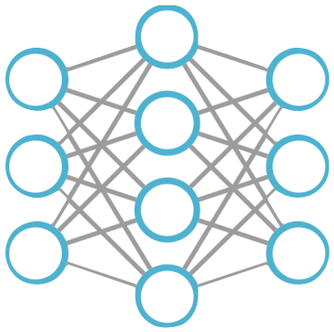
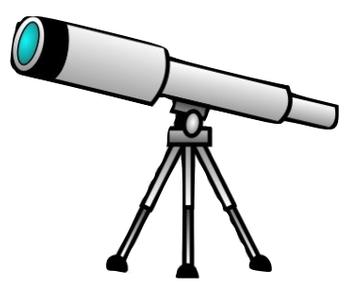


Neural Network Renormalization Group

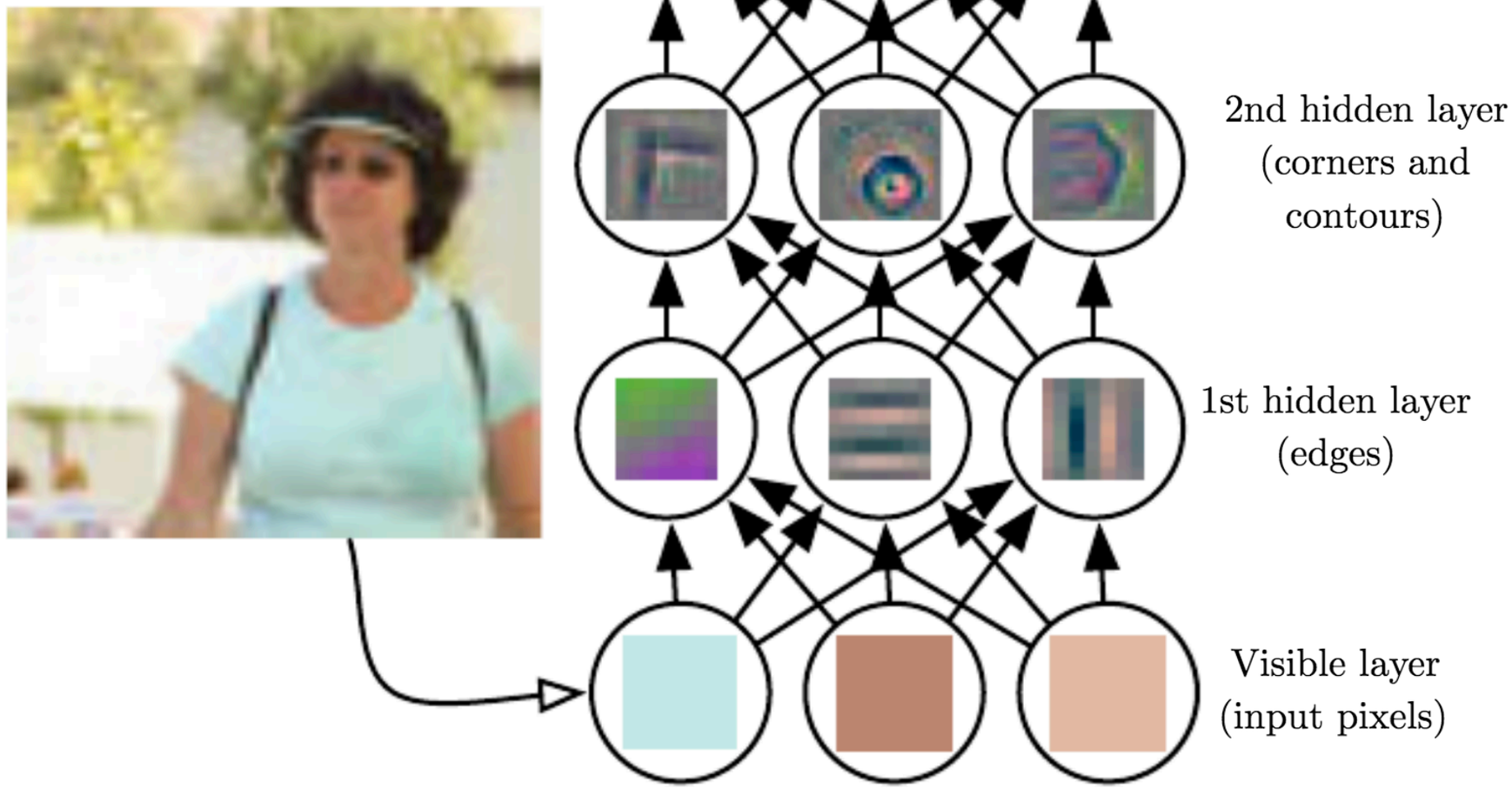


Lei Wang (王磊)
Institute of Physics, CAS
https://wangleiphy.github.io

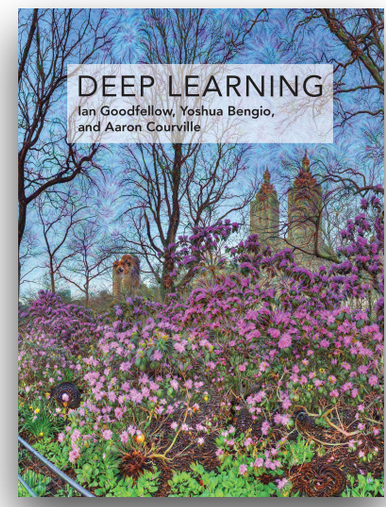
Collaborator

Shuo-Hui Li (李烁辉)
PhD. student at IOP, CAS

Deep Neural Network and RG



Page 6
Figure 1.2



Goodfellow, Bengio, Courville, http://www.deeplearningbook.org/

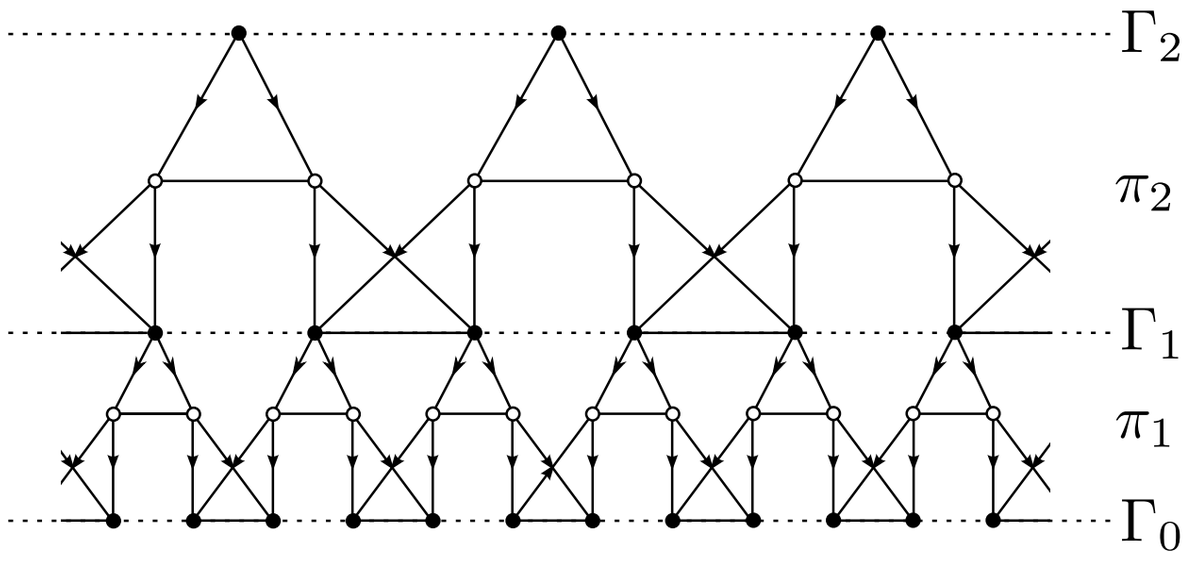
Deep learning and the renormalization group

Cédric Bény

15 Jan 2013 ICLR 2013 conference submission readers: everyone

Decision: reject

Abstract: Renormalization group methods, which analyze the way in which the effective behavior of a system depends on the scale at which it is observed, are key to modern condensed-matter theory and particle physics. The aim of this paper is to compare and contrast the ideas behind the renormalization group (RG) on the one hand and deep machine learning on the other, where depth and scale play a similar role. In order to illustrate this connection, we review a recent numerical method based on the RG---the multiscale entanglement renormalization ansatz (MERA)---and show how it can be converted into a learning algorithm based on a generative hierarchical Bayesian network model. Under the assumption---common in physics---that the distribution to be learned is fully characterized by local correlations, this algorithm involves only explicit evaluation of probabilities, hence doing away with sampling.



arxiv:1301.3124

Deep learning and the renormalization group

Cédric Bény

15 Jan 2013 ICLR 2013 conference submission readers: everyone

Decision: reject

Yann LeCun

05 Apr 2013 ICLR 2013 submission review readers: everyone

Review: It seems to me like there could be an interesting connection between approximate inference in graphical models and the renormalization methods.

There is in fact a long history of interactions between condensed matter physics and graphical models. For example, it is well known that the loopy belief propagation algorithm for inference minimizes the Bethe free energy (an approximation of the free energy in which only pairwise interactions are taken into account and high-order interactions are ignored). More generally, variational methods inspired by statistical physics have been a very popular topic in graphical model inference.

The renormalization methods could be relevant to deep architectures in the sense that the grouping of random variable resulting from a change of scale could be be made analogous with the pooling and subsampling operations often used in deep models.

It's an interesting idea, but it will probably take more work (and more tutorial expositions of RG) to catch the attention of this community.

QUANTA MAGAZINE illuminating science

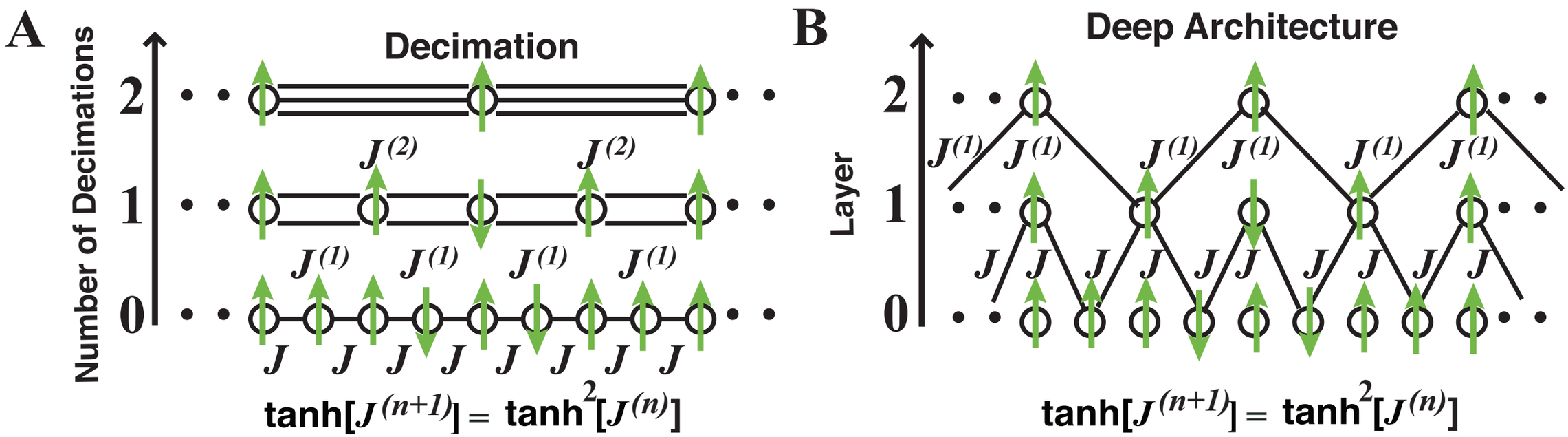
A Common Logic to Seeing Cats and Cosmos

Olena Shmahalo / Quanta Magazine

There may be a universal logic to how physicists, computers and brains tease out important features from among other irrelevant bits of data.

"An exact mapping between the Variational Renormalization Group and Deep Learning", Mehta and Schwab, 1410.3831

“Exact Mapping”



$$e^{-H(\boldsymbol{h})} = \sum_{\boldsymbol{x}} e^{T(\boldsymbol{x},\boldsymbol{h})-H(\boldsymbol{x})}$$

RG Transformation

$$e^{-E(\boldsymbol{h})} = \sum_{\boldsymbol{x}} e^{-E(\boldsymbol{x},\boldsymbol{h})}$$

Boltzmann Machine

Harsh comments below the Quanta Magazine article

Noah says:
December 26, 2014 at 9:54 am

I just spend an hour reading Mehta-Schwab paper from the beginning to end. Let me say that "A Common Logic to Seeing Cats and Cosmos" is a sensationalist article about a trivial paper, which will have no impact whatsoever. The whole M-S paper is based on the fact that couplings of two systems appear in more than one context and that distributions can sometimes appear as marginal distributions on product spaces. There is no one-to-one mappings between renormalization group (RG) scheme of Kadanoff and Restricted Boltzmann Machines (RBM) in Deep Neural Networks (DNN) in their paper. What they show is that RBM can be represented as a RG scheme with a very specific choice of coupling function T in equation (18). Conveniently, this coupling function depends on the Hamiltonian of the spin system, which it normally should not. Equivalence in equations (8) and (9) is also not correct. Condition (9) of course implies that the scheme is exact, but not the other way around, unless the authors make some implicit assumptions about coupling function T not mentioned in the paper. The paper contains no non-trivial ideas, it does not "open up a door to something very exciting", and I will not hold my breath expecting new breakthroughs because of this connection.

https://www.quantamagazine.org/20141204-a-common-logic-to-seeing-cats-and-cosmos/

Dictionary: RG vs Deep Learning

Property	Variational RG	Deep Belief Networks
How input distribution is defined	Hamiltonian defining P(v)	Data samples drawn from P(v)
How interactions are defined	T(v,h)	E(v,h)
Exact transformation	$Tr_{\mathbf{h}} e^{T(\mathbf{v},\mathbf{h})} = 1$	KL divergence between P(v) and variational distribution is zero
Approximations	Minimize or bound free energy differences	Minimize the KL divergence
Method	Analytic (mostly)	Numerical
What happens under coarse-graining	Relevant operators grow/irrelevant shrink	New features emerge

From Schwab's talk at PI: http://pirsa.org/displayFlash.php?id=16080006

More on the DL-RG Connections

- “Why does deep and cheap learning work so well ”, Lin, Tegmark, Rolnick, 1608.08225

- Comment on the above paper, Schwab and Mehta, 1609.03541

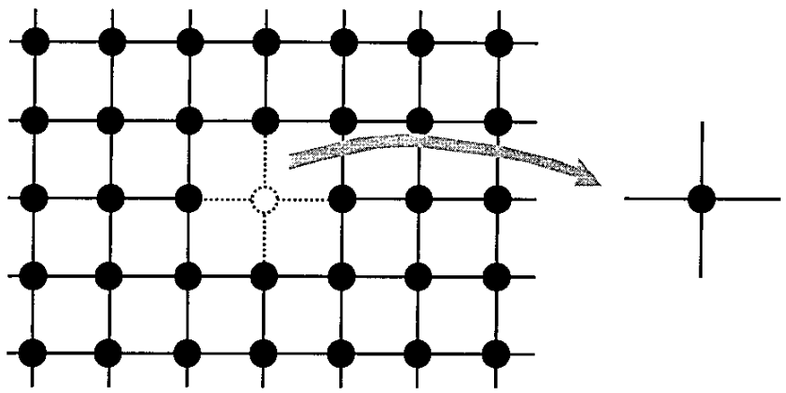
- PCA meets RG, Bradde and Bialek, 1610.09733

- Mutual information RG, Koch-Janusz and Ringel, 1704.06279

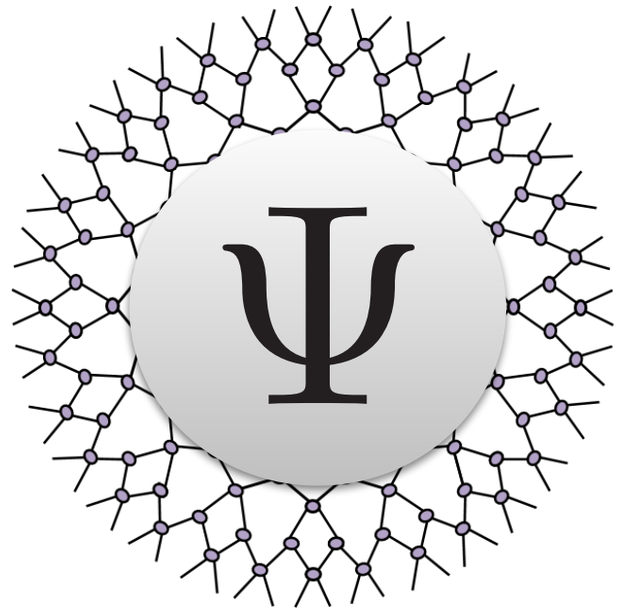
- Media coverage/blog posts/student term papers etc

