

Quantum Variational AutoEncoder



Quantum Computing for High
Energy Particle Physics

CERN, Geneva, November 5-6

Motivation

How to use current (and near-term) quantum devices to perform ‘state-of-the-art’ computations

Motivation

How to use current (and near-term) quantum devices to perform '**state-of-the-art**' computations



Develop quantum-classical hybrid algorithms

A Hybrid Approach to Quantum Machine Learning

Use early quantum devices to process encoded and compressed representations of the original data

A Hybrid Approach to Quantum Machine Learning

Use early quantum devices to process encoded and compressed representations of the original data



Use classical Deep Learning techniques to learn efficient encodings suitable to the quantum device

A Hybrid Approach to Quantum Machine Learning

Use early quantum devices to process encoded and compressed representations of the original data

+

Use classical Deep Learning techniques to learn efficient encodings suitable to the quantum device

=

Quantum Generative Models with Latent Variables

Quantum Generative Models with Latent Variables

- Reproduce data distribution by marginalizing over a set of latent (or unobserved) variables:

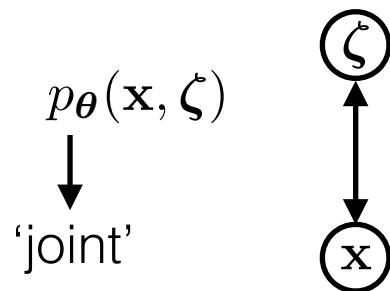
$$p_{\text{data}}(\mathbf{x}) \sim p_{\boldsymbol{\theta}}(\mathbf{x}) = \sum_{\zeta} p_{\boldsymbol{\theta}}(\mathbf{x}, \zeta)$$

Quantum Generative Models with Latent Variables

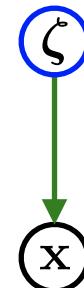
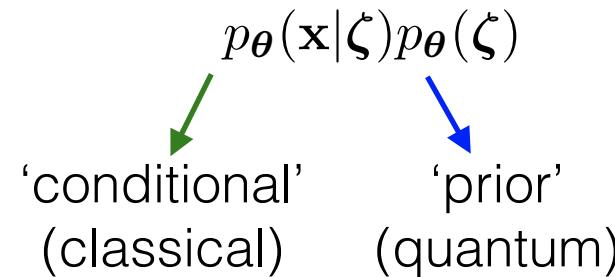
- Reproduce data distribution by marginalizing over a set of latent (or unobserved) variables:

$$p_{\text{data}}(\mathbf{x}) \sim p_{\boldsymbol{\theta}}(\mathbf{x}) = \sum_{\zeta} p_{\boldsymbol{\theta}}(\mathbf{x}, \zeta)$$

Undirected:



Directed:

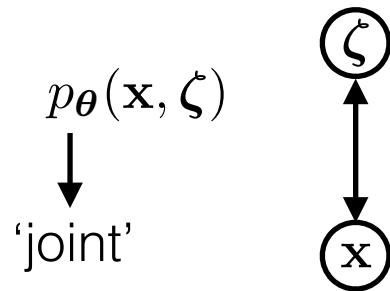


Quantum Generative Models with Latent Variables

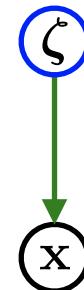
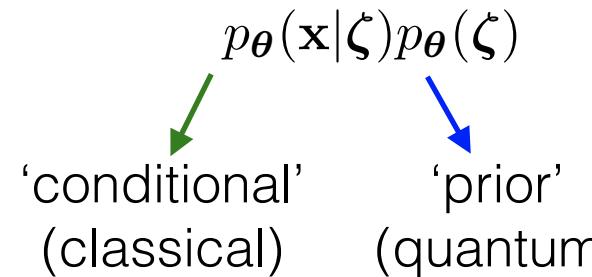
- Reproduce data distribution by marginalizing over a set of latent (or unobserved) variables:

$$p_{\text{data}}(\mathbf{x}) \sim p_{\boldsymbol{\theta}}(\mathbf{x}) = \sum_{\zeta} p_{\boldsymbol{\theta}}(\mathbf{x}, \zeta)$$

Undirected:



Directed:



- ‘Quantum Supremacy’ could be achieved in the gate model with sampling applications [Harrow and Montanaro, Nature 549]
- Sampling with quantum annealers can replace costly quantum Monte Carlo simulations

[Harris et al., Science 361; King et al., Nature 560]

Quantum Generative Models for HEP

1. Quantum priors may be ideally suited to model quantum processes such as high-energy particle collisions

Quantum Generative Models for HEP

1. Quantum priors may be ideally suited to model quantum processes such as high-energy particle collisions
2. Quantum generative models may provide a computational advantage in training state-of-the-art generative models

Quantum Generative Models for HEP

1. Quantum priors may be ideally suited to model quantum processes such as high-energy particle collisions
2. Quantum generative models may provide a computational advantage in training state-of-the-art generative models
 - a. Unsupervised learning: improve data analyses exploiting huge amount of unlabeled data produced in HEP collisions

[Andreassen et al., arXiv:1804.09720]

Quantum Generative Models for HEP

1. Quantum priors may be ideally suited to model quantum processes such as high-energy particle collisions
2. Quantum generative models may provide a computational advantage in training state-of-the-art generative models
 - a. Unsupervised learning: improve data analyses exploiting huge amount of unlabeled data produced in HEP collisions
[Andreassen et al., arXiv:1804.09720]
 - b. Learning new physics (e.g., anomaly detection)
[D'Agnolo and Wulzer, arXiv:1806.02350]

Quantum Generative Models for HEP

1. Quantum priors may be ideally suited to model quantum processes such as high-energy particle collisions
2. Quantum generative models may provide a computational advantage in training state-of-the-art generative models
 - a. Unsupervised learning: improve data analyses exploiting huge amount of unlabeled data produced in HEP collisions
[Andreassen et al., arXiv:1804.09720]
 - b. Learning new physics (e.g., anomaly detection)
[D'Agnolo and Wulzer, arXiv:1806.02350]
 - c. Detector simulations: compress raw (and sparse) data from detectors in a controlled (but unsupervised) way
[Paganini et al., PRL 120, 042003]

Quantum Generative Models for HEP

1. Quantum priors may be ideally suited to model quantum processes such as high-energy particle collisions
2. Quantum generative models may provide a computational advantage in training state-of-the-art generative models
 - a. Unsupervised learning: improve data analyses exploiting huge amount of unlabeled data produced in HEP collisions
[Andreassen et al., arXiv:1804.09720]
 - b. Learning new physics (e.g., anomaly detection)
[D'Agnolo and Wulzer, arXiv:1806.02350]
 - c. Detector simulations: compress raw (and sparse) data from detectors in a controlled (but unsupervised) way
[Paganini et al., PRL 120, 042003]
 - d. Lossy data compression!?

Goals

1. Demonstrate QVAE is a class of potentially state-of-the-art generative models



Goals

1. Demonstrate QVAE is a class of potentially state-of-the-art generative models 
2. Demonstrate QVAE can be trained with quantum annealers on non-trivial datasets (MNIST, ..., HEP!?) 

Goals

1. Demonstrate QVAE is a class of potentially state-of-the-art generative models 
2. Demonstrate QVAE can be trained with quantum annealers on non-trivial datasets (MNIST, ..., HEP!?) 
3. Identify a path to quantum advantage by working on realistic applications and with existing quantum devices 

Quantum Variational AutoEncoder

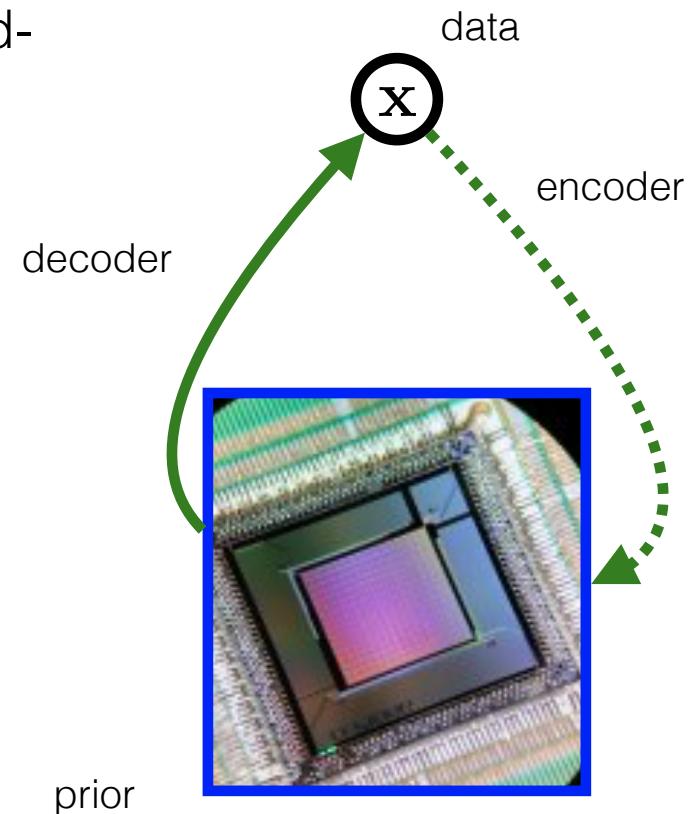
Quantum Variational AutoEncoder (QVAE):
quantum-classical hybrid generative model with latent variables
suitable for quantum annealing devices

[Khoshaman, Vinci et al., QST, 4, 1]

Quantum Variational AutoEncoder

Quantum Variational AutoEncoder (QVAE):
quantum-classical hybrid generative model with latent variables
suitable for quantum annealing devices

