text{Beta}(p|\tilde{\alpha}, \tilde{\beta}).$$

## SIVI:

$$q(r, p|\psi) = \text{LogNorm}(r|\mu_r, \sigma_0^2)\text{LogitNorm}(p|\mu_p, \sigma_0^2), \\ \psi = (\mu_r, \mu_p) \sim q(\psi) \text{ is MLP-based.}$$

The model is trained with  $K = 1000$ .



We see that  $K = 20$  achieves a nice compromise between complexity and accuracy.

# A Continuous relaxation of discrete random variables<sup>4</sup>

**Challenge:** discrete random variables lack useful reparameterizations due to the discontinuous nature of discrete states.

**Solution:** introduce Concrete random variables.

**Background: Reparametrization trick:**

$$\mathbb{E}_{x \sim p_\phi(x)} f(x) \rightarrow \min_{\phi}.$$

Assume that  $f$  is differentiable w.r.t  $x$ ,  $x \sim p_\phi(x) \Leftrightarrow z \sim p(z)$ ,  $x = g_\phi(z)$ , and  $g_\phi(\cdot)$  is differentiable w.r.t.  $\phi$ .

$$\hat{\nabla}_{\phi} \mathbb{E}_{x \sim p_\phi(x)} f(x) = \frac{\partial f}{\partial g_\phi(z)} \frac{\partial g_\phi(z)}{\partial \phi}.$$

## Gumbel-Max trick

$$d \sim \text{Cat}(\alpha_1, \dots, \alpha_n) \Leftrightarrow \\ d = \arg \max_{k=1, n} (\log \alpha_k - \underbrace{\log(-\log u_k))}_{g_k \sim \text{Gumbel}(0,1)}),$$

$$u_k \stackrel{i.i.d.}{\sim} \mathcal{U}[0, 1].$$

**Concrete random variable:** given  $\alpha \in \mathbb{R}_{++}^n$

$$\mathbf{x} := \text{softmax}((\log \alpha + \mathbf{g})/\lambda) \sim \text{Concrete}(\alpha, \lambda),$$

$$\{g_i\} \stackrel{i.i.d.}{\sim} \text{Gumbel}(0, 1),$$

$\lambda \in \mathbb{R}_{++}$  temperature parameter.

The softmax approaches arg max as  $\lambda \rightarrow 0$ .

<sup>4</sup>Maddison C. et. al, The Concrete Distribution: A Continuous relaxation of discrete random variables, 2017

# Properties of Concrete distribution

## Theorem

The following is true for  $\mathbf{x} \sim \text{Concrete}(\boldsymbol{\alpha}, \lambda)$

- (Rounding)  $\mathbb{P}(x_k > x_i, i \neq k) = \frac{\alpha_k}{\sum_{i=1}^n \alpha_i}$ .
- (Zero Temperature)  
 $\mathbb{P}(\lim_{\lambda \rightarrow 0} x_k = 1) = \frac{\alpha_k}{\sum_{i=1}^n \alpha_i}$ .
- (Convex eventually) if  $\lambda \leq (n-1)^{-1}$ , then  $p(\mathbf{x}|\boldsymbol{\alpha}, \lambda)$  is log-convex in  $\mathbf{x}$ .

**Note:** for any  $\lambda > 0$  the gradient estimator is biased. The temperature sets a tradeoff between the bias and the variance of the estimator. The higher  $\lambda$ , the greater the bias.

**VAE with a single discrete latent variable:**

$$q_{\phi}(d|\mathbf{x}) = \text{Cat}(d|f_{\theta}(\mathbf{x})),$$

$$\mathbb{E}_{d \sim q_{\phi}(d|\mathbf{x})} \log \frac{p_{\theta}(\mathbf{x}, d)}{q_{\phi}(d)} \overset{\text{relax}}{\rightsquigarrow}$$

$$\mathbb{E}_{\mathbf{z} \sim q_{\alpha, \lambda}(\mathbf{z}|\mathbf{x}, \phi)} \log \frac{p_{\theta}(\mathbf{x}, \mathbf{z})}{q_{\alpha, \lambda}(\mathbf{z}|\mathbf{x}, \phi)}.$$

Note that we assumed that  $p(\mathbf{x}, \mathbf{z})$  is feasible, i.e. the decoder is able to condition on relaxed variable  $\mathbf{z}$ .

**Setup:** VAE with discrete latent variables. More specifically, the task is to predict the bottom half  $\mathbf{x}_1$  of a MNIST image given the upper one  $\mathbf{x}_2$ . Consider IWAE objective with the prior  $p_\theta(\mathbf{z}|\mathbf{x}_2)$  as the variational distribution.

$$\mathcal{L}_m(\mathbf{x}_1, \mathbf{x}_2) = \mathbb{E}_{\mathbf{z}_{1:m} \sim p_\theta(\mathbf{z}|\mathbf{x}_2)} \log \left( \frac{1}{m} \sum_{i=1}^m p_\theta(\mathbf{x}_1|\mathbf{z}_i) \right).$$

## Comparison with a baseline

binary model	$m$	Test NLL		Train NLL	
		Concrete	VIMCO	Concrete	VIMCO
(392V–240H –240H–392V)	1	<b>58.5</b>	61.4	<b>54.2</b>	59.3
	5	<b>54.3</b>	54.5	<b>49.2</b>	52.7
	50	53.4	<b>51.8</b>	<b>48.2</b>	49.6
(392V–240H –240H–240H –392V)	1	<b>56.3</b>	59.7	<b>51.6</b>	58.4
	5	<b>52.7</b>	53.5	<b>46.9</b>	51.6
	50	52.0	<b>50.2</b>	<b>45.9</b>	47.9

It could be clearly seen that the proposed Concrete estimator outperforms the baseline.