Physics-DL connection is more general than we thought

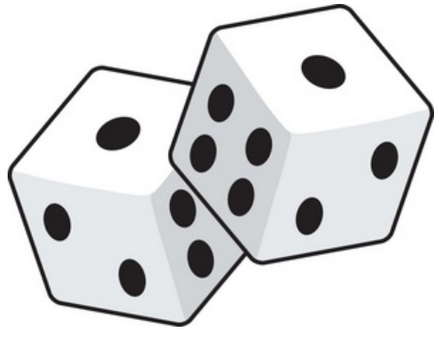
Variational Mean Field



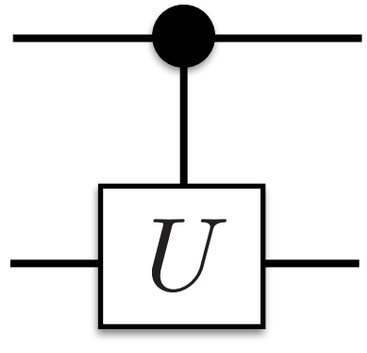
Tensor Networks



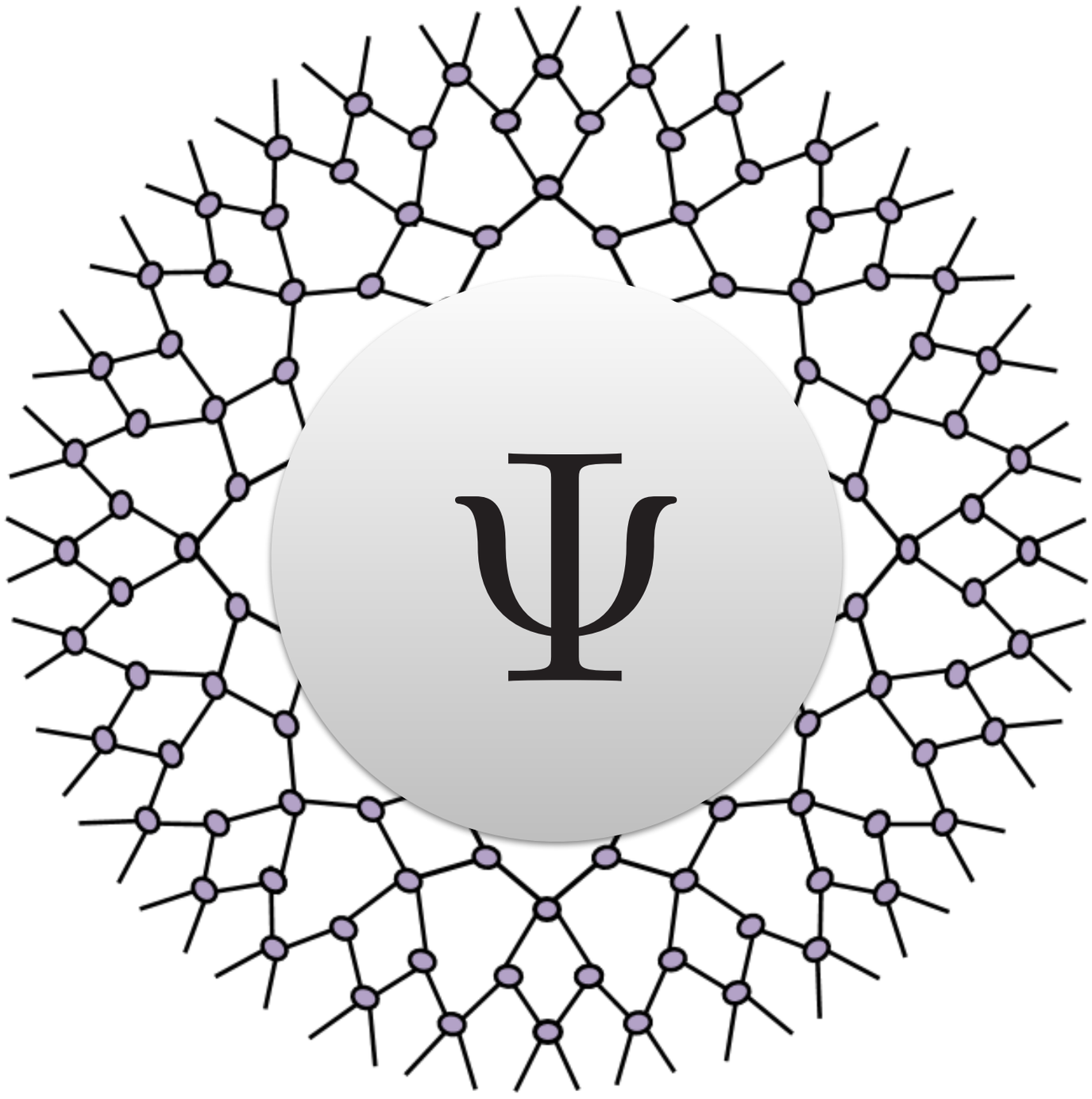
Monte Carlo Methods



Quantum Computing

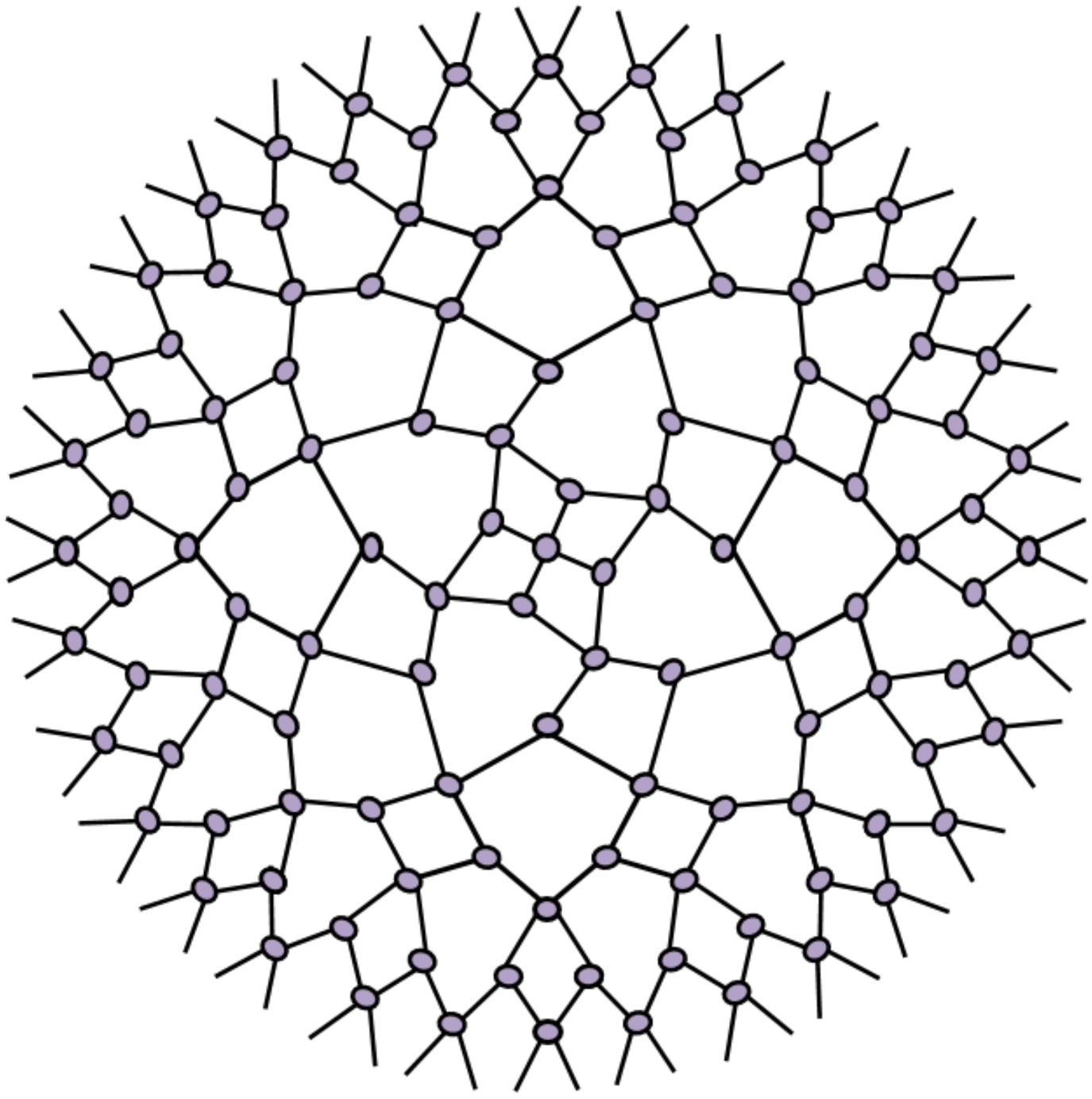


Multi-Scale Entanglement Renormalization Ansatz



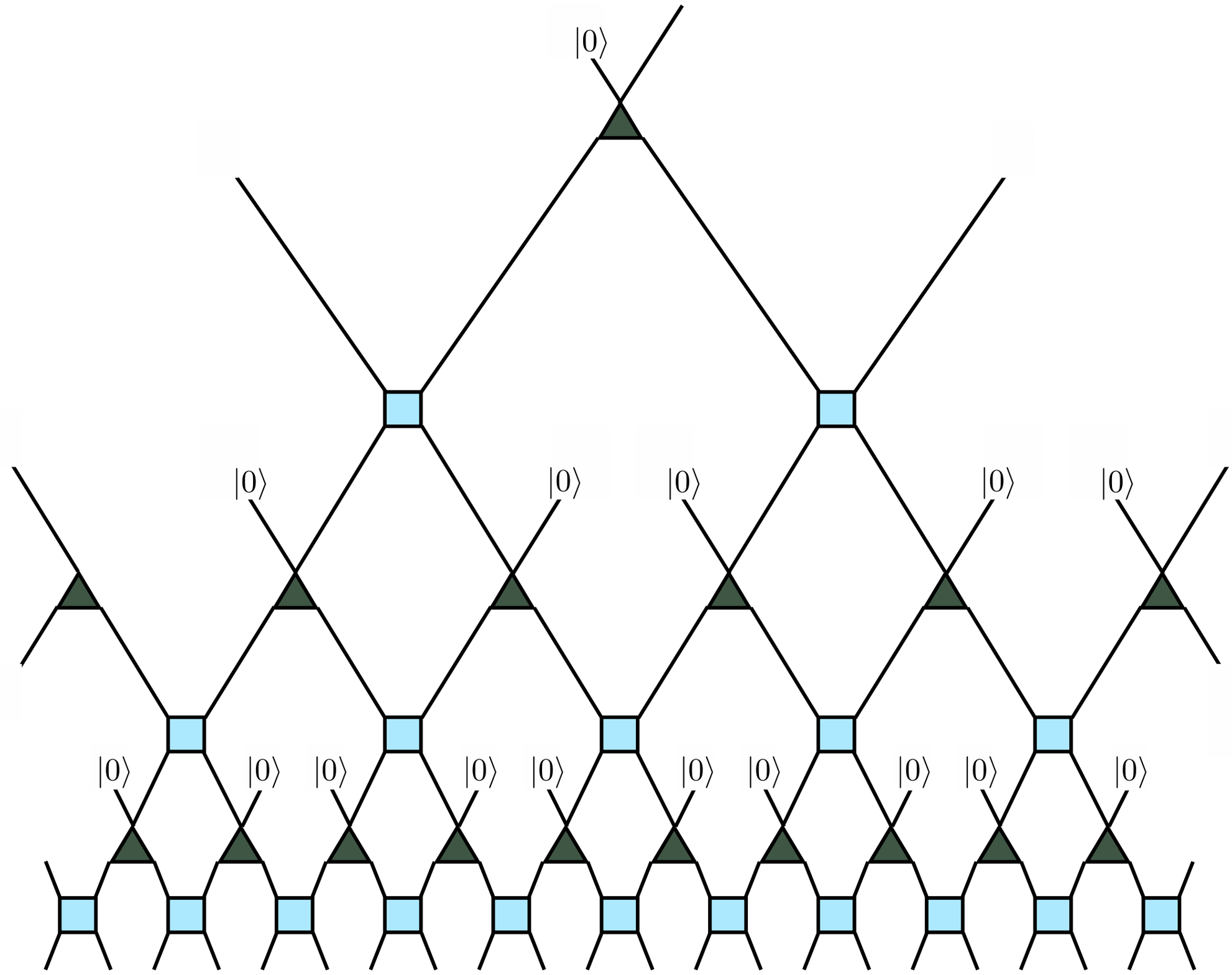
Vidal 2006

Multi-Scale Entanglement Renormalization Ansatz

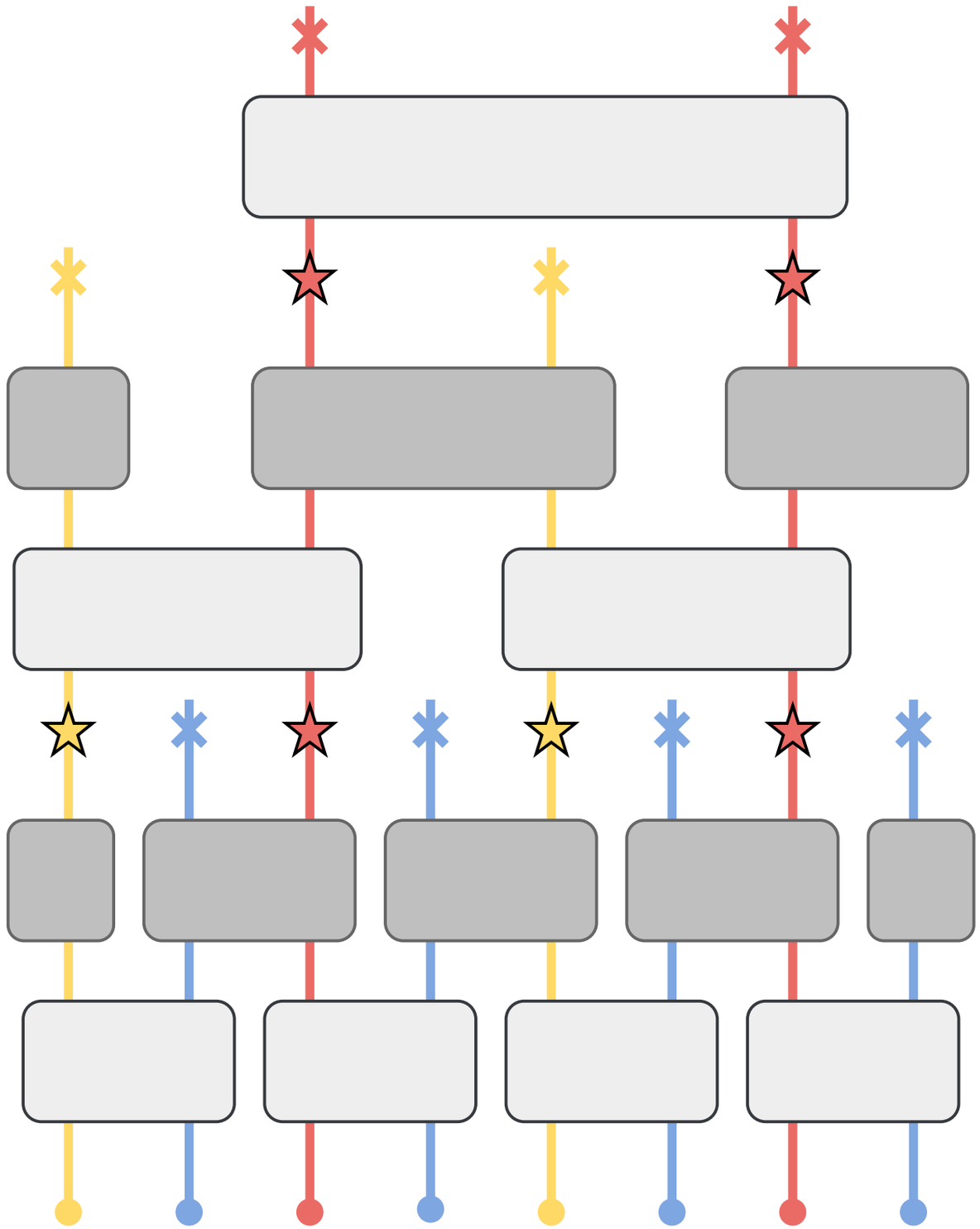


Vidal 2006

MERA as a quantum circuit

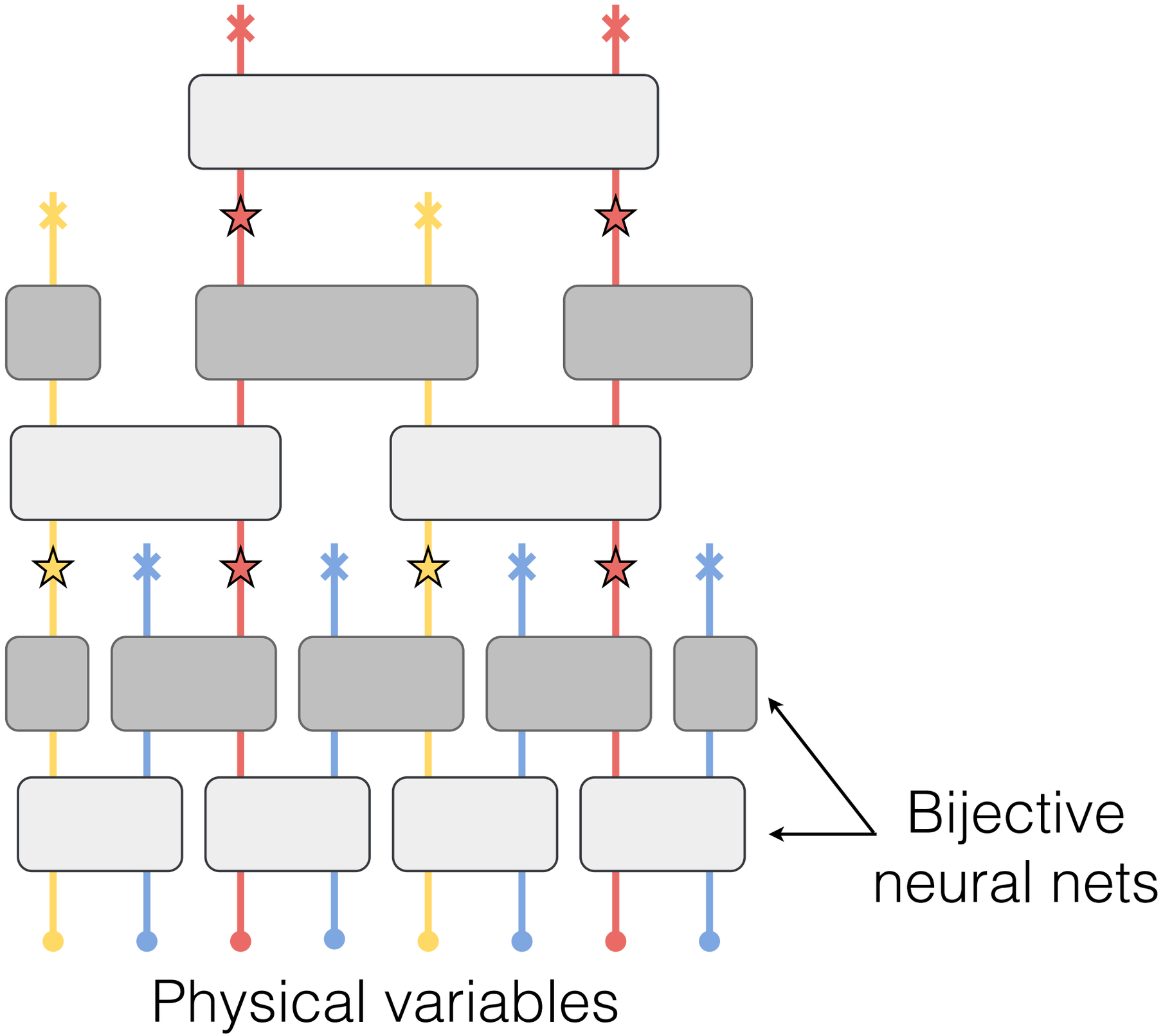


Neural Network Renormalization Group

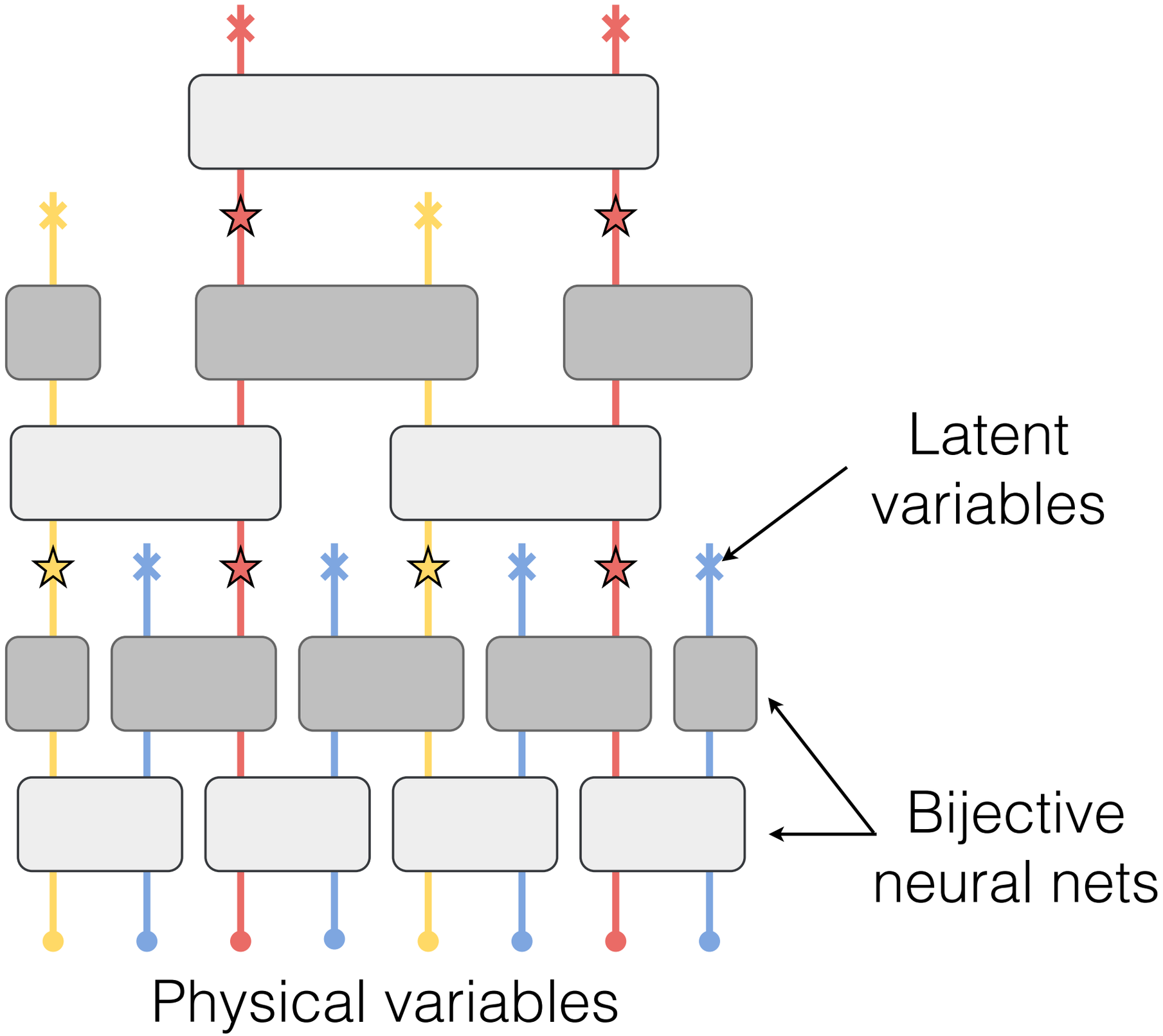


Physical variables

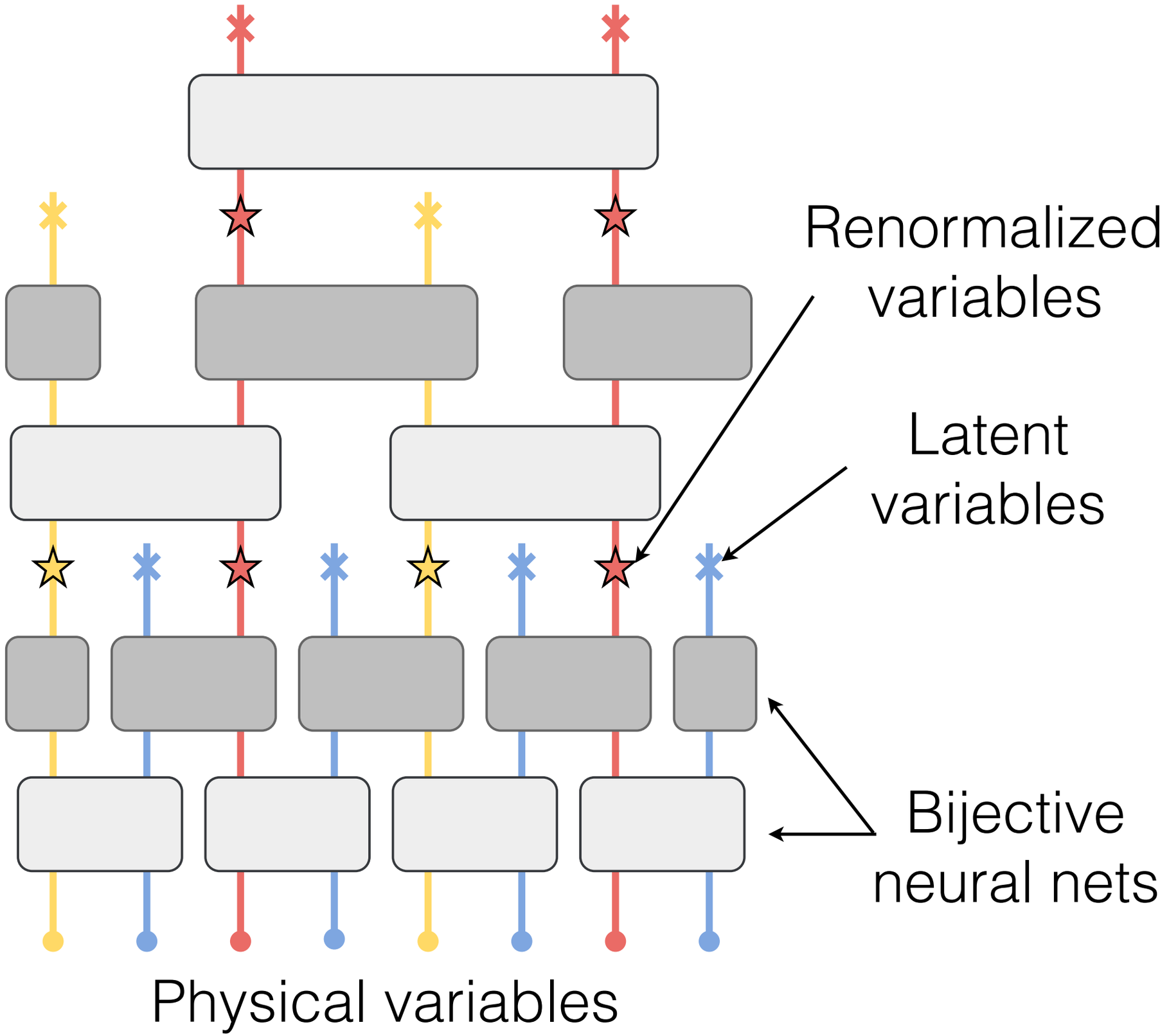
Neural Network Renormalization Group



Neural Network Renormalization Group

