Quantum Variational Autoencoder and its applications

Hossein Sadeghi

September 16th, 2019

TRIUMF-Helmholtz Workshop

D-Wave

The Quantum Computing Company™

Overview

Quantum Annealing

Physics discovery

Generative modeling with Quantum Annealing

Performance of simulated quantum variational autoencoder

Quantum Annealing

Quantum annealing

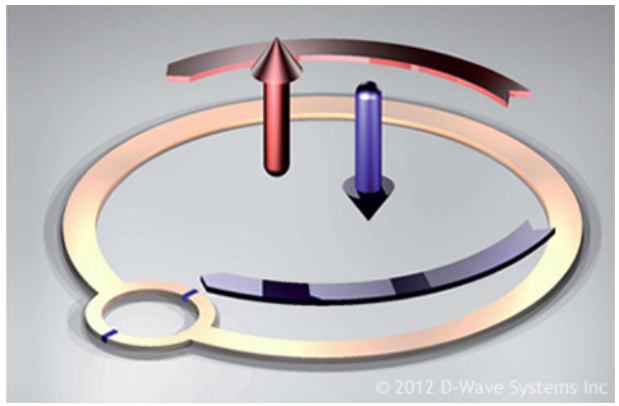
- Superconducting circuit as a qubit
- effective Hamiltonian

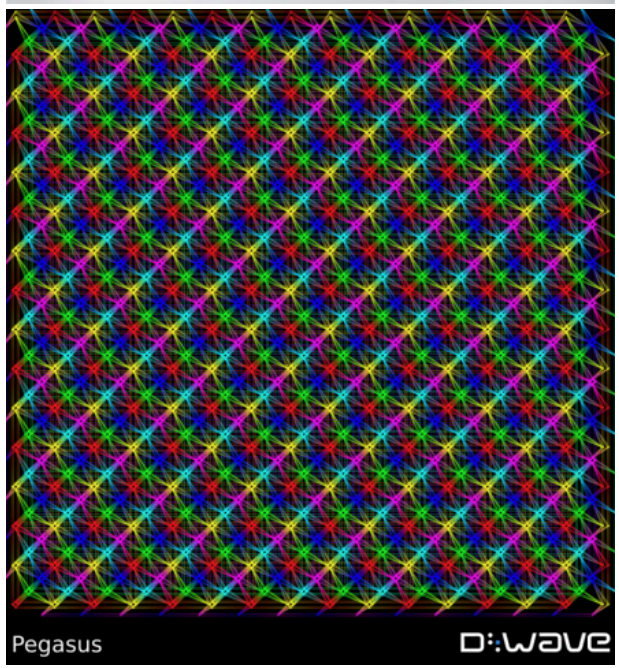
$$H_p = \sum_i h_i \sigma_i^z + \sum_{ij} J_{ij} \sigma_i^z \sigma_j^z \tag{1}$$

- quantum annealing

$$H = A(s) \sum_i \sigma_i^x + B(s) H_p$$

- 2048 qubits, 6k couplers
- **Next generation 5k qubits, 40K couplers**





Optimization & Sampling

- Many complex problems can be mapped to energy optimization over binary variables

$$E(\mathbf{s}) = \sum_i h_i s_i + \sum_{ij} J_{ij} s_i s_j$$

$$\mathbf{s} = \operatorname*{argmin}_{\mathbf{s}} \sum_i h_i s_i + \sum_{ij} J_{ij} s_i s_j$$

- Sampling. Quantum annealer is a stochastic sampler and it follows "a" Boltzmann distribution.

$$p(\mathbf{s}) = \frac{e^{-\beta E(\mathbf{s})}}{\sum_{\mathbf{s}'} e^{-\beta E(\mathbf{s}')}}$$

$$\mathbf{s} \sim p(\mathbf{s})$$


Physics discovery

Programmable quantum spin glass simulator

REPORT

PHYSICS

Phase transitions in a programmable quantum spin glass simulator

R. Harris[1*], Y. Sato[1], A. J. Berkley[1], M. Reis[1], F. Altomare[1], M. H. Amin[1,2], K. Boothby[1], P. Bunyk[1], C. Deng[1], C. Enderud[1], S. Huang[1], E. Hoskinson[1], M. W. Johnson[1], E. Ladizinsky[1], N. Ladizinsky[1], T. Lanting[1], R. Li[1], T. Medina[1], R. Molavi[1,3], R. Neufeld[1], T. Oh[1], I. Pavlov[1], I. Perminov[1], G. Poulin-Lamarre[1], C. Rich[1], A. Smirnov[1], L. Swenson[1], N. Tsai[1], M. Volkmann[1], J. Whittaker[1], J. Yao[1]