# Black-box gradient estimation<sup>5</sup>

**Challenge:** backpropagation w.r.t. encoder parameters could not be applied in case of discrete latent variables

**Solution:** introduce a learnable free-form control variate parameterized by a neural network.

**Background: gradient estimators**

$$\mathbb{E}_{p(b|\theta)}[f(b)] \rightarrow \min_{\theta} \Leftarrow \mathbb{E}_{q_{\phi}(z|x)} \log p_{\theta}(x|z)$$

$$\hat{g}_{\text{reinf}}[f] := f(b) \frac{\partial}{\partial \theta} \log p(b|\theta), \quad b \sim p(b|\theta),$$

$$\hat{g}_{\text{reparam}}[f] := \frac{\partial}{\partial \theta} f(b) = \frac{\partial f}{\partial T} \frac{\partial T(\theta, \epsilon)}{\partial \theta}, \quad \epsilon \sim p(\epsilon).$$

**Control variates:** reduces the variance of a stochastic estimator.

$$\hat{g}_{\text{new}}(b) := \hat{g}(b) - c(b) + \mathbb{E}_{p(b|\theta)}[c(b)].$$

**Constructing a differentiable surrogate:** Assume that  $b$  - continuous and  $b = T(\theta, \epsilon)$ ,  $\epsilon \sim p(\epsilon)$ ; given  $c_{\phi}$ , a differentiable surrogate of  $f$ , but  $f$  cannot be differentiated.

$$\begin{aligned} \hat{g}_{\text{LAX}} &:= \hat{g}_{\text{reinf}}[f] - \hat{g}_{\text{reinf}}[c_{\phi}] + \hat{g}_{\text{reparam}}[c_{\phi}] = \\ &[f(b) - c_{\phi}(b)] \frac{\partial}{\partial \theta} \log p(b|\theta) + \frac{\partial}{\partial \theta} c_{\phi}(b). \end{aligned}$$

Note that the estimator is unbiased for any  $c_{\phi}$ .

---

<sup>5</sup>Grathwohl W. et. al, Backpropagation Through The Void: Optimizing Control Variates For Black-Box Gradient Estimation, 2018

# Black-box gradient estimation: discrete random variables

**Gradient-based optimization of the control variate:**

**Theorem:** The proposed  $\hat{g}_{\text{relax}}$  is unbiased for any  $c_\phi$ .

$$\frac{\partial}{\partial \phi} \mathbb{V}[\hat{g}] = \frac{\partial}{\partial \phi} \mathbb{E}[\hat{g}^2] - \underbrace{\frac{\partial}{\partial \phi} \mathbb{E}[\hat{g}]^2}_{=0} = \mathbb{E} \left[ \frac{\partial}{\partial \phi} \hat{g}^2 \right]. \quad \underbrace{[f(b) - c_\phi(\tilde{z})] \frac{\partial}{\partial \theta} \log p(b|\theta) + \frac{\partial}{\partial \theta} c_\phi(z) - \frac{\partial}{\partial \theta} c_\phi(\tilde{z})}_{\hat{g}_{\text{relax}}},$$

So, we can directly minimize the variance of a gradient estimator. We alternate between  $\theta$  and  $\phi$  updates. where  $b = H(z)$ ,  $z \sim p(z|\theta)$ ,  $\tilde{z} \sim p(z|b, \theta)$ .

**Example:** when  $p(b|\theta) = \text{Be}(\theta)$ ,  $H(z) = \mathbf{1}[z > 0]$ .

**Discrete random variables and conditional reparametrization:** let  $b$  be a discrete random variable. Introduce a "relaxed" continuous

**reparametrizable**  $z \sim p(z|\theta)$ ,  $H(z) = b$ ,  $b \sim$

$p(b|\theta)$ , where  $H(\cdot)$  is a deterministic mapping.  $v' = v[(1 - \theta)(1 - b) + \theta b]$ ,  $v \sim \mathcal{U}[0, 1]$ .



