Variational Autoencoder

Mark Chang

Original Paper

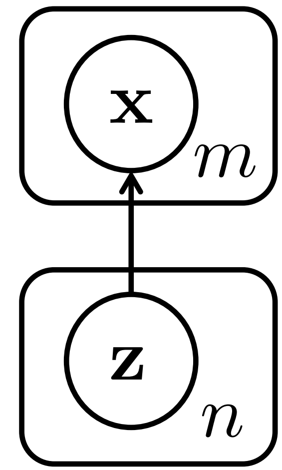
- Title:
  – Auto-Encoding Variational Bayes
- Author:
  – Diederik P. Kingma
  – Max Welling
- Organization:
  – Machine Learning Group, Universiteit van Amsterdam

Outlines

- Variational Inference
- Variational Autoencoder
- Experiment
- Further Research

Variational Inference

- Problem Definition
  - Observable Data: $\mathbf{x} = \{x_1, x_2, ..., x_m\}$
  - Hidden Variable: $\mathbf{z} = \{z_1, z_2, ..., z_n\}$
  - Posterior Distribution of hidden variable given some data:



$$p(\mathbf{z}|\mathbf{x}) = \frac{p(\mathbf{z}, \mathbf{x})}{p(\mathbf{x})} = \frac{p(\mathbf{x}|\mathbf{z})p(\mathbf{z})}{\int p(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z}}$$

Intractable to compute

Variational Inference

- Solutions for Intractable Posterior
  - Monte Carlo Sampling
    - Metropolis Hasting
    - Gibbs Sampling
  - **Variational Inference**

Variational Inference

- Approximate $p(\mathbf{z}|\mathbf{x})$ by $q(\mathbf{z})$
- Minimize the KL Divergence:

$$D_{KL}[q(\mathbf{z})||p(\mathbf{z}|\mathbf{x})] = \int q(\mathbf{z}) \log \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{x})} d\mathbf{z}$$

Evidence(Variational) Lower Bound

$$D_{KL}[q(\mathbf{z})||p(\mathbf{z}|\mathbf{x})] = \int q(\mathbf{z})\log\frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{x})}d\mathbf{z}$$

$$= \int q(\mathbf{z})\log\frac{q(\mathbf{z})p(\mathbf{x})}{p(\mathbf{z},\mathbf{x})}d\mathbf{z}$$

$$= \int q(\mathbf{z})\log\frac{q(\mathbf{z})}{p(\mathbf{z},\mathbf{x})}d\mathbf{z} + \int q(\mathbf{z})\log p(\mathbf{x})d\mathbf{z}$$

$$= \int q(\mathbf{z})(\log q(\mathbf{z}) - \log p(\mathbf{z},\mathbf{x}))d\mathbf{z} + \log p(\mathbf{x})$$

$$= -\underbrace{(E_{q(\mathbf{z})}[\log p(\mathbf{z},\mathbf{x})] - E_{q(\mathbf{z})}[\log q(\mathbf{z})])}_{\text{Evidence Lower Bound (ELBO): } L[q(\mathbf{z})]} + \log p(\mathbf{x})$$

Evidence Lower Bound

$$D_{KL}[q(\mathbf{z})||p(\mathbf{z}|\mathbf{x})] = -L[q(\mathbf{z})] + \log p(\mathbf{x})$$

$$\log p(\mathbf{x}) = D_{KL}[q(\mathbf{z})||p(\mathbf{z}|\mathbf{x})] + L[q(\mathbf{z})]$$

Minimize $D_{KL}[q(\mathbf{z})||p(\mathbf{z}|\mathbf{x})]$

is equal to Maximize $L[q(\mathbf{z})]$

Mean-Field Variational Inference

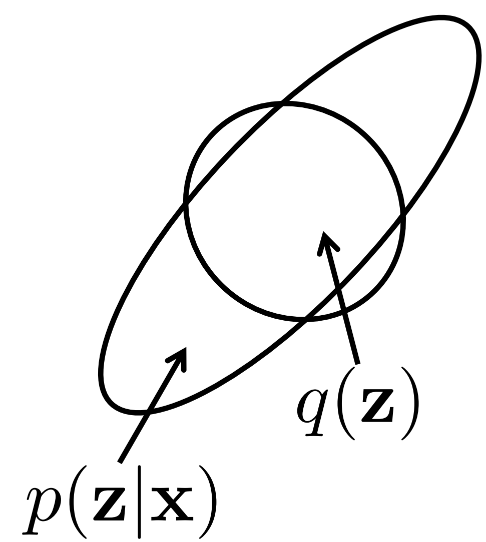
- Q can be factorized:

$$q(\mathbf{z}) = \prod_i q(z_i|\theta_i)$$

$$\forall i, \int q(z_i|\theta_i) dz_i = 1$$

Minimize

$$D_{KL}[q(\mathbf{z})||p(\mathbf{z}|\mathbf{x})]$$

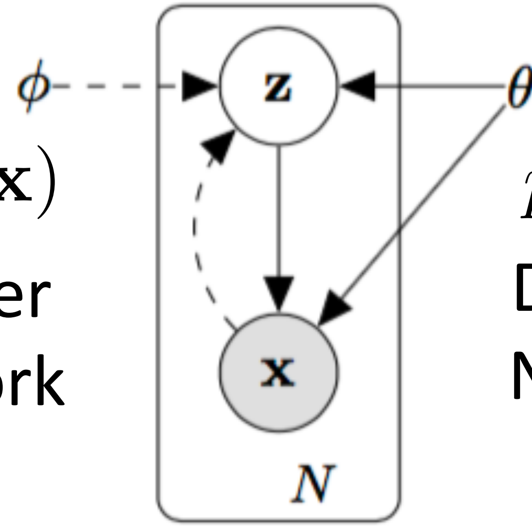


http://cpmarkchang.logdown.com/posts/737247-pgm-variational-inference

Variational Autoencoder

$q_\phi(\mathbf{z}|\mathbf{x})$

Encoder Network



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

Decoder Network

Minimize: $D_{KL}[q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})]$

Intractable: $p_\theta(\mathbf{z}|\mathbf{x}) = \dfrac{p_\theta(\mathbf{x}|\mathbf{z})p_\theta(\mathbf{z})}{p_\theta(\mathbf{x})}$

Variational Autoencoder

Marginal Likelihood:

$$\log p_\theta(\mathbf{x}) = D_{KL}[q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})] + L(\theta, \phi, \mathbf{x})$$

Variational Lower Bound:

$$L(\theta, \phi, \mathbf{x}) = E_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}, \mathbf{z}) - \log q_\phi(\mathbf{z}|\mathbf{x})]$$

$$= E_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{z}) + \log p_\theta(\mathbf{x}|\mathbf{z}) - \log q_\phi(\mathbf{z}|\mathbf{x})]$$

$$= E_{q_\phi(\mathbf{z}|\mathbf{x})}[\log \frac{p_\theta(\mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})} + p_\theta(\mathbf{x}|\mathbf{z})]$$

$$= -D_{KL}[q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z})] + E_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})]$$

Monte Carlo Gradient Estimator

Gradient of $L(\theta, \phi, \mathbf{x})$ contains $\nabla_\phi E_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})]$ which is Intractable

Use Monte Carlo Gradient Estimator :

$$\nabla_\phi E_{q_\phi(\mathbf{z})}[f(\mathbf{z})] = \nabla_\phi \int q_\phi(\mathbf{z}) f(\mathbf{z}) d\mathbf{z}$$

$$= \int q_\phi(\mathbf{z}) f(\mathbf{z}) \frac{\nabla_\phi q_\phi(\mathbf{z})}{q_\phi(\mathbf{z})} d\mathbf{z} \quad = \int q_\phi(\mathbf{z}) f(\mathbf{z}) \nabla_\phi \log q_\phi(\mathbf{z}) d\mathbf{z}$$

$$= E_{q_\phi(\mathbf{z})}[f(\mathbf{z}) \nabla_\phi \log q_\phi(\mathbf{z})]$$

$$\approx \frac{1}{L} \sum_{l=1}^{L} f(\mathbf{z}) \nabla_\phi \log q_\phi(\mathbf{z}^{(l)}) \quad \text{where} \quad \mathbf{z}^{(l)} \sim q_\phi(\mathbf{z})$$

Objective Function

$$L(\theta, \phi, \mathbf{x}^{(i)}) = -D_{KL}[q_\phi(\mathbf{z}|\mathbf{x}^{(i)})||p_\theta(\mathbf{z})] + E_{q_\phi(\mathbf{z}|\mathbf{x}^{(i)})}[\log p_\theta(\mathbf{x}^{(i)}|\mathbf{z})]$$

Monte Carlo Gradient Estimator

$$\tilde{L}(\theta, \phi, \mathbf{x}^{(i)}) \approx L(\theta, \phi, \mathbf{x}^{(i)})$$

$$\tilde{L}(\theta, \phi, \mathbf{x}^{(i)}) = -D_{KL}[q_\phi(\mathbf{z}|\mathbf{x}^{(i)})||p_\theta(\mathbf{z})] + \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(\mathbf{x}^{(i)}|\mathbf{z}^{(i,l)})$$

where $\mathbf{z}^{(l)} \sim q_\phi(\mathbf{z}|\mathbf{x}^{(i,l)})$

Reparameterization Trick

$$\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})$$

auxiliary variable

$$\epsilon \sim p(\epsilon)$$

deterministic variable

$$\mathbf{z} = g_\phi(\epsilon, \mathbf{x})$$

Example:

$$z \sim p(z|x) = \mathcal{N}(\mu, \sigma^2)$$

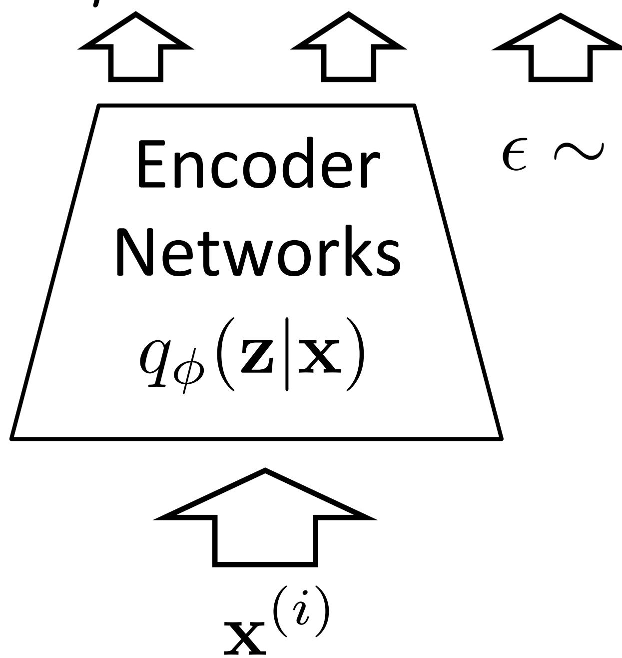
$$\epsilon \sim \mathcal{N}(0, 1)$$

$$z = \mu + \sigma\epsilon$$

Reparameterization Trick

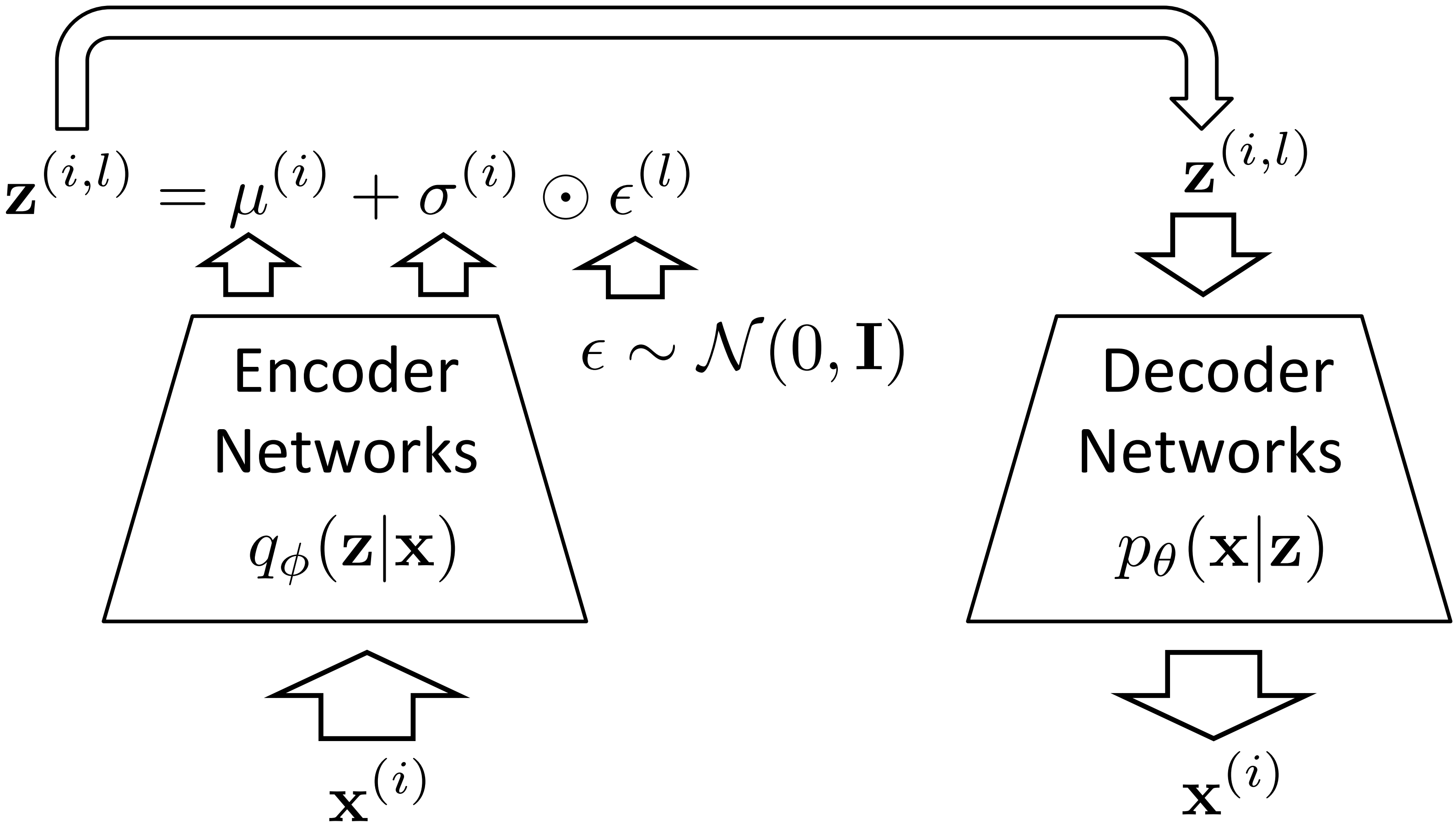
$$\log q_\phi(\mathbf{z}|\mathbf{x}^{(i)}) = \log \mathcal{N}(\mathbf{z}, \mu^{(i)}, \sigma^{2(i)}\mathbf{I})$$