1. A classical 'AutoEncoding' structure: forward-propagation through deep neural networks



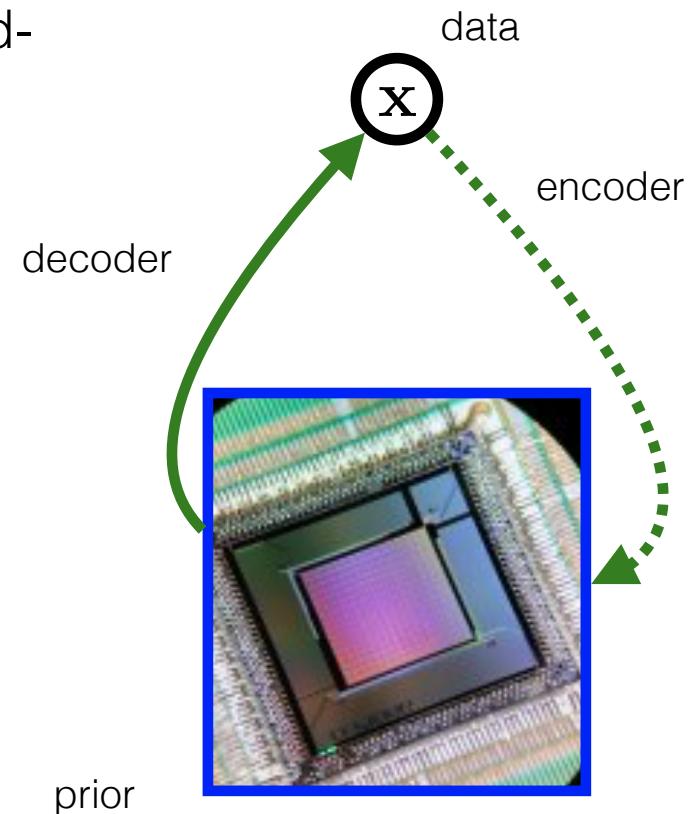
[Khoshaman, Vinci et al., QST, 4, 1]

Quantum Variational AutoEncoder

Quantum Variational AutoEncoder (QVAE):
quantum-classical hybrid generative model with latent variables
suitable for quantum annealing devices

1. A classical ‘AutoEncoding’ structure: forward-propagation through deep neural networks

2. A quantum ‘Prior’: generative process involving sampling from quantum Boltzmann distributions

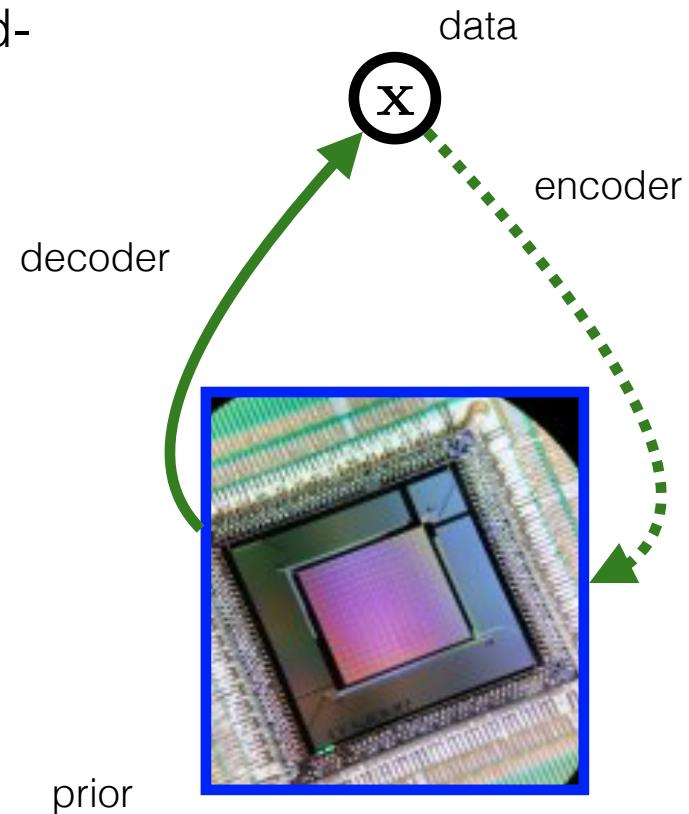


[Khoshaman, Vinci et al., QST, 4, 1]

Quantum Variational AutoEncoder

Quantum Variational AutoEncoder (QVAE):
quantum-classical hybrid generative model with latent variables
suitable for quantum annealing devices

1. A classical ‘AutoEncoding’ structure: forward-propagation through deep neural networks
2. A quantum ‘Prior’: generative process involving sampling from quantum Boltzmann distributions
3. A hybrid framework efficiently trained using Stochastic Gradient Descent

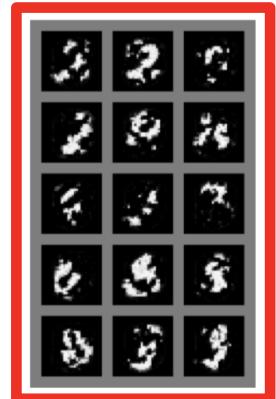


[Khoshaman, Vinci et al., QST, 4, 1]

Training Generative Models with Quantum Annealers

- (Quantum) Boltzmann Machine:
 - Not-hybrid, local connectivity major limitation

[Dumoulin et al.,
arXiv:1312.5258]

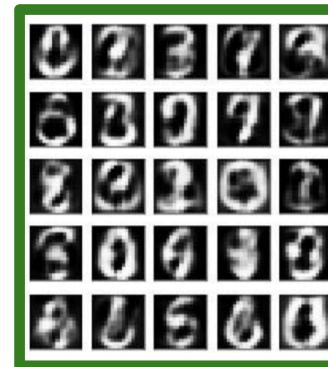
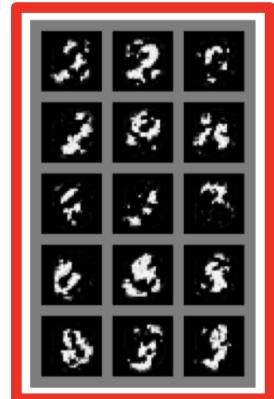


[MNIST 50k handwritten digits]

Training Generative Models with Quantum Annealers

- (Quantum) Boltzmann Machine:
 - Not-hybrid, local connectivity major limitation
- (Quantum Assisted) Helmholtz Machine:
 - Hybrid, lacks a well-defined loss function, inefficient training

[Dumoulin et al.,
arXiv:1312.5258]



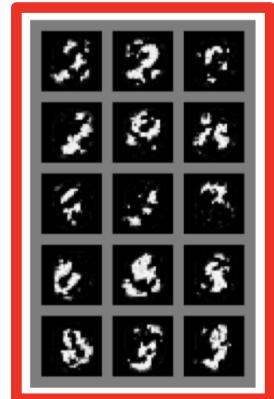
[Benedetti et al.,
QST, 3, 3]

[MNIST 50k handwritten digits]

Training Generative Models with Quantum Annealers

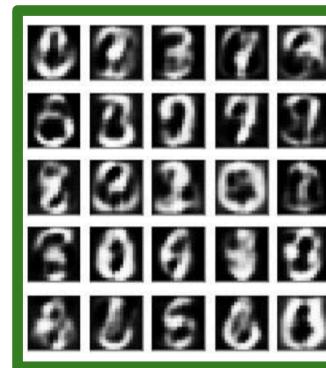
- (Quantum) Boltzmann Machine:
 - Not-hybrid, local connectivity major limitation

[Dumoulin et al.,
arXiv:1312.5258]



- (Quantum Assisted) Helmholtz Machine:
 - Hybrid, lacks a well-defined loss function, inefficient training

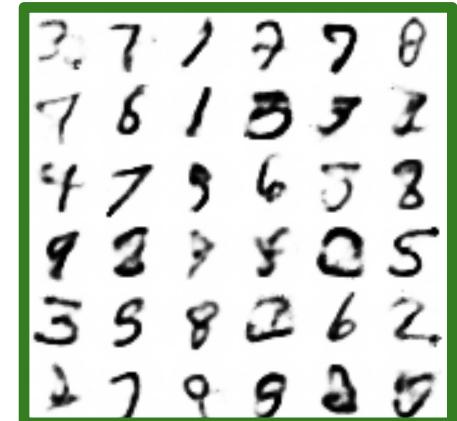
[Benedetti et al.,
QST, 3, 3]



- QVAE:
 - Hybrid, efficient training, enable state-of-the-art generative modelling

[MNIST 50k handwritten digits]

[Khoshaman, Vinci
et al., QST, 4, 1]



Quantum Annealers as Boltzmann Samplers

- Quantum annealers simulate a transverse field Ising model...

$$H(s) = -A(s) \sum_i \sigma_i^x + B(s) \left(\sum_{i < j} J_{ij} \sigma_i^z \sigma_j^z + \sum_i h_i \sigma_i^z \right)$$

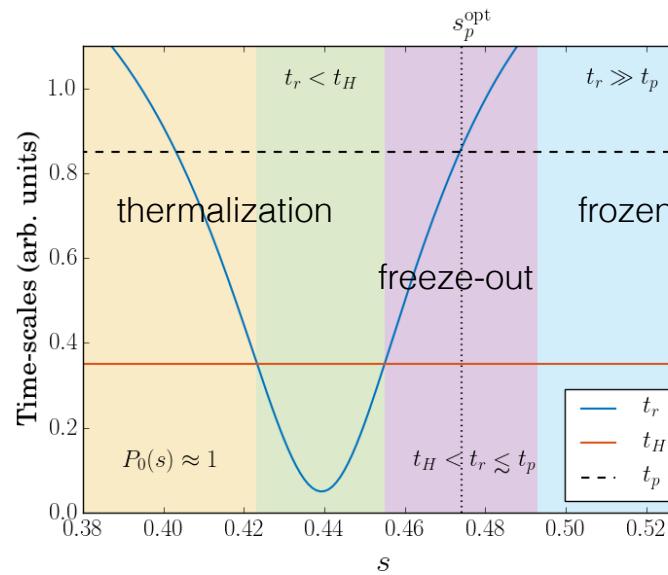
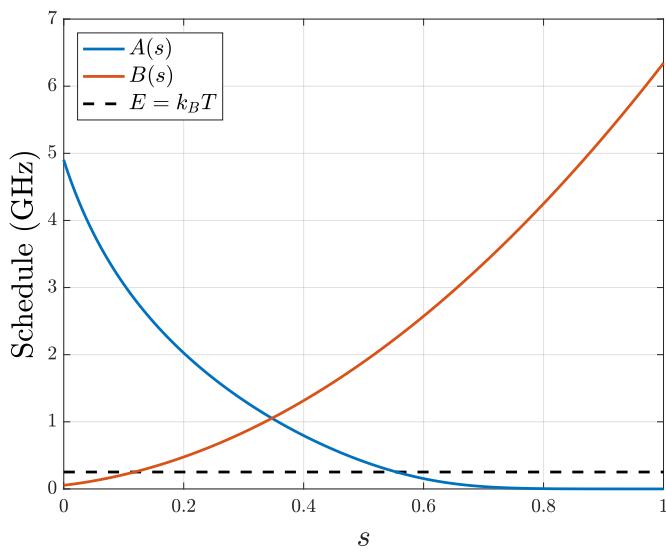
...immersed in a thermal bath (noisy environment)

