Towards interpretable and scalable machine learning for lattice field theory

Chanju Park, Gert Aarts, Biagio Lucini

c.j.park@swansea.ac.uk

Department of Physics, Swansea University

YTF, Durham, 14 Dec 2023

Lattice Field Theory simulation

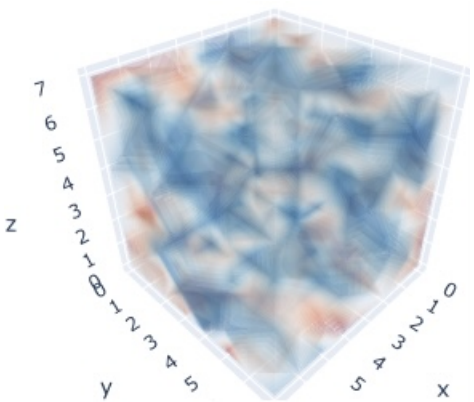


Figure: One time slice of 4D U(1) gauge theory configuration.

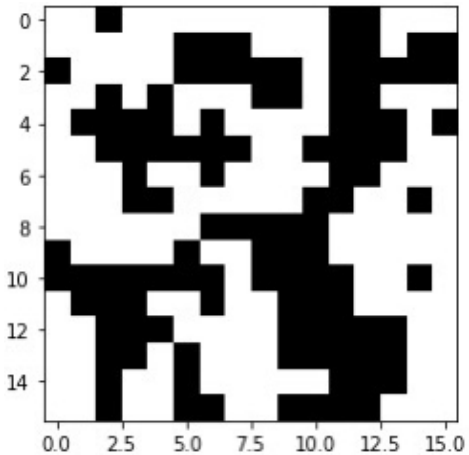


Figure: 2D Ising model configuration.

- The distribution of particles across the space and time are described by **Boltzmann distribution**.

$$p(\psi, \bar{\psi}, A) = \frac{1}{Z} e^{-S[\psi, \bar{\psi}, A]} \tag{1}$$

- The **configurations**, which are snapshots of the discretized space-time with fields, are sampled from the Boltzmann distribution using **Monte Carlo algorithm**.
- Then the expectation value of a observable is calculated by averaging over the value of the observable in each configuration.

$$\langle O \rangle \simeq \frac{1}{N} \sum_{i=1}^{N} O_i \tag{2}$$

Some limitations of Monte Carlo simulation

Critical slowing down

- Computational cost diverges near the critical point of the system due to the diverging correlation length.