$$\mathbf{z}^{(i,l)} = \mu^{(i)} + \sigma^{(i)} \odot \epsilon^{(l)}$$



$$\epsilon \sim \mathcal{N}(0, \mathbf{I})$$

Reparameterization Trick



Objective Function

$$\tilde{L}(\theta, \phi, \mathbf{x}^{(i)}) = -D_{KL}[q_\phi(\mathbf{z}|\mathbf{x}^{(i)})||p_\theta(\mathbf{z})] + \frac{1}{L}\sum_{l=1}^{L}(\log p_\theta(\mathbf{x}^{(i)}|\mathbf{z}^{(i,l)})$$

$$q_\phi(\mathbf{z}|\mathbf{x}^{(i)}) = \mathcal{N}(\mathbf{z}, \mu^{(i)}, \sigma^{2(i)}\mathbf{I})$$

$$p_\theta(\mathbf{z}) = \mathcal{N}(\mathbf{z}, 0, \mathbf{I})$$

$$\Downarrow$$

$$\tilde{L}(\theta, \phi, \mathbf{x}^{(i)}) = \frac{1}{2}\sum_{j=1}^{J}(1+\log((\sigma_j^{(i)})^2)-(\mu_j^{(i)})^2-(\sigma_j^{(i)})^2)+\frac{1}{L}\sum_{l=1}^{L}(\log p_\theta(\mathbf{x}^{(i)}|\mathbf{z}^{(i,l)})$$