Quantum Annealers as Boltzmann Samplers

- Quantum annealers simulate a transverse field Ising model...

$$H(s) = -A(s) \sum_i \sigma_i^x + B(s) \left(\sum_{i < j} J_{ij} \sigma_i^z \sigma_j^z + \sum_i h_i \sigma_i^z \right)$$

...immersed in a thermal bath (noisy environment)



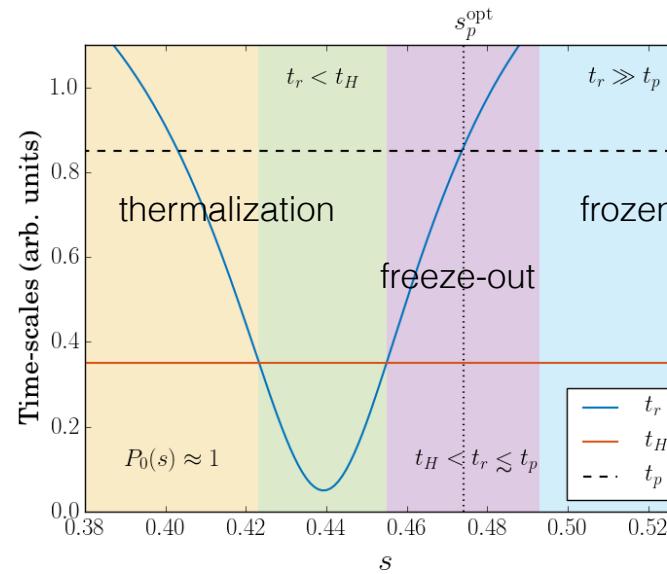
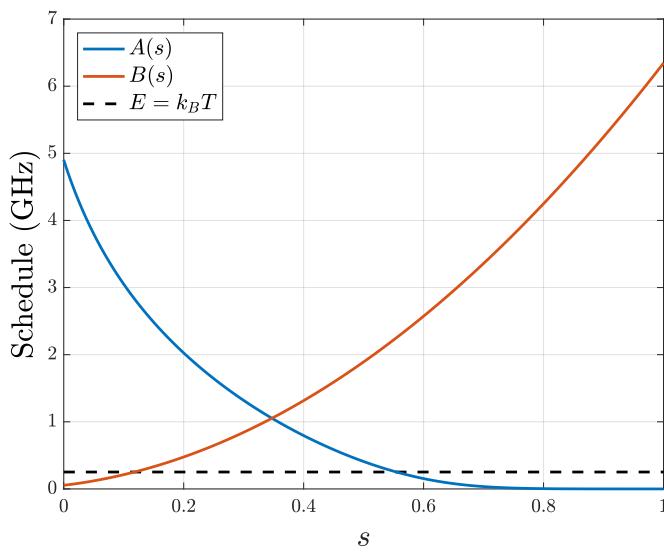
[Amin, PRA 92;
Marshall et al.,
arXiv:1810.05881]

Quantum Annealers as Boltzmann Samplers

- Quantum annealers simulate a transverse field Ising model...

$$H(s) = -A(s) \sum_i \sigma_i^x + B(s) \left(\sum_{i < j} J_{ij} \sigma_i^z \sigma_j^z + \sum_i h_i \sigma_i^z \right)$$

...immersed in a thermal bath (noisy environment)



[Amin, PRA 92;
Marshall et al.,
arXiv:1810.05881]

- Requires additional control of annealing schedules: pauses, fast quenches, reverse anneals...
- The use of D-Wave quantum annealers as quantum Boltzmann samplers recently demonstrated in material simulations

Variational AutoEncoders (VAE)

- VAE is a class of directed generative models with latent variables:

$$p_{\theta}(\mathbf{x}, \zeta) = p_{\theta}(\mathbf{x}|\zeta)p_{\theta}(\zeta)$$

‘decoder’ ‘prior’



θ : parameters
of the generative
model

- Encode useful representations of the data in the latent space

Variational AutoEncoders (VAE)

- VAE is a class of directed generative models with latent variables:

$$p_{\theta}(\mathbf{x}, \zeta) = p_{\theta}(\mathbf{x}|\zeta)p_{\theta}(\zeta)$$

'decoder' 'prior'



θ : parameters
of the generative
model

- Encode useful representations of the data in the latent space



- Exact inference (marginalization over latent variables) is intractable

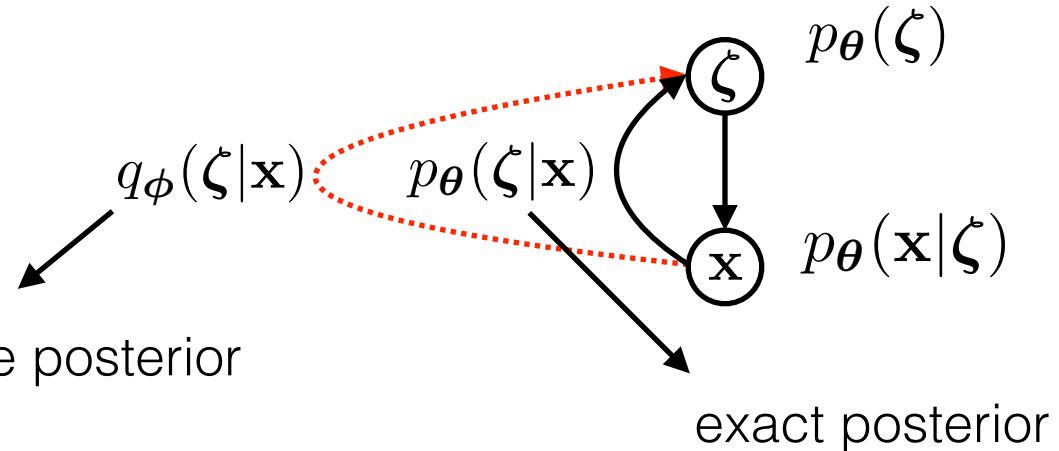


Variational AutoEncoders

1. Variational inference:

ϕ : parameters of the inference model

approximate posterior
(‘encoder’)

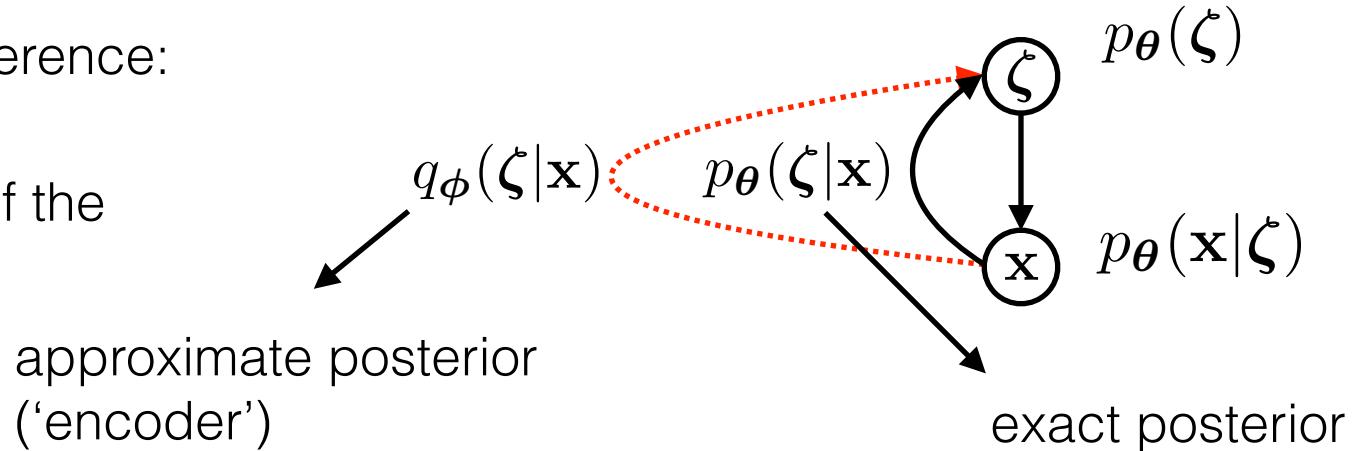


exact posterior

Variational AutoEncoders

1. Variational inference:

ϕ : parameters of the inference model



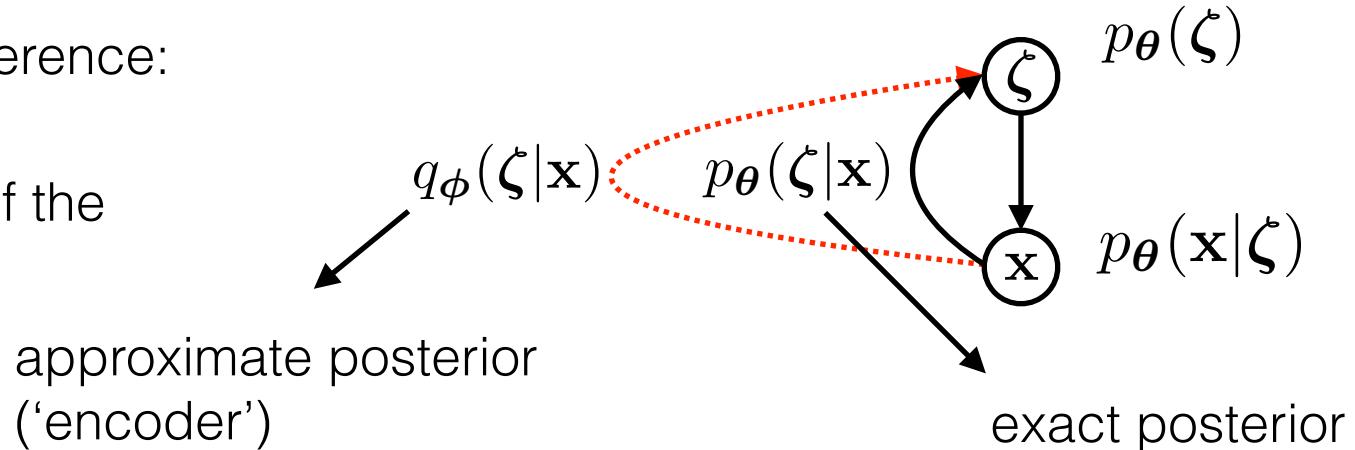
- Evidence Lower BOund (ELBO)

