Deep Generative Models: Variational Auto Encoders

Fall Semester 2025

René Vidal
Director of the Center for Innovation in Data Engineering and Science (IDEAS)
Rachleff University Professor, University of Pennsylvania
Amazon Scholar & Chief Scientist at NORCE

The story up till now

- **Step 1**: We set out with our original goal of learning a model p_θ that gives maximum likelihood to our datapoints $\boldsymbol{x}_i$

- **Step 2**: We introduced latent variables $\boldsymbol{z}$ such that $\boldsymbol{z} \sim p(\boldsymbol{z})$ and $\boldsymbol{x} \mid \boldsymbol{z} \sim p(\boldsymbol{x} \mid \boldsymbol{z})$, which gave us the marginalization $p(\boldsymbol{x}) = \int p(\boldsymbol{x} \mid \boldsymbol{z}) p(\boldsymbol{z})\, d\boldsymbol{z}$
  - Step 2a: When we assumed $p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}; \boldsymbol{0}, \boldsymbol{I})$, and $p(\boldsymbol{x} \mid \boldsymbol{z}) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{W}\boldsymbol{z} + \boldsymbol{b}, \sigma^2 \boldsymbol{I})$, we could solve for $(\boldsymbol{W}, \boldsymbol{b}, \sigma^2)$ in closed form! This gave us PPCA.

- **Step 3**: We set up variational inference because sadly, not everything in life is Gaussian and linear. This gave us a new Evidence Lower Bound (ELBO) objective:

$$\max_\theta \sum_{i=1}^{N} \log p_\theta(\boldsymbol{x}_i) = \max_\theta \max_{q(\cdot|\boldsymbol{x}_i), \forall i} \sum_{i=1}^{N} \int q(\boldsymbol{z}|\boldsymbol{x}_i) \log \frac{p_\theta(\boldsymbol{x}_i, \boldsymbol{z})}{q\,(\boldsymbol{z}|\boldsymbol{x}_i)}\, dz$$

  - Step 3a: When the integral is easy to evaluate, we can alternate between optimizing w.r.t. θ with $q(\boldsymbol{x} \mid \boldsymbol{z})$ fixed and vice versa, leading to the Expectation Maximization (EM) algorithm.

The story continues with Variational Autoencoders

- Before introducing VAEs formally, let us decompose the ELBO further

$$\max_{\theta} \log p_{\theta}(\boldsymbol{x}_i) = \max_{\theta} \max_{q} \sum_{i=1}^{N} \int q(\boldsymbol{z}|\boldsymbol{x}_i) \log \frac{p_{\theta}(\boldsymbol{x}_i, \boldsymbol{z})}{q(\boldsymbol{z}|\boldsymbol{x}_i)} d\boldsymbol{z}$$

$$\geq \max_{\theta,\psi} \sum_{i} \mathbb{E}_{\boldsymbol{z}\sim q_{\psi}(\boldsymbol{z}\,|\boldsymbol{x}_i)} \log \frac{p_{\theta}(\boldsymbol{x}_i, \boldsymbol{z})}{q_{\psi}(\boldsymbol{z}\,|\boldsymbol{x}_i)}$$

Evidence Lower Bound (ELBO)

$$= \max_{\theta,\psi} \sum_{i} \mathbb{E}_{\boldsymbol{z}\sim q_{\psi}(\boldsymbol{z}\,|\boldsymbol{x}_i)} \log p_{\theta}(\boldsymbol{x}_i|\boldsymbol{z}) \ \frac{p(\boldsymbol{z})}{q_{\psi}(\boldsymbol{z}\,|\boldsymbol{x}_i)}$$

$$= \max_{\theta,\psi} \sum_{i} \ \mathbb{E}_{\boldsymbol{z}\sim q_{\psi}(\boldsymbol{z}\,|\boldsymbol{x}_i)} \log p_{\theta}(\boldsymbol{x}_i|\boldsymbol{z}) + \mathbb{E}_{\boldsymbol{z}\sim q_{\psi}(\boldsymbol{z}|\boldsymbol{x}_i)} \log \frac{p(\boldsymbol{z})}{q_{\psi}(\boldsymbol{z}|\boldsymbol{x}_i)}$$

$$-D_{KL}(q_{\psi}(z|x)||p(z))$$

Variational AutoEncoders (VAEs): Setup

- We have three models we need to define for VAE model
  1. Inference model $q_\psi(\boldsymbol{z} \mid \boldsymbol{x})$: We will define as $q_\psi(\boldsymbol{z} \mid \boldsymbol{x}) = \mathcal{N}(\boldsymbol{z};\ \mu_\psi(\boldsymbol{x}), \sigma_\psi^2(\boldsymbol{x})\boldsymbol{I})$, i.e., a normal distribution with learned mean and covariance