Regularization

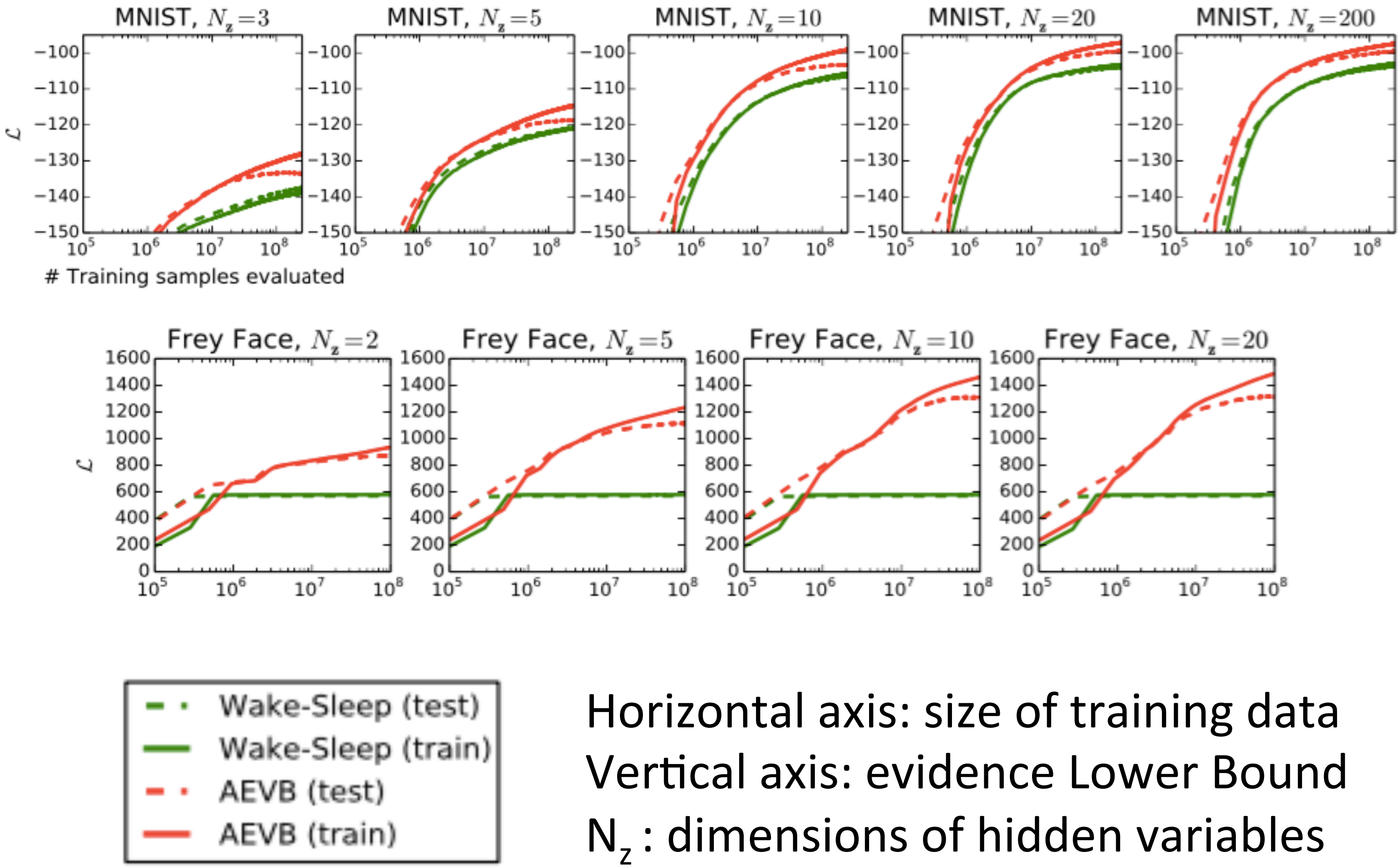
Reconstruction Error

Training

Algorithm 1 Minibatch version of the Auto-Encoding VB (AEVB) algorithm. Either of the two SGVB estimators in section 2.3 can be used. We use settings $M = 100$ and $L = 1$ in experiments.

$\boldsymbol{\theta}, \boldsymbol{\phi} \leftarrow$ Initialize parameters
repeat
  $\mathbf{X}^M \leftarrow$ Random minibatch of M datapoints (drawn from full dataset)
  $\boldsymbol{\epsilon} \leftarrow$ Random samples from noise distribution $p(\boldsymbol{\epsilon})$
  $\mathbf{g} \leftarrow \nabla_{\boldsymbol{\theta},\boldsymbol{\phi}} \widetilde{\mathcal{L}}^M(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{X}^M, \boldsymbol{\epsilon})$ (Gradients of minibatch estimator (8))
  $\boldsymbol{\theta}, \boldsymbol{\phi} \leftarrow$ Update parameters using gradients $\mathbf{g}$ (e.g. SGD or Adagrad [DHS10])
until convergence of parameters $(\boldsymbol{\theta}, \boldsymbol{\phi})$
return $\boldsymbol{\theta}, \boldsymbol{\phi}$