$$\text{ELBO} = \text{LL} - \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}} \left[\mathbb{E}_{\zeta \sim q_{\phi}(\zeta|\mathbf{x})} \left[\log \frac{q_{\phi}(\zeta|\mathbf{x})}{p_{\theta}(\zeta|\mathbf{x})} \right] \right] \leq \text{LL}$$

Variational AutoEncoders

1. Variational inference:

ϕ : parameters of the inference model



- Evidence Lower BOund (ELBO)

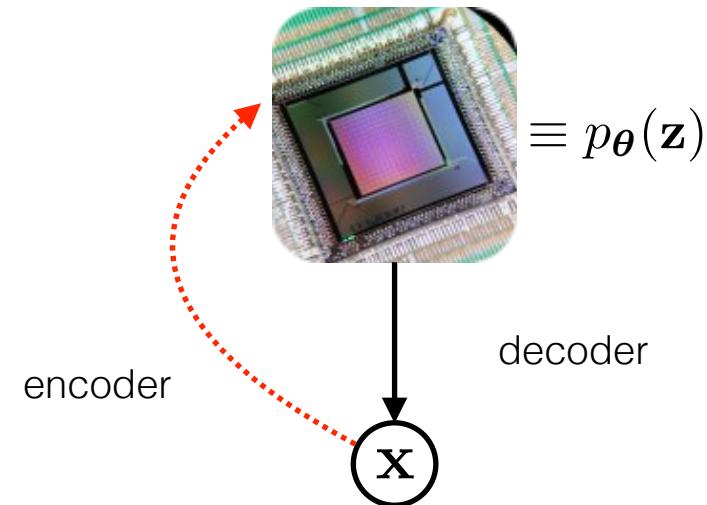
$$\text{ELBO} = \mathbf{LL} - \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}} \left[\mathbb{E}_{\zeta \sim q_{\phi}(\zeta|\mathbf{x})} \left[\log \frac{q_{\phi}(\zeta|\mathbf{x})}{p_{\theta}(\zeta|\mathbf{x})} \right] \right] \leq \mathbf{LL}$$

2. Reparameterization Trick:

$$\partial_{\phi} \mathbb{E}_{\zeta \sim q_{\phi}} [f(\zeta)] = \mathbb{E}_{\rho \sim p(\rho)} [\partial_{\phi} f(\zeta(\phi, \rho))]$$

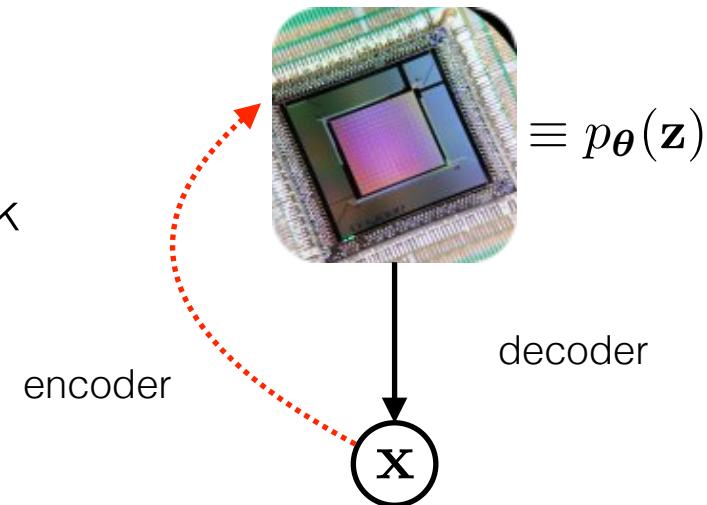
Quantum Variational AutoEncoders

- QVAE: prior is implemented as a Quantum Boltzmann Machine
 - Physical implementation with quantum annealers [Amin et al., PRX 8]



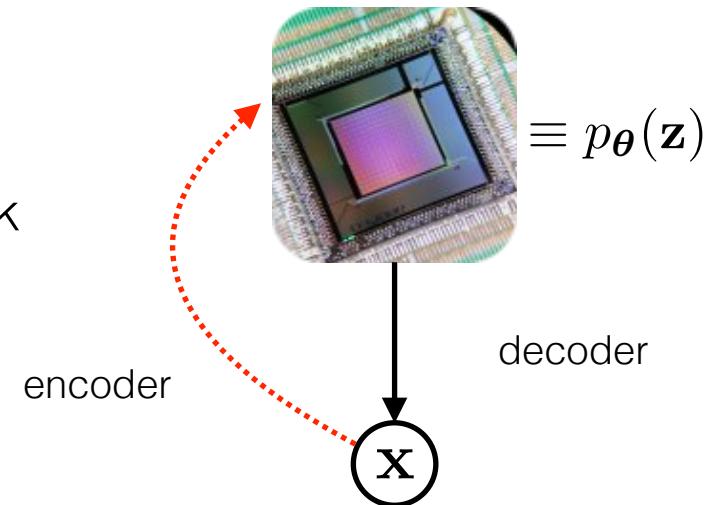
Quantum Variational AutoEncoders

- QVAE: prior is implemented as a Quantum Boltzmann Machine
 - Physical implementation with quantum annealers [Amin et al., PRX 8]
- Need to generalize the reparameterization trick to discrete variables ($\zeta \rightarrow \mathbf{z}$)



Quantum Variational AutoEncoders

- QVAE: prior is implemented as a Quantum Boltzmann Machine
 - Physical implementation with quantum annealers [Amin et al., PRX 8]



- Need to generalize the reparameterization trick to discrete variables ($\zeta \rightarrow \mathbf{z}$)

- Developed at D-Wave:

- Back-propagation through smoothing distributions (DVAE, DVAE++) [Rolfe, arXiv:1609.02200; Vahdat et al., arXiv:1802.04920]
- Back-propagation through smoothed, continuous variables (Gumbel/Softmax, GumBolt)

[Jang and Maddison, arXiv:1611.00712; Khoshaman and Amin, arXiv:1805.07349]

QVAE: VAE with Quantum Boltzmann Machine

- QBM: reproduces data distribution as a thermal distribution of a quantum spin-system

$$p_{\boldsymbol{\theta}}(\mathbf{z}) = \text{Tr}[\Lambda_{\mathbf{z}} e^{-\mathcal{H}_{\boldsymbol{\theta}}}] / Z_{\boldsymbol{\theta}}, \quad Z_{\boldsymbol{\theta}} = \text{Tr}[e^{-\mathcal{H}_{\boldsymbol{\theta}}}]$$

$\Lambda_{\mathbf{z}}$: projector on the state \mathbf{z}

$$\mathcal{H}_{\boldsymbol{\theta}} = \sum_l \sigma_l^x \Gamma_l + \sum_l \sigma_l^z h_l + \sum_{l < m} W_{lm} \sigma_l^z \sigma_m^z$$

$\sigma^{x,y,z}$: Pauli matrices

[Amin et al., PRX 8]

QVAE: VAE with Quantum Boltzmann Machine

- QBM: reproduces data distribution as a thermal distribution of a quantum spin-system

$$p_{\boldsymbol{\theta}}(\mathbf{z}) = \text{Tr}[\Lambda_{\mathbf{z}} e^{-\mathcal{H}_{\boldsymbol{\theta}}}] / Z_{\boldsymbol{\theta}}, \quad Z_{\boldsymbol{\theta}} = \text{Tr}[e^{-\mathcal{H}_{\boldsymbol{\theta}}}]$$

$\Lambda_{\mathbf{z}}$: projector on the state \mathbf{z}

$$\mathcal{H}_{\boldsymbol{\theta}} = \sum_l \sigma_l^x \Gamma_l + \sum_l \sigma_l^z h_l + \sum_{l < m} W_{lm} \sigma_l^z \sigma_m^z$$

$\sigma^{x,y,z}$: Pauli matrices

- Maximize negative quantum cross-entropy

$$-H(q_{\phi}, p_{\boldsymbol{\theta}}) = \mathbb{E}_{\mathbf{z} \sim q_{\phi}} [\log p_{\boldsymbol{\theta}}(\mathbf{z})] = \mathbb{E}_{\mathbf{z} \sim q_{\phi}} [\log(\text{Tr}[\Lambda_{\mathbf{z}} e^{-\mathcal{H}_{\boldsymbol{\theta}}}])] - \log Z_{\boldsymbol{\theta}}$$

[Amin et al., PRX 8]

QVAE: VAE with Quantum Boltzmann Machine

- QBM: reproduces data distribution as a thermal distribution of a quantum spin-system

$$p_{\boldsymbol{\theta}}(\mathbf{z}) = \text{Tr}[\Lambda_{\mathbf{z}} e^{-\mathcal{H}_{\boldsymbol{\theta}}}] / Z_{\boldsymbol{\theta}}, \quad Z_{\boldsymbol{\theta}} = \text{Tr}[e^{-\mathcal{H}_{\boldsymbol{\theta}}}]$$