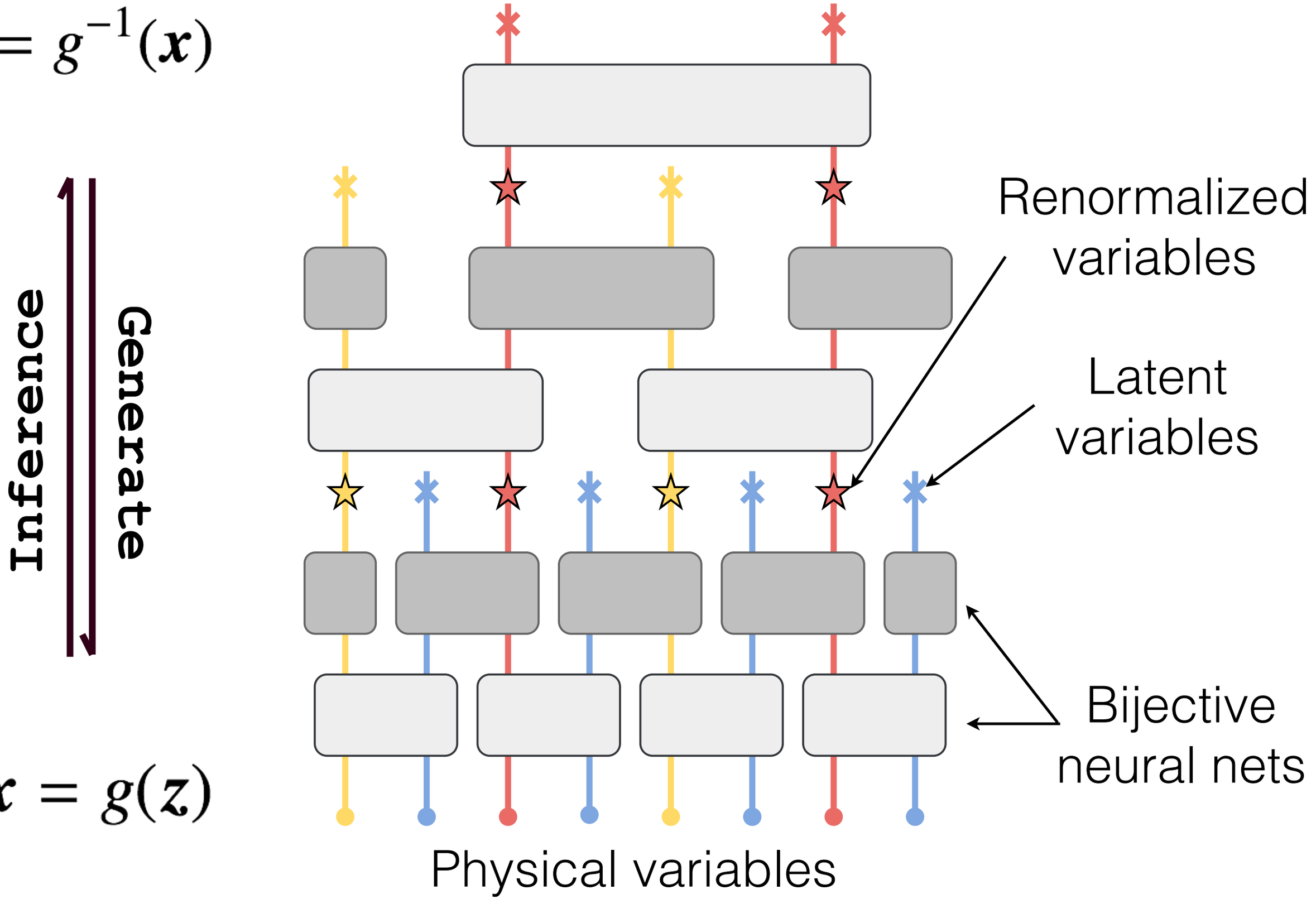


Neural Network Renormalization Group



Neural Network Renormalization Group

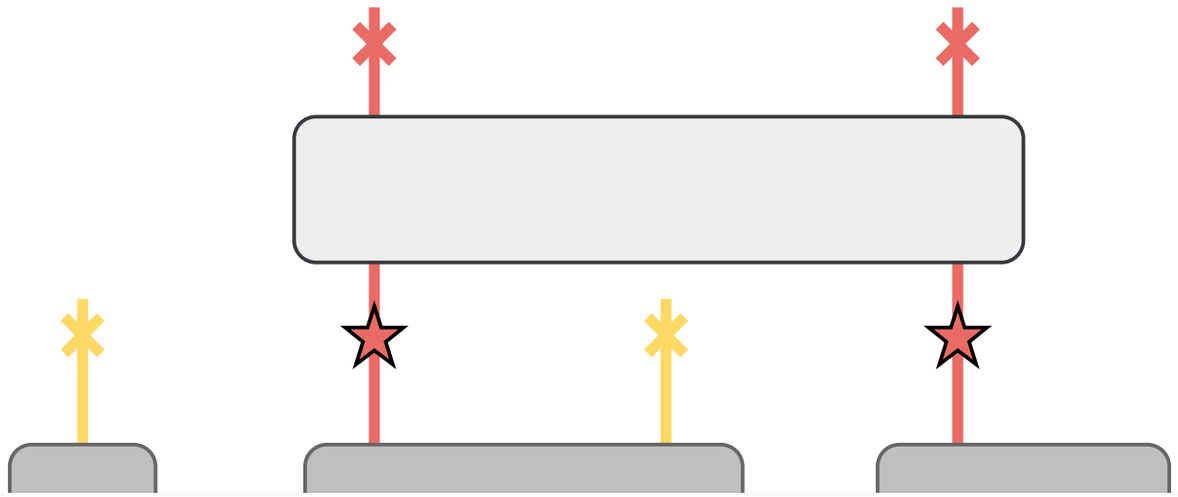
$$z = g^{-1}(x)$$



$$x = g(z)$$

Neural Network Renormalization Group

$$z = g^{-1}(x)$$



Renormalized variables

Latent variables

Probability Transformation

$$\ln q(\boldsymbol{x}) = \ln p(z) - \ln \left| \det \left(\frac{\partial \boldsymbol{x}}{\partial z} \right) \right|$$

Inference

Generate

$$x = g(z)$$

Bijective neural nets

Physical variables

Bijector Block

Bijective & Differentiable map, i.e., Diffeomorphism

Forward

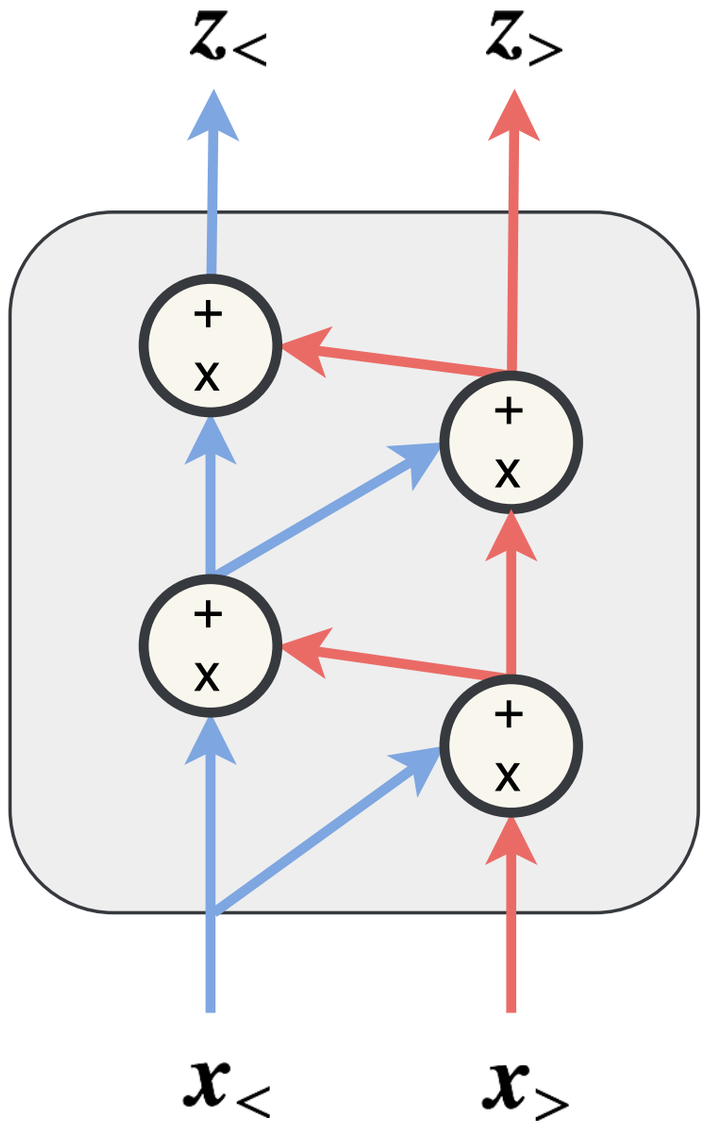
$$\begin{cases} \boldsymbol{x}_< = \boldsymbol{z}_< \\ \boldsymbol{x}_> = \boldsymbol{z}_> \odot e^{s(\boldsymbol{z}_<)} + t(\boldsymbol{z}_<) \end{cases}$$