# RELAX: evaluation

## VAE with Bernoulli latent variables

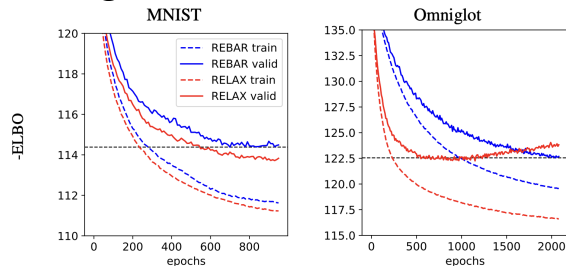
One -layer linear model

$$q(b_i|\mathbf{x}) = \sigma(\mathbf{W}_q \mathbf{x} + b_q),$$

$$p(\mathbf{x}|\mathbf{b}) = \sigma(\mathbf{W}_p \mathbf{b} + \mathbf{b}_p)$$

Dataset	Model	Concrete	NVIL	MuProp	REBAR	RELAX
MNIST	Nonlinear	-102.2	-101.5	-101.1	-81.01	<b>-78.13</b>
	linear one-layer	-111.3	-112.5	-111.7	-111.6	<b>-111.20</b>
	linear two-layer	-99.62	-99.6	-99.07	-98.22	<b>-98.00</b>
Omniglot	Nonlinear	-110.4	-109.58	-108.72	-56.76	<b>-56.12</b>
	linear one-layer	-117.23	-117.44	-117.09	-116.63	<b>-116.57</b>
	linear two-layer	-109.95	-109.98	-109.55	-108.71	<b>-108.54</b>

## Training curves



The proposed approach improved validation performance as well increased convergence speed.

# Vector Quantised - Variational AutoEncoder<sup>6</sup>

**Challenge:** the latent codes  $\mathbf{z}$  are ignored when they are paired with a powerful decoder  $p(\mathbf{x}|\mathbf{z})$  in VAE framework.

**Solution:** introduce a discrete latent model VQ-VAE.

## Posterior collapse

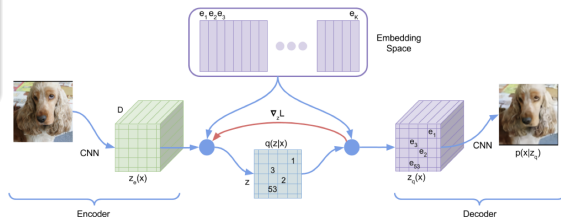
The posterior of  $\mathbf{z}$  is collapses if  $q_\phi(\mathbf{z}|\mathbf{x}) = p(\mathbf{z})$ . So, when posterior collapse occurs, it prevents the latent variable from providing meaningful summary of the dataset.

**The variational distribution** Given a latent embeddings space  $\mathbf{E} \in \mathbb{R}^{K \times D}$  and a deterministic encoder  $\mathbf{z}_e : \mathbb{R}^{\dim \mathbf{x}} \rightarrow \mathbb{R}^D$ .

Define the variational posterior by a nearest-neighbour look-up using  $\mathbf{E}$ :

$$q(z = k|\mathbf{x}) = \begin{cases} 1, & k = \arg \min_j \|\mathbf{z}_e(\mathbf{x}) - \mathbf{e}_j\|_2, \\ 0, & \text{otherwise.} \end{cases}$$

Define a latent variable  $\mathbf{z}_q = \mathbf{e}_k$ ,  $k = \arg \min_j \|\mathbf{z}_e(\mathbf{x}) - \mathbf{e}_j\|_2$ .



<sup>6</sup>van den Oord A. et. al, Neural Discrete Representation Learning, 2018

## Deriving the ELBO

$$\begin{aligned}\mathcal{L}_{\theta,\phi} &= \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})}[\log p_{\theta}(\mathbf{x}, \mathbf{z}) - \log q_{\phi}(\mathbf{z}|\mathbf{x})] \\ &= \log p_{\theta}(\mathbf{x}|\mathbf{z}_q)p(\mathbf{z}_q) \propto \log p_{\theta}(\mathbf{x}|\mathbf{z}_q) \rightarrow \max_{\phi,\theta,\mathbf{E}}.\end{aligned}$$

Note that  $\arg \max$  is not differentiable, so copy gradients from decoder input  $\mathbf{z}_q$  to encoder output  $\mathbf{z}_e$ .

**Challenge:**  $\mathbf{E}$  receive no gradients due to the straight-through gradient.

**Solution:** Vector Quantization

$$\begin{aligned}\mathcal{L}_{\text{total}}(\mathbf{x}) &= \log p_{\theta}(\mathbf{x}|\mathbf{z}_q) + \|\text{sg}(\mathbf{z}_e(\mathbf{x})) - \mathbf{e}_k\|_2^2 \\ &\quad + \beta \|\mathbf{z}_e(\mathbf{x}) - \text{sg}(\mathbf{e}_k)\|_2^2,\end{aligned}$$

where  $\text{sg}(\cdot)$  is the stop gradient operation. The last term is a *commitment loss*. The loss ensures that the outputs of the encoder do not grow.

**The prior:**

During training  $p(\mathbf{z}_q)$  kept constant and uniform. Subsequently, learn an autoregressive prior  $p(\mathbf{z})$  when there are more than one latent variable.

# Summary

- Latent Variable Model
- An introduction to Variational Autoencoders
- Semi-Implicit Variational Inference
- Continuous relaxation and Concrete distribution
- Black-box gradient estimator and RELAX
- VQ-VAE