Material simulation

nature
International journal of science

Letter | Published: 22 August 2018

Observation of topological phenomena in a programmable lattice of 1,800 qubits

Andrew D. King ✉, Juan Carrasquilla, […] Mohammad H. Amin

Higgs Boson optimization problem

nature
International journal of science

Letter | Published: 18 October 2017

Solving a Higgs optimization problem with quantum annealing for machine learning

Alex Mott, Joshua Job, Jean-Roch Vlimant, Daniel Lidar & Maria Spiropulu

Generative modeling with Quantum Annealing

Quantum Boltzmann Machine

- A Boltzmann machine with a transverse field Ising Hamiltonian, with

$$\Gamma = A/B$$

$$H = \Gamma \sum_i \sigma_i^x + \left(\sum_i h_i \sigma_i^z + \sum_{i,j} J_{ij} \sigma_i^z \sigma_j^z \right) \tag{2}$$

- Objective function: negative log-Likelihood function

$$\mathcal{L} = -\mathbb{E}_{\mathbf{x} \sim D} \log(p(\mathbf{x})) \tag{3}$$

- Training:

$$\delta h_i = \eta \left(\overline{\langle \sigma_i^z \rangle_{\mathbf{x}}} - \langle \sigma_i^z \rangle \right), \tag{4}$$

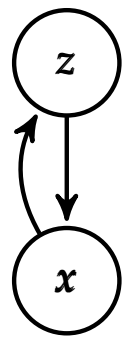
$$\delta J_{ij} = \eta \left(\overline{\langle \sigma_i^z \sigma_j^z \rangle_{\mathbf{x}}} - \langle \sigma_i^z \sigma_j^z \rangle \right). \tag{5}$$

Quantum Boltzmann Machine

- For a powerful RBM, NM connections are needed. N binary variables = size of binary feature space. M hidden units. For the MNIST dataset with 784 variables and an RBM with 100 hidden units, the number of pairwise couplers is 78400. A bipartite graph with at least the degree 784.
- QBM only works with binary data, or data needs to be binarized.
- A quantum bound (Golden-Thompson inequality) is needed to computed clamped expectations.

Quantum Variational Autoencoders

- A discrete variational autoencoder with a BM or QBM prior
- Quantum annealing is used for sampling from the Boltzmann prior
- The approximate inference is a factorial Bernoulli distribution
- Intractable likelihood evaluation, but a tractable evidence lower bound for optimization



$$\log p_\theta(\boldsymbol{x}) \geq \mathbb{E}_{z\sim q_\phi(\boldsymbol{z}|\boldsymbol{x})} \log \frac{p_\theta(\boldsymbol{x}|\boldsymbol{z})p_\theta(\boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})} \tag{6}$$

Quantum Variational Autoencoders

- With the right encoder and decoder, QVAE can be applied to any type of data
- The number of qubits and the connectivity of the qubits can be controlled.

Back-propagation with discrete variables

$$\nabla_\theta \mathbb{E}_{z\sim q_\phi(\boldsymbol{z}|\boldsymbol{x})} \log \frac{p_\theta(\boldsymbol{x}|\boldsymbol{z})p_\theta(\boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})} \tag{7}$$

$$= \mathbb{E}_{z\sim q_\phi(\boldsymbol{z}|\boldsymbol{x})} \nabla_\theta \log \frac{p_\theta(\boldsymbol{x}|\boldsymbol{z})p_\theta(\boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})} \tag{8}$$

$$\nabla_\phi \mathbb{E}_{z\sim q_\phi(\boldsymbol{z}|\boldsymbol{x})} \log \frac{p_\theta(\boldsymbol{x}|\boldsymbol{z})p_\theta(\boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})} \tag{9}$$