Backward

$$\begin{cases} \boldsymbol{z}_< = \boldsymbol{x}_< \\ \boldsymbol{z}_> = (\boldsymbol{x}_> - t(\boldsymbol{x}_<)) \odot e^{-s(\boldsymbol{x}_<)} \end{cases}$$

Log-Det-Jacobian

$$\ln \left| \det \left(\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{z}} \right) \right| = \sum_i [s(\boldsymbol{z}_<)]_i$$



Normalizing flow, Rezende et al,1505.05770

Special case: Real NVP, Dinh et al,1605.08803

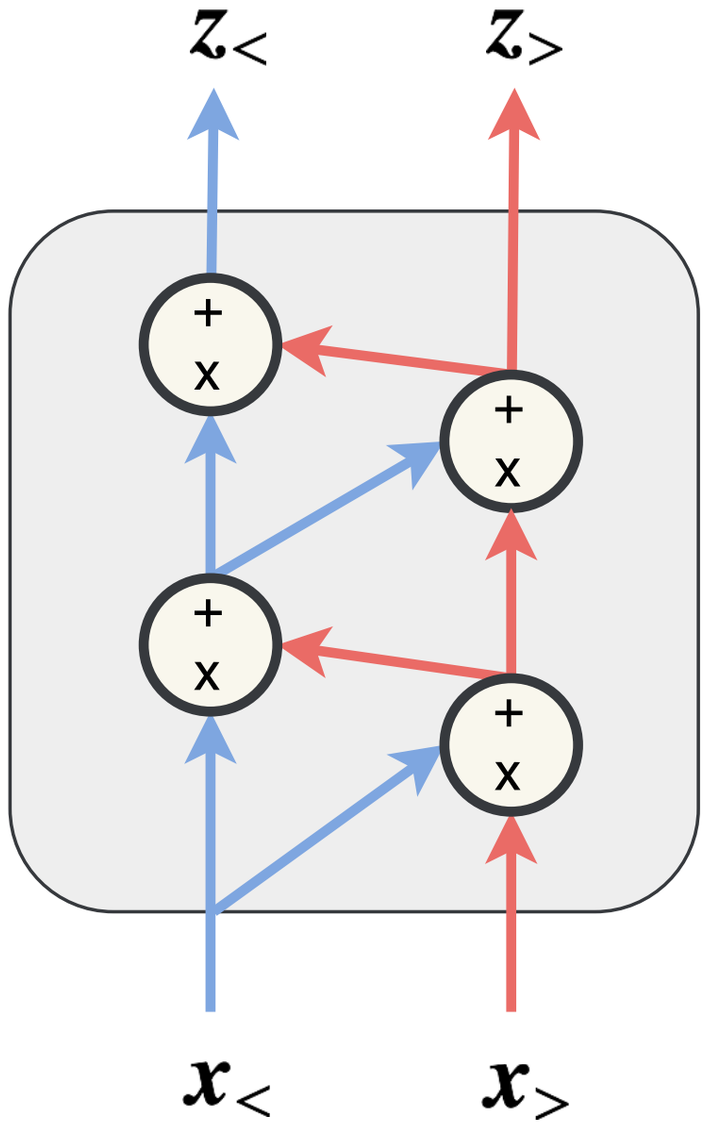
Bijector Block

Bijective & Differentiable map, i.e., Diffeomorphism

Forward

Arbitrary neural nets

$$\begin{cases} \boldsymbol{x}_< = \boldsymbol{z}_< \\ \boldsymbol{x}_> = \boldsymbol{z}_> \odot e^{s(\boldsymbol{z}_<)} + t(\boldsymbol{z}_<) \end{cases}$$



Backward

$$\begin{cases} \boldsymbol{z}_< = \boldsymbol{x}_< \\ \boldsymbol{z}_> = (\boldsymbol{x}_> - t(\boldsymbol{x}_<)) \odot e^{-s(\boldsymbol{x}_<)} \end{cases}$$

Log-Det-Jacobian

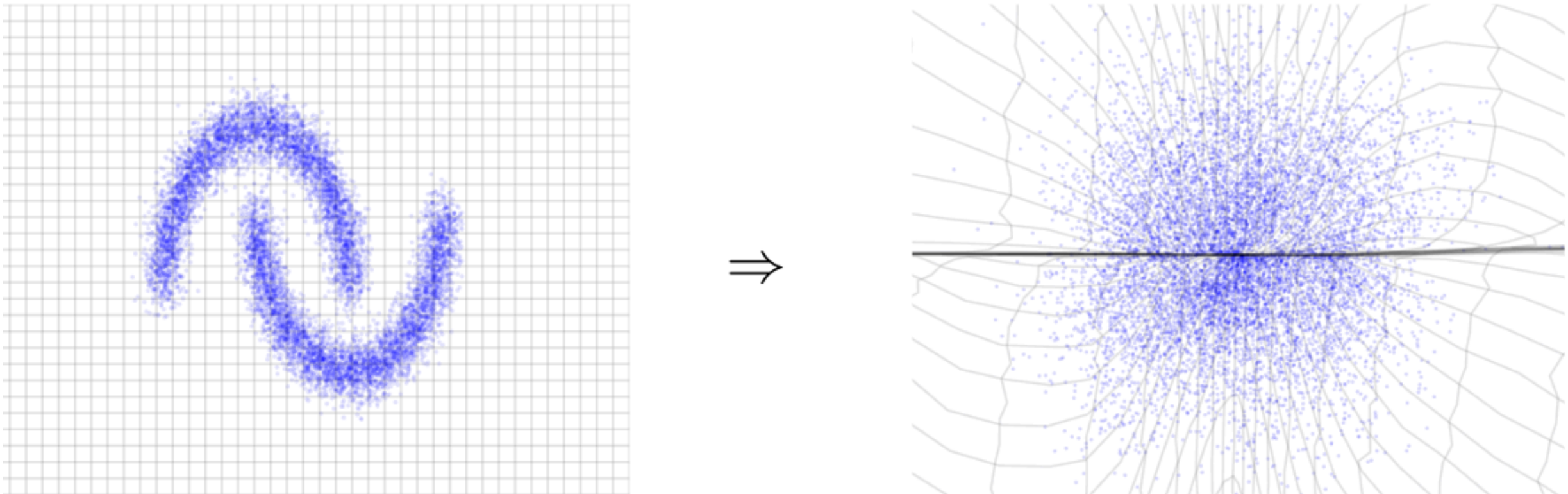
$$\ln \left| \det \left(\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{z}} \right) \right| = \sum_i [s(\boldsymbol{z}_<)]_i$$

Normalizing flow, Rezende et al,1505.05770

Special case: Real NVP, Dinh et al,1605.08803

Inference

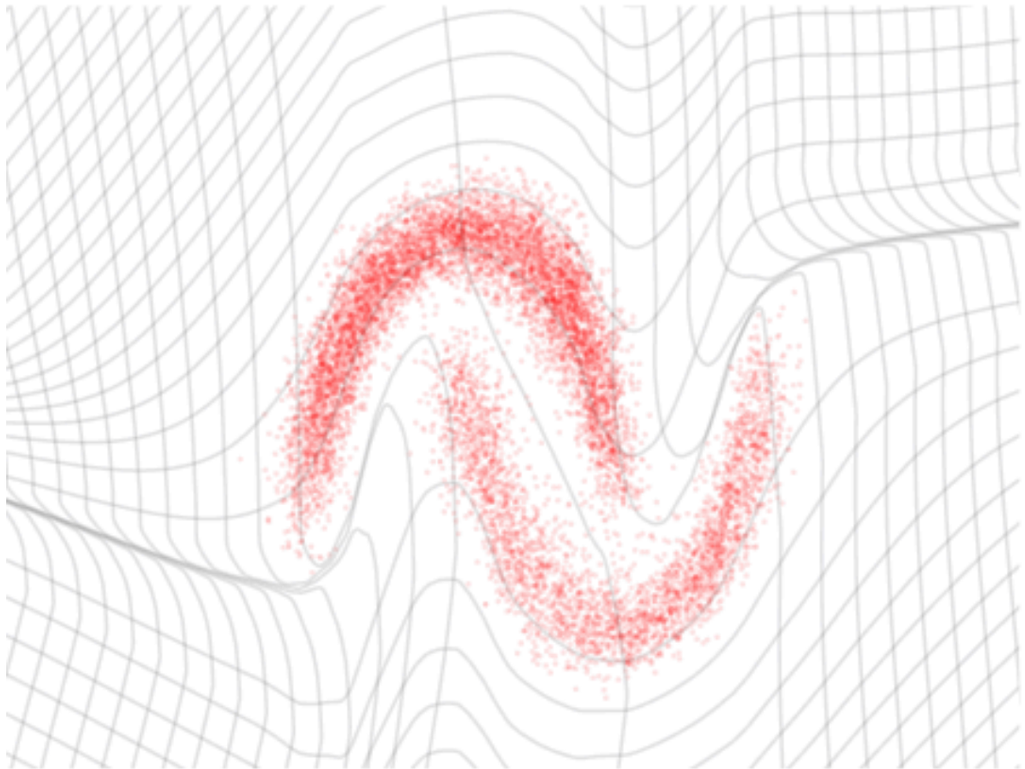
Data space $g^{-1}(\boldsymbol{x}) = \boldsymbol{z}$ Latent space

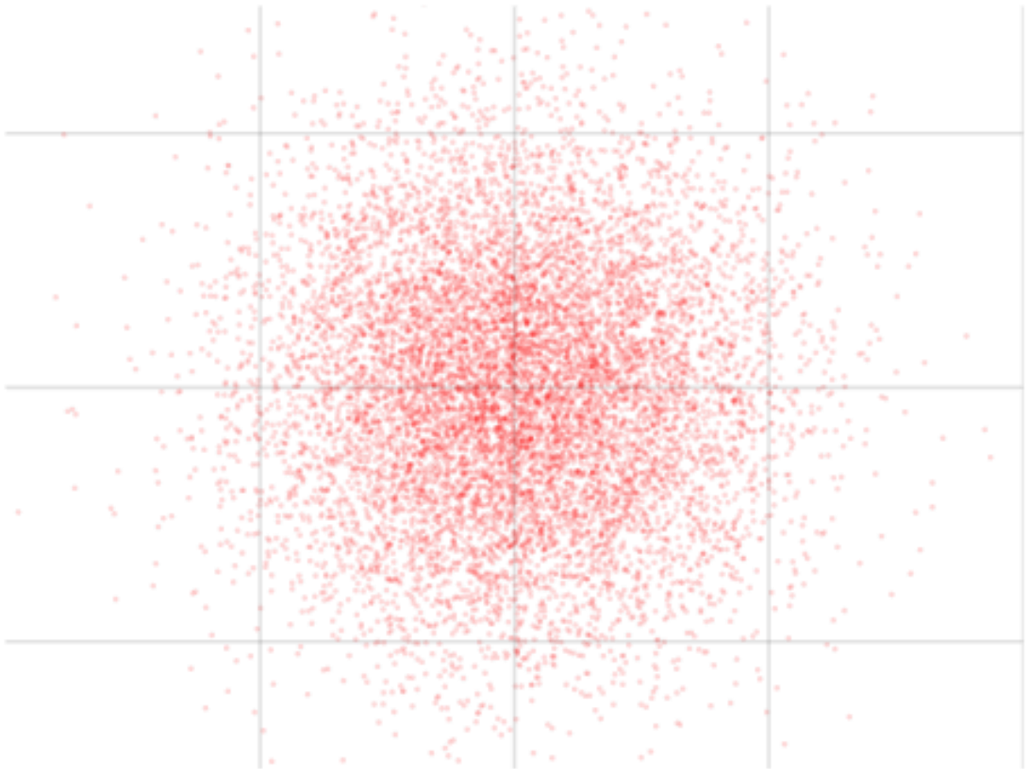


Real NVP, Dinh et al, 1605.08803

Generate

Data space $\boldsymbol{x} = g(\boldsymbol{z})$ Latent space





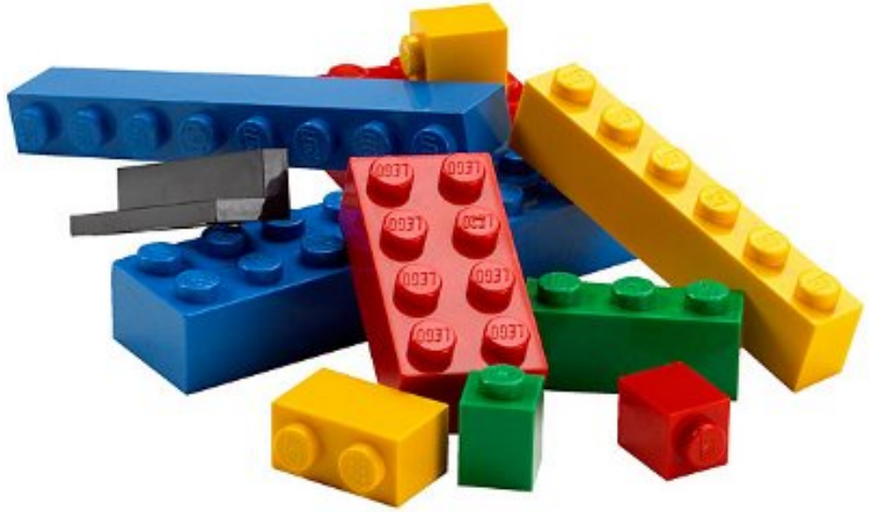
Real NVP, Dinh et al, 1605.08803

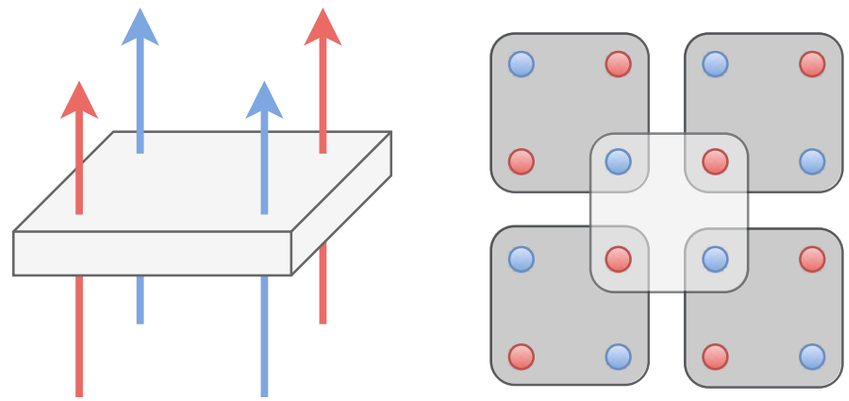
Bijectors form a group

$$\boldsymbol{x} = g(\boldsymbol{z})$$

$$g = \cdots \circ g_2 \circ g_1$$

$$\ln \left| \det \left(\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{z}} \right) \right| = \sum_i \ln \left| \det \left(\frac{\partial g_{i+1}}{\partial g_i} \right) \right|$$