Experiment

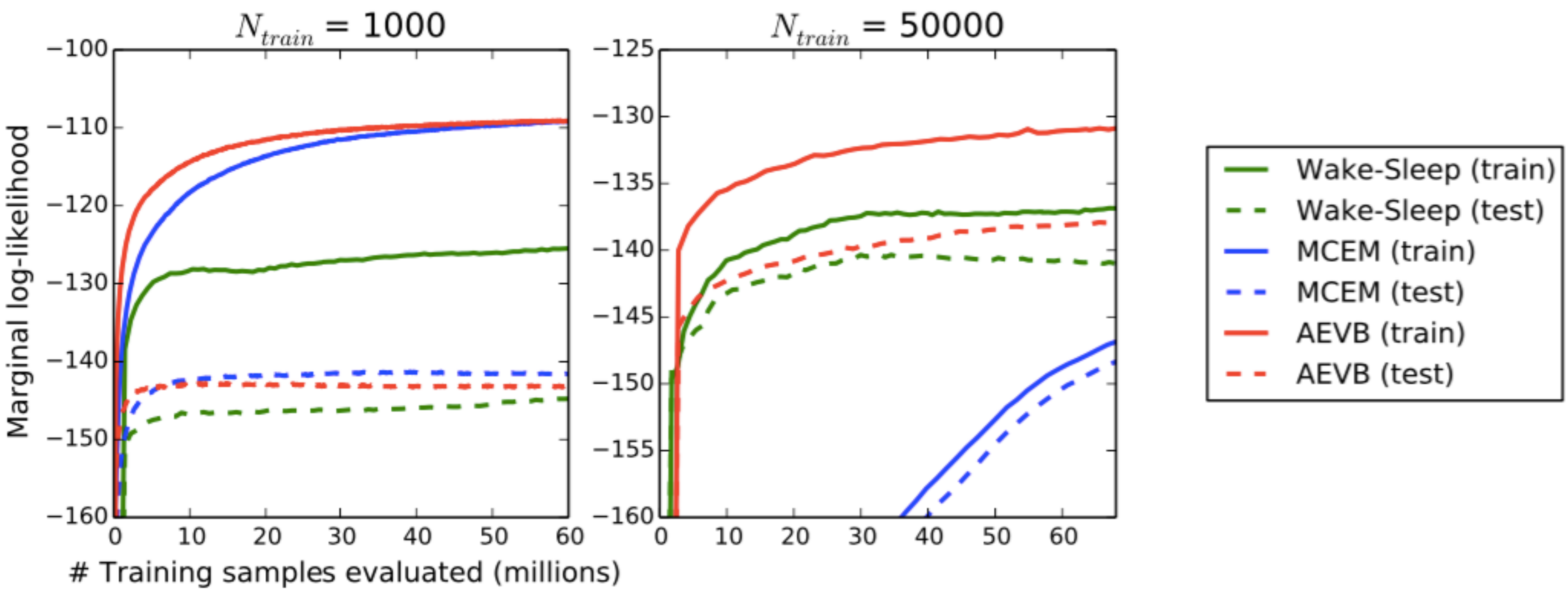


Horizontal axis: size of training data

Vertical axis: evidence Lower Bound

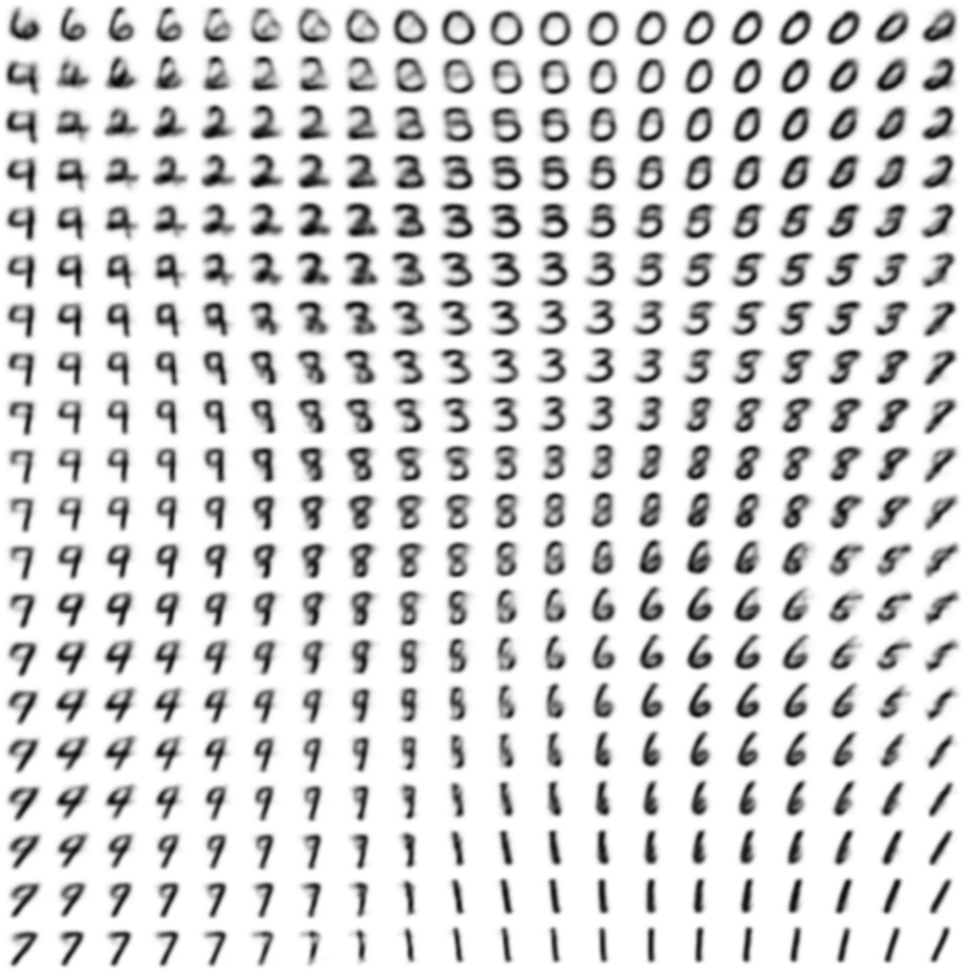
N_z : dimensions of hidden variables

Experiment



N_{train} = 1000
N_{train} = 50000
Marginal log-likelihood
Training samples evaluated (millions)
−100
−110
−120
−130
−140
−150
−160
−125
−130
−135
−140
−145
−150
−155
−160
0
10
20
30
40
50
60
Wake-Sleep (train)
Wake-Sleep (test)
MCEM (train)
MCEM (test)
AEVB (train)
AEVB (test)

