Deep Generative Models
Lecture 3

Roman Isachenko

Ozon Masters

Spring, 2021

Recap of previous lecture

MLE problem

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} p(\mathbf{X}|\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\theta}} \prod_{i=1}^{n} p(\mathbf{x}_i|\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \log p(\mathbf{x}_i|\boldsymbol{\theta}).$$

Challenge

$p(\mathbf{x}|\boldsymbol{\theta})$ could be intractable.

LVM

Introduce latent variable $\mathbf{z}$ for each sample $\mathbf{x}$

$$p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) = p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})p(\mathbf{z}); \quad \log p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) = \log p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) + \log p(\mathbf{z}).$$

$$p(\mathbf{x}|\boldsymbol{\theta}) = \int p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) d\mathbf{z} = \int p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})p(\mathbf{z}) d\mathbf{z}.$$

Motivation

The distributions $p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})$ and $p(\mathbf{z})$ could be quite simple.

Recap of previous lecture

Incomplete likelihood maximization

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \log p(\mathbf{X}|\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\theta}} \log \sum_{i=1}^{n} \int p(\mathbf{x}_i|\mathbf{z}_i, \boldsymbol{\theta}) p(\mathbf{z}_i) d\mathbf{z}_i.$$

Variational lower bound

$$\log p(\mathbf{x}|\boldsymbol{\theta}) = \mathcal{L}(q, \boldsymbol{\theta}) + KL(q(\mathbf{z}) || p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})) \geq \mathcal{L}(q, \boldsymbol{\theta}).$$

Evidence Lower Bound (ELBO)

$$\mathcal{L}(q, \boldsymbol{\theta}) = \mathbb{E}_q \log p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) - KL(q(\mathbf{z}) || p(\mathbf{z})).$$

Instead of maximizing incomplete likelihood, maximize ELBO (equivalently minimize KL)

$$\max_{\boldsymbol{\theta}} \log p(\mathbf{x}|\boldsymbol{\theta}) \quad \rightarrow \quad \max_{q, \boldsymbol{\theta}} \mathcal{L}(q, \boldsymbol{\theta}) \equiv \min_{q, \boldsymbol{\theta}} KL(q(\mathbf{z}) || p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})).$$

Recap of previous lecture

EM algorithm (block-coordinate optimization)

- Initialize $\boldsymbol{\theta}^*$;
- E-step
$$q(\mathbf{z}) = \arg\max_q \mathcal{L}(q, \boldsymbol{\theta}^*) = \arg\min_q KL(q||p) = p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}^*);$$
    - $p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}^*)$ could be **intractable**;
    - $q(\mathbf{z})$ is different for each object $\mathbf{x}$.
- M-step
$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \mathcal{L}(q, \boldsymbol{\theta});$$
- Repeat E-step and M-step until convergence.

Amortized variational inference

Restrict a family of all possible distributions $q(\mathbf{z})$ to a particular parametric class $q(\mathbf{z}|\mathbf{x}, \phi)$ conditioned on samples $\mathbf{x}$ with parameters ϕ.

Variational EM-algorithm

ELBO

$$\log p(\mathbf{x}|\boldsymbol{\theta}) = \mathcal{L}(\phi, \boldsymbol{\theta}) + KL(q(\mathbf{z}|\mathbf{x}, \phi)||p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})) \geq \mathcal{L}(\phi, \boldsymbol{\theta}).$$

- E-step
$$\phi_k = \phi_{k-1} + \eta \nabla_{\phi} \mathcal{L}(\phi, \boldsymbol{\theta}_{k-1})\big|_{\phi=\phi_{k-1}},$$
where ϕ – parameters of variational distribution $q(\mathbf{z}|\mathbf{x}, \phi)$.
- M-step
$$\boldsymbol{\theta}_k = \boldsymbol{\theta}_{k-1} + \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}(\phi_k, \boldsymbol{\theta})\big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{k-1}},$$
where $\boldsymbol{\theta}$ – parameters of the generative distribution $p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})$.

Now all we have to do is to obtain two gradients $\nabla_{\phi} \mathcal{L}(\phi, \boldsymbol{\theta})$, $\nabla_{\boldsymbol{\theta}} \mathcal{L}(\phi, \boldsymbol{\theta})$.