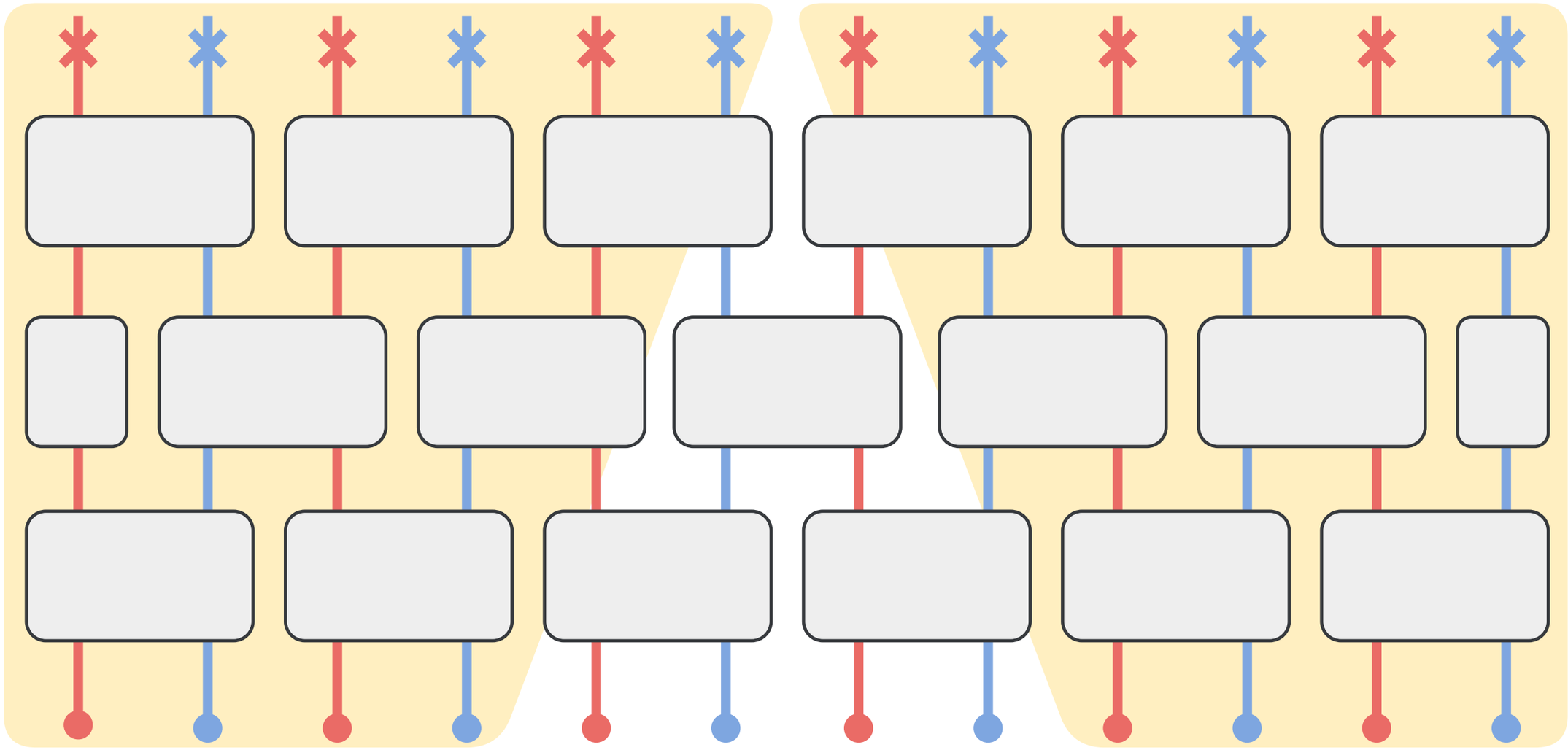




Modular design

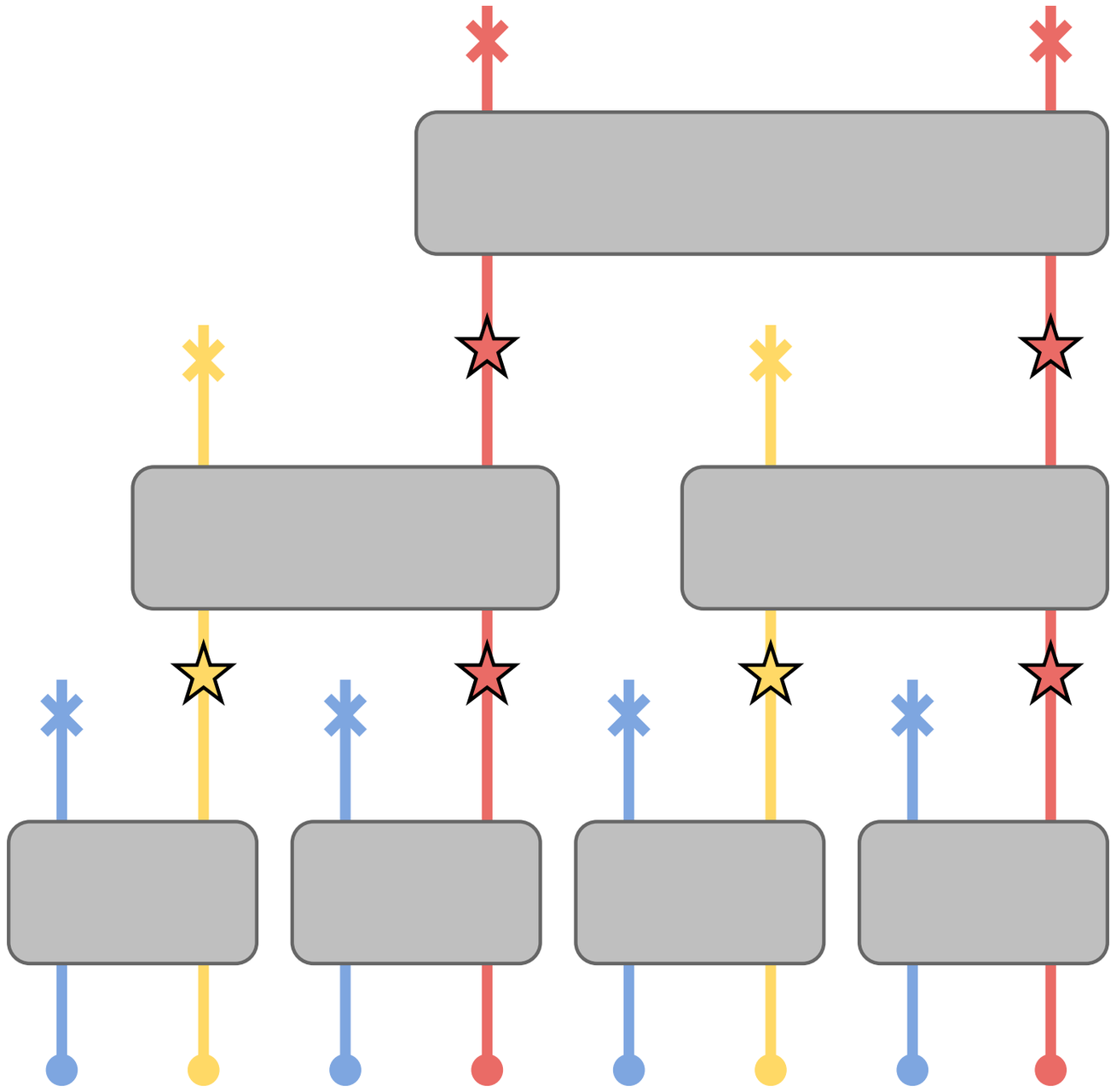
Flexible blocks and stacking

“Disentangler” only architecture

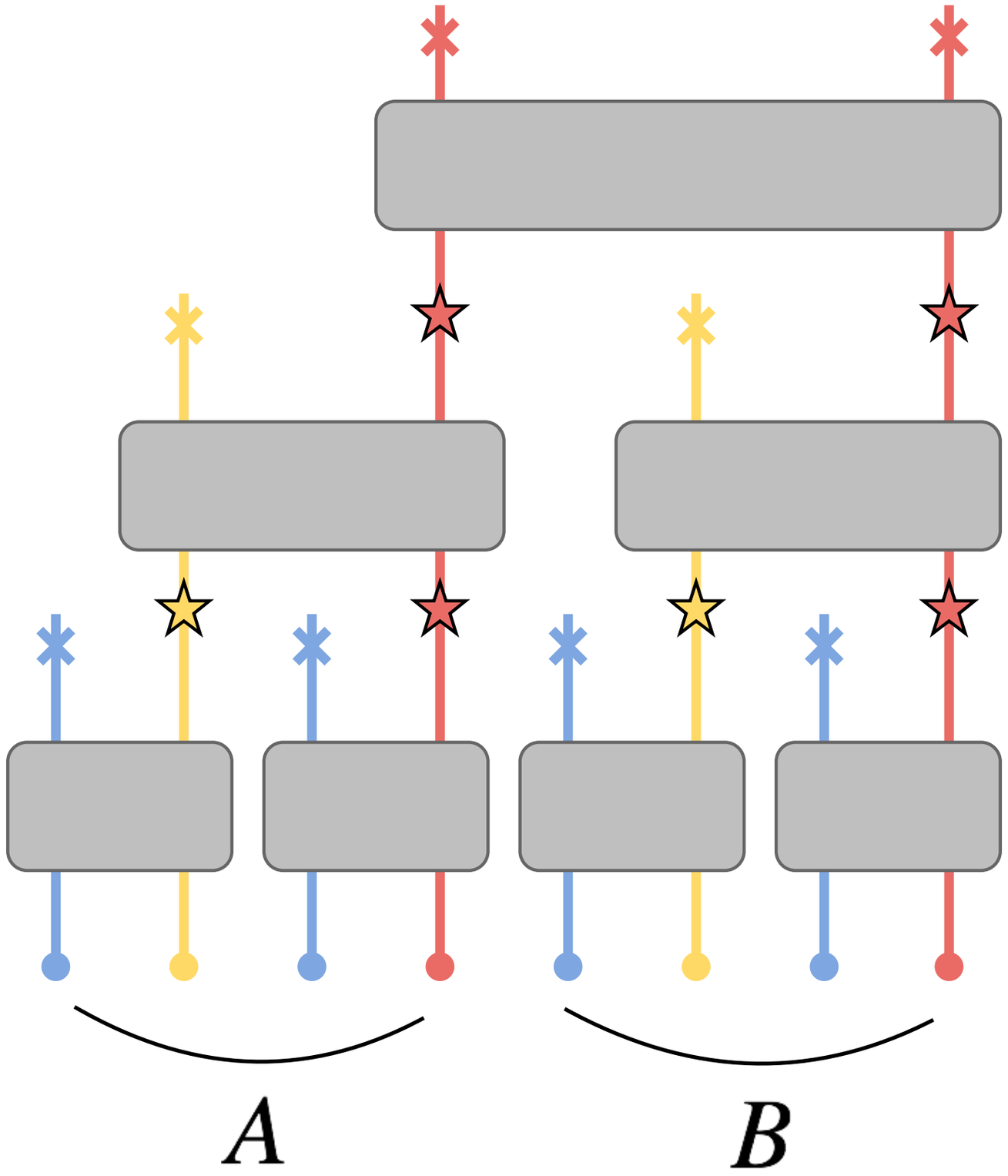


Correlation length ~ Network depth

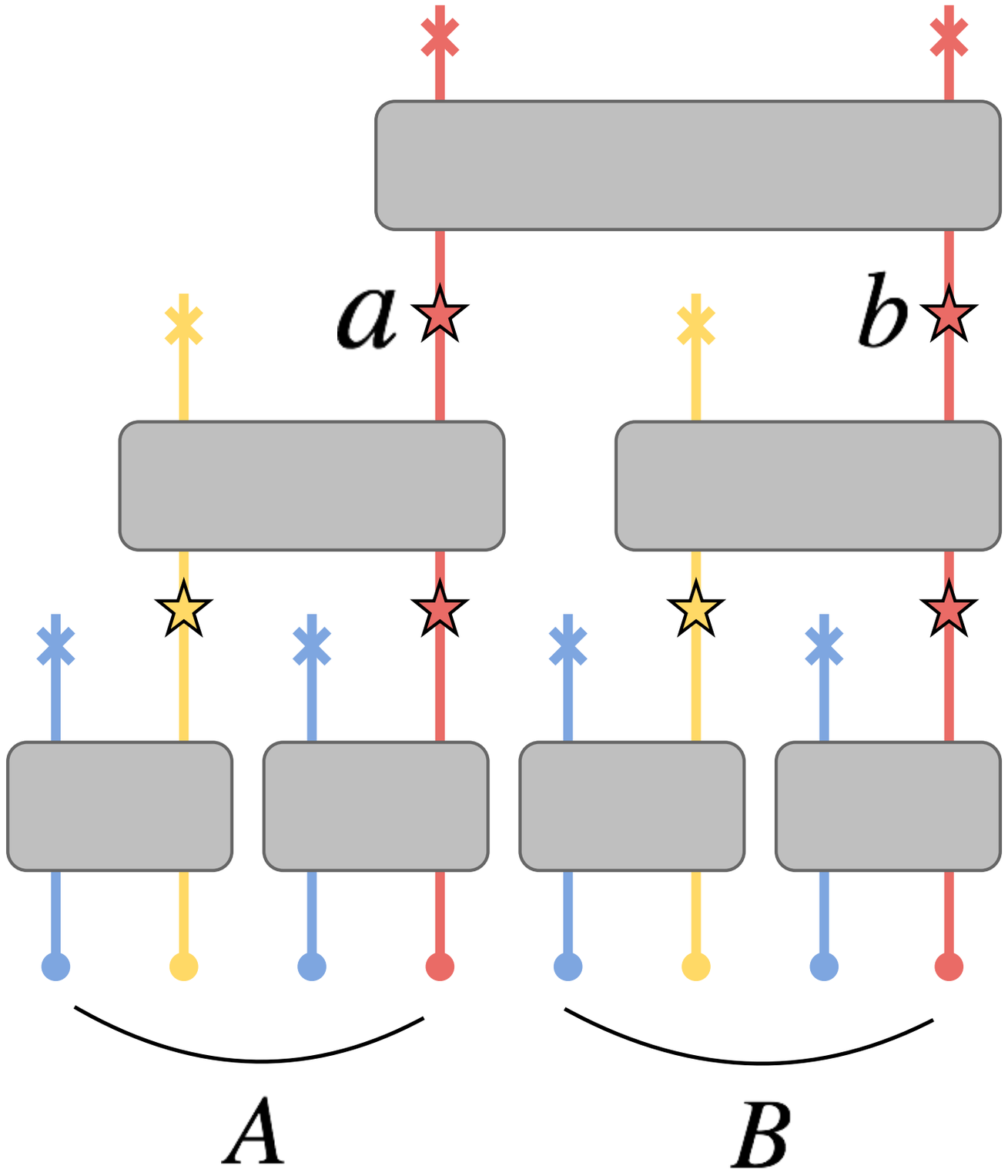
“Decimator” only architecture



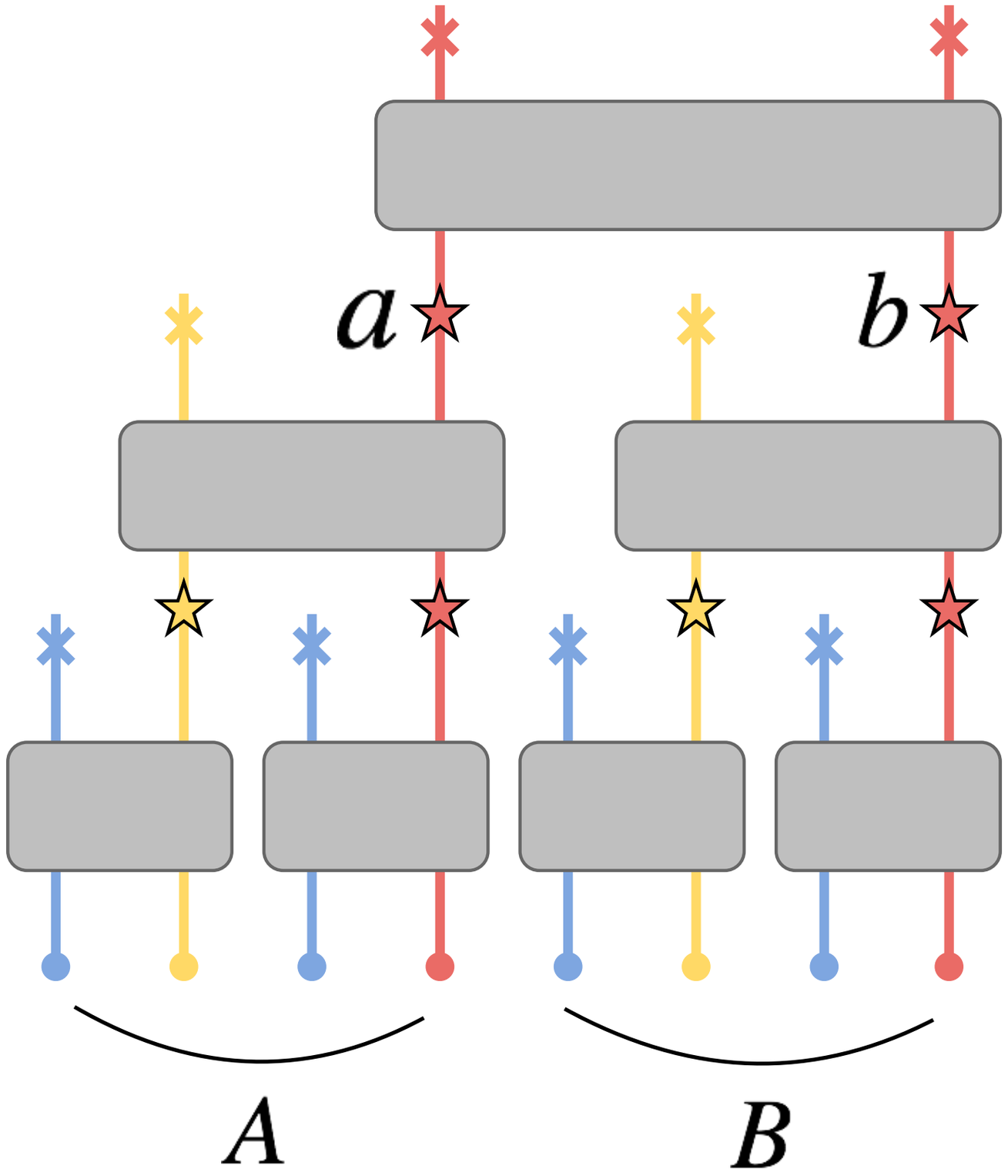
“Decimator” only architecture



“Decimator” only architecture



“Decimator” only architecture



$I(A:B) = I(a:b)$ Mutual Information Bottleneck

Probability Density *Estimation*

Given a dataset, learn its probability density by minimizing the negative likelihood

$$\mathrm{NLL}_{\boldsymbol{\theta}} = -\sum_{\boldsymbol{x} \in \text{dataset}} \ln q_{\boldsymbol{\theta}}(\boldsymbol{x})$$

Network parameters

Equivalent to reduce the Kullback–Leibler divergence

$$\mathbb{KL}\left(\frac{\pi(\boldsymbol{x})}{Z} \,\middle\|\, q_{\boldsymbol{\theta}}(\boldsymbol{x})\right)$$