Reparameterization trick:

$$\nabla_\phi \mathbb{E}_{\rho\sim\mathcal{U}} \log \frac{p(\boldsymbol{x}|\boldsymbol{z}(\rho,\phi))p(\boldsymbol{z}(\rho,\phi))}{q(\boldsymbol{z}(\rho,\phi)|\boldsymbol{x})} \tag{10}$$

$$= \mathbb{E}_{\rho\sim\mathcal{U}} \nabla_\phi \log \frac{p(\boldsymbol{x}|\boldsymbol{z}(\rho,\phi))p(\boldsymbol{z}(\rho,\phi))}{q(\boldsymbol{z}(\rho,\phi)|\boldsymbol{x})} \tag{11}$$

Back-propagation with discrete variables

The gradient of inverse CDF for discrete variables is undefined.

$$\mathbb{E}_{\rho\sim\mathcal{U}}\nabla_\phi \log \frac{p(\boldsymbol{x}|\boldsymbol{z}(\rho,\phi))p(\boldsymbol{z}(\rho,\phi))}{q(\boldsymbol{z}(\rho,\phi)|\boldsymbol{x})} \tag{12}$$

Continuous relaxation $z \to \zeta$. Note: biased low-variance estimate of the gradient.

$$\mathbb{E}_{\rho\sim\mathcal{U}}\nabla_\phi \log \frac{p(\boldsymbol{x}|\zeta(\rho,\phi))p(\zeta(\rho,\phi))}{q(\zeta(\rho,\phi)|\boldsymbol{x})} \tag{13}$$

$$\zeta(\rho,\phi) = \sigma(\frac{l_\phi + \rho}{\tau}), \quad \rho \sim \mathcal{U} \tag{14}$$

Performance of simulated quantum variational autoencoder

Generative performance of QVAE+QBM

(b) Generated MNIST: QVAE, $\mathrm{QBM}_{64\times64}$, $\Gamma = 0$,

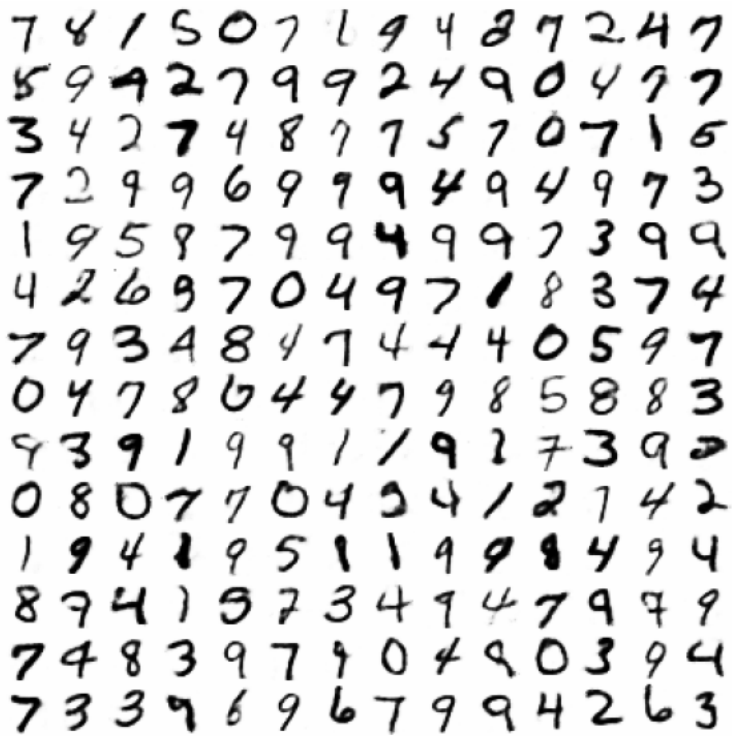
(d) Generated MNIST: QVAE, $\mathrm{QBM}_{64\times64}$, $\Gamma = 1$,