Challenge

Number of samples n could be huge (we heed to derive unbiased stochastic gradients).

ELBO interpretations

$$\log p(\mathbf{x}|\boldsymbol{\theta}) = \mathcal{L}(q, \boldsymbol{\theta}) + KL(q(\mathbf{z}|\mathbf{x}, \phi)||p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})).$$

$$\mathcal{L}(q, \boldsymbol{\theta}) = \int q(\mathbf{z}|\mathbf{x}, \phi) \log \frac{p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})}{q(\mathbf{z}|\mathbf{x}, \phi)} d\mathbf{z}.$$

- Evidence minus posterior KL

$$\mathcal{L}(q, \boldsymbol{\theta}) = \log p(\mathbf{x}|\boldsymbol{\theta}) - KL(q(\mathbf{z}|\mathbf{x}, \phi)||p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})).$$

- Average negative energy plus entropy

$$\begin{aligned}\mathcal{L}(q, \boldsymbol{\theta}) &= \mathbb{E}_{q(\mathbf{z}|\mathbf{x}, \phi)} \left[\log p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) - \log q(\mathbf{z}|\mathbf{x}, \phi)\right] \\ &= \mathbb{E}_{q(\mathbf{z}|\mathbf{x}, \phi)} \log p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) + \mathbb{H}\left[q(\mathbf{z}|\mathbf{x}, \phi)\right].\end{aligned}$$

- Average reconstruction minus KL to prior

$$\begin{aligned}\mathcal{L}(q, \boldsymbol{\theta}) &= \mathbb{E}_{q(\mathbf{z}|\mathbf{x}, \phi)} \left[\log p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) + \log p(\mathbf{z}) - \log q(\mathbf{z}|\mathbf{x}, \phi)\right] \\ &= \mathbb{E}_{q(\mathbf{z}|\mathbf{x}, \phi)} \log p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) - KL(q(\mathbf{z}|\mathbf{x}, \phi)||p(\mathbf{z})).\end{aligned}$$

Monte-Carlo estimation

$$\sum_{i=1}^{n} \mathbb{E}_q f(\mathbf{z}_i) \approx n \cdot \mathbb{E}_q f(\mathbf{z}) = n \cdot \int q(\mathbf{z}) f(\mathbf{z}) d\mathbf{z} \approx n \cdot f(\mathbf{z}^*), \text{where } \mathbf{z}^* \sim q(\mathbf{z}).$$

ELBO gradients

$$\nabla_{\boldsymbol{\theta}} \sum_{i=1}^{n} \mathcal{L}_i(\boldsymbol{\phi}, \boldsymbol{\theta}) \approx n \cdot \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta}); \quad \nabla_{\boldsymbol{\phi}} \sum_{i=1}^{n} \mathcal{L}_i(\boldsymbol{\phi}, \boldsymbol{\theta}) \approx n \cdot \nabla_{\boldsymbol{\phi}} \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta})$$

ELBO

$$\mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta}) = \mathbb{E}_q \left[\log p(\mathbf{x}, \mathbf{z} | \boldsymbol{\theta}) - \log q(\mathbf{z} | \mathbf{x}, \boldsymbol{\phi}) \right] \to \max_{\boldsymbol{\phi}, \boldsymbol{\theta}}.$$

ELBO gradient (M-step, $\nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta})$)

$$\nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta}) = \int q(\mathbf{z} | \mathbf{x}, \boldsymbol{\phi}) \nabla_{\boldsymbol{\theta}} \log p(\mathbf{x} | \mathbf{z}, \boldsymbol{\theta}) d\mathbf{z} \approx$$

$$\approx \nabla_{\boldsymbol{\theta}} \log p(\mathbf{x} | \mathbf{z}^*, \boldsymbol{\theta}), \quad \mathbf{z}^* \sim q(\mathbf{z} | \mathbf{x}, \boldsymbol{\phi}).$$

ELBO gradient (E-step, $\nabla_{\boldsymbol{\phi}} \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta})$)

$$\mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta}) = \mathbb{E}_q \left[\log p(\mathbf{x}, \mathbf{z} | \boldsymbol{\theta}) - \log q(\mathbf{z} | \mathbf{x}, \boldsymbol{\phi}) \right] \to \max_{\boldsymbol{\phi}, \boldsymbol{\theta}}.$$