Datapoint
$\boldsymbol{x}$

Inference Model
$q_\psi(\boldsymbol{z} \mid \boldsymbol{x})$

Can be any NN

$\mu_\psi(\boldsymbol{x})$

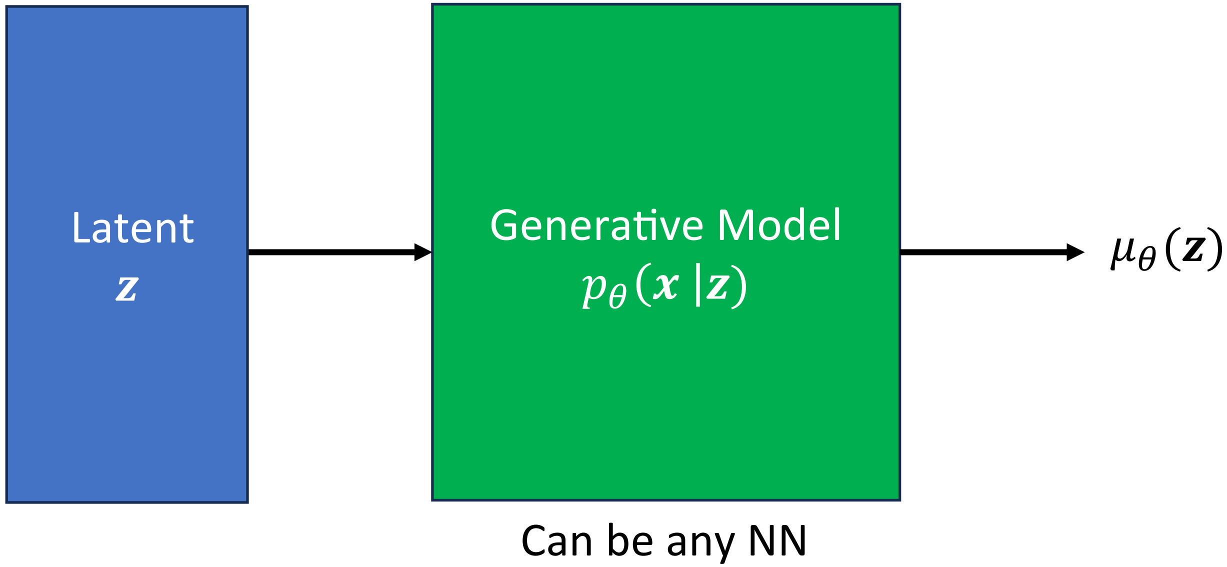
$\log \sigma_\psi^2(\boldsymbol{x})$

Ensures variance is always positive and improves stability of training!

  2. Prior model $p(\boldsymbol{z})$: We will define prior for latent variables as $p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}; \boldsymbol{0}, \boldsymbol{I})$

Variational AutoEncoders (VAEs): Setup

- We have three models we need to define for VAE model
  3. Generative model $p_\theta(\mathbf{x} \mid \mathbf{z})$: We will define as
     - $p_\theta(\boldsymbol{x} \mid \boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}; \mu_\theta(\boldsymbol{z}), \eta^2 \boldsymbol{I})$, i.e., a normal distribution with learned mean and variance
     - $p_\theta(\boldsymbol{x} \mid \boldsymbol{z}) = Cat(\boldsymbol{z}; \pi_\theta(\boldsymbol{z}))$, i.e., a categorical distribution with learned class probabilities



- Note this can be defined in many different ways, yielding different models (such as a categorical distribution over 255 values of each pixel)

Variational Autoencoders: Training

Datapoint x

μ

$\log \sigma^2$

Inference Model
$q_\psi(\boldsymbol{z} \mid \boldsymbol{x})$

Sample
$\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{z} \mid \mu_\psi(\boldsymbol{x}), \sigma_\psi^2(\boldsymbol{x}) I)$

Generative Model
$p_\theta(\boldsymbol{x} \mid \boldsymbol{z})$

$\mu_\theta(\boldsymbol{z})$

ELBO Objective

$$\mathbb{E}_{\boldsymbol{z} \sim q_\psi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x} \mid \boldsymbol{z}) - KL\left(q_\psi(\boldsymbol{z} \mid \boldsymbol{x}) \,||\, p(\boldsymbol{z})\right)$$

Variational Autoencoders after Training

- Suppose we have learned VAE using the ELBO loss (details to follow).
- Then, as a generative model, we just sample $\boldsymbol{z} \sim p(\boldsymbol{z})$ and use the fixed generative model $p_\theta(\boldsymbol{x} \mid \boldsymbol{z})$

Sample
$\boldsymbol{z} \sim p(\boldsymbol{z})$
→
Generative Model
$p_\theta(\boldsymbol{x} \mid \boldsymbol{z})$
→ $\mu_\theta(\boldsymbol{z})$ →
Sample
$\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{x} \mid \mu_\theta(\boldsymbol{z}), \eta \boldsymbol{I})$
→

Computing the ELBO Loss

$$L_{\theta,\psi}(\boldsymbol{x}) \coloneqq \boxed{\mathbb{E}_{\boldsymbol{z}\sim q_\psi(\boldsymbol{z}\,|\boldsymbol{x})} \log p_\theta(\boldsymbol{x} \mid \boldsymbol{z})} + \boxed{\mathbb{E}_{\boldsymbol{z}\sim q_\psi(\boldsymbol{z}\,|\,\boldsymbol{x})} \log \frac{p(\boldsymbol{z})}{q_\psi(\boldsymbol{z} \mid \boldsymbol{x})}}$$

- Term 1 (Reconstruction Error)
  - Because $p_\theta(\boldsymbol{x}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{x}|\mu_\theta(\boldsymbol{z}), \eta \boldsymbol{I})$, we have $\log p_\theta(\boldsymbol{x}|\boldsymbol{z}) = -\frac{1}{2\eta}\left|\left|\boldsymbol{x} - \mu_\theta(\boldsymbol{z})\right|\right|_2^2 + const.$
  - We can approximate the expectation over $\boldsymbol{z} \sim q_\psi(\boldsymbol{z}\,|\boldsymbol{x})$ by an average over $q_\psi(\boldsymbol{z}\,|\boldsymbol{x})$
  - Recall that the expectation of a function $f(\boldsymbol{z})$ w.r.t. a random variable $\boldsymbol{z} \sim p$ can be approximated from i.i.d. samples $\boldsymbol{z}_j \sim \boldsymbol{p}, j = 1, \dots, M$, via Monte Carlo averages

$$\mathbb{E}_{\boldsymbol{z}\sim p}[f(\boldsymbol{z})] = \int_{\boldsymbol{z}} f(\boldsymbol{z}) p(\boldsymbol{z}) dx \approx \frac{1}{M}\sum_j f(\boldsymbol{z}_j)$$