However, typical Stat-Mech problem has access only to the energy function

Probability Density *Distillation*

Learn from the data generated by the network itself

$$\mathcal{L}_{\theta} = \int \mathrm{d}\boldsymbol{x}\, q_{\theta}(\boldsymbol{x}) \left[\ln q_{\theta}(\boldsymbol{x}) - \ln \pi(\boldsymbol{x})\right]$$

Probability Density *Distillation*

Learn from the data generated by the network itself

$$\mathcal{L}_{\theta} = \int \mathrm{d}\boldsymbol{x}\, q_{\theta}(\boldsymbol{x}) \left[\ln q_{\theta}(\boldsymbol{x}) - \ln \pi(\boldsymbol{x})\right]$$

Negentropy

Probability Density *Distillation*

Learn from the data generated by the network itself

$$\mathcal{L}_{\theta} = \int \mathrm{d}\boldsymbol{x}\, q_{\theta}(\boldsymbol{x}) \left[\ln q_{\theta}(\boldsymbol{x}) - \ln \pi(\boldsymbol{x})\right]$$

Negentropy Energy

Probability Density *Distillation*

Learn from the data generated by the network itself

$$\mathcal{L}_\theta = \int \mathrm{d}\boldsymbol{x}\, q_\theta(\boldsymbol{x}) \left[\ln q_\theta(\boldsymbol{x}) - \ln \pi(\boldsymbol{x})\right]$$

"Variational Free Energy"

Negentropy

Energy

Probability Density *Distillation*

Learn from the data generated by the network itself

$$\mathcal{L}_\theta = \int \mathrm{d}\boldsymbol{x}\, q_\theta(\boldsymbol{x}) \left[\ln q_\theta(\boldsymbol{x}) - \ln \pi(\boldsymbol{x})\right]$$

"Variational Free Energy"

Negentropy

Energy

$$Z = \int \mathrm{d}\boldsymbol{x}\, \pi(\boldsymbol{x})$$

Partition function

Probability Density *Distillation*

Learn from the data generated by the network itself

$$\mathcal{L}_\theta = \int \mathrm{d}\boldsymbol{x}\, q_\theta(\boldsymbol{x}) \left[\ln q_\theta(\boldsymbol{x}) - \ln \pi(\boldsymbol{x})\right]$$

"Variational Free Energy" — Negentropy — Energy

$$\mathcal{L}_\theta + \ln Z = \mathbb{KL}\left(q_\theta(\boldsymbol{x}) \,\middle\|\, \frac{\pi(\boldsymbol{x})}{Z}\right) \geq 0$$

The loss function is lower bounded by the physical free energy (Gibbs-Bogoliubov-Feynman inequality)

“Reparametrization trick”

How to compute the gradient w.r.t random sampling ?

$$\mathcal{L}_\theta = \mathop{\mathbb{E}}_{z \sim p(z)} \left[\ln q(g_\theta(z)) - \ln \pi(g_\theta(z)) \right]$$

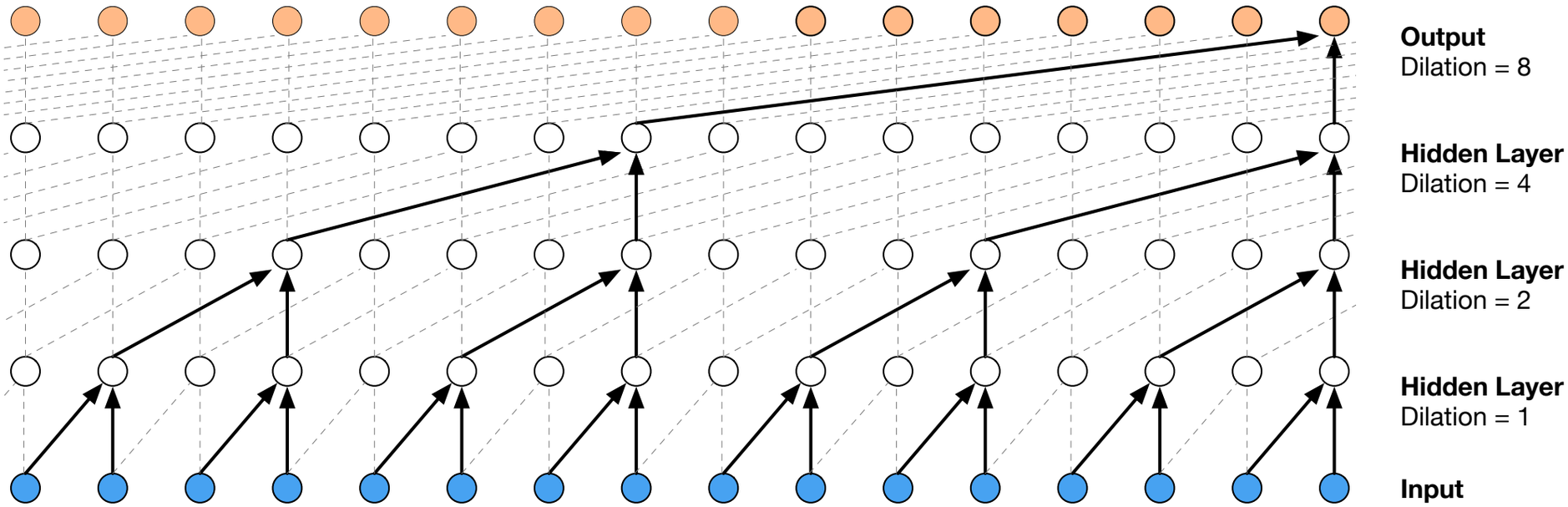
Sample from the prior distribution

Network parameters

End-to-end training using **back-propagation**

Interlude: WaveNet Story

generated signal



input noise

Given a parallel WaveNet student $p_S(\boldsymbol{x})$ and WaveNet teacher $p_T(\boldsymbol{x})$ which has been trained on a dataset of audio, we define the *Probability Density Distillation* loss as follows:

$$D_{\mathrm{KL}}\left(P_S||P_T\right) = H(P_S, P_T) - H(P_S) \tag{6}$$

DeepMind

https://deepmind.com/blog/wavenet-generative-model-raw-audio/ 1609.03499
https://deepmind.com/blog/high-fidelity-speech-synthesis-wavenet/ 1711.10433

Demo: Ising model

$$\pi(\boldsymbol{s}) = \exp\left(\frac{1}{2}\boldsymbol{s}^T K \boldsymbol{s}\right)$$

Demo: Ising model

$$\pi(\boldsymbol{s}) = \exp\left(\frac{1}{2}\boldsymbol{s}^T K \boldsymbol{s}\right)$$

decouple

$$\propto \int \mathrm{d}\boldsymbol{x} \exp\left(-\frac{1}{2}\boldsymbol{x}^T \left(K + \alpha I\right)^{-1} \boldsymbol{x} + \boldsymbol{s}^T \boldsymbol{x}\right)$$

Demo: Ising model

$$\pi(\boldsymbol{s}) = \exp\left(\frac{1}{2}\boldsymbol{s}^T K \boldsymbol{s}\right)$$

decouple

$$\propto \int \mathrm{d}\boldsymbol{x} \exp\left(-\frac{1}{2}\boldsymbol{x}^T \left(K + \alpha I\right)^{-1} \boldsymbol{x} + \boldsymbol{s}^T \boldsymbol{x}\right)$$

trace out s

$$\pi(\boldsymbol{x}) = \exp\left(-\frac{1}{2}\boldsymbol{x}^T \left(K + \alpha I\right)^{-1} \boldsymbol{x}\right) \prod_i \cosh(x_i)$$

Demo: Ising model

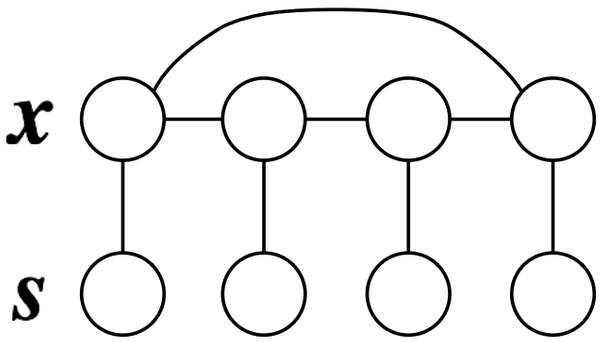
$$\pi(\boldsymbol{s}) = \exp\left(\frac{1}{2}\boldsymbol{s}^T K \boldsymbol{s}\right)$$

decouple

$$\propto \int \mathrm{d}\boldsymbol{x} \exp\left(-\frac{1}{2}\boldsymbol{x}^T \left(K + \alpha I\right)^{-1} \boldsymbol{x} + \boldsymbol{s}^T \boldsymbol{x}\right)$$

trace out s

$$\pi(\boldsymbol{x}) = \exp\left(-\frac{1}{2}\boldsymbol{x}^T \left(K + \alpha I\right)^{-1} \boldsymbol{x}\right) \prod_i \cosh(x_i)$$

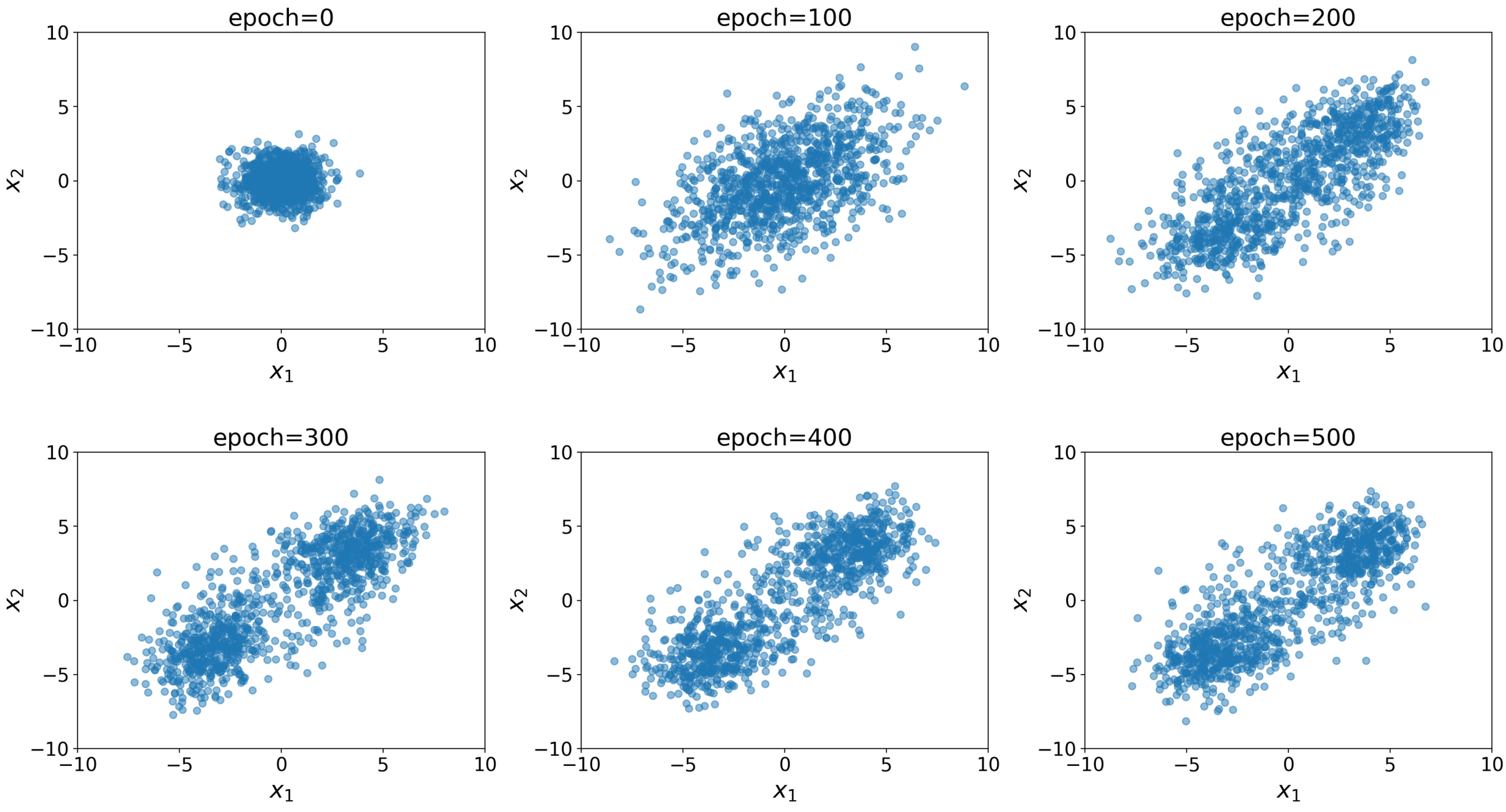


$$\pi(\boldsymbol{s}|\boldsymbol{x}) = \prod_i \left(1 + e^{-2 s_i x_i}\right)^{-1}$$

continuous dual of the Ising model

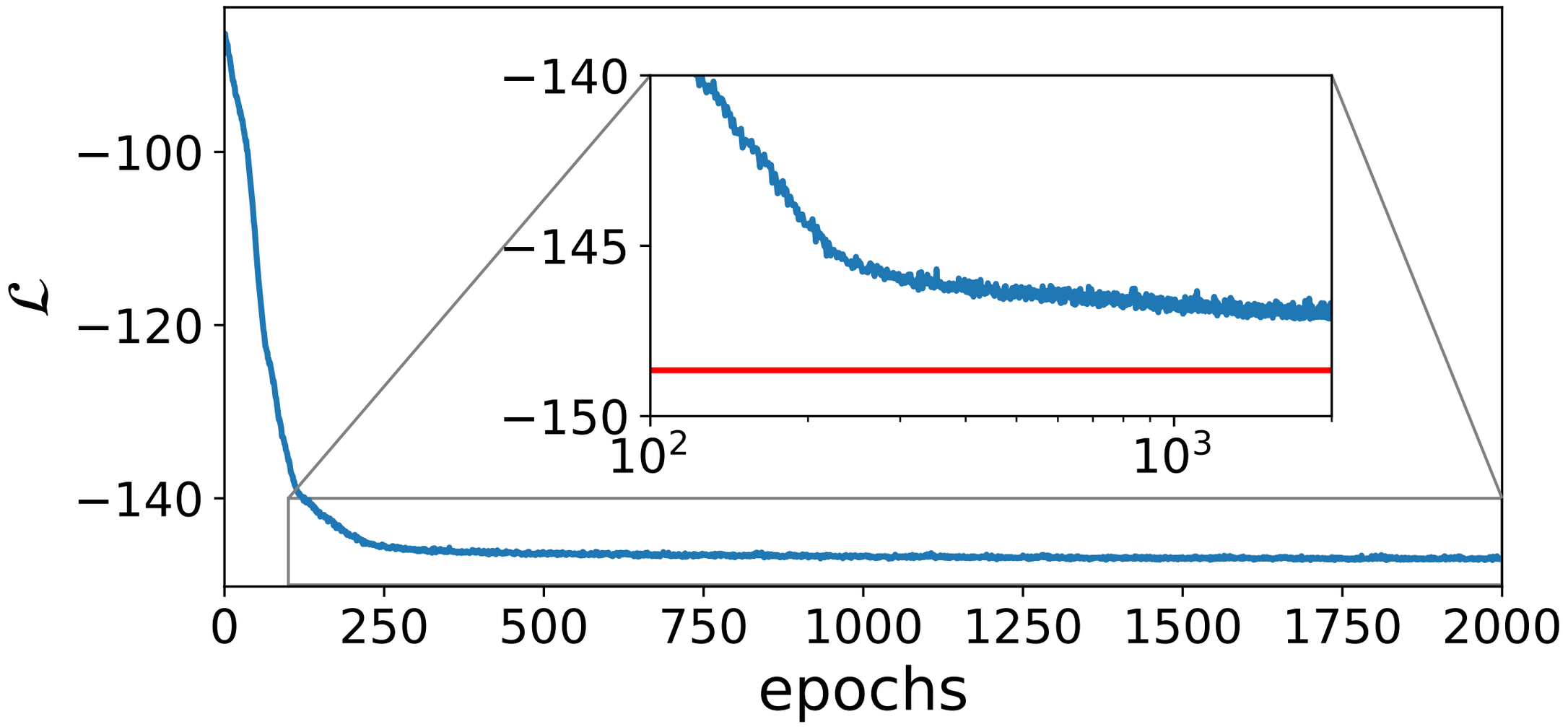
"Gaussian-Bernoulli Boltzmann Machine" Zhang, Sutton, Storkey, Ghahramani, NIPS 2012