$\Lambda_{\mathbf{z}}$: projector on the state \mathbf{z}

$$\mathcal{H}_{\boldsymbol{\theta}} = \sum_l \sigma_l^x \Gamma_l + \sum_l \sigma_l^z h_l + \sum_{l < m} W_{lm} \sigma_l^z \sigma_m^z$$

$\sigma^{x,y,z}$: Pauli matrices

- Maximize negative quantum cross-entropy

$$-H(q_{\phi}, p_{\boldsymbol{\theta}}) = \mathbb{E}_{\mathbf{z} \sim q_{\phi}} [\log p_{\boldsymbol{\theta}}(\mathbf{z})] = \mathbb{E}_{\mathbf{z} \sim q_{\phi}} [\log(\text{Tr}[\Lambda_{\mathbf{z}} e^{-\mathcal{H}_{\boldsymbol{\theta}}}])] - \log Z_{\boldsymbol{\theta}}$$

- Negative phase: estimated via sampling $\partial \log Z_{\boldsymbol{\theta}} = \mathbb{E}_{\mathbf{z} \sim p_{\boldsymbol{\theta}}} [\partial \mathcal{H}_{\boldsymbol{\theta}}(\mathbf{z})]$

[Amin et al., PRX 8]

QVAE: VAE with Quantum Boltzmann Machine

- QBM: reproduces data distribution as a thermal distribution of a quantum spin-system

$$p_{\boldsymbol{\theta}}(\mathbf{z}) = \text{Tr}[\Lambda_{\mathbf{z}} e^{-\mathcal{H}_{\boldsymbol{\theta}}}] / Z_{\boldsymbol{\theta}}, \quad Z_{\boldsymbol{\theta}} = \text{Tr}[e^{-\mathcal{H}_{\boldsymbol{\theta}}}]$$

$\Lambda_{\mathbf{z}}$: projector on the state \mathbf{z}

$$\mathcal{H}_{\boldsymbol{\theta}} = \sum_l \sigma_l^x \Gamma_l + \sum_l \sigma_l^z h_l + \sum_{l < m} W_{lm} \sigma_l^z \sigma_m^z$$

$\sigma^{x,y,z}$: Pauli matrices

- Maximize negative quantum cross-entropy

$$-H(q_{\phi}, p_{\boldsymbol{\theta}}) = \mathbb{E}_{\mathbf{z} \sim q_{\phi}} [\log p_{\boldsymbol{\theta}}(\mathbf{z})] = \mathbb{E}_{\mathbf{z} \sim q_{\phi}} [\log(\text{Tr}[\Lambda_{\mathbf{z}} e^{-\mathcal{H}_{\boldsymbol{\theta}}}])] - \log Z_{\boldsymbol{\theta}}$$

- Negative phase: estimated via sampling $\partial \log Z_{\boldsymbol{\theta}} = \mathbb{E}_{\mathbf{z} \sim p_{\boldsymbol{\theta}}} [\partial \mathcal{H}_{\boldsymbol{\theta}}(\mathbf{z})]$
- Positive phase: intractable 

[Amin et al., PRX 8]

QVAE: Quantum ELBO (Q-ELBO)

- Golden-Thompson inequality

- Given any two Hermitian matrices: $\text{Tr}[e^A e^B] \geq \text{Tr}[e^{A+B}]$

$$\mathbb{E}_{\mathbf{z} \sim q_\phi} [\log(\text{Tr}[\Lambda_{\mathbf{z}} e^{-\mathcal{H}_\theta}])] \geq \mathbb{E}_{\mathbf{z} \sim q_\phi} [\log(\text{Tr}[e^{-\mathcal{H}_\theta + \log \Lambda_{\mathbf{z}}}])] = -\mathbb{E}_{\mathbf{z} \sim q_\phi} [\mathcal{H}_\theta(\mathbf{z})]$$

Golden-Thompson

reduces to a sum
over classical states

QVAE: Quantum ELBO (Q-ELBO)

- Golden-Thompson inequality

- Given any two Hermitian matrices: $\text{Tr}[e^A e^B] \geq \text{Tr}[e^{A+B}]$

$$\mathbb{E}_{\mathbf{z} \sim q_{\phi}} [\log(\text{Tr}[\Lambda_{\mathbf{z}} e^{-\mathcal{H}_{\theta}}])] \geq \mathbb{E}_{\mathbf{z} \sim q_{\phi}} [\log(\text{Tr}[e^{-\mathcal{H}_{\theta} + \log \Lambda_{\mathbf{z}}}])] = -\mathbb{E}_{\mathbf{z} \sim q_{\phi}} [\mathcal{H}_{\theta}(\mathbf{z})]$$

Golden-Thompson

reduces to a sum
over classical states

- QVAE is trained maximizing the Q-ELBO: $\text{LL} \geq \text{ELBO} \geq \text{Q} - \text{ELBO}$

- Biased, transverse field cannot be trained 

- Derivatives easily evaluated via sampling 

$$\mathbb{E}_{\mathbf{z} \sim q_{\phi}} [\partial \mathcal{H}_{\theta}(\mathbf{z})] - \mathbb{E}_{\mathbf{z} \sim p_{\theta}} [\partial \mathcal{H}_{\theta}(\mathbf{z})]$$

sample from decoder  sample with Quantum Monte Carlo, quantum annealers 

QVAE: Testing Training via Q-ELBO

- QVAE can be trained well despite the use of a looser bound
 - Used population-annealed CT-QMC (best method, still very slow)
 - Scale-up with quantum annealers

Size	Γ	MNIST (static binarization)		Epochs
		ELBO	Q-ELBO	
QBM _{16×16} :	0	-109.3	-109.3	800
	1	-110.5	-120.6	
	2	-115.3	-135.8	
QBM _{32×32} :	0	-101.8	-101.8	250
	1	-103.6	-117.9	
	2	-112.1	-139.7	
QBM _{64×64} :	0	-105.7	-105.7	50
	1	-108.7	-133.9	
	2	-120.0	-165.2	

QVAE: Testing Training via Q-ELBO

- QVAE can be trained well despite the use of a looser bound
 - Used population-annealed CT-QMC (best method, still very slow)
 - Scale-up with quantum annealers

Size	Γ	MNIST (static binarization)		Epochs
		ELBO	Q-ELBO	
QBM _{16×16} :	0	-109.3	-109.3	800
	1	-110.5	-120.6	
	2	-115.3	-135.8	
QBM _{32×32} :	0	-101.8	-101.8	250
	1	-103.6	-117.9	
	2	-112.1	-139.7	
QBM _{64×64} :	0	-105.7	-105.7	50
	1	-108.7	-133.9	
	2	-120.0	-165.2	

QBM_{64×64}

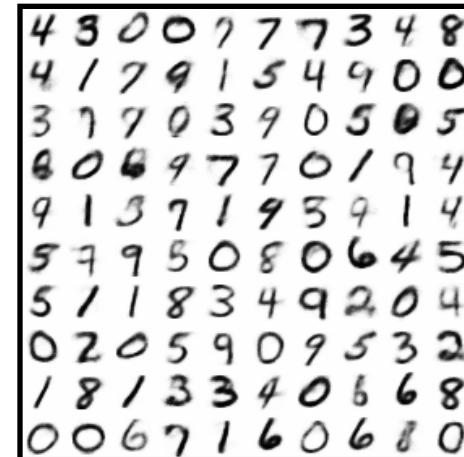
[Khoshaman, Vinci et al., QST, 4, 1]



$\Gamma = 0$



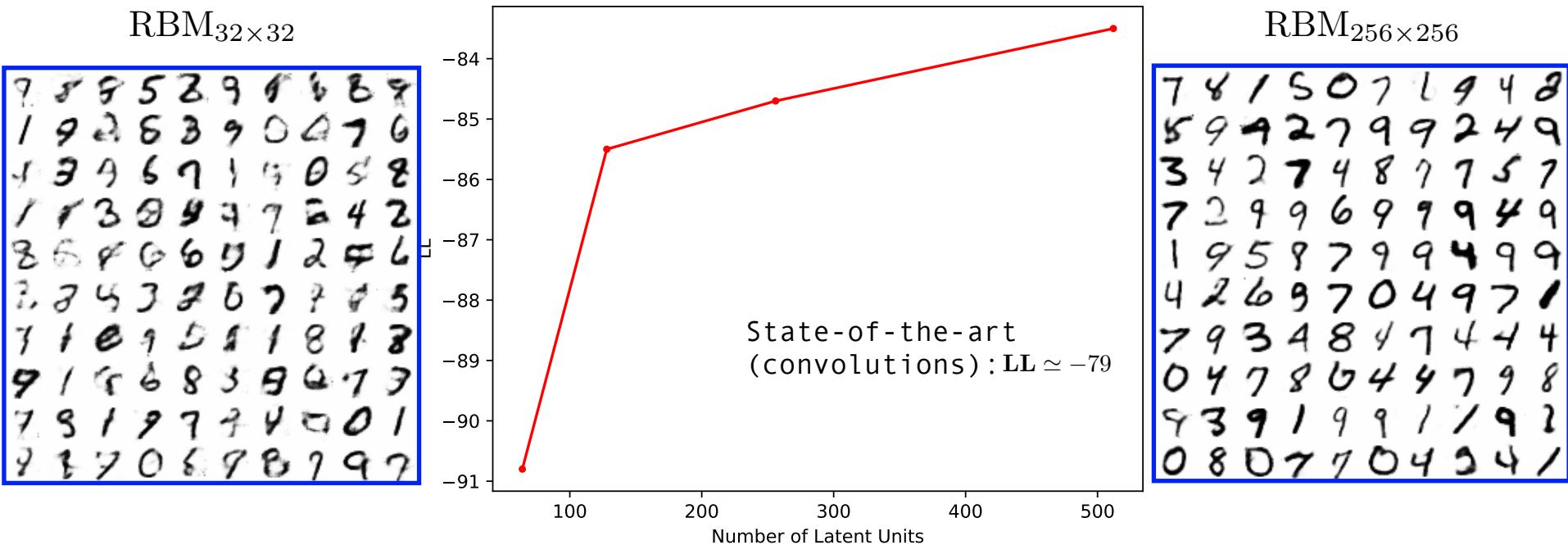
$\Gamma = 1$



$\Gamma = 2$

Classical Limit (DVAE): Testing the Limits of the Model

- VAE equipped with RBM is a powerful generative model

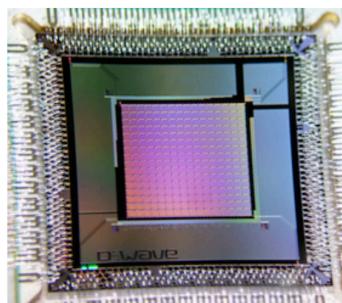


- ‘Vanilla’ implementation of DVAE:
 - AutoEncoder: fully connected deep neural nets
 - RBM samples generated with ‘persistent contrasting divergence’