  - Applying the above formula to M i.i.d. samples $\boldsymbol{z}_j \sim q_\psi(\boldsymbol{z} \mid x)$ we get reconstruction error

$$\mathbb{E}_{\boldsymbol{z}\sim q_\psi}[\log p_\theta(\boldsymbol{x} \mid \boldsymbol{z})] \approx \frac{1}{M}\sum_j \log p_\theta\left(\boldsymbol{x} \mid \boldsymbol{z}_j\right) = \frac{1}{2\eta M}\sum_j \left|\left|\boldsymbol{x} - \mu_\theta(\boldsymbol{z}_j)\right|\right|_2^2$$

Computing the ELBO Loss

$$L_{\theta,\psi}(\boldsymbol{x}) := \boxed{\mathbb{E}_{\boldsymbol{z}\sim q_\psi(\boldsymbol{z}\,|\boldsymbol{x})} \log p_\theta(\boldsymbol{x} \mid \boldsymbol{z})} + \boxed{\mathbb{E}_{\boldsymbol{z}\sim q_\psi(\boldsymbol{z}\,|\,\boldsymbol{x})} \log \frac{p(\boldsymbol{z})}{q_\psi(\boldsymbol{z} \mid \boldsymbol{x})}}$$

- Term 2 (Regularization to Prior)
  - Since $\mathbb{E}_{\boldsymbol{z}\sim q_\psi(\boldsymbol{z}|\boldsymbol{x})} \log \frac{p(\boldsymbol{z})}{q_\psi(\boldsymbol{z}|\boldsymbol{x})} = -KL(q_\psi(\boldsymbol{z}|\boldsymbol{x}) \,||\, p(\boldsymbol{z}))$, $q_\psi(\boldsymbol{z} \mid \boldsymbol{x}) = \mathcal{N}(\boldsymbol{z} \mid \mu_\psi(\boldsymbol{x}), \sigma_\psi^2(\boldsymbol{x})\boldsymbol{I})$, $p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, the second term is a KL divergence between two d-dimensional Gaussians, which has a closed form solution

$$KL(\mathcal{N}(\mu_1, \sigma_1^2 I) \,||\, \mathcal{N}(\mu_2, \sigma_2^2 I)) = \log\left(\frac{\sigma_2}{\sigma_1}\right) - \frac{d}{2} + \frac{d\sigma_1^2 + ||\mu_1 - \mu_2||_2^2}{2\sigma_2^2}$$

  - Applying the above formula to our VAE model yields,

$$KL\big(q_\psi(\boldsymbol{z} \mid \boldsymbol{x}) \,||\, p(\boldsymbol{z})\big) = -\log\left(\sigma_\psi(\boldsymbol{x})\right) + \frac{d\sigma_\psi^2(\boldsymbol{x}) + ||\mu_\psi(\boldsymbol{x})\,||_2^2}{2} + constant$$

  - Note this term does not require any sampling w.r.t. $\boldsymbol{z}$ because the expectation is computed in closed form thanks for the KL formula for Gaussians.

Maximizing ELBO: How to Optimize?

- **ELBO objective**: We want to solve the following optimization problem:

$$\max_{\theta,\psi} \sum_i L_{\theta,\psi}(\boldsymbol{x}_i) = \sum_i \mathbb{E}_{\boldsymbol{z}\sim q_\psi(\boldsymbol{z}|\boldsymbol{x}_i)} \log \frac{p_\theta(\boldsymbol{x}_i, \boldsymbol{z})}{q_\psi(\boldsymbol{z} \mid \boldsymbol{x}_i)}$$

- **Simple idea**: Just alternate gradient ascent wrt θ, ψ on objective function

$$\theta_{k+1} = \theta_k + \alpha \frac{\delta L}{\delta \theta}(\theta_k, \psi_k)$$

$$\psi_{k+1} = \psi_k + \alpha \frac{\delta L}{\delta \psi}(\theta_k, \psi_k)$$

Stochastic Optimization of ELBO wrt $\boldsymbol{\theta}$

- **Issue**: Computing ∇_θ (ELBO) is not easy because of expectation w.r.t. latent z.
- **Solution**: Compute an unbiased estimator

$$\nabla_\theta L_{\theta,\psi}(\boldsymbol{x}) = \nabla_\theta \mathbb{E}_{\boldsymbol{z}\sim q_\psi(\boldsymbol{z}|\boldsymbol{x})}\left[\log(p_\theta(\boldsymbol{x} \mid \boldsymbol{z}))\right] + \nabla_\theta \mathbb{E}_{\boldsymbol{z}\sim q_\psi(\boldsymbol{z}|\boldsymbol{x})}\left[\log\left(\frac{p(\boldsymbol{z})}{q_\psi(\boldsymbol{z} \mid \boldsymbol{x})}\right)\right]$$

$$= \mathbb{E}_{\boldsymbol{z}\sim q_\psi(\boldsymbol{z}|\boldsymbol{x})}\left[\nabla_\theta \log p_\theta(\boldsymbol{x} \mid \boldsymbol{z})\right]$$

$$= -\frac{1}{2\eta}\mathbb{E}_{z\sim q_\psi(\boldsymbol{z}|\boldsymbol{x})}\nabla_\theta \left|\left|\boldsymbol{x} - \mu_\theta(\boldsymbol{z})\right|\right|_2^2$$

- We can take sample averages to compute an unbiased estimator
  - For each datapoint $\boldsymbol{x}$, compute $q_\psi(\boldsymbol{z} \mid \boldsymbol{x})$ through encoder, which gives $\mu_\psi(\boldsymbol{x}), \sigma_\psi^2(\boldsymbol{x})$
  - Sample $\boldsymbol{z}_j \sim \mathcal{N}\left(\mu_\psi(\boldsymbol{x}), \sigma_\psi^2(\boldsymbol{x})\boldsymbol{I}\right)$, find $\mu_\theta\left(\boldsymbol{z}_j\right)$ through decoder
  - Estimate gradient from M samples: $\nabla_\theta L_{\theta,\psi}(\boldsymbol{x}) \approx -\frac{1}{2\eta M}\sum_j \nabla_\theta \left|\left|\boldsymbol{x} - \mu_\theta\left(\boldsymbol{z}_j\right)\right|\right|_2^2$