Challenge

Difference from M-step: density function $q(\mathbf{z}|\mathbf{x}, \boldsymbol{\phi})$ depends on the parameters $\boldsymbol{\phi}$, it is impossible to use the Monte-Carlo estimation:

$$\begin{aligned}
\nabla_{\boldsymbol{\phi}} \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta}) &= \nabla_{\boldsymbol{\phi}} \int q(\mathbf{z}|\mathbf{x}, \boldsymbol{\phi}) \left[\log p(\mathbf{x}, \mathbf{z} | \boldsymbol{\theta}) - \log q(\mathbf{z}|\mathbf{x}, \boldsymbol{\phi}) \right] d\mathbf{z} \\
&\neq \int q(\mathbf{z}|\mathbf{x}, \boldsymbol{\phi}) \nabla_{\boldsymbol{\phi}} \left[\log p(\mathbf{x}, \mathbf{z} | \boldsymbol{\theta}) - \log q(\mathbf{z}|\mathbf{x}, \boldsymbol{\phi}) \right] d\mathbf{z}
\end{aligned}$$

Solution

Reparametrization trick for $q(\mathbf{z}|\mathbf{x}, \boldsymbol{\phi})$ allows the expectation to become independent of parameters $\boldsymbol{\phi}$.

Reparametrization trick

$$f(\xi) = \mathbb{E}_{q(\eta|\xi)} h(\eta) = \int q(\eta|\xi) h(\eta) d\eta$$

Let $\eta = g(\xi, \epsilon)$, where g is a deterministic function, ϵ is a random variable with a density function $r(\epsilon)$.

$$f(\xi) = \int q(\eta|\xi) h(\eta) d\eta = \int r(\epsilon) h(g(\xi, \epsilon)) d\epsilon \approx h(g(\xi, \epsilon^*)), \quad \epsilon^* \sim r(\epsilon).$$

Examples

- $r(\epsilon) = \mathcal{N}(\epsilon|0, 1)$, $\eta = \sigma \cdot \epsilon + \mu$, $q(\eta|\xi) = \mathcal{N}(\eta|\mu, \sigma^2)$, $\xi = [\mu, \sigma]$.
- $\boldsymbol{\epsilon}^* \sim r(\boldsymbol{\epsilon}), \quad \mathbf{z} = g(\mathbf{x}, \boldsymbol{\epsilon}, \phi), \quad \mathbf{z} \sim q(\mathbf{z}|\mathbf{x}, \phi)$

$$\begin{aligned} \nabla_\phi \int q(\mathbf{z}|\mathbf{x}, \phi) f(\mathbf{z}) d\mathbf{z} &= \nabla_\phi \int r(\boldsymbol{\epsilon}) f(\mathbf{z}) d\boldsymbol{\epsilon} \\ &= \int r(\boldsymbol{\epsilon}) \nabla_\phi f(g(\mathbf{x}, \boldsymbol{\epsilon}, \phi)) d\boldsymbol{\epsilon} \approx \nabla_\phi f(g(\mathbf{x}, \boldsymbol{\epsilon}^*, \phi)) \end{aligned}$$

ELBO gradient (E-step, $\nabla_{\phi}\mathcal{L}(\phi, \boldsymbol{\theta})$)

$$\nabla_{\phi}\mathcal{L}(\phi, \boldsymbol{\theta}) = \nabla_{\phi} \int q(\mathbf{z}|\mathbf{x}, \phi) \left[\log p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) - \log q(\mathbf{z}|\mathbf{x}, \phi)\right] d\mathbf{z}$$
$$= \int r(\boldsymbol{\epsilon})\nabla_{\phi}\big[\log p(\mathbf{x}, g(\mathbf{x}, \boldsymbol{\epsilon}, \phi)|\boldsymbol{\theta}) - \log q\big(g(\mathbf{x}, \boldsymbol{\epsilon}, \phi)|\mathbf{x}, \phi\big)\big] d\boldsymbol{\epsilon}$$
$$\approx \nabla_{\phi}\big[\log p(\mathbf{x}, g(\mathbf{x}, \boldsymbol{\epsilon}^*, \phi)|\boldsymbol{\theta}) - \log q\big(g(\mathbf{x}, \boldsymbol{\epsilon}^*, \phi)|\mathbf{x}, \phi\big)\big]$$