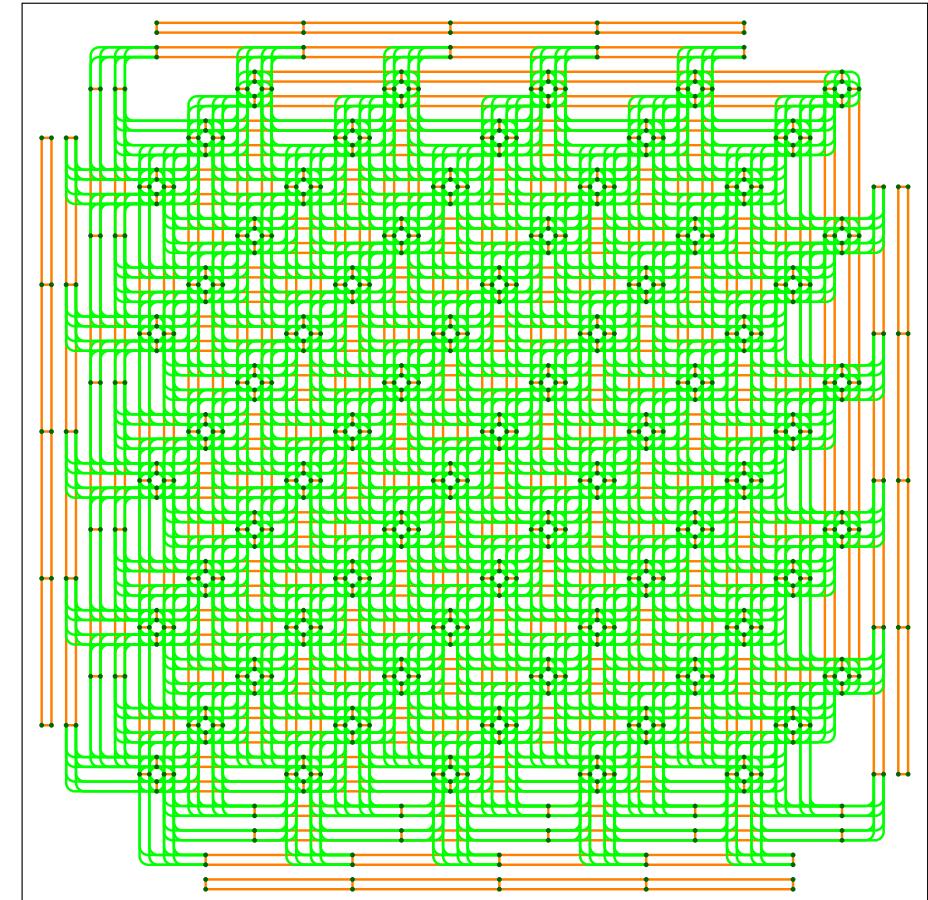
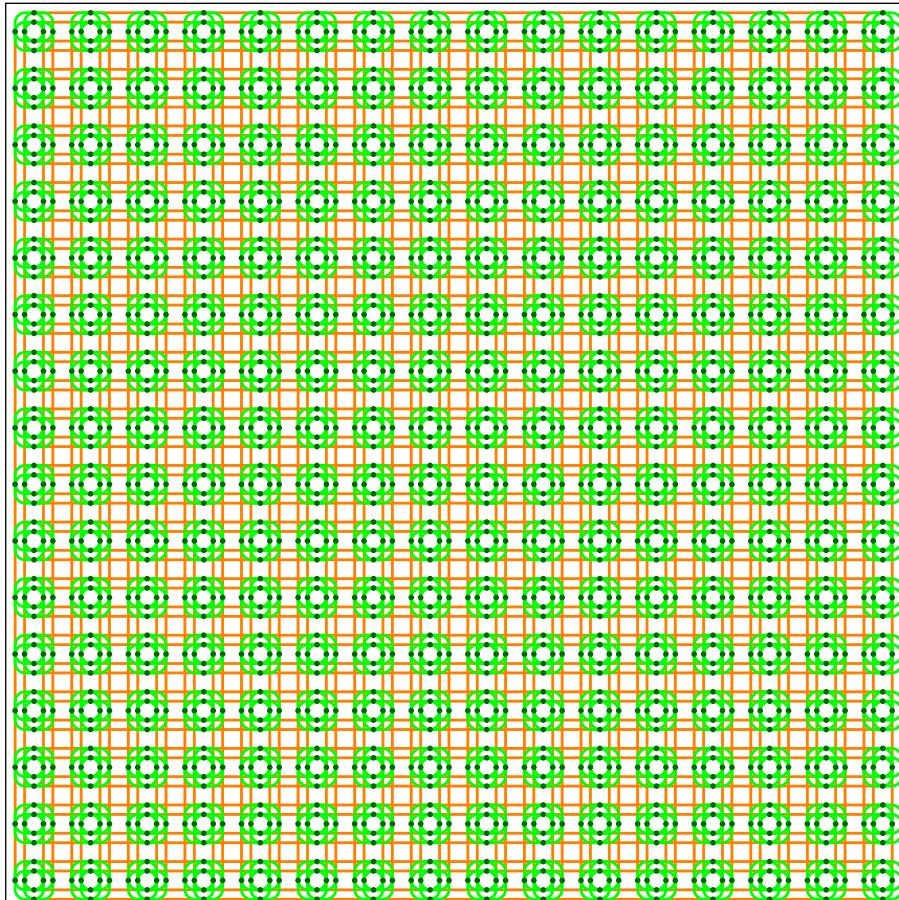
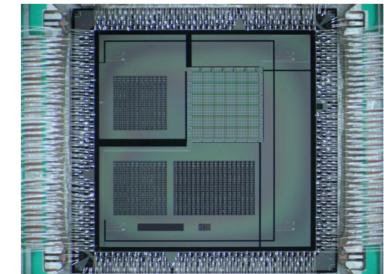
[Khoshaman, Vinci et al., QST, 4, 1]