Stochastic Optimization of ELBO wrt ψ

- Here, we cannot just switch gradient and expectation because both are wrt to ψ

$$\nabla_\psi \mathbb{E}_{\boldsymbol{z}\sim q_\psi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}\,,\boldsymbol{z}) - \log q_\psi(\boldsymbol{z}\mid\boldsymbol{x})] \neq \mathbb{E}_{\boldsymbol{z}\sim q_\psi(\boldsymbol{z}|\boldsymbol{x})}\nabla_\psi[\log p_\theta(\boldsymbol{x},\boldsymbol{z}) - \log q_\psi(\boldsymbol{z}\mid\boldsymbol{x})]$$

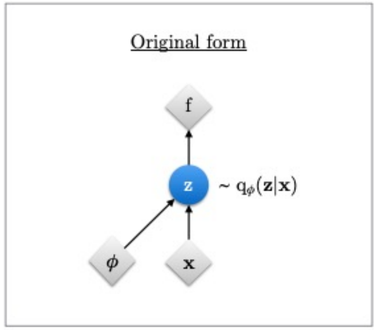
- Reparameterization trick
  - Because $q_\psi(\boldsymbol{z}\mid\boldsymbol{x}) = N(\boldsymbol{z};\mu_\psi(\boldsymbol{x}),\sigma_\psi(\boldsymbol{x})I)$, we can rewrite samples $\boldsymbol{z}\sim q_\psi(\boldsymbol{z}\mid\boldsymbol{x})$ as

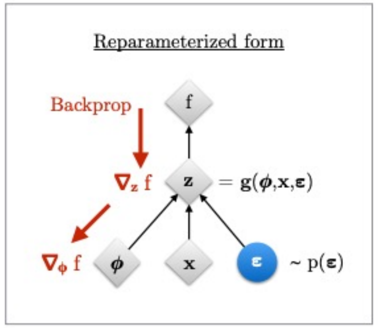
$$\boldsymbol{z}_\psi = g(\epsilon,\psi,\boldsymbol{x}) = \mu_\psi(\boldsymbol{x}) + \sigma_\psi(\boldsymbol{x})\boldsymbol{\epsilon} \quad \text{for} \quad \boldsymbol{\epsilon}\sim N(\boldsymbol{0},\boldsymbol{I})$$

  - With this change of variables, we can rewrite the gradient of the loss as:

$$\nabla_\psi L_{\theta,\psi}(\boldsymbol{x}) = \nabla_\psi \mathbb{E}_{\boldsymbol{z}\sim q_\psi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x},\boldsymbol{z}) - \log q_\psi(\boldsymbol{z}\mid\boldsymbol{x})]$$

$$= \nabla_\psi \mathbb{E}_{\boldsymbol{\epsilon}\sim N(\boldsymbol{0},\boldsymbol{I})}[\log p_\theta(\boldsymbol{x},\boldsymbol{z}_\psi) - \log q_\psi(\boldsymbol{z}_\psi\mid\boldsymbol{x})]$$

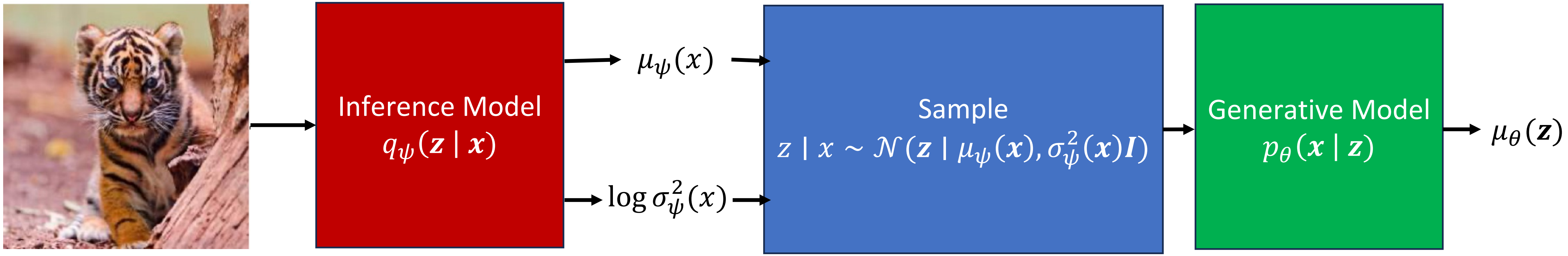




Gradient and expectation can now be switched! So as before, we compute an unbiased estimator by sampling many ϵ

Putting it all together

- Variational Autoencoder
  - We modelled inference and generative model as deep networks
  - We interpreted ELBO as an expected reconstruction error plus a KL-regularization to prior
  - Then, we rewrote the sampling in the latent space using the reparameterization trick
  - Finally, we derived stochastic gradient estimates to optimize the ELBO and learn a VAE



Datapoint $\boldsymbol{x}$

ELBO Objective

$$\mathbb{E}_{\boldsymbol{z} \sim q_\psi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x} \mid \boldsymbol{z}) - KL\left(q_\psi(\boldsymbol{z} \mid \boldsymbol{x}) \,||\, p(\boldsymbol{z})\right)$$

VAE's in Action

- $q_\psi(\boldsymbol{z} \mid \boldsymbol{x}) = \mathcal{N}(\boldsymbol{z} \mid \mu_\psi(\boldsymbol{x}), \sigma_\psi^2(\boldsymbol{x}))$
- $p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z} \mid \boldsymbol{0}, \boldsymbol{I})$
- $p_\theta(\boldsymbol{x} \mid \boldsymbol{z}) = Categorical\ (\boldsymbol{x} \mid \pi_\theta\ (\boldsymbol{z}))$ Note this is different from the model considered up till now!