Variational assumption

$$r(\boldsymbol{\epsilon}) = \mathcal{N}(0, \mathbf{I}); \quad q(\mathbf{z}|\mathbf{x}, \phi) = \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{\phi}}(\mathbf{x}), \boldsymbol{\sigma}^2_{\boldsymbol{\phi}}(\mathbf{x})).$$
$$\mathbf{z} = g(\mathbf{x}, \boldsymbol{\epsilon}, \phi) = \boldsymbol{\sigma}_{\boldsymbol{\phi}}(\mathbf{x}) \cdot \boldsymbol{\epsilon} + \boldsymbol{\mu}_{\boldsymbol{\phi}}(\mathbf{x}).$$

Here $\boldsymbol{\mu}_{\boldsymbol{\phi}}(\cdot), \boldsymbol{\sigma}_{\boldsymbol{\phi}}(\cdot)$ are parameterized functions (outputs of neural network).

If we could calculate $\log p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})$ and $\log q(\mathbf{z}|\mathbf{x}, \phi)$, we are done. Could we?

ELBO gradient (E-step, $\nabla_{\phi}\mathcal{L}(\phi, \boldsymbol{\theta})$)

$$\nabla_{\phi}\mathcal{L}(\phi, \boldsymbol{\theta}) \approx \nabla_{\phi}\left[\log p(\mathbf{x}, g(\mathbf{x}, \boldsymbol{\epsilon}^*, \phi)|\boldsymbol{\theta}) - \log q\big(g(\mathbf{x}, \boldsymbol{\epsilon}^*, \phi)|\mathbf{x}, \phi\big)\right]$$

First term

$$\log p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) = \log p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) + \log p(\mathbf{z}).$$

- $p(\mathbf{z})$ – prior distribution on latent variables $\mathbf{z}$. We could specify any distribution that we want. Let say $p(\mathbf{z}) = \mathcal{N}(0, \mathbf{I})$.
- $p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})$ - generative distibution. Since it is a parameterized function let it be neural network with parameters $\boldsymbol{\theta}$.

Second term

Function $\mathbf{z} = g(\mathbf{x}, \epsilon, \phi) = \boldsymbol{\sigma}_{\phi}(\mathbf{x}) \cdot \boldsymbol{\epsilon} + \boldsymbol{\mu}_{\phi}(\mathbf{x})$ is invertible.

$$q(\mathbf{z}|\mathbf{x}, \phi) = r(\boldsymbol{\epsilon}) \cdot \left|\frac{\partial \boldsymbol{\epsilon}}{\partial \mathbf{z}}\right| \quad \Rightarrow \quad \log q(\mathbf{z}|\mathbf{x}, \phi) = \log r(\boldsymbol{\epsilon}) - \sum_{i=1}^{d} \log\left[\boldsymbol{\sigma}_{\phi}(\mathbf{x})\right]_i$$

Variational autoencoder (VAE)

Final algorithm

- pick $i \sim U[1, n]$;
- compute a stochastic gradient w.r.t. $\boldsymbol{\phi}$

$$\nabla_{\boldsymbol{\phi}} \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta}) \approx \nabla_{\boldsymbol{\phi}} \Big[\log p(\mathbf{x}, g(\mathbf{x}, \boldsymbol{\epsilon}^*, \boldsymbol{\phi}) | \boldsymbol{\theta}) - \\ - \log q\big(g(\mathbf{x}, \boldsymbol{\epsilon}^*, \boldsymbol{\phi}) | \mathbf{x}, \boldsymbol{\phi}\big) \Big], \quad \boldsymbol{\epsilon}^* \sim r(\boldsymbol{\epsilon});$$

- compute a stochastic gradient w.r.t. $\boldsymbol{\theta}$

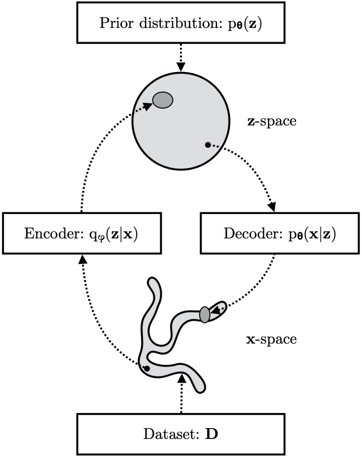
$$\nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta}) \approx \nabla_{\boldsymbol{\theta}} \log p(\mathbf{x} | \mathbf{z}^*, \boldsymbol{\theta}), \quad \mathbf{z}^* \sim q(\mathbf{z} | \mathbf{x}, \boldsymbol{\phi});$$