Effects of Increased Connectivity

D-Wave 2000Q

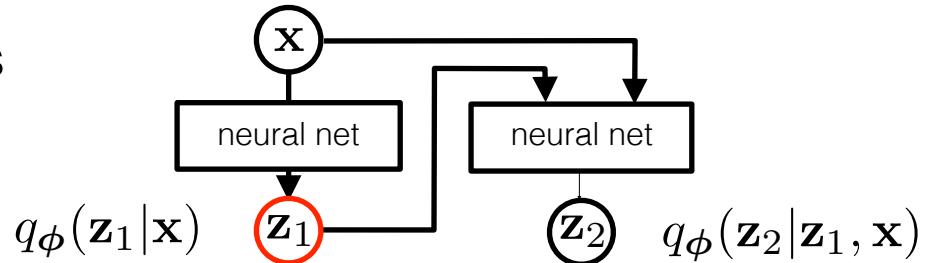


Pegasus prototype



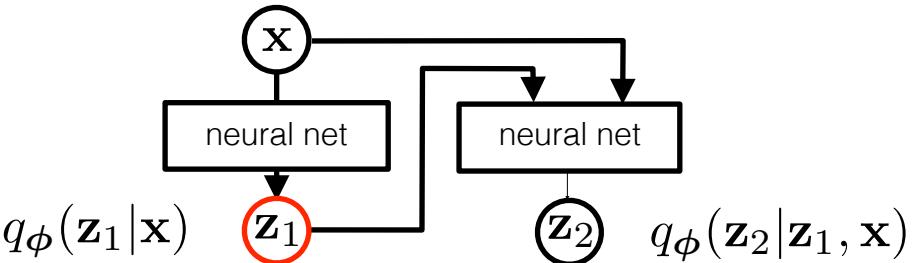
Model Optimization for Sparse Graphs

- Hierarchical conditional relationships for a more powerful encoder

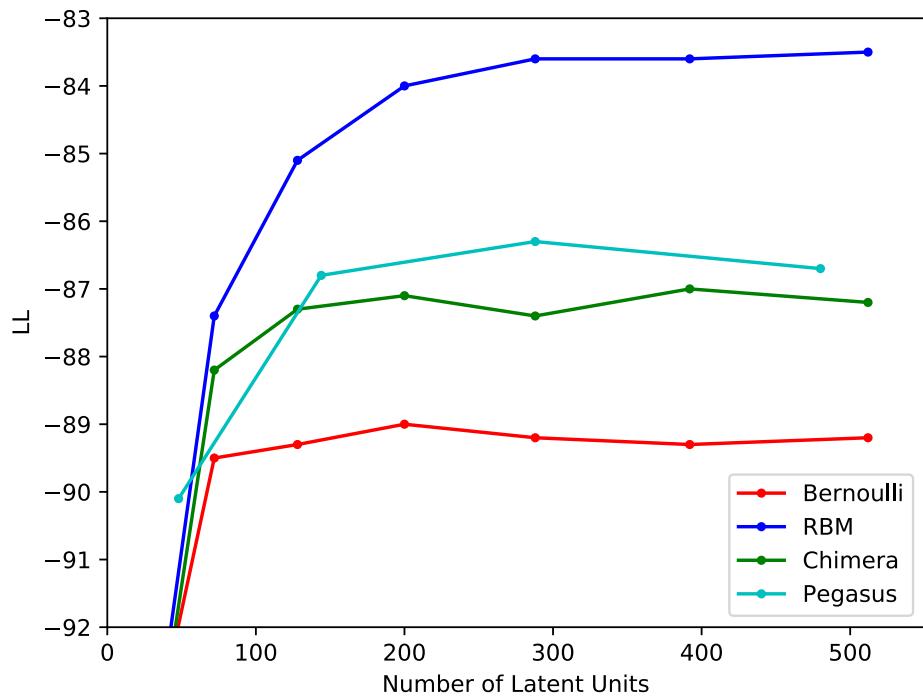


Model Optimization for Sparse Graphs

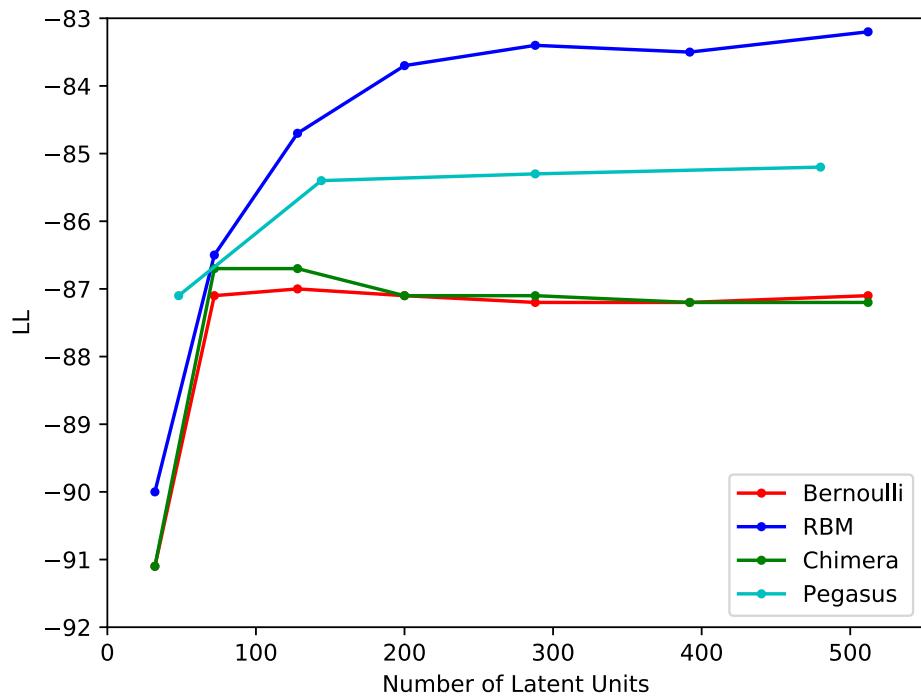
- Hierarchical conditional relationships for a more powerful encoder



1 hierarchy



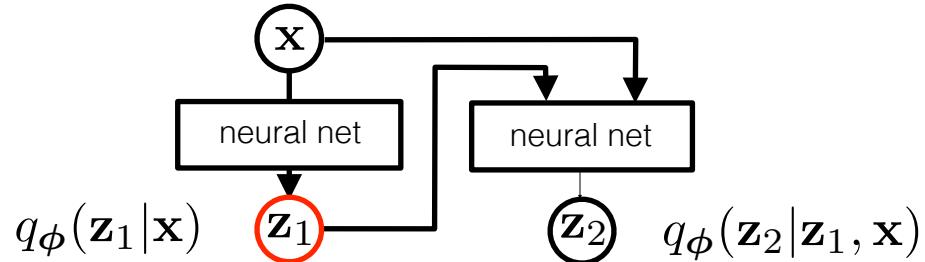
2 hierarchies



[Samples generated with population annealing]

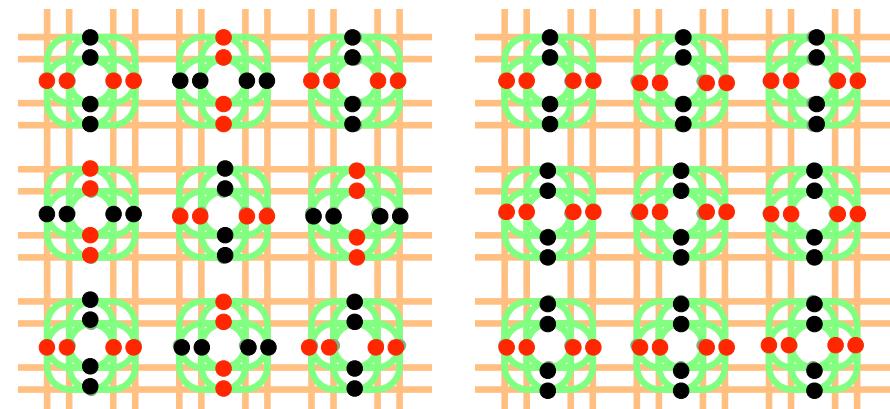
Model Optimization for Sparse Graphs

- Engineer the classical AutoEncoding structure to fit hardware specifications



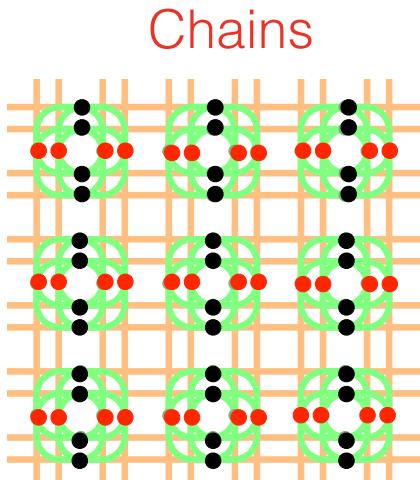
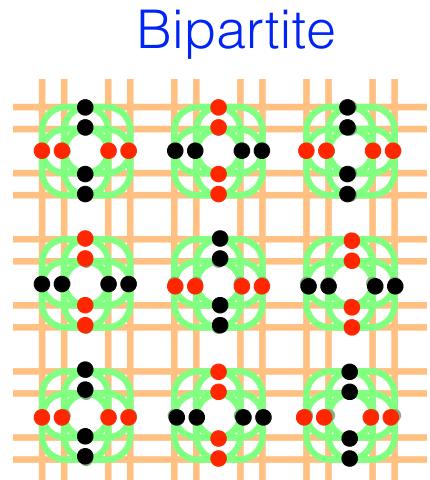
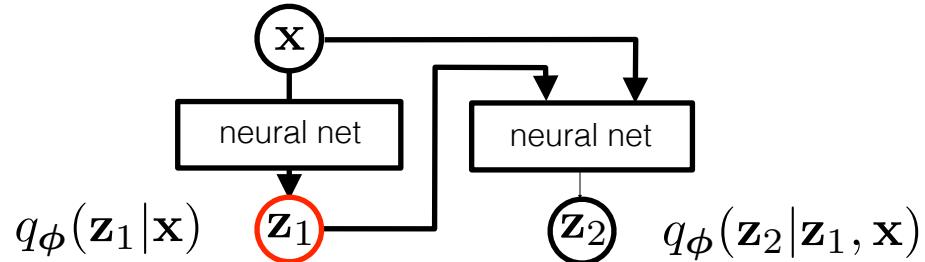
Bipartite

Chains

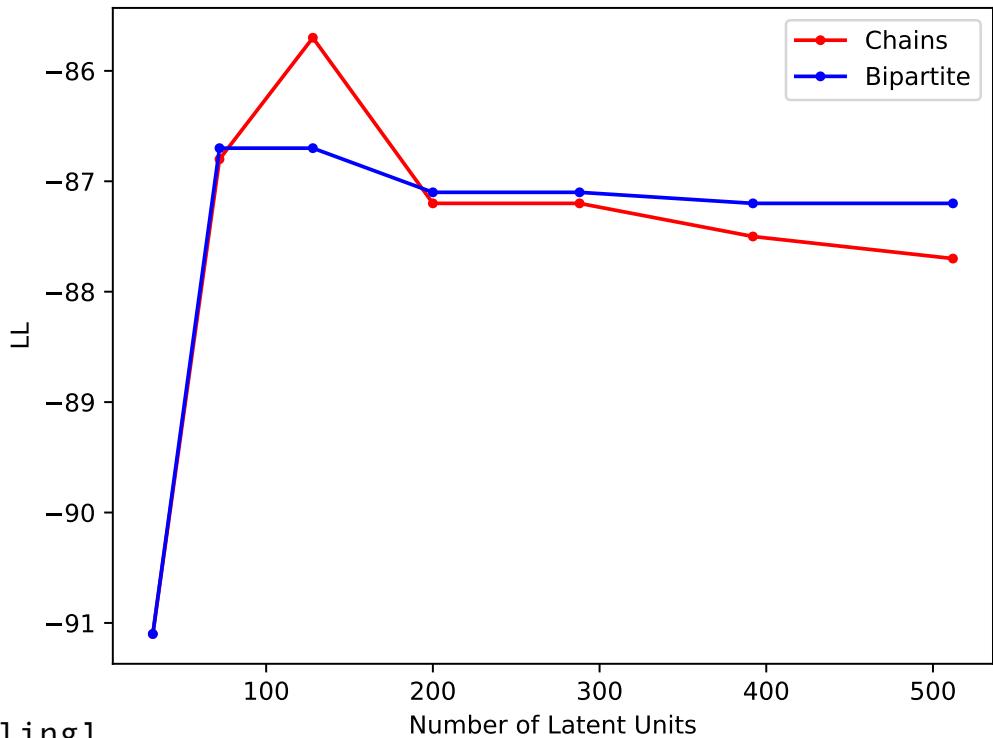


Model Optimization for Sparse Graphs

- Engineer the classical AutoEncoding structure to fit hardware specifications



[Samples generated with population annealing]