Quantitative performance

MNIST (static binarization)				
			ELBO	**Q-ELBO**
QVAE:	$\Gamma = 0$	$\text{RBM}_{16\times16}$	-109.3 ± 0.2	-109.3 ± 0.2
	$\Gamma = 1$	$\text{QBM}_{16\times16}$	-110.5	-120.6
	$\Gamma = 2$		-115.3	-135.8
QVAE:	$\Gamma = 0$	$\text{RBM}_{32\times32}$	-101.8	-101.8
	$\Gamma = 1$	$\text{QBM}_{32\times32}$	-103.6	-117.9
	$\Gamma = 2$		-112.1	-139.7
QVAE:	$\Gamma = 0$	$\text{RBM}_{64\times64}$	-105.7	-105.7
	$\Gamma = 1$	$\text{QBM}_{64\times64}$	-108.7	-133.9
	$\Gamma = 2$		-120.0	-165.2

Generative performance of QVAE+RBM

(a) Generated MNIST: DVAE, $\text{RBM}_{32\times32}$



(c) Generated MNIST: DVAE, $\text{RBM}_{256\times256}$

Quantitative performance

MNIST (static binarization)			
		ELBO	**LL**
DVAE	$\text{RBM}_{32\times32}$	-99.3 ± 0.2	-90.8 ± 0.2
	$\text{RBM}_{64\times64}$	-92.4	-85.5
	$\text{RBM}_{128\times128}$	-90.4	-84.7
	$\text{RBM}_{256\times256}$	$\mathbf{-89.2}$	$\mathbf{-83.5}$
VIMCO [88]			-91.9
NVIL [75]			-93.5
CONCRETE [82]			-85.7
GS [76]		-101.5	
RWS [89]			-88.9
REBAR [90]		-98.8	

PixelQVAE

- Combination of a QVAE with PixelCNN decoder
- Autoregressive decoder, latent variable model, variational inference

$$\log p(\boldsymbol{x}|\boldsymbol{z}) = \sum_i \log p(x_i|\boldsymbol{x}_{<i}, \boldsymbol{z}) \tag{15}$$

$$\log p(\boldsymbol{x}) \geq \mathbb{E}_{z\sim q(\boldsymbol{z}|\boldsymbol{x})} \log \frac{p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})}{q(\boldsymbol{z}|\boldsymbol{x})} \tag{16}$$

Results

	MNIST (Static)		MNIST (Dynamic)	
	NLL	KL	NLL	KL
PixelVAE++ Gaussian	78.66	6.86	78.01	4.2
PixelVAE++ RBM	**78.65**	7.62	**78.00**	5.05
VLAE	79.03		78.53	

	OMNIGLOT		Caltech 101	
	NLL	KL	NLL	KL
PixelVAE++ Gaussian	88.65	1.63	79.52	4.00
PixelVAE++ RBM	**88.29**	2.56	77.46	6.85
VLAE	89.83		**77.36**	

bpd	CIFAR10	
	NLL	KL
PixelVAE++ Gaussian	2.92	0.005
PixelVAE++ RBM	2.90	0.016
VLAE	2.95	
PixelCNN++	2.92	
PixelSNAIL	**2.85**	

Summary

- Powerful latent variables models can be hybridized with Quantum (assisted) Boltzmann Machines.
- The range of applications extends to that of all VAE models
- Quantum annealing can circumvent the issue of sampling from Boltzmann distributions and replace expensive MCMC methods.
- Currently, quantum annealing can achieve similar performance
- As the decoder gets more powerful, the role of latent variables fades away
- Sampling bias can prevent gaining any advantage from a QVAE.

Thank you!

Quantum Boltzmann Machine

- A hybrid ML algorithm that makes use of deep neural networks for encoding and decoding latent variables
- The prior distribution is a classical/quantum Boltzmann machine

$$E_{\mathbf{z}} = -\sum_{a} b_a z_a - \sum_{a,b} w_{ab} z_a z_b. \tag{17}$$

$$P_{\mathbf{v}} = Z^{-1} \sum_{\mathbf{h}} e^{-E_{\mathbf{z}}}, \qquad Z = \sum_{\mathbf{z}} e^{-E_{\mathbf{z}}}, \tag{18}$$

$$\mathcal{L} = -\sum_{\mathbf{v}} P_{\mathbf{v}}^{\text{data}} \log P_{\mathbf{v}}, \tag{19}$$