Generated Samples



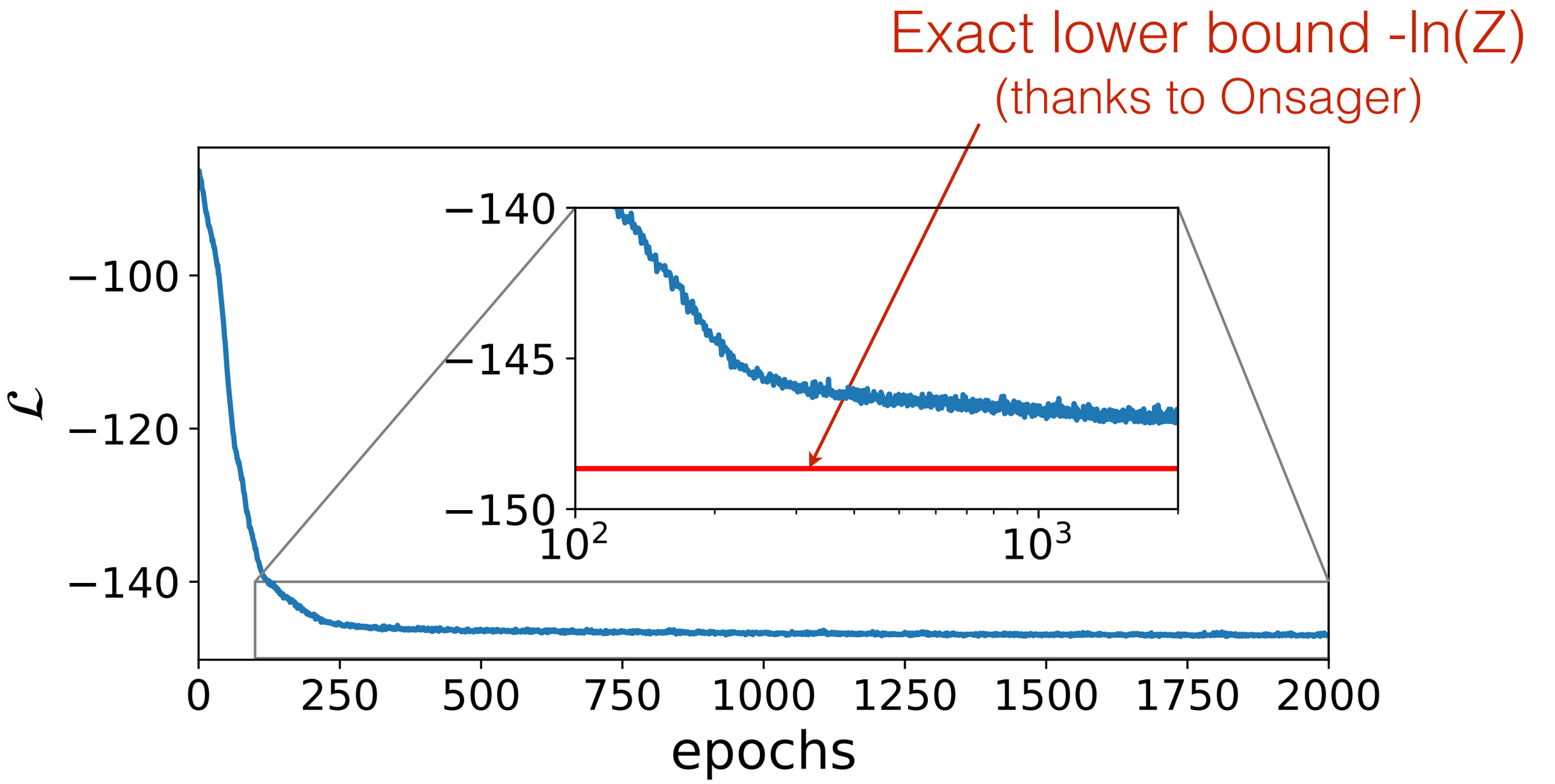
x’s are continuous fields dual to Ising spins

Variational Loss



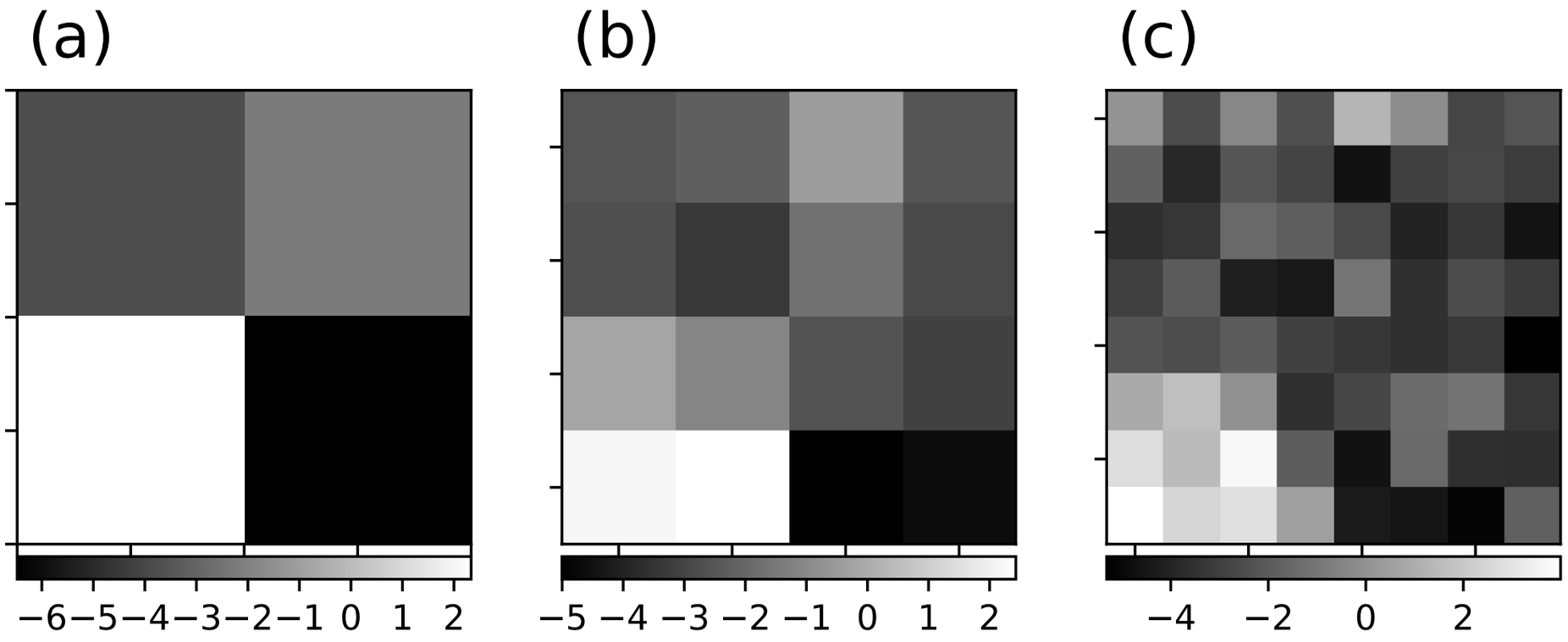
Loss can be further improved by using more expressive networks

Variational Loss



Loss can be further improved by using more expressive networks

Renormalized Collective Variables



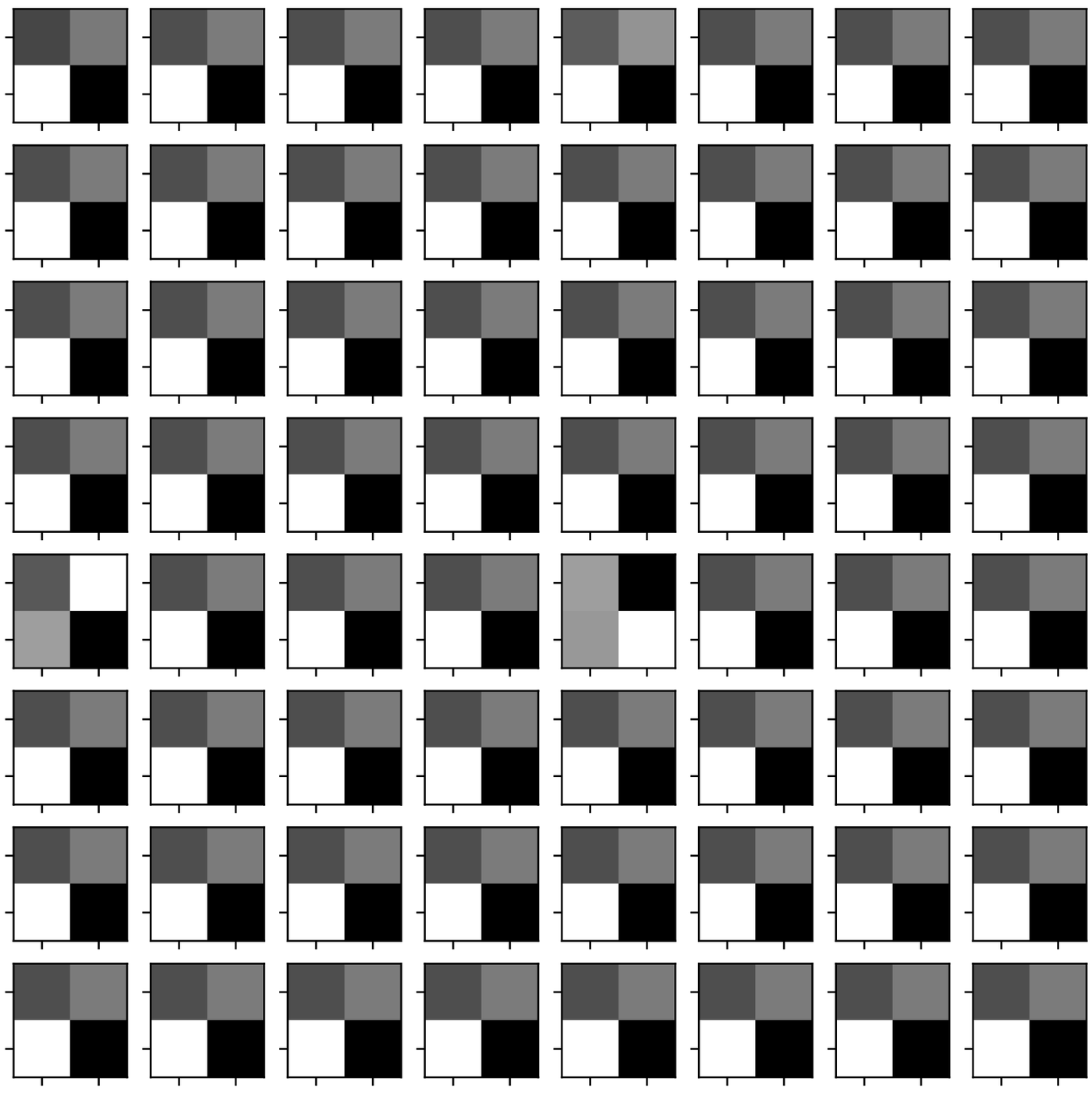
2x2

4x4

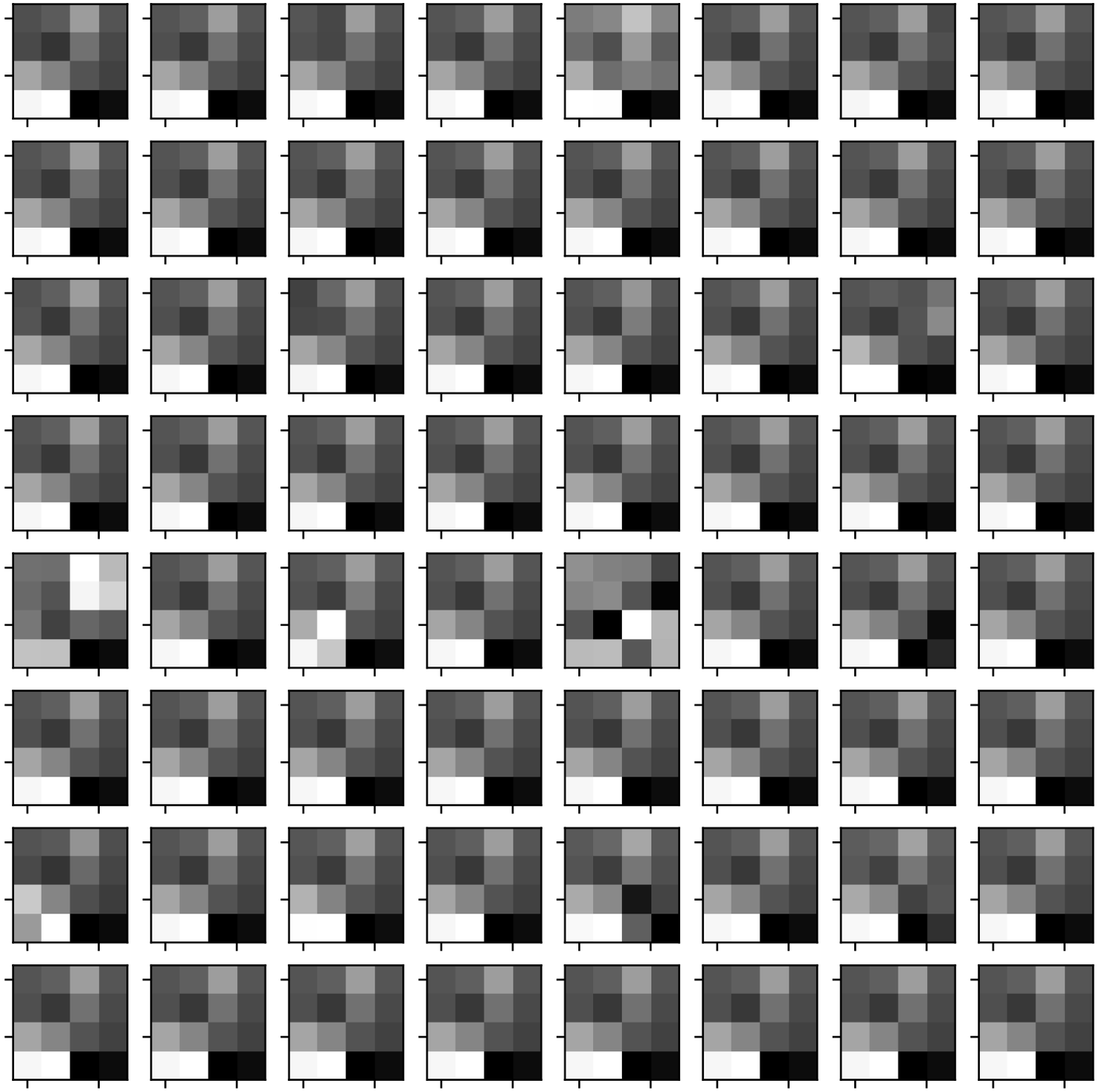
Physical Variables

Also know their effective couplings
=> renormalized energy function

Wander in the Latent Space



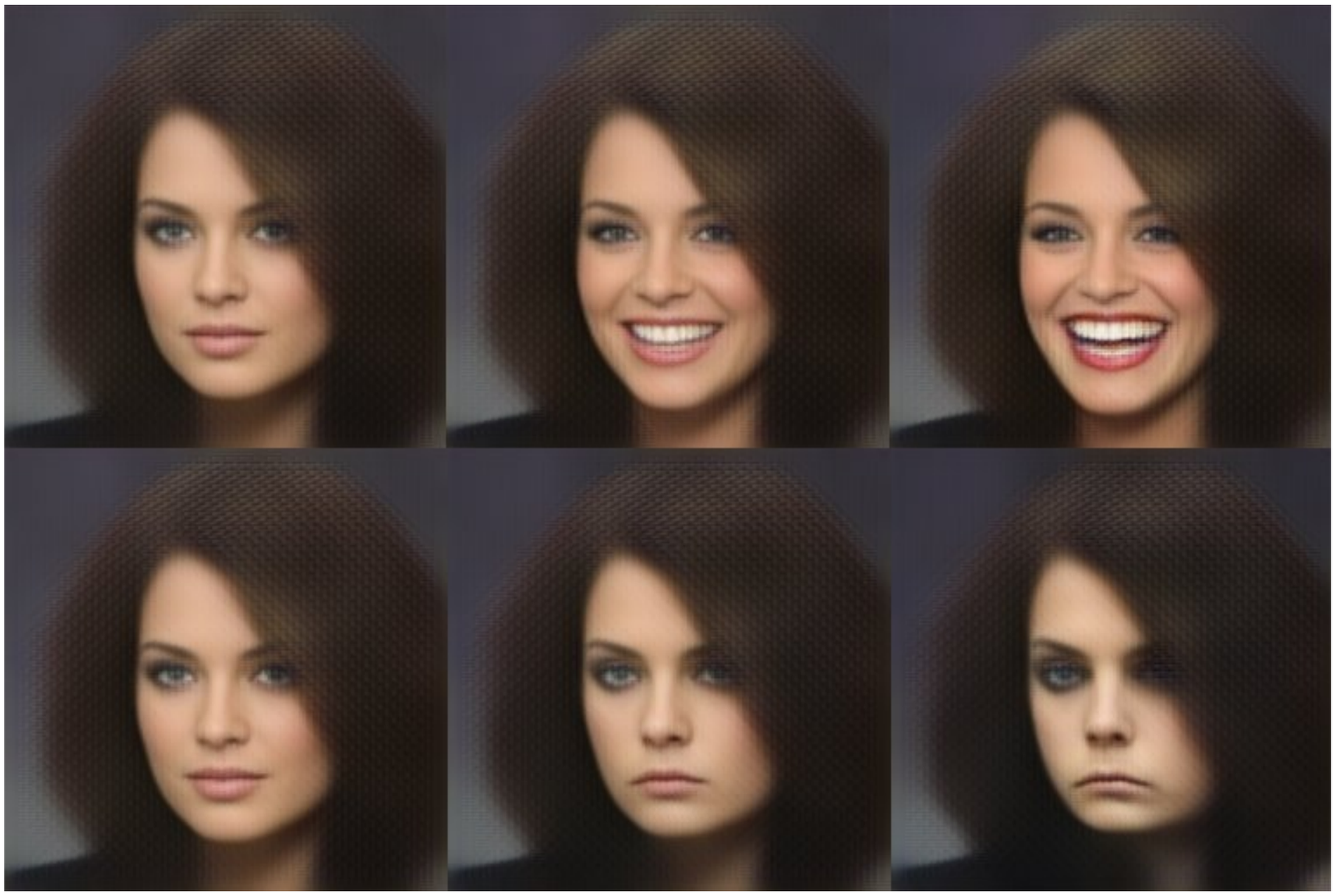
Wander in the Latent Space



Wander in the Latent Space

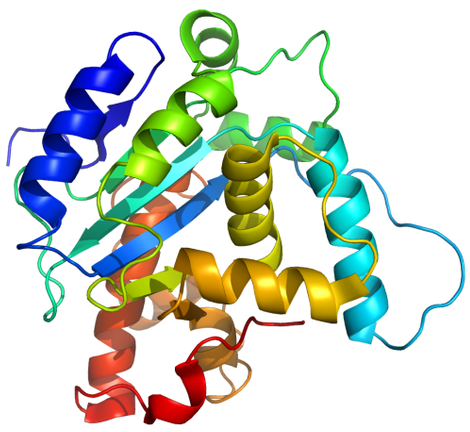
Wander in the Latent Space

Arithmetics of the “smile vector”

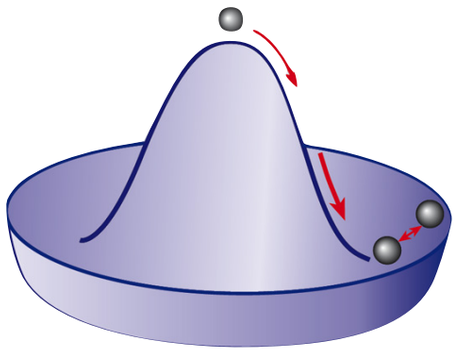


White,1609.04468 implemented using the variational autoencoder by Kingma and Welling,1312.6114

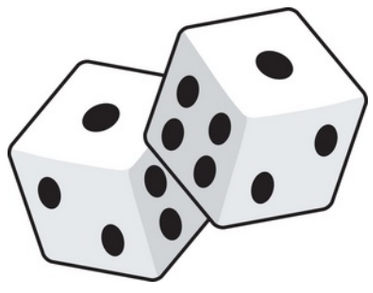
How is it useful ?