Training QVAE with D-Wave 2000Q

- QVAE trained with D-Wave 2000Q
 - Chimera subgraph with 128 qubits
 - Forward anneal, 10μs pause at s = 0.5

9	3	6	3	2	1	9	7	0	8
3	6	4	7	2	0	8	8	7	1
8	9	1	6	3	9	9	1	9	3
7	3	1	3	0	9	8	8	9	9
3	8	2	9	4	3	2	6	8	8
3	5	9	9	7	8	5	8	9	8
9	1	8	5	2	8	3	1	8	1
7	9	1	1	4	8	9	1	8	7
6	9	9	2	5	5	8	0	4	8
7	8	4	9	8	7	4	1	7	9

[Samples generated with D-Wave 2000Q]

Training QVAE with D-Wave 2000Q

- QVAE trained with D-Wave 2000Q
 - Chimera subgraph with 128 qubits
 - Forward anneal, 10μs pause at $s = 0.5$

- Validation of training with QPU on Chimera is subtle:

- ELBO and LL are not available
- ‘Visual’ validation also not available

9	3	6	3	2	1	9	7	8	8
3	6	4	7	2	0	8	8	7	9
8	9	1	6	3	9	9	1	9	3
7	3	1	3	0	9	8	8	9	9
3	8	2	9	4	3	2	6	8	8
3	5	9	9	7	8	5	8	9	8
9	1	8	5	2	8	3	1	8	1
7	9	1	1	4	8	9	1	8	7
6	9	9	2	5	5	8	0	4	8
7	8	4	9	8	7	4	1	7	9

[Samples generated with D-Wave 2000Q]

8	9	3	7	8	2	3	5	9	2
1	9	0	0	8	3	5	5	9	7
9	2	2	5	3	4	4	4	5	0
3	0	0	3	2	1	7	6	9	3
2	8	5	1	7	4	9	9	2	4
5	3	9	4	3	8	1	2	3	8
8	6	8	2	5	6	7	8	4	6
3	3	4	2	0	3	9	9	1	6
2	0	1	9	1	8	6	8	3	7
3	8	9	5	7	0	9	2	2	0

Bernoulli

9	9	1	8	8	0	6	6	9	2
4	4	2	3	3	0	2	1	4	5
7	7	7	0	8	8	9	9	3	9
5	9	5	9	7	4	3	6	6	7
8	4	9	0	8	2	9	8	3	0
8	8	8	4	7	5	4	9	6	6
3	5	0	1	9	1	3	3	6	8
5	4	8	3	5	6	8	6	0	5
1	5	7	7	5	0	9	3	5	8
7	8	0	9	9	8	8	7	7	7

Chimera

[Gumbolt model, 500k gradient updates]

[Samples generated with population annealing]

Training QVAE with D-Wave 2000Q

- QVAE trained with D-Wave 2000Q
 - Chimera subgraph with 128 qubits
 - Forward anneal, 10μs pause at $s = 0.5$
- Validation of training with QPU on Chimera is subtle:
 - ELBO and LL are not available
 - ‘Visual’ validation also not available
- Evidence of successful training:
 - Validation via ‘auxiliary’ RBM

[Gumbolt model, 500k gradient updates]

9	3	6	3	2	1	9	7	8	8
3	6	4	7	2	0	8	8	7	9
8	9	1	6	3	9	9	1	9	3
7	3	1	3	0	9	8	8	9	9
3	8	2	9	4	3	2	6	8	8
3	5	9	9	7	8	5	8	9	8
9	1	8	5	2	8	3	1	8	1
7	9	1	1	4	8	9	1	8	7
6	9	9	2	5	5	8	0	4	8
7	8	4	9	8	7	4	1	7	9

[Samples generated with D-Wave 2000Q]

9	9	3	7	8	2	3	5	2	2
1	9	0	0	8	3	5	5	8	7
9	2	2	5	3	4	4	4	5	0
3	0	2	3	2	1	7	6	9	3
2	8	5	1	7	2	4	9	9	4
5	3	9	4	3	8	1	2	3	8
8	6	8	2	5	6	7	8	4	6
3	3	4	2	0	3	9	9	1	5
2	0	1	9	1	8	6	8	3	7
3	8	9	5	7	0	9	2	2	0
9	9	1	8	8	0	2	5	7	2
7	7	7	0	3	8	9	9	9	3
5	9	5	4	7	4	3	6	6	7
8	4	9	7	0	8	2	9	8	3
8	8	4	7	5	4	9	6	6	5
3	5	0	1	9	1	3	3	6	8
5	4	8	3	5	6	8	6	0	5
1	5	7	4	5	0	9	3	5	6
7	8	0	9	9	8	8	7	7	7

Bernoulli

Chimera

[Samples generated with population annealing]

Validation via Auxiliary RBM

- To estimate ELBO and LL, need to estimate cross-entropy term:

$$-H(q_\phi, p_\theta) \equiv \mathbb{E}_{\mathbf{z} \sim q_\phi} [\log p_\theta(\mathbf{z})]$$

QPU probabilities not known

Validation via Auxiliary RBM

- To estimate ELBO and LL, need to estimate cross-entropy term:

$$-H(q_\phi, p_{\theta}) \equiv \mathbb{E}_{\mathbf{z} \sim q_\phi} [\log p_{\theta}(\mathbf{z})]$$

- Train an auxiliary RBM on QPU samples:

QPU probabilities not known

$$-H(q_\phi, p_{\theta}^{\text{DW}}) \rightsquigarrow -H(q_\phi, p_{\theta'}^{\text{RBM}})$$

$$p_{\theta}^{\text{DW}}(\mathbf{z}) \sim p_{\theta'}^{\text{RBM}}(\mathbf{z})$$

Validation via Auxiliary RBM

- To estimate ELBO and LL, need to estimate cross-entropy term:

$$-H(q_\phi, p_{\theta}) \equiv \mathbb{E}_{\mathbf{z} \sim q_\phi} [\log p_{\theta}(\mathbf{z})]$$

- Train an auxiliary RBM on QPU samples:

QPU probabilities not known

$$-H(q_\phi, p_{\theta}^{\text{DW}}) \rightsquigarrow -H(q_\phi, p_{\theta'}^{\text{RBM}}) \quad p_{\theta}^{\text{DW}}(\mathbf{z}) \sim p_{\theta'}^{\text{RBM}}(\mathbf{z})$$



Chimera

MNIST (static binarization)		
Size	Sampler	LL (± 0.1)
128 (C4)	DW2000Q	-86.0

[GumBolt model, 500k gradient updates]

[Samples generated with D-Wave 2000Q]



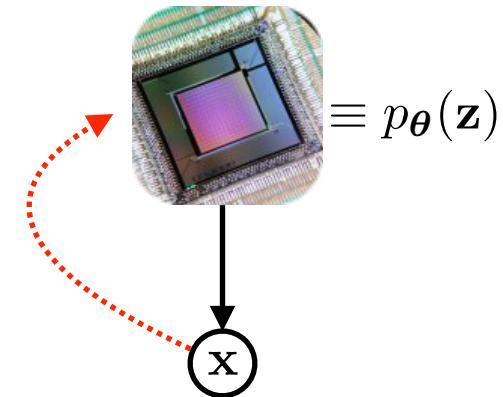
Bernoulli

A Path Towards Quantum Advantage with QVAE

- Processor development: improve sampling quality
 - Pauses, fast quenches, smaller control errors, denser connectivities
 - Reverse-annealed sampling, larger tunnelling rates

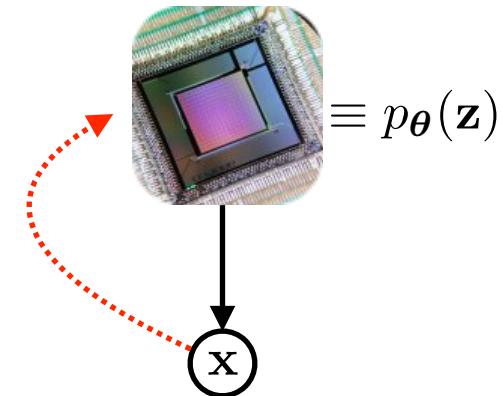
A Path Towards Quantum Advantage with QVAE

- Processor development: improve sampling quality
 - Pauses, fast quenches, smaller control errors, denser connectivities
 - Reverse-annealed sampling, larger tunnelling rates
- Model building:
 - Generative capacity: $p_{\theta}(\mathbf{x}|\mathbf{z})p_{\theta}(\mathbf{z})$ ← quantum
 - Exploit highly representative quantum distributions
 - Exploit larger number of latent units



A Path Towards Quantum Advantage with QVAE

- Processor development: improve sampling quality
 - Pauses, fast quenches, smaller control errors, denser connectivities
 - Reverse-annealed sampling, larger tunnelling rates
- Model building:
 - Generative capacity: $p_{\theta}(\mathbf{x}|\mathbf{z})p_{\theta}(\mathbf{z})$ ← quantum
 - Exploit highly representative quantum distributions
 - Exploit larger number of latent units
- Do not be shy in using (a lot of) classical resources!
 - Deep convolutional networks: Bernoulli: LL=-85.5, Chimera: LL=83.9
 - Fully autoregressive decoders (pixelCNN...): LL < -79



Conclusions

- QVAE...
 - ... are powerful generative models that can achieve state-of-the-art performance on complex datasets
 - ... can be effectively trained in presence of strong quantum effects
 - ... can be seamlessly integrated with current-generation quantum annealers
 - ... was successfully trained using D-Wave 2000Q quantum annealing devices as ‘quantum generators’

Conclusions

- QVAE...
 - ... are powerful generative models that can achieve state-of-the-art performance on complex datasets
 - ... can be effectively trained in presence of strong quantum effects
 - ... can be seamlessly integrated with current-generation quantum annealers
 - ... was successfully trained using D-Wave 2000Q quantum annealing devices as ‘quantum generators’

How can we use it in HEP ?

Thank you!

Thank you!