- Encoder network: $\boldsymbol{x} \in \mathbb{R}^D$ -> Linear(D, 256) -> LeakyReLU -> Linear (256, $2d$) -> split into $\mu \in \mathbb{R}^d, \log \sigma^2 \in \mathbb{R}^d$

- Decoder network: $\boldsymbol{z} \in \mathbb{R}^d$ -> Linear(d, 256) -> LeakyReLU -> Linear(256, D) -> softmax

VAE's for Generation of MNIST Digits

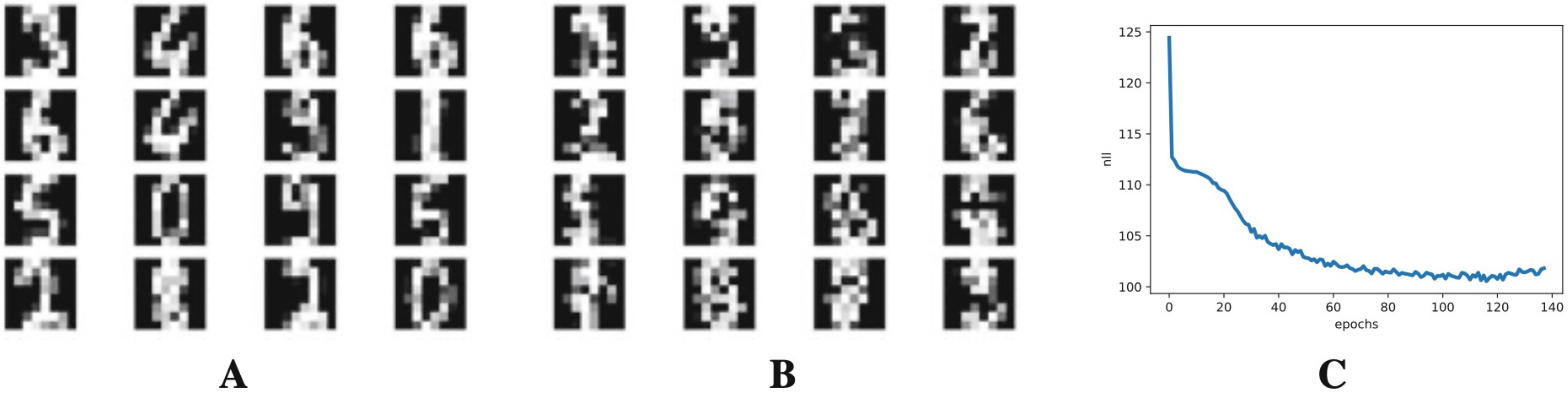


Fig. 4.4 An example of outcomes after the training: (**a**) Randomly selected real images. (**b**) Unconditional generations from the VAE. (**c**) The validation curve during training

VAEs: extensions

- So far, we have seen how to model $p(\boldsymbol{x})$ via VAEs.

- Many machine learning applications requires beyond learning $p(\boldsymbol{x})$:
  - Conditional generation, i.e., sampling from $p(\boldsymbol{x} \mid \boldsymbol{y})$ with a given class label $\boldsymbol{y}$
  - Classification, i.e., sampling from $p(\boldsymbol{y} \mid \boldsymbol{x})$ with a given sample $\boldsymbol{x}$

- Answer: We can extend the VAE paradigm to model $p(\boldsymbol{x}, \boldsymbol{y})$

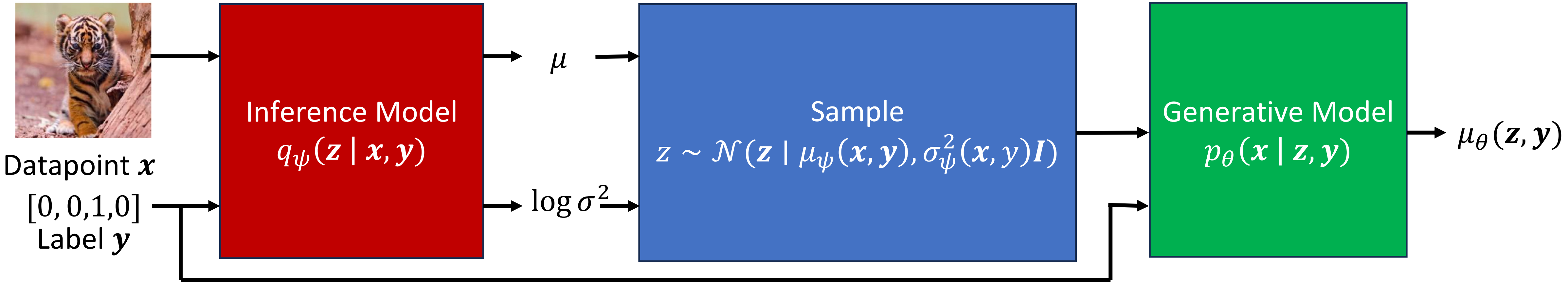
Joint VAE

- Regular VAE: start from $\log p_\theta(x)$ and derive its ELBO $\int q(\boldsymbol{z} \mid \boldsymbol{x}) \log \frac{p_\theta(\boldsymbol{x},\boldsymbol{z})}{q(\boldsymbol{z}|\boldsymbol{x})} dz$
- Here, since we want to model $p_\theta(x, y)$, let us derive ELBO for $\log p_\theta(x, y)$

$$\max_\theta \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}\sim p_{\text{data}}} \log p_\theta(x, \boldsymbol{y})$$

$$= \max_\theta \max_\psi \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}\sim p_{\text{data}}} \left[\mathbb{E}_{q_\psi(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})} \log p_\theta(\boldsymbol{x} \mid \boldsymbol{z}, \boldsymbol{y}) - KL[q_\psi(\boldsymbol{z} \mid \boldsymbol{x}, \boldsymbol{y}) \,||\, p_\theta(\boldsymbol{z} \mid \boldsymbol{y})] + \log p_\theta(\boldsymbol{y})\right]$$

  - Details in the next slide. Spoiler alert: the ELBO derivation is almost the same as before
- Architecture:



Joint VAE: ELBO

- Let $q_\psi(z|x,y)$ be the variational distribution. Observe that

$$\log p_\theta(x,y) = \int q_\psi(\mathbf{z} \mid \mathbf{x},\mathbf{y}) \log p_\theta(\mathbf{x},\mathbf{y}) d\mathbf{z} = \int q_\psi(\mathbf{z} \mid \mathbf{x},\mathbf{y}) \log \frac{p_\theta(\mathbf{x},\mathbf{y},\mathbf{z})}{p_\theta(\mathbf{z}|\mathbf{x},\mathbf{y})} d\mathbf{z}$$

$$= \int q_\psi(\mathbf{z} \mid \mathbf{x},\mathbf{y}) \log \frac{p_\theta(\mathbf{x},\mathbf{y},\mathbf{z})}{q_\psi(\mathbf{z}|\mathbf{x},\mathbf{y})} \frac{q_\psi(\mathbf{z}|\mathbf{x},\mathbf{y})}{p_\theta(\mathbf{z}|\mathbf{x},\mathbf{y})} d\mathbf{z}$$

$$= \underbrace{\boxed{\int q_\psi(\mathbf{z} \mid \mathbf{x},\mathbf{y}) \log \frac{p_\theta(\mathbf{x},\mathbf{y},\mathbf{z})}{q_\psi(\mathbf{z}|\mathbf{x},\mathbf{y})} d\mathbf{z}}}_{\text{Evidence Lower Bound (ELBO)}} + \underbrace{\boxed{\int q_\psi(\mathbf{z} \mid \mathbf{x},\mathbf{y}) \log \frac{q_\psi(\mathbf{z}|\mathbf{x},\mathbf{y})}{p_\theta(\mathbf{z}|\mathbf{x},\mathbf{y})} d\mathbf{z}}}_{\mathrm{KL}[q(z|x,y)\ ||\ p_\theta(z|x,y)]}$$

Therefore, $\log p_\theta(x \mid \mathbf{y}) = \max_\psi \mathbb{E}_{q_\psi(\mathbf{z}|\mathbf{x},\mathbf{y})} \log \frac{p_\theta(\mathbf{x},\mathbf{y},\mathbf{z})}{q_\psi(\mathbf{z}|\mathbf{x},\mathbf{y})}$

$$= \max_\psi \mathbb{E}_{q_\psi(\mathbf{z}|\mathbf{x},\mathbf{y})} \left[\log \frac{p_\theta(\mathbf{z}|\mathbf{y})}{q_\psi(\mathbf{z}|\mathbf{x},\mathbf{y})} + \log p_\theta(\mathbf{x} \mid \mathbf{y},\mathbf{z}) + \log p_\theta(\mathbf{y})\right]$$

$$= \max_\psi \mathbb{E}_{q_\psi(\mathbf{z}|\mathbf{x},\mathbf{y})} \log p_\theta(\mathbf{x} \mid \mathbf{y},\mathbf{z}) - KL\left[q_\psi(\mathbf{z} \mid \mathbf{x},\mathbf{y}), p_\theta(\mathbf{z} \mid \mathbf{y})\right] + \log p_\theta(\mathbf{y})$$

Joint VAE: Training objective

- The training objective is given by

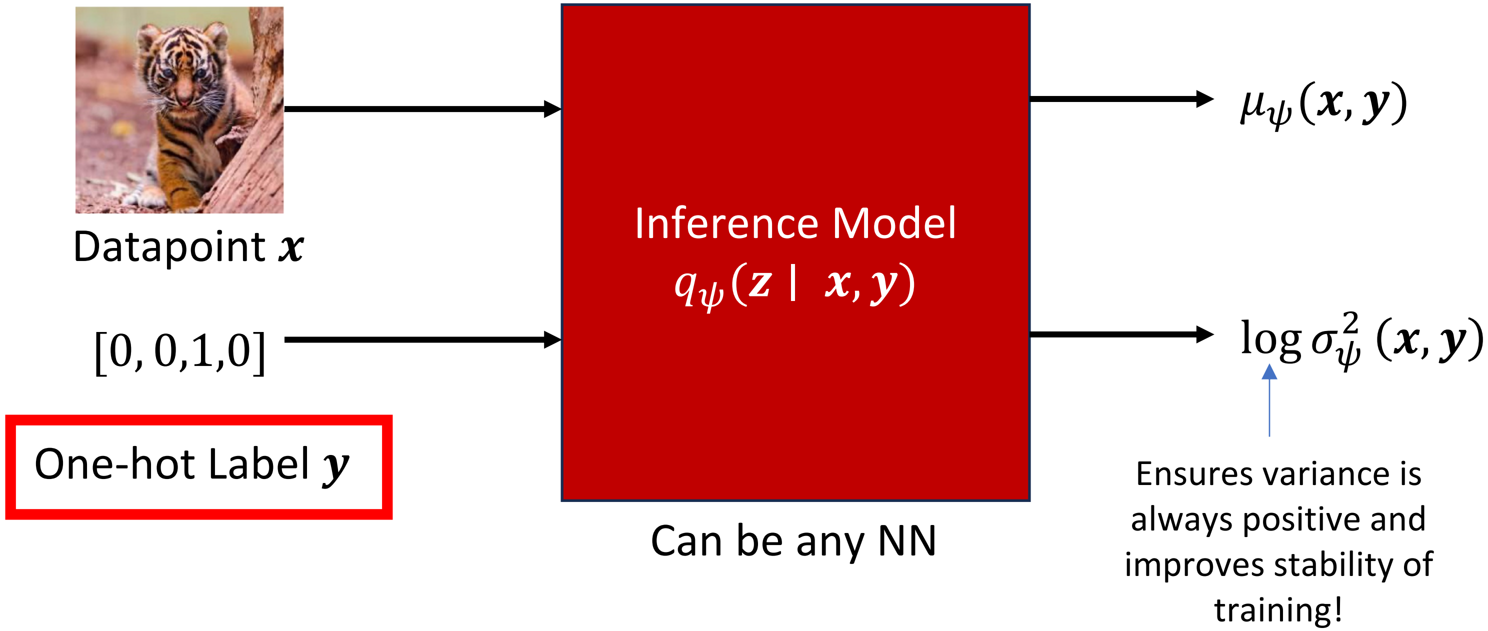
$$\max_{\theta} \mathrm{J}_{\mathrm{cond}} \coloneqq \mathbb{E}_{\boldsymbol{x},\boldsymbol{y} \sim p_{\mathrm{data}}} \log p_{\theta}(x, \boldsymbol{y})$$