Experiment

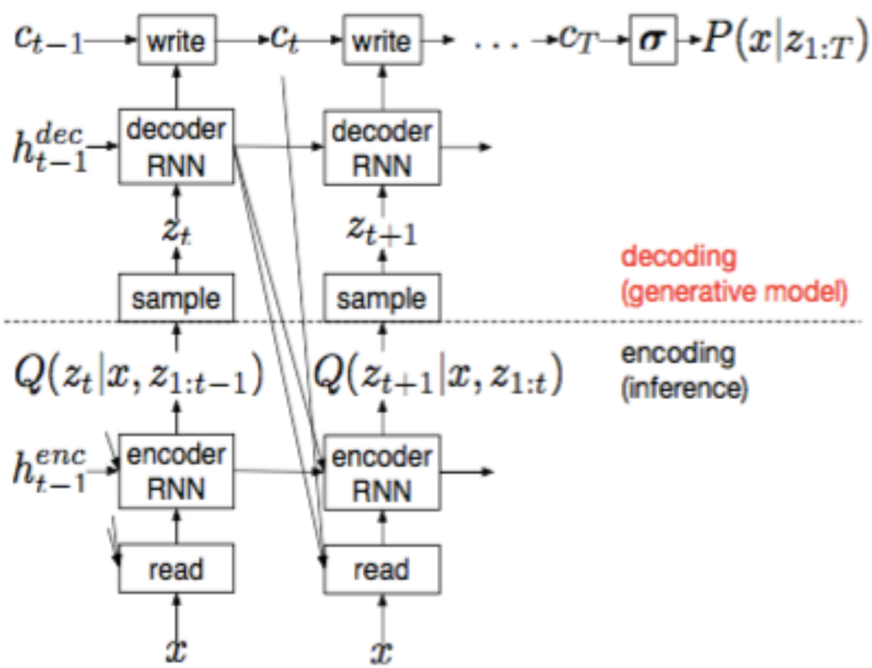


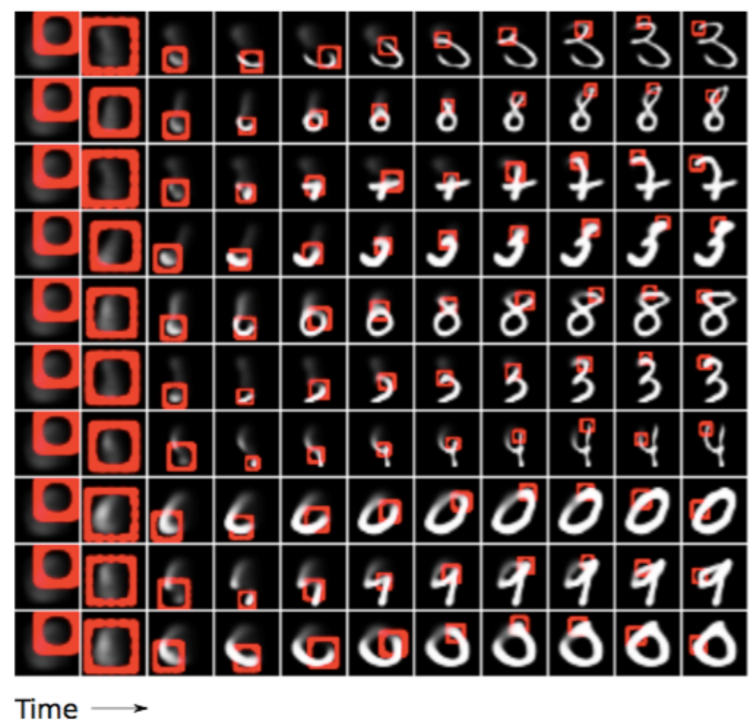
Visualization of 2d latent space

Further Research

DRAW: A Recurrent Neural Network For Image Generation

Karol Gregor, Ivo Danihelka, Alex Graves,Danilo Jimenez Rezende and Daan Wierstra