- update $\boldsymbol{\theta}, \boldsymbol{\phi}$ according to the selected optimization method (SGD, Adam, RMSProp):

$$\boldsymbol{\phi} := \boldsymbol{\phi} + \eta \nabla_{\boldsymbol{\phi}} \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta}),$$
$$\boldsymbol{\theta} := \boldsymbol{\theta} + \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\theta}).$$

Variational autoencoder (VAE)

- VAE learns stochastic mapping between $\mathbf{x}$-space, from complicated distribution $\pi(\mathbf{x})$, and a latent $\mathbf{z}$-space, with simple distribution.
- The generative model learns a joint distribution $p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) = p(\mathbf{z})p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})$, with a prior distribution $p(\mathbf{z})$, and a stochastic decoder $p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})$.
- The stochastic encoder $q(\mathbf{z}|\mathbf{x}, \phi)$ (inference model), approximates the true but intractable posterior $p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})$ of the generative model.



Kingma D. P., Welling M. An introduction to variational autoencoders, 2019

Variational Autoencoder

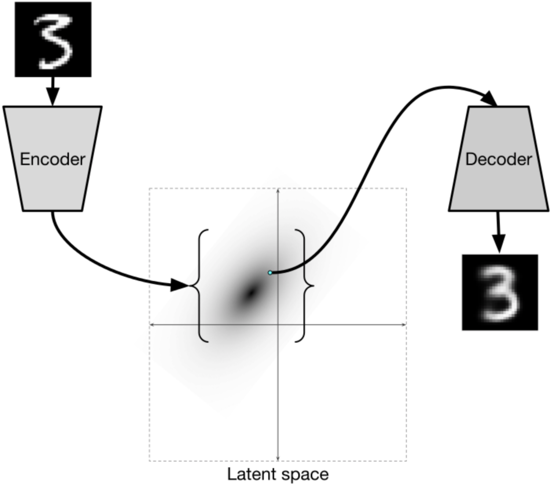


image credit: http://ijdykeman.github.io/ml/2016/12/21/cvae.html

Variational autoencoder (VAE)

- Encoder $q(\mathbf{z}|\mathbf{x}, \phi) = \text{NN}_e(\mathbf{x}, \phi)$ outputs $\boldsymbol{\mu}_\phi(\mathbf{x})$ and $\boldsymbol{\sigma}_\phi(\mathbf{x})$.
- Decoder $p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) = \text{NN}_d(\mathbf{z}, \boldsymbol{\theta})$ outputs parameters of the sample distribution.

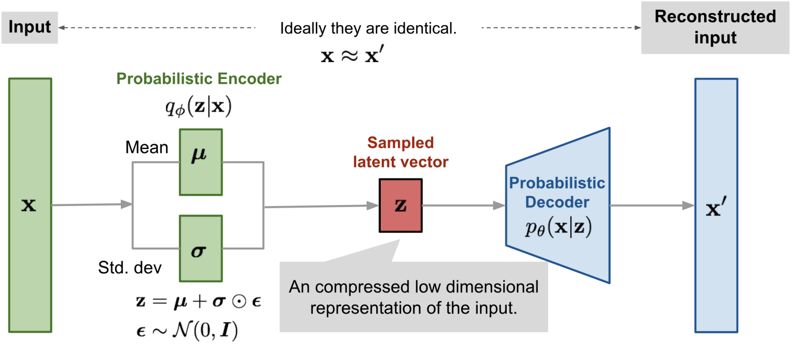


image credit: https://lilianweng.github.io/lil-log/2018/08/12/from-autoencoder-to-beta-vae.html

Variational Autoencoder

Generated images for latent objects $\mathbf{z}$ sampled from prior $\mathcal{N}(0, \mathbf{I})$