$$= \max_{\theta} \max_{\psi} \mathbb{E}_{\boldsymbol{x},\boldsymbol{y} \sim p_{\mathrm{data}}} \left[\mathbb{E}_{q_{\psi}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})} \log p_{\theta}(\boldsymbol{x} \mid \boldsymbol{z}, \boldsymbol{y}) - KL[q_{\psi}(\boldsymbol{z} \mid \boldsymbol{x}, \boldsymbol{y}) \,||\, p_{\theta}(\boldsymbol{z} \mid \boldsymbol{y})] + \log p_{\theta}(\boldsymbol{y}) \right]$$

- What are $q_{\psi}(\boldsymbol{z} \mid \boldsymbol{x}, \boldsymbol{y})$, $p_{\theta}(\boldsymbol{x} \mid \boldsymbol{z}, \boldsymbol{y})$, $p_{\theta}(\boldsymbol{z} \mid \boldsymbol{y})$?
  - They are similar to what we needed in regular VAEs, with an additional input $\boldsymbol{y}$
- What is $p_{\theta}(\boldsymbol{y})$?
  - This is a new term denoting the prior probability on class labels
- Answer: next two slides

Joint VAE: Setup

- We have three models we need to define for VAE model
  1. Inference model $q_\psi(\boldsymbol{z} \mid \boldsymbol{x}, \boldsymbol{y})$: We will define as $q_\psi(\boldsymbol{z} \mid \boldsymbol{x}, \boldsymbol{y}) = \mathcal{N}(\boldsymbol{z};\ \mu_\psi(\boldsymbol{x}, \boldsymbol{y}), \sigma_\psi^2(\boldsymbol{x}, \boldsymbol{y})\boldsymbol{I})$, i.e., a normal distribution with learned mean and covariance

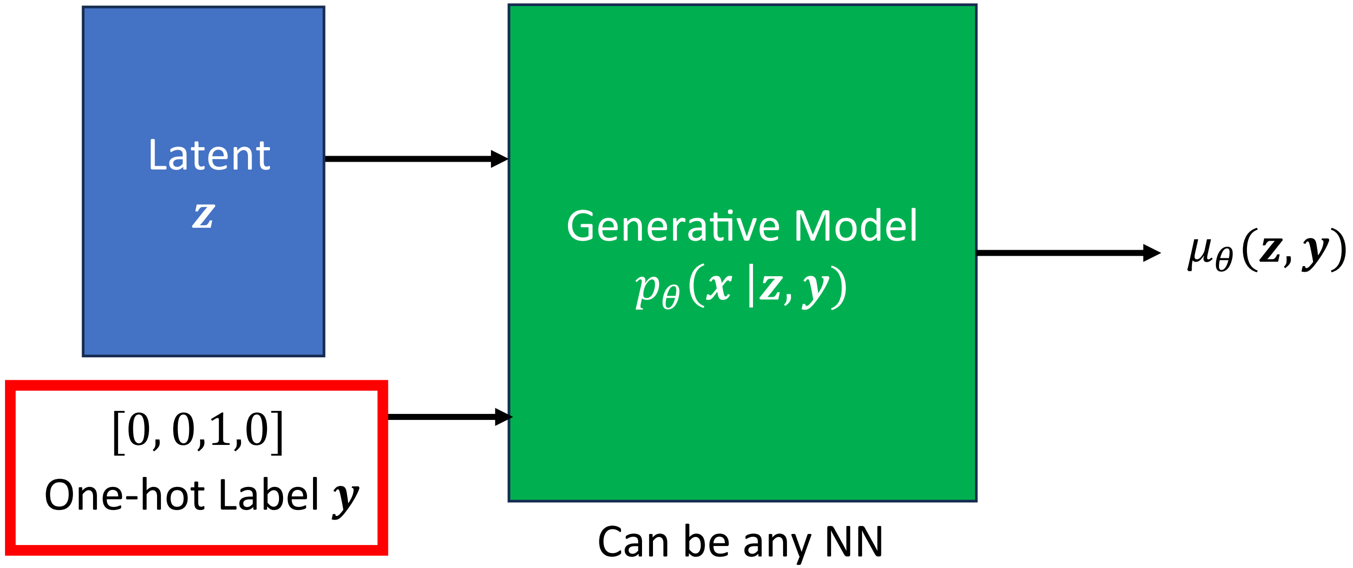


  2. Latent prior: $p(\boldsymbol{z} \mid \boldsymbol{y}) = \mathcal{N}(\boldsymbol{z}; \boldsymbol{0}, \boldsymbol{I})$
  3. Class prior: $p_\theta(\boldsymbol{y}) = Categorical\ (\pi)$

Joint VAE: Setup

- We have three models we need to define for VAE model

4. Generative model $p_\theta(\mathbf{x} \mid \mathbf{z}, \mathbf{y})$: We will define as
    - $p_\theta(\boldsymbol{x} \mid \boldsymbol{z}, \boldsymbol{y}) = \mathcal{N}(\boldsymbol{z}; \mu_\theta(\boldsymbol{z}, \boldsymbol{y}), \eta^2 \boldsymbol{I})$, i.e., a normal distribution with learned mean and variance
    - $p_\theta(\boldsymbol{x} \mid \boldsymbol{z}, \boldsymbol{y}) = Cat(\boldsymbol{z}; \pi_\theta(\boldsymbol{z}, \boldsymbol{y}))$, i.e., a categorical distribution with learned class probabilities