$$\mathcal{L} = -\sum_{\mathbf{v}} P_{\mathbf{v}}^{\text{data}} \log \frac{\sum_{\mathbf{h}} e^{-E_{\mathbf{z}}}}{\sum_{\mathbf{z}'} e^{-E_{\mathbf{z}'}}}. \tag{20}$$

$$\delta\theta = -\eta \partial_\theta \mathcal{L}, \tag{21}$$

$$\begin{aligned}
\partial_\theta \mathcal{L} &= \sum_{\mathbf{v}} P^{\text{data}}_{\mathbf{v}} \frac{\sum_{\mathbf{h}} \partial_\theta E_{\mathbf{z}} e^{-E_{\mathbf{z}}}}{\sum_{\mathbf{h}} e^{-E_{\mathbf{z}}}} - \frac{\sum_{\mathbf{z}} \partial_\theta E_{\mathbf{z}} e^{-E_{\mathbf{z}}}}{\sum_{\mathbf{z}} e^{-E_{\mathbf{z}}}} \\
&= \overline{\langle \partial_\theta E_{\mathbf{z}} \rangle_{\mathbf{v}}} - \langle \partial_\theta E_{\mathbf{z}} \rangle, \qquad (22)
\end{aligned}$$

$$\delta b_a = \eta \left(\overline{\langle z_a \rangle_{\mathbf{v}}} - \langle z_a \rangle \right), \qquad (23)$$

$$\delta w_{ab} = \eta \left(\overline{\langle z_a z_b \rangle_{\mathbf{v}}} - \langle z_a z_b \rangle \right). \qquad (24)$$

$$H = -\sum_a b_a \sigma_a^z - \sum_{a,b} w_{ab} \sigma_a^z \sigma_b^z. \tag{25}$$

$$\sigma_a^z \equiv \overbrace{I \otimes \ldots \otimes I}^{a-1} \otimes \sigma_z \otimes \overbrace{I \otimes \ldots \otimes I}^{N-a} \tag{26}$$

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \qquad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{27}$$

$$\rho = Z^{-1} e^{-H}. \tag{28}$$

$$P_{\mathbf{v}} = \mathrm{Tr}[\Lambda_{\mathbf{v}} \rho], \tag{29}$$

$$\Lambda_{\mathbf{v}} = \mathbf{v}\mathbf{v} \otimes \mathcal{I}_{\mathbf{h}}, \tag{30}$$

$$\mathbf{vv} \equiv \prod_{\nu} \left(\frac{1 + \mathbf{v}_\nu \sigma_\nu^z}{2} \right) \tag{31}$$

$$\sigma_a^x \equiv \overbrace{I \otimes \ldots \otimes I}^{a-1} \otimes \sigma_x \otimes \overbrace{I \otimes \ldots \otimes I}^{N-a}, \qquad \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},$$

$$\mathcal{L} = -\sum_{\mathbf{v}} P_{\mathbf{v}}^{\text{data}} \log \frac{\mathrm{Tr}\left[\Lambda_{\mathbf{v}} e^{-H}\right]}{\mathrm{Tr}\left[e^{-H}\right]}. \tag{32}$$

$$\partial_\theta \mathcal{L} = \sum_{\mathbf{v}} P_{\mathbf{v}}^{\text{data}} \left(\frac{\mathrm{Tr}\left[\Lambda_{\mathbf{v}} \partial_\theta e^{-H}\right]}{\mathrm{Tr}\left[\Lambda_{\mathbf{v}} e^{-H}\right]} - \frac{\mathrm{Tr}\left[\partial_\theta e^{-H}\right]}{\mathrm{Tr}\left[e^{-H}\right]} \right). \tag{33}$$

$$\partial_\theta e^{-H} = \sum_{m=1}^{n} e^{-m\delta\tau H} \left(-\partial_\theta H \delta\tau\right) e^{-(n-m)\delta\tau H}. \tag{34}$$

$$\partial_\theta e^{-H} = -\int_0^1 d\tau e^{-\tau H} \partial_\theta H e^{(\tau-1)H}. \tag{35}$$

$$\mathrm{Tr}\left[\partial_\theta e^{-H}\right] = -\mathrm{Tr}\left[\partial_\theta H e^{-H}\right], \tag{36}$$