Automatically identify collective variables
(metadynamics molecular simulation)



Automatically derive effective field theory
(free energy surface)



Monte Carlo update proposals

Sampling in the latent space

Change-of-variables in a learnable way

$$Z = \int \mathrm{d}\boldsymbol{x}\, \pi(\boldsymbol{x}) = \int \mathrm{d}\boldsymbol{z}\, \pi(g(\boldsymbol{z})) \left| \det \left(\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{z}} \right) \right|$$

Physical Prob. Dist.

Latent variable Prob. Dist.

Latent space is less correlated, therefore, easier to sample

Metropolized Independent Sampler

Acceptance rate with detailed balance condition

$$A(\boldsymbol{x} \to \boldsymbol{x}') = \min\left[1, \frac{q(\boldsymbol{x})}{q(\boldsymbol{x}')} \cdot \frac{\pi(\boldsymbol{x}')}{\pi(\boldsymbol{x})}\right]$$

Propose Ratio

Physical Probability

- Unbiased physics even for imperfect proposals
- Proposals are independent

Surrogate energy function:

Li Huang and LW, 1610.02746
Liu, Qi, Meng, Fu, 1610.03137

Trainable transition kernel:

Song, Zhao, Ermon, 1706.07561
Levy, Hoffman, Sohl-Dickstein, 1711.09268

Remarks on TNS Connection

- What we had is a classical downgrade of MERA (Bény 2013)

  Probability Density~ Quantum Wavefuntion

  Classical Mutual Information ~ Entanglement Entropy

  “Decorrelator" ~ Disentangler

  Decimator~Isometry

  Bijector~Unitary

- Deep Learning machinery provides structural flexibility, modular abstraction, end-to-end training

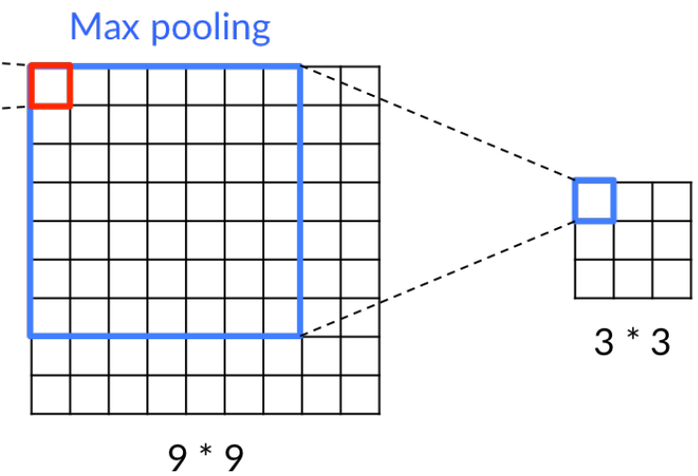
- We give back to DL understandings of what are they doing (and hopefully, how to do better)

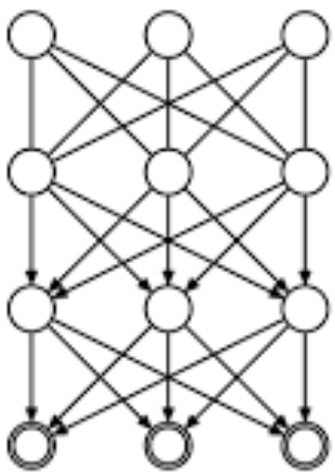
Remarks on DL

Old Wisdoms

Pooling layer in ConvNets
~ Decimation

Hidden nodes in RBM/DBN
~ Renormalized Variables

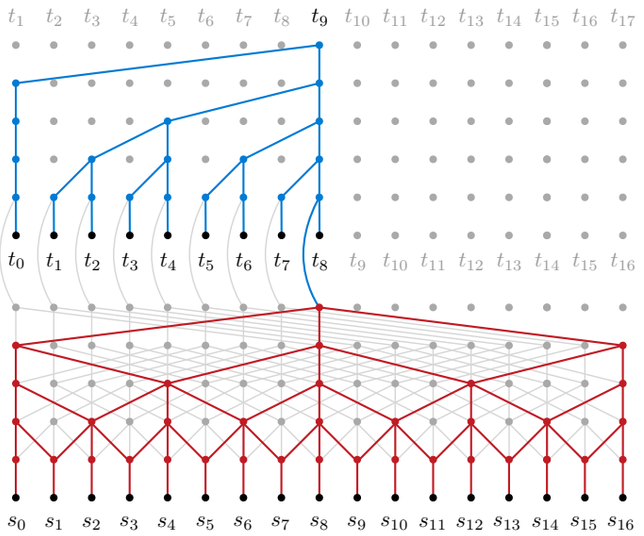


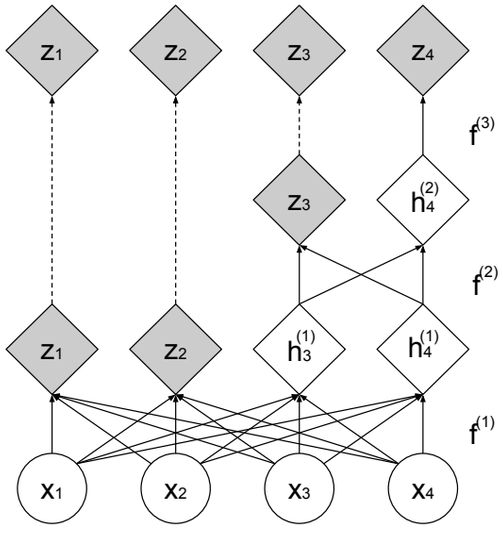


New Insights

Dialed convolution + Factor out layers = Decimation

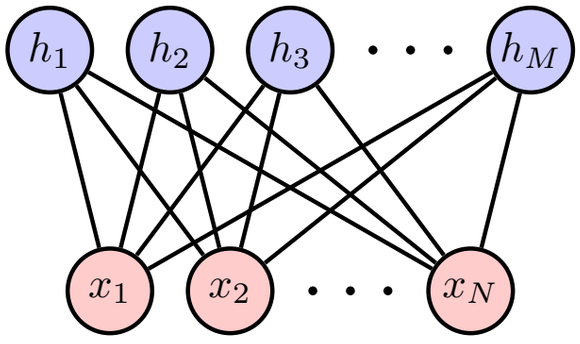
Kept latent variables = Renormalized Variables

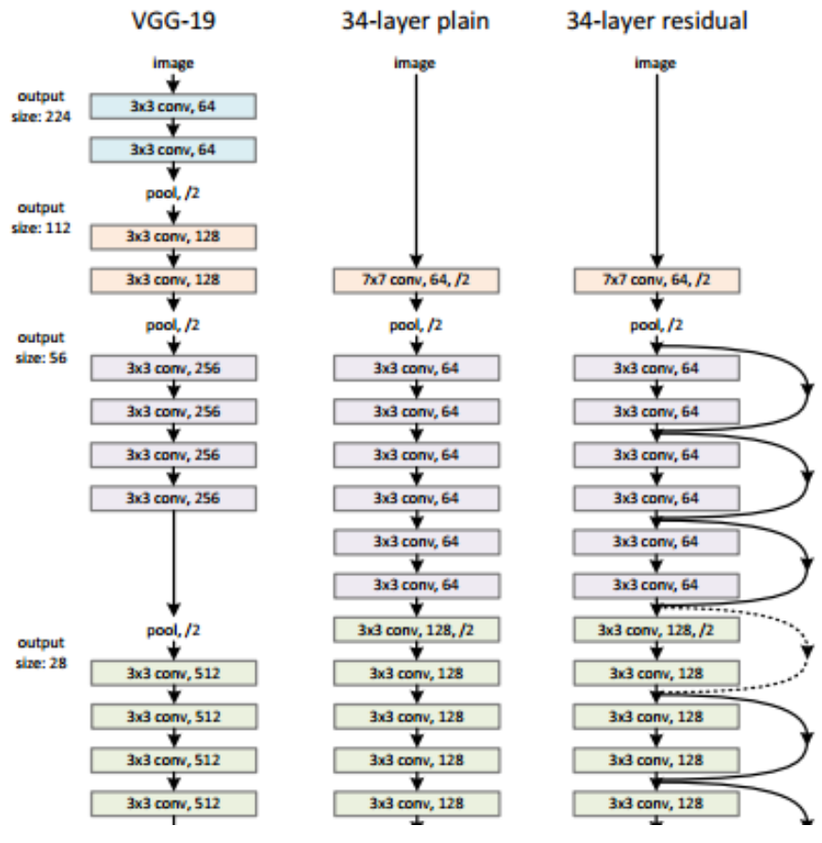




Spherical chicken
in vacuum

Animals
in the wild

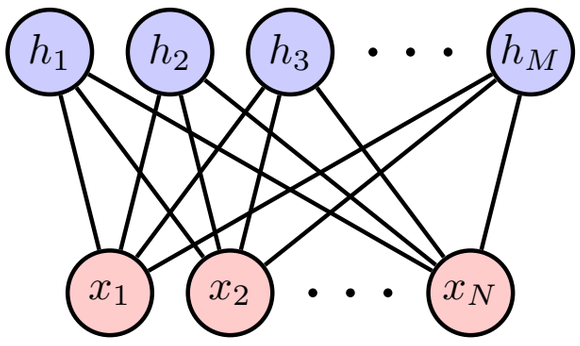


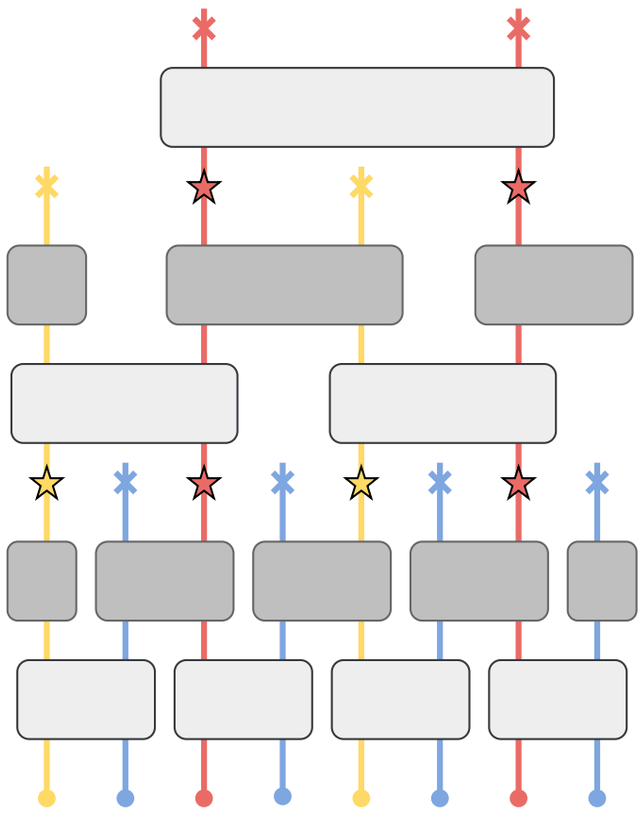


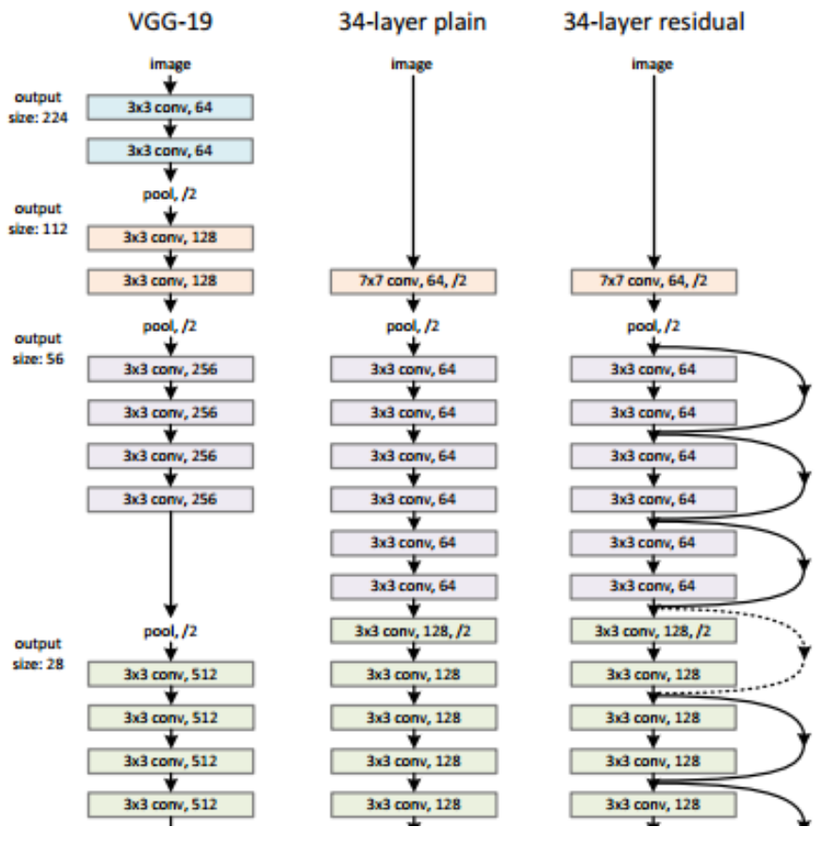
Here we are

Spherical chicken in vacuum

Animals in the wild







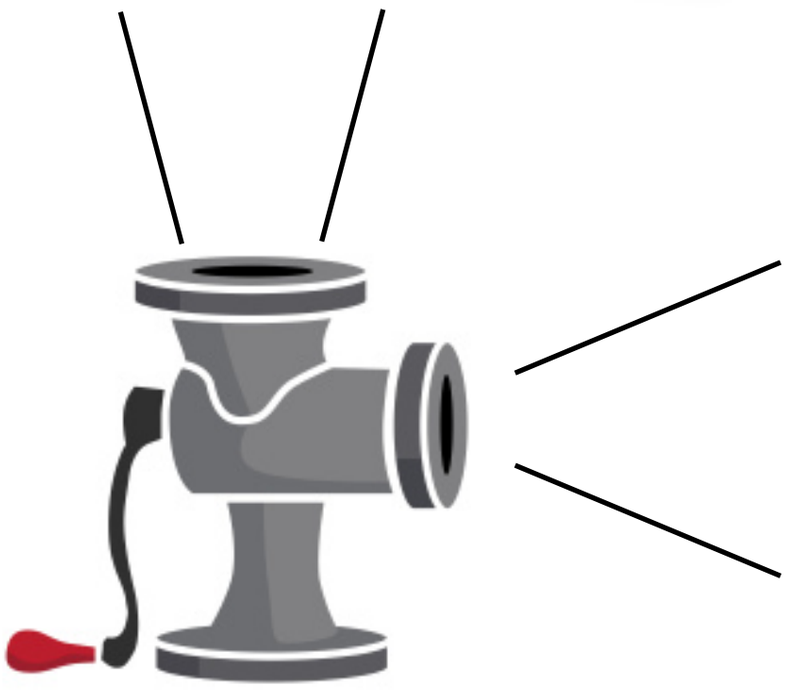
Simplified, but not oversimplified model with balanced interpretability and expressibility

Remarks on RG

- Conventionally, RG is a semi-group, not a group
- NeuralRG builds on bijectors, hence a group (coarse-graining due to the multiscale structure)
- Probabilistic (Jona-Lasinio 75') and Information Theory (Apenko 09') views on RG (same is true for neural & tensor networks)
- Diffeomorphism does not change topology of the manifolds, therefore, may be limited.

The Universe as a Generative Model

$$\mathcal{L} = \int d^4x \sqrt{-g} \Big[\frac{m_p^2}{2} R - \frac{1}{4} F^a_{\mu\nu} F_a^{\mu\nu} + i\bar{\psi}^i \Gamma^\mu D_\mu \psi^i + \left(\bar{\psi}^i_L V_{ij} \Phi \psi^j_R + \text{h.c.} \right) - |D_\mu \Phi|^2 - V(\Phi) \Big]$$



RG = Infer the Effective Field Theory

Thank you!