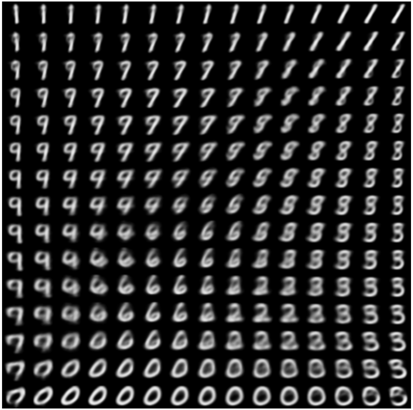


image credit: https://habr.com/ru/post/331664

Bayesian framework

Posterior distribution

$$p(\boldsymbol{\theta}|\mathbf{X}) = \frac{p(\mathbf{X}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{X})} = \frac{p(\mathbf{X}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{\int p(\mathbf{X}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}}$$

Bayesian inference

$$p(\mathbf{x}|\mathbf{X}) = \int p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{X})d\boldsymbol{\theta}$$

Maximum a posteriori (MAP) estimation

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta}|\mathbf{X}) = \arg\max_{\boldsymbol{\theta}} \big(\log p(\mathbf{X}|\boldsymbol{\theta}) + \log p(\boldsymbol{\theta})\big)$$

MAP inference

$$p(\mathbf{x}|\mathbf{X}) = \int p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{X})d\boldsymbol{\theta} = \int p(\mathbf{x}|\boldsymbol{\theta})\delta(\boldsymbol{\theta} - \boldsymbol{\theta}^*)d\boldsymbol{\theta} \approx p(\mathbf{x}|\boldsymbol{\theta}^*).$$

VAE as Bayesian model

Posterior distribution

$$p(\boldsymbol{\theta}|\mathbf{X}) = \frac{p(\mathbf{X}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{X})}$$

ELBO

$$\begin{aligned}\log p(\boldsymbol{\theta}|\mathbf{X}) &= \log p(\mathbf{X}|\boldsymbol{\theta}) + \log p(\boldsymbol{\theta}) - \log p(\mathbf{X}) \\ &= \mathcal{L}(q, \boldsymbol{\theta}) + KL(q||p) + \log p(\boldsymbol{\theta}) - \log p(\mathbf{X}) \\ &\geq [\mathcal{L}(q, \boldsymbol{\theta}) + \log p(\boldsymbol{\theta})] - \log p(\mathbf{X}).\end{aligned}$$

EM-algorithm

- E-step

$$q(\mathbf{z}) = \arg\max_q \mathcal{L}(q, \boldsymbol{\theta}^*) = \arg\min_q KL(q||p) = p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}^*);$$

- M-step

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \left[\mathcal{L}(q, \boldsymbol{\theta}) + \log p(\boldsymbol{\theta})\right].$$

VAE limitations

- Poor variational posterior distribution (inference model encoder)
$$q(\mathbf{z}|\mathbf{x}, \phi) = \mathcal{N}(\mathbf{z}|\boldsymbol{\mu}_{\boldsymbol{\phi}}(\mathbf{x}), \boldsymbol{\sigma}^2_{\boldsymbol{\phi}}(\mathbf{x})).$$
- Poor prior distribution
$$p(\mathbf{z}) = \mathcal{N}(0, \mathbf{I}).$$
- Poor probabilistic model (generative model, decoder)
$$p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{z}), \boldsymbol{\sigma}^2_{\boldsymbol{\theta}}(\mathbf{z})).$$
- Loose lower bound
$$\log p(\mathbf{x}|\boldsymbol{\theta}) - \mathcal{L}(q, \boldsymbol{\theta}) = (?).$$

Summary

- Amortized inference allows to efficiently compute stochastic gradients for ELBO and to use deep neural networks for $q(\mathbf{z}|\mathbf{x}, \phi)$ and $p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})$.

- ELBO gradients are computed using Monte-Carlo estimation.

- The reparametrization trick allows to get unbiased gradients w.r.t to a variational posterior distribution.

- The VAE model is an LVM with an encoder network for $q(\mathbf{z}|\mathbf{x}, \phi)$ and a decoder network for $p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})$.

- VAE is not a "true" bayesian model since parameters $\boldsymbol{\theta}$ do not have a prior distribution.

- Standart VAE has several limitations that we will address later in the course.