$$\frac{\mathrm{Tr}\left[\partial_\theta e^{-H}\right]}{\mathrm{Tr}\left[e^{-H}\right]} = -\langle \partial_\theta H \rangle, \tag{37}$$

$$\frac{\mathrm{Tr}\left[\Lambda_{\mathbf{v}} \partial_\theta e^{-H}\right]}{\mathrm{Tr}\left[\Lambda_{\mathbf{v}} e^{-H}\right]} = -\int_0^1 dt \frac{\mathrm{Tr}\left[\Lambda_{\mathbf{v}} e^{-tH} \partial_\theta H e^{-(1-t)H}\right]}{\mathrm{Tr}\left[\Lambda_{\mathbf{v}} e^{-H}\right]}, \tag{38}$$

$$\mathrm{Tr}\left[e^A e^B\right] \geq \mathrm{Tr}\left[e^{A+B}\right], \tag{39}$$

$$P_{\mathbf{v}} = \frac{\mathrm{Tr}\left[e^{-H} e^{\ln \Lambda_{\mathbf{v}}}\right]}{\mathrm{Tr}\left[e^{-H}\right]} \geq \frac{\mathrm{Tr}\left[e^{-H+\ln \Lambda_{\mathbf{v}}}\right]}{\mathrm{Tr}\left[e^{-H}\right]}. \tag{40}$$

$$H_{\mathbf{v}} = H - \ln \Lambda_{\mathbf{v}}, \tag{41}$$

$$P_{\mathbf{v}} \geq \frac{\mathrm{Tr}\left[e^{-H_{\mathbf{v}}}\right]}{\mathrm{Tr}\left[e^{-H}\right]}. \tag{42}$$

$$H_{\mathbf{v}} \equiv H(\sigma_\nu^x = 0, \sigma_\nu^z = \mathbf{v}_\nu). \tag{43}$$

$$\mathcal{L} \leq \tilde{\mathcal{L}} \equiv -\sum_{\mathbf{v}} P_{\mathbf{v}}^{\mathrm{data}} \log \frac{\mathrm{Tr}\left[e^{-H_{\mathbf{v}}}\right]}{\mathrm{Tr}\left[e^{-H}\right]}. \tag{44}$$

$$\begin{aligned}\partial_\theta \tilde{\mathcal{L}} &= \sum_{\mathbf{v}} P^{\text{data}}_{\mathbf{v}} \left(\frac{\text{Tr}[e^{-H_{\mathbf{v}}} \partial_\theta H_{\mathbf{v}}]}{\text{Tr}[e^{-H_{\mathbf{v}}}]} - \frac{\text{Tr}[e^{-H} \partial_\theta H]}{\text{Tr}[e^{-H}]} \right), \\ &= \left(\overline{\langle \partial_\theta H_{\mathbf{v}} \rangle_{\mathbf{v}}} - \langle \partial_\theta H \rangle \right), \end{aligned} \tag{45}$$

where

$$\overline{\langle ... \rangle_{\mathbf{v}}} = \sum_{\mathbf{v}} P^{\text{data}}_{\mathbf{v}} \langle ... \rangle_{\mathbf{v}} = \sum_{\mathbf{v}} P^{\text{data}}_{\mathbf{v}} \frac{\text{Tr} e^{-H_{\mathbf{v}}} ...}{\text{Tr} e^{-H_{\mathbf{v}}}}. \tag{46}$$

Taking θ to be b_a, w_{ab}, and using $\delta\theta = -\eta \partial_\theta \tilde{\mathcal{L}}$, we obtain

$$\delta b_a = \eta \left(\overline{\langle \sigma^z_a \rangle_{\mathbf{v}}} - \langle \sigma^z_a \rangle \right), \tag{47}$$

$$\delta w_{ab} = \eta \left(\overline{\langle \sigma^z_a \sigma^z_b \rangle_{\mathbf{v}}} - \langle \sigma^z_a \sigma^z_b \rangle \right). \tag{48}$$

$$\delta\Gamma_a = \eta \left(\overline{\langle \sigma_a^x \rangle_{\mathbf{v}}} - \langle \sigma_a^x \rangle \right). \tag{49}$$