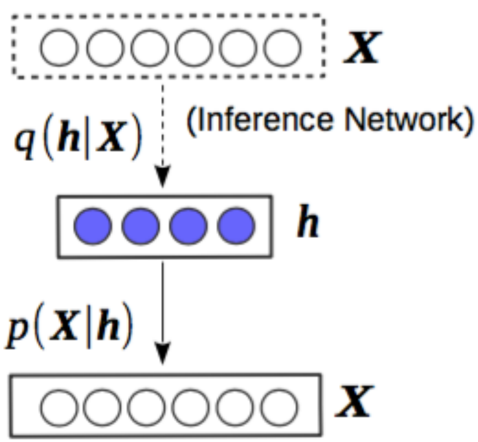




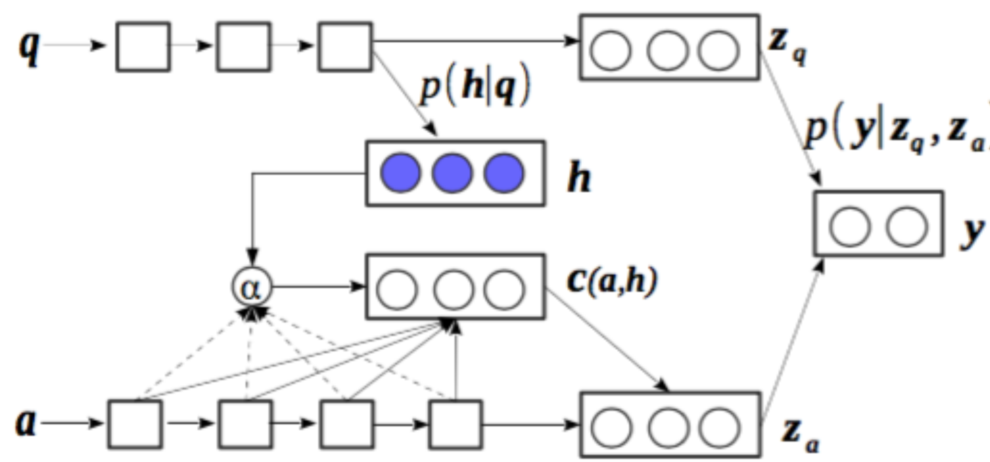
Further Research

Neural Variational Inference for Text Processing

Yishu Miao, Lei Yu & Phil Blunsom



Neural Variational Document Model

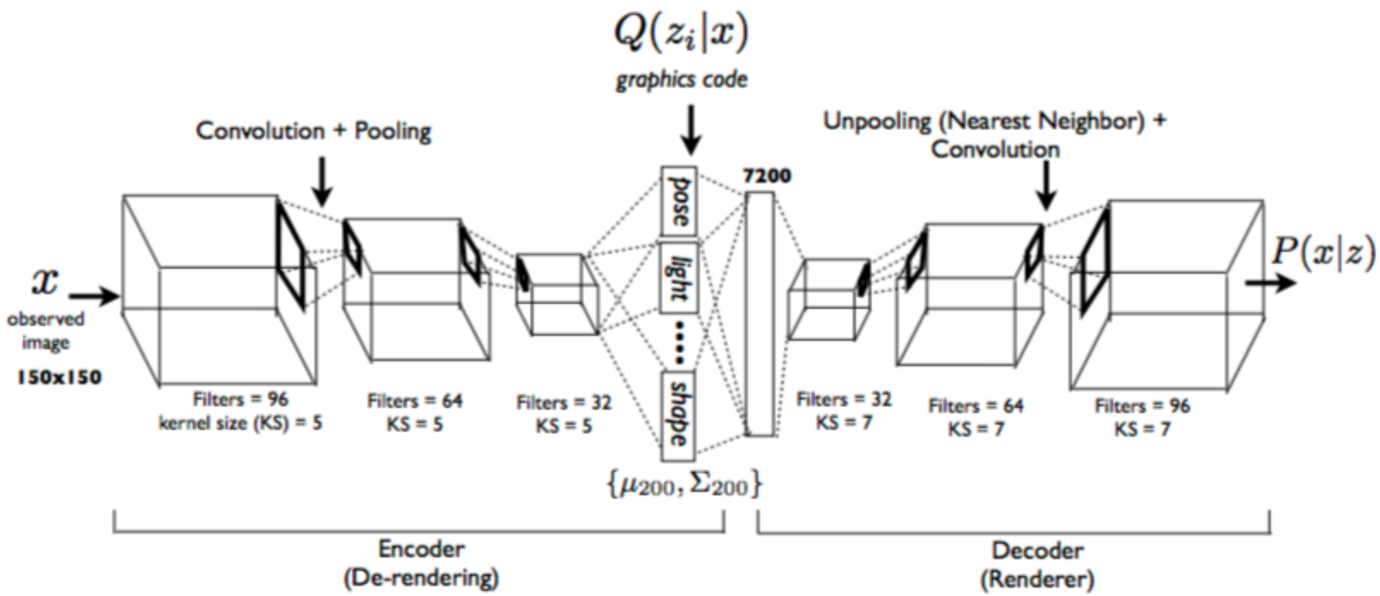


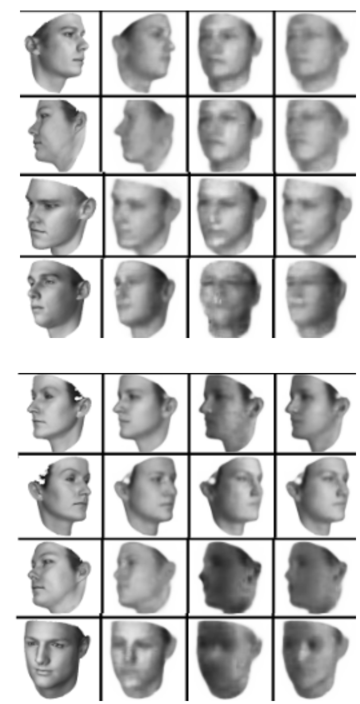
Neural Answer Selection Model

Further Research

Deep Convolutional Inverse Graphics Network

Tejas D. Kulkarni, William F. Whitney, Pushmeet Kohli, Joshua B. Tenenbaum





Source Code

- https://jmetzen.github.io/2015-11-27/vae.html

Reference

- Charles Fox, Stephen Roberts. A Tutorial on Variational Bayesian Inference.
  - http://www.orchid.ac.uk/eprints/40/1/fox_vbtut.pdf
- Diederik P Kingma, Max Welling. Auto-Encoding Variational Bayes.
  - https://arxiv.org/pdf/1312.6114v10.pdf