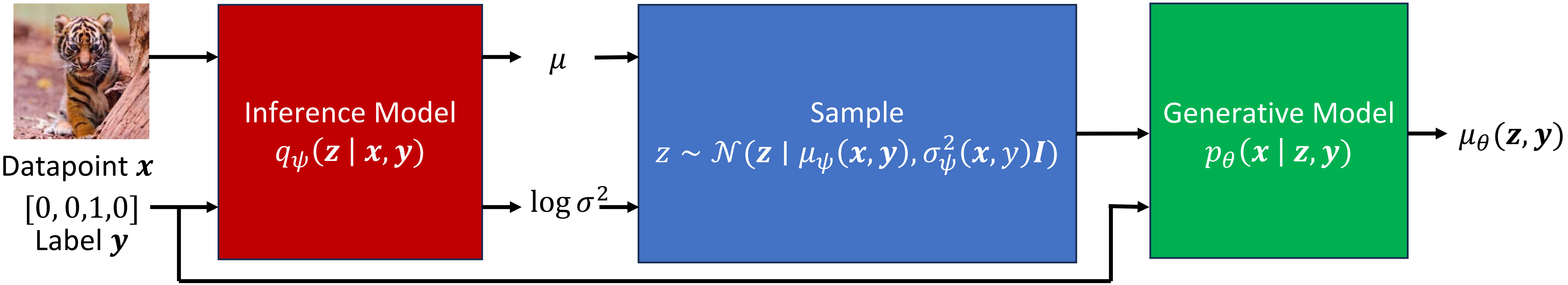
- Note this can be defined in many different ways, yielding different models (such as a categorical distribution over 255 values of each pixel)

Joint VAE: Training objective

- So far, the training objective is given by

$$\max_\theta J_{\text{cond}} \coloneqq \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}\sim p_{\text{data}}} \log p_\theta(x, \boldsymbol{y})$$

$$= \max_\theta \max_\psi \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}\sim p_{\text{data}}} \left[\mathbb{E}_{q_\psi(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})} \log p_\theta(\boldsymbol{x} \mid \boldsymbol{z}, \boldsymbol{y}) - KL[q_\psi(\boldsymbol{z} \mid \boldsymbol{x}, \boldsymbol{y}) \,||\, p_\theta(\boldsymbol{z} \mid \boldsymbol{y})] + \log p_\theta(\boldsymbol{y}) \right]$$



- This allows us to do conditional generation
  - Specify a class $\boldsymbol{y}$ that you want to sample from
  - Sample from $p_\theta(\boldsymbol{z} \mid \boldsymbol{y})$, which is a Gaussian $\mathcal{N}(\boldsymbol{z}; \boldsymbol{0}, \boldsymbol{I})$
  - Sample from the conditional generation $p_\theta(\boldsymbol{x} \mid \boldsymbol{z}, \boldsymbol{y})$

Joint VAE: Training objective

- So far, the training objective is given by

$$\max_{\theta} \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}\sim p_{\text{data}}} \log p_{\theta}(x, \boldsymbol{y})$$

$$= \max_{\theta} \max_{\psi} \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}\sim p_{\text{data}}} \left[\mathbb{E}_{q_{\psi}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})} \log p_{\theta}(\boldsymbol{x} \mid \boldsymbol{z}, \boldsymbol{y}) - KL[q_{\psi}(\boldsymbol{z} \mid \boldsymbol{x}, \boldsymbol{y}) \,||\, p_{\theta}(\boldsymbol{z} \mid \boldsymbol{y})] + \log p_{\theta}(\boldsymbol{y})\right] =: \mathrm{J}_{\text{cond}}$$

- Hold on… Classification, i.e., the distribution of $\boldsymbol{y}$ given $\boldsymbol{x}$ is not taken care of
- How can we also model classification?
  - Again, let us start with MLE principle: $\mathbb{E}_{\boldsymbol{x},\boldsymbol{y}\sim p_{\text{data}}} \log q_{\psi}(\boldsymbol{x}, \boldsymbol{y})$
  - Define $q_{\psi}(\boldsymbol{x}, \boldsymbol{y}) = q_{\psi}(\boldsymbol{y} \mid \boldsymbol{x}) p_{\text{data}}(\boldsymbol{x})$, where $q_{\psi}(\boldsymbol{y} \mid \boldsymbol{x})$ is a classifier
  - $\mathbb{E}_{\boldsymbol{x},\boldsymbol{y}\sim p_{\text{data}}} \log q_{\psi}(\boldsymbol{x}, \boldsymbol{y}) = \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}\sim p_{\text{data}}} \log q_{\psi}(\boldsymbol{y} \mid \boldsymbol{x}) + \text{const. not depending on } \psi$
  - This gives the classification loss $\mathrm{J}_{\text{cls}} \coloneqq \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}\sim p_{\text{data}}} \log q_{\psi}(\boldsymbol{y} \mid \boldsymbol{x})$
- The final objective is given by $\mathrm{J}_{\text{cond}} + \lambda \mathrm{J}_{\text{cls}}$
  - They originate from the MLE principles, since it is the sum of two log-likelihoods

Joint VAE: Summary

$$\max_{\theta}\max_{\psi}\mathbb{E}_{\boldsymbol{x},\boldsymbol{y}\sim p_{\text{data}}}\left[\mathbb{E}_{q_\psi(\boldsymbol{z}\mid\boldsymbol{x},\boldsymbol{y})}\log p_\theta(\boldsymbol{x}\mid\boldsymbol{z},\boldsymbol{y}) - KL\left[q_\psi(\boldsymbol{z}\mid\boldsymbol{x},\boldsymbol{y})\,||\,p_\theta(\boldsymbol{z}\mid\boldsymbol{y})\right] + \log p_\theta(\boldsymbol{y}) + \lambda\log q_\psi(\boldsymbol{y}\mid\boldsymbol{x})\right]$$