$$H_{\mathbf{v}} = -\sum_i \left(\Gamma_i \sigma_i^x + b_i^{\text{eff}}(\mathbf{v}) \sigma_i^z \right), \tag{50}$$

where $b_i^{\text{eff}}(\mathbf{v}) = b_i + \sum_\nu w_{i\nu} \mathbf{v}_\nu$. Expectations $\langle \sigma_i^z \rangle_{\mathbf{v}}$ entering (47) can be computed exactly:

$$\langle \sigma_i^z \rangle_{\mathbf{v}} = \frac{b_i^{\text{eff}}}{D_i} \tanh D_i, \tag{51}$$

where $D_i = \sqrt{\Gamma_i^2 + (b_i^{\text{eff}})^2}$. Notice that (51) reduces to the classical RBM expression,

$$\langle \sigma_i^z \rangle_{\mathbf{v}} = \tanh b_i^{\text{eff}}, \tag{52}$$

Results

	MNIST (Static)		MNIST (Dynamic)	
	NLL	KL	NLL	KL
PixelVAE++ Gaussian	78.66	6.86	78.01	4.2
PixelVAE++ RBM	**78.65**	7.62	**78.00**	5.05
VLAE	79.03		78.53	

	OMNIGLOT		Caltech 101	
	NLL	KL	NLL	KL
PixelVAE++ Gaussian	88.65	1.63	79.52	4.00
PixelVAE++ RBM	**88.29**	2.56	77.46	6.85
VLAE	89.83		**77.36**	

bpd	CIFAR10	
	NLL	KL
PixelVAE++ Gaussian	2.92	0.005
PixelVAE++ RBM	2.90	0.016
VLAE	2.95	
PixelCNN++	2.92	
PixelSNAIL	**2.85**	

Mutual information

- VAEs are notorious for deactivating latent variables
- various methods are used to overcome this issue, but when decoder gets stronger (PixelVAE) things get worse
- This issue can be expressed in terms of mutual information between the observations and latent variables. (quantified in terms of KL)
- Training PixelVAE naively will result in a **redundant** VAE
- Scheduled sampling, noisy auto-regressive channel, warm-up, ...
- Using discrete variables seems to help

