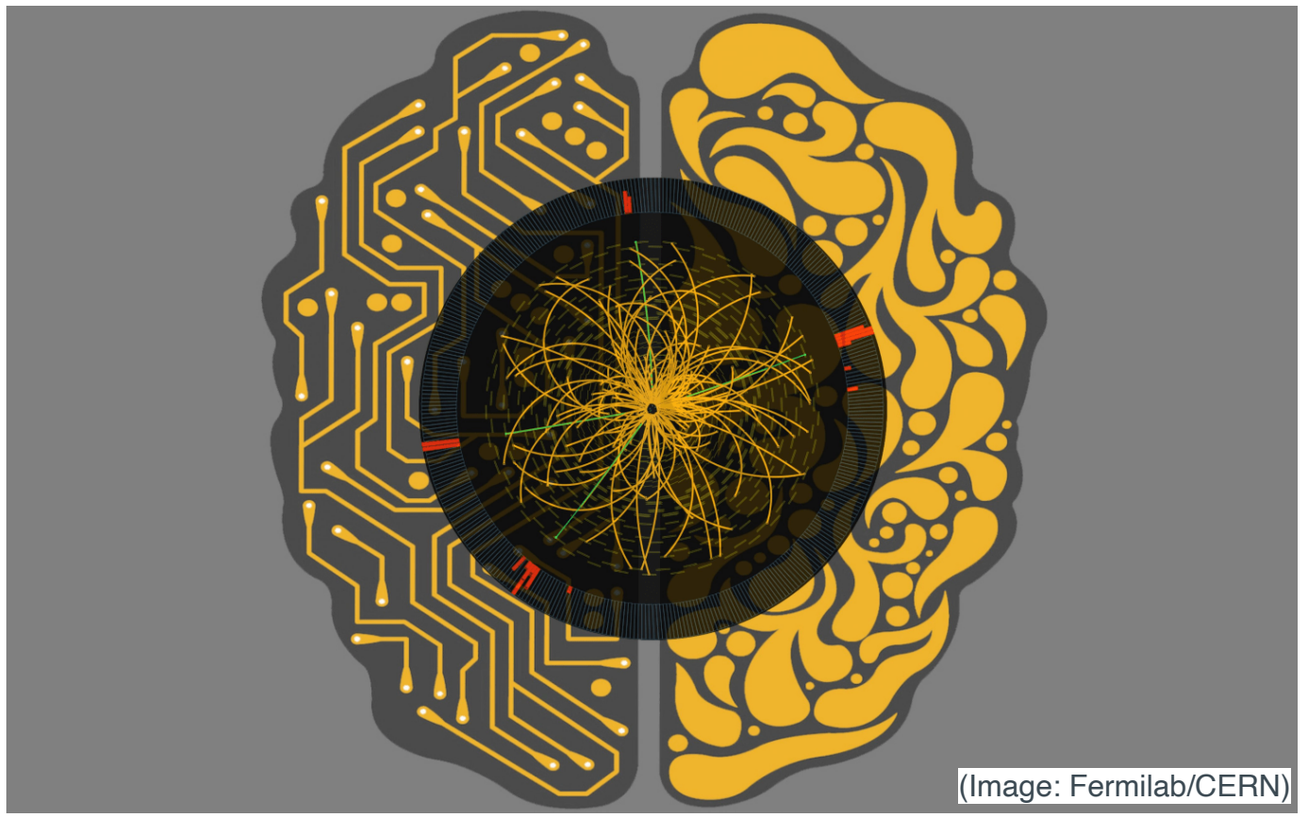
PHY 835: Collider Physics Phenomenology

Machine Learning in Fundamental Physics

Gary Shiu, UW-Madison



Lecture 17: Variational Autoencoder

Recap of Lecture 16

- Generative Adversarial Networks (GANs)
- Limitations of Maximizing Likelihood
- Adversarial Learning
- Wasserstein Loss and WGAN

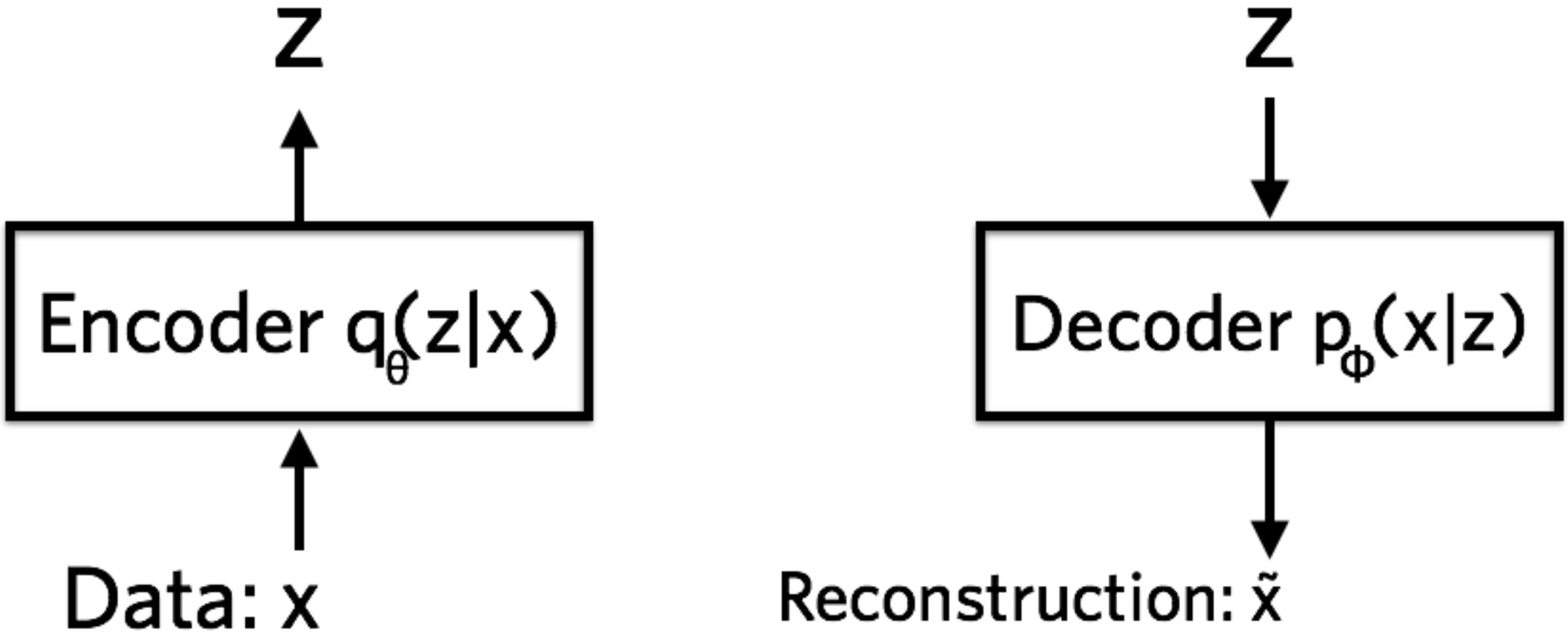
Outline for today

- Variational Autoencoder (VAE)
  - Neural Net Perspective
  - Probability Model Perspective
    - Connecting the two perspectives
- Reparametrization trick

References: 1803.08823, Deep Learning Book
https://arxiv.org/abs/1606.05908
https://arxiv.org/abs/1312.6114

VAE: Neural Net Perspective

- In neural net language, a variational autoencoder consists of an encoder, a decoder, and a loss function.



- The latent (hidden) space z has a much smaller dimension than the data space x (hence the name “bottleneck”).

VAE: Neural Net Perspective

- The encoder is a neural net with weights θ and the decoder is another neural net with weights ϕ.

- As an example, consider a 28 by 28-pixel black and white photo of a handwritten number. x = 784 dimensional vector with 0 or 1 entries.

- The decoder ‘decodes’ the real-valued numbers in latent space z into real-valued numbers between 0 and 1 (Bernoulli distribution).

- Information cannot be fully transmitted. How much information is lost?

- Reconstruction log-likelihood $\log p_\phi(x \mid z)$ measures how effectively the decoder has learned to reconstruct an input image x given its latent representation z.

VAE: Neural Net Perspective

- The loss function of the variational autoencoder is the negative log-likelihood with a regularizer.

- Because there are no global representations that are shared by all datapoints, we can decompose the loss function into only terms that depend on a single datapoint l_i. The total loss is the sum of l_i:

$$l_i(\theta, \phi) = -\mathbb{E}_{z \sim q_\theta(z|x_i)}[\log p_\phi(x_i \mid z)] + \mathbb{KL}(q_\theta(z \mid x_i) \mid\mid p(z))$$

- The first term is the **reconstruction loss**, with expectation taken with respect to the encoder's distribution over the representations. This term encourages the decoder to learn to reconstruct the data.

- The second term is a regularizer (derived later). The KL divergence measures how much information is lost when using q to represent p.

VAE: Neural Net Perspective

- In the variational autoencoder, p is specified as a standard Normal distribution with mean zero and variance one, or $p(z) = \text{Norm}(0,1)$.

- The regularizer means 'keep the representation z sufficiently diverse'. Without the regularizer, the encoder could learn to cheat & give each datapoint a representation in a different region of Euclidean space.

- Otherwise, two images of the same number (say a 2 written by 2_{Alice} and 2_{Bob}) could end up with different representations z_{Alice} and z_{Bob}.

- The regularizer has the effect of keeping similar numbers' representations close together.

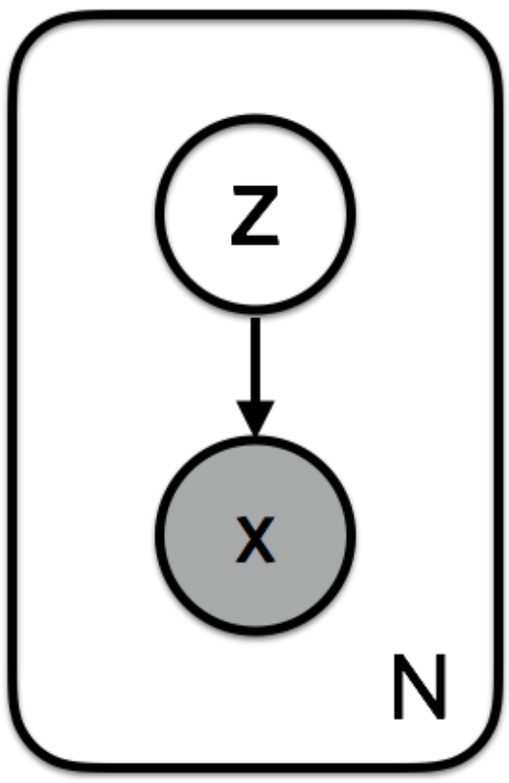
- We train the variational autoencoder using gradient descent to optimize the loss with respect to the parameters of the encoder & decoder θ & ϕ.

VAE: Probability Model Perspective

- In the probability model framework, a variational autoencoder contains a specific probability model of data x and latent variables z.
- The joint probability of the model:

$$p(x, z) = p(x \mid z)p(z).$$

- The decoder can be graphically represented as follows:



For each datapoint i:

- Draw latent variables $z_i \sim p(z)$
- Draw datapoint $x_i \sim p(x \mid z)$

VAE: Probability Model Perspective

- The latent variables are drawn from a **prior** $p(z)$. The data x have a likelihood $p(x \mid z)$ that is conditioned on latent variables z.
- The model defines a joint probability distribution $p(x, z)$:

$$p(x, z) = p(x \mid z)p(z)$$

- The goal is to infer good values of the latent variables given observed data, or to calculate the posterior $p(z \mid x)$. Using Bayes' theorem:

$$p(z \mid x) = \frac{p(x \mid z)p(z)}{p(x)}$$

- $p(x)$ is called the evidence, and we can calculate it by marginalizing out the latent variables: $p(x) = \int dx p(x \mid z)p(z)dz$.
- Unfortunately, this integral requires exponential time to compute as it needs to be evaluated over all configurations of latent variables.

VAE: Probability Model Perspective

- Variational inference approximates the posterior with a family of distributions $q_\lambda(z \mid x)$.
- The variational parameter λ indexes the family of distributions. For example, if q were Gaussian, it would be the mean and variance of the latent variables for each datapoint $\lambda_{x_i} = (\mu_{x_i}, \sigma^2_{x_i})$.
- We can use the KL divergence to measure how well our variational posterior $q_\lambda(z \mid x)$ approximates the true posterior $p(z \mid x)$.

$$\mathbb{KL}(q_\lambda(z \mid x) \,||\, p(z \mid x)) =$$

$$\mathbf{E}_q[\log q_\lambda(z \mid x)] - \mathbf{E}_q[\log p(x, z)] + \log p(x)$$

- Our goal is to find the variational parameters λ that minimize this divergence. The optimal approximate posterior is thus

$$q^*_\lambda(z \mid x) = \arg\min_\lambda \mathbb{KL}(q_\lambda(z \mid x) \,||\, p(z \mid x)).$$

VAE: Probability Model Perspective

- This is impossible to compute directly because the pesky evidence $p(x)$ (which is intractable) appears in the divergence.
- Introduce a new function:

$$ELBO(\lambda) = \mathbf{E}_q[\log p(x, z)] - \mathbf{E}_q[\log q_\lambda(z \mid x)]$$

- We can combine this with the KL divergence & rewrite the evidence:

$$\log p(x) = ELBO(\lambda) + \mathbb{KL}(q_\lambda(z \mid x) \,||\, p(z \mid x))$$

- Since the KL divergence is positive semi-definite, minimizing the KL divergence is equivalent to maximizing the ELBO.
- ELBO = Evidence Lower BOund allows us to do approximate posterior inference in a computationally tractable way.

VAE: Probability Model Perspective

- In the VAE model, there are only local latent variables (no datapoint shares its latent z with the latent variable of another datapoint).
- We can decompose the ELBO into a sum where each term depends on a single datapoint. This allows us to use stochastic gradient descent with respect to the parameters λ (which are shared).
- The ELBO for a single datapoint in the VAE is

$$ELBO_i(\lambda) = \mathbb{E}q_\lambda(z \mid x_i)[\log p(x_i \mid z)] - \mathbb{KL}(q_\lambda(z \mid x_i) \mid\mid p(z)).$$

- To see that this is equivalent to our previous definition of the ELBO, expand the log joint into the prior and likelihood terms and use the product rule for the logarithm.

Connecting the two perspectives

- To make the connection to NN language, parametrize the approximate posterior $q_\theta(z \mid x, \lambda)$ with an *inference network* (**encoder**).
- We parametrize the likelihood $p(x \mid z)$ with a *generative network* (or **decoder**) that takes latent variables and outputs parameters to the data distribution $p_\phi(x \mid z)$.
- We optimize the model parameters to maximize the ELBO using SGD:

$$ELBO_i(\theta, \phi) = \mathbb{E}q_\theta(z \mid x_i)[\log p_\phi(x_i \mid z)] - \mathbb{KL}(q_\theta(z \mid x_i) \mid\mid p(z))$$

- This evidence lower bound is the negative of the loss function for VAE we discussed from the NN perspective, $ELBO_i(\theta, \phi) = -l_i(\theta, \phi)$.
- The probability model approach makes clear why the “reconstruction cost” and “regularizer” terms exist: to minimize the KL divergence between the approximate posterior $q_\lambda(z \mid x)$ and model posterior $p(z \mid x)$.

Connecting the two perspectives

- What about the model parameters?
- The term ‘variational inference’ usually refers to maximizing the ELBO with respect to the variational parameters λ.
- We can also maximize the ELBO with respect to the model parameters ϕ (e.g. the weights and biases of the generative NN parameterizing the likelihood). This technique is called variational EM (expectation maximization) discussed earlier.
- The recipe for variational inference involves defining:
  - a probability model p of latent variables and data
  - a variational family q for the latent variables to approximate our posterior
  - Then we used the variational inference algorithm to learn the variational parameters (gradient ascent on the ELBO to learn λ).

Reparametrization Trick

- Maximizing ELBO w.r.t. λ is tricky because it appears in both terms:

$$ELBO_i(\lambda) = \mathbb{E}q_\lambda(z \mid x_i)[\log p(x_i \mid z)] - \mathbb{KL}(q_\lambda(z \mid x_i) \mid\mid p(z)).$$

- Use backpropagation to calculate the gradient of the first term?
- Change of variables (**reparametrization trick** **[Kingma and Welling, 13]**).
- Idea: express the random variable $z \sim q_\theta(z \mid x)$ as some differentiable and invertible transformation of random variable ϵ:

$$\mathbf{z} = g(\boldsymbol{\epsilon}, \theta, \mathbf{x}),$$

where the distribution of ϵ is independent of x and θ. We can replace:

$$\mathbb{E}_{q_\theta(\mathbf{z}|\mathbf{x})}[f(\mathbf{z})] = \mathbb{E}_{p_\epsilon}[f(\mathbf{z})].$$

Reparametrization Trick

- Evaluating the derivative of the expectation becomes straightforward:

$$\nabla_\theta \mathbb{E}_{q_\theta(\mathbf{z}|\mathbf{x})}[f(\mathbf{z})] \sim \mathbb{E}_{p_\epsilon}[\nabla_\theta f(\mathbf{z})].$$

- The Jacobian of this change of variables:

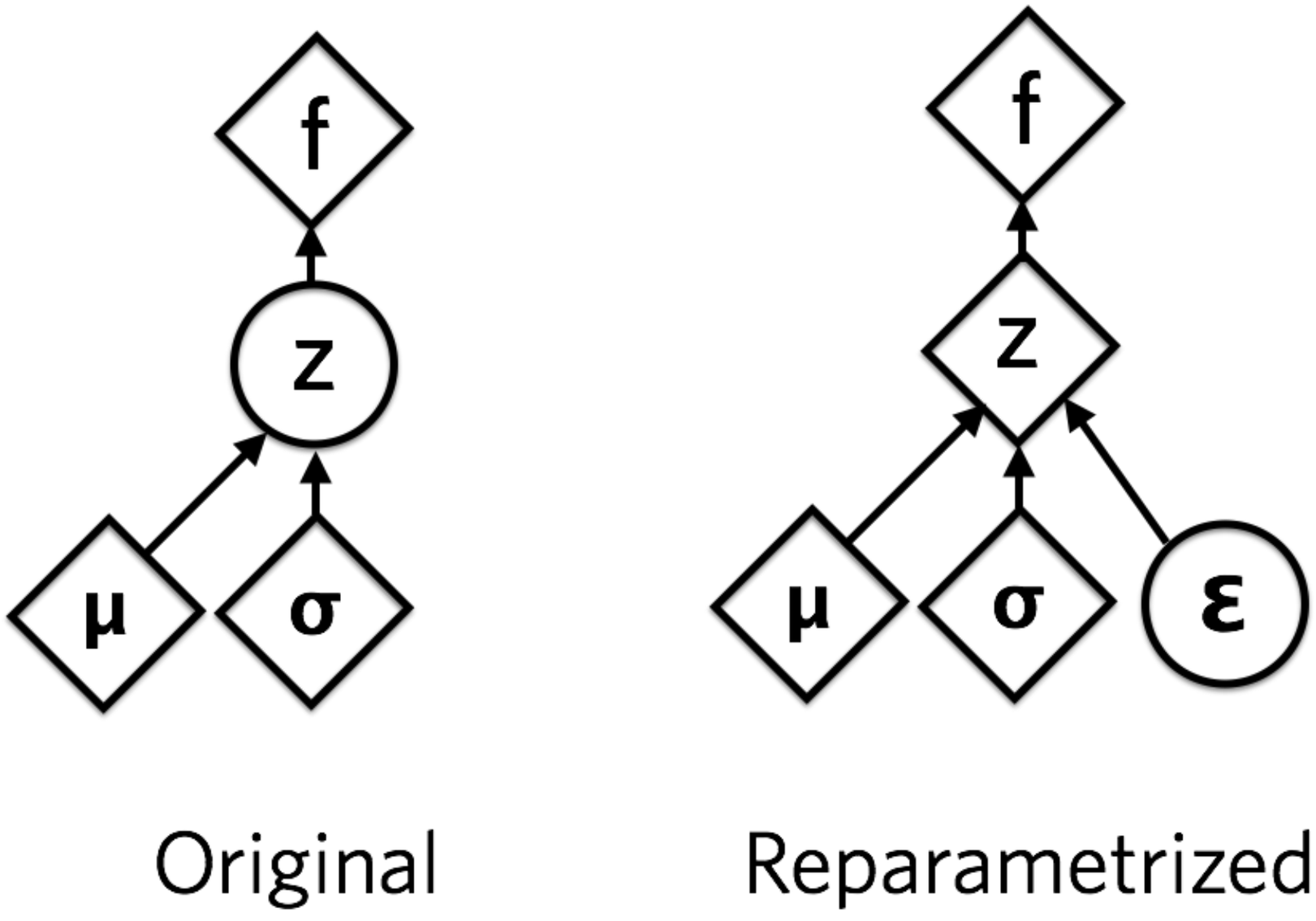
$$d_\theta(\mathbf{x}, \theta) = \mathrm{Det}\left|\frac{\partial \mathbf{z}}{\partial \boldsymbol{\epsilon}}\right|$$

- The distributions are related by:

$$\log q_\theta(\mathbf{z}|\mathbf{x}) = \log p(\boldsymbol{\epsilon}) - \log d_\theta(\mathbf{x}, \theta).$$

- We can now use backpropagation on the full ELBO objective function.

Reparametrization Trick



Original Reparametrized

Diamonds indicate deterministic dependencies, circles indicate random variables.

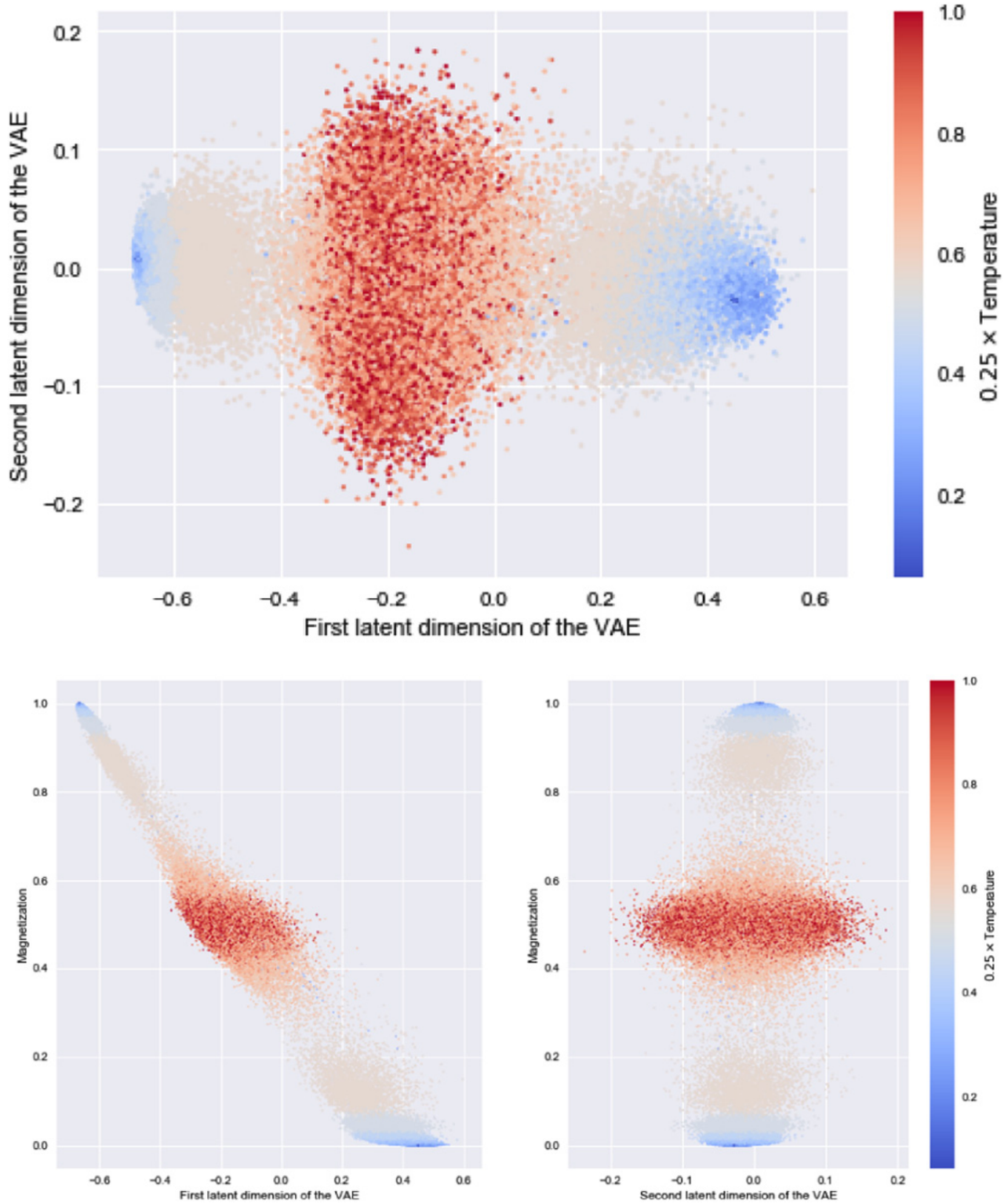
Training VAE

- A common problem in training VAEs by stochastic optimization of the ELBO is that it often gets stuck in undesirable local minima.
- The reason is that the ELBO can be improved in two qualitatively different ways: by minimizing the reconstruction error or by making the posterior distribution $q_\theta(z|x)$ to be close to $p(z)$.
- For complex dataset, at the beginning of training when the reconstruction is extremely poor, the model quickly learns to make $q(z|x) \approx p(z)$ and gets stuck in this local minimum.
- Modify the ELBO objective:

$$\mathbb{E}_{q_\theta(z|x)}[\log p_\phi(x,z)] - \beta D_{KL}\left(q_\theta(z|x)||p(z)\right)$$

with β slowly annealed from 0 to 1 (**[Bowman et al, 15]; [Sonderby et al, 16]**). An alternative regularization is the "method of free bits" **[Kingma et al, 17]**.

Training VAE for Ising Models



https://physics.bu.edu/~pankajm/ML-Notebooks/HTML/NB20_CXVII-Keras_VAE_ising.html

Summary

- Variational Autoencoder (VAE)
  - Neural Net Perspective
  - Probability Model Perspective
    - Connecting the two perspectives
- Reparametrization trick