Image generation

Graph visualization

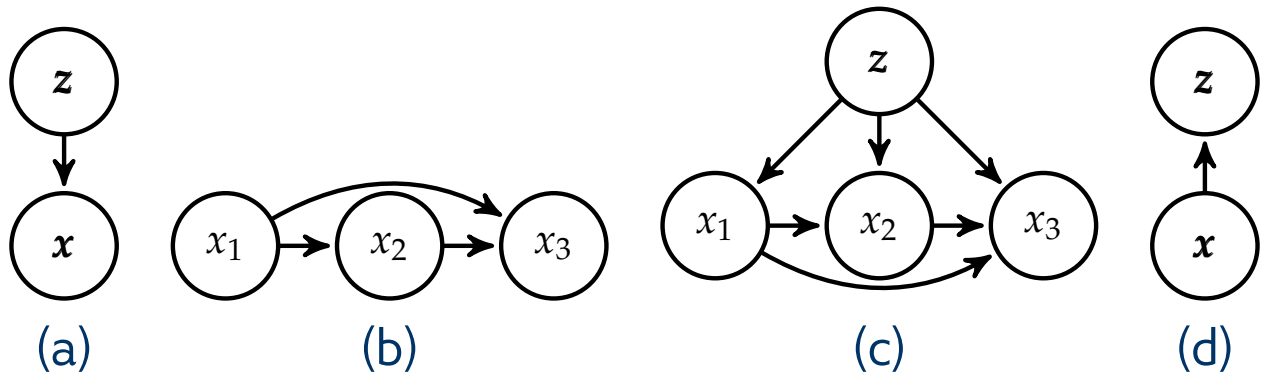
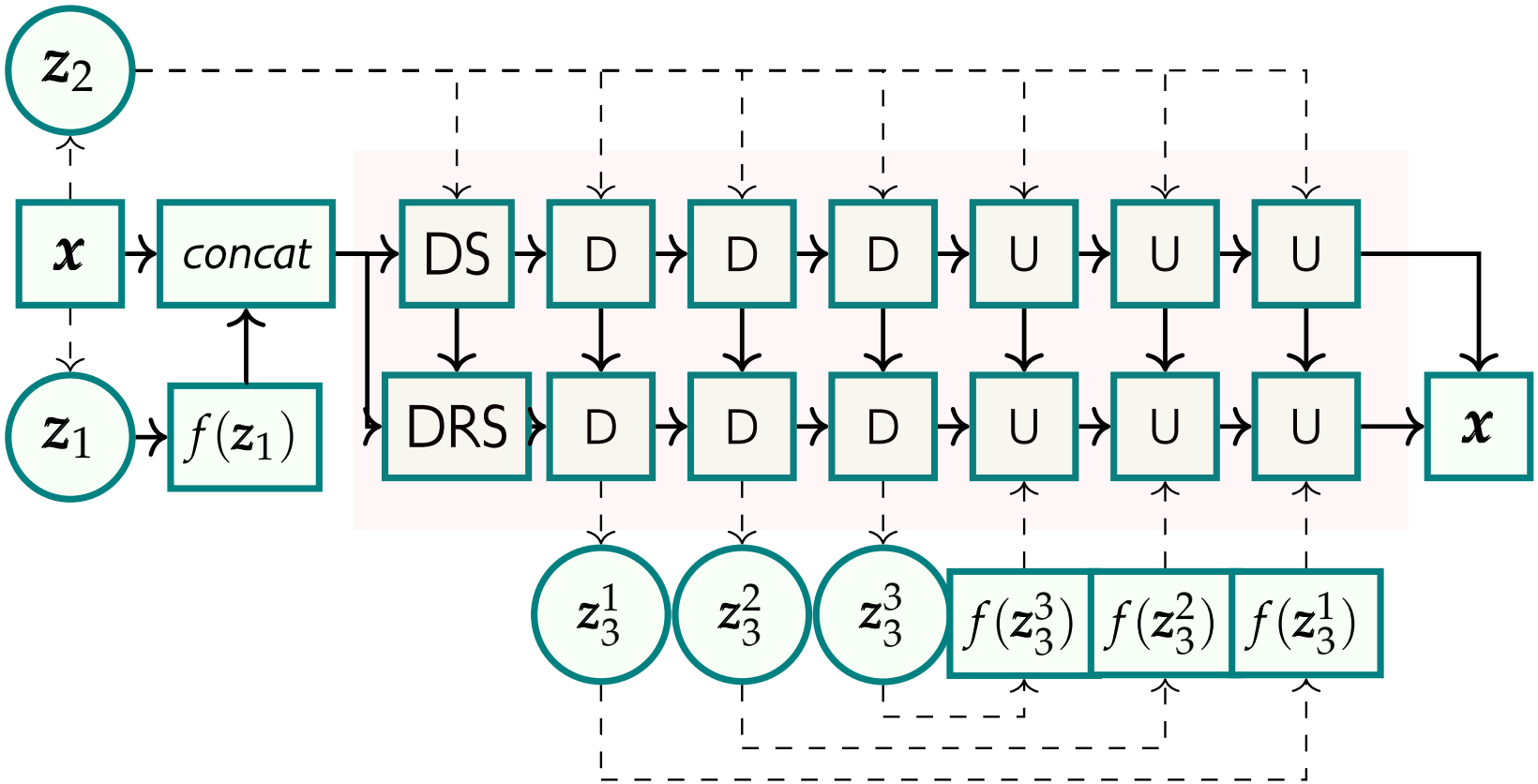


Figure: (a) A VAE model, (b) An autoregressive model, (c) A VAE with an autoregressive decoder, and (d) Inference model.

PixelVAE++



(a) PixelVAE++ architecture

PixelCNN (an Autoregressive model)

- Factorizes the joint distribution of the features to a product of conditionals
- Tractable likelihood. Fast training. High capacity model
- Slow sample generation. There are ways to improve.
- Conditional probabilities are parameterized by masked convolutional neural networks with shared parameters
- Useful when there are local temporal or spatial correlations

$$\log p(\boldsymbol{x}) = \sum_i \log p(x_i | \boldsymbol{x}_{<i}) \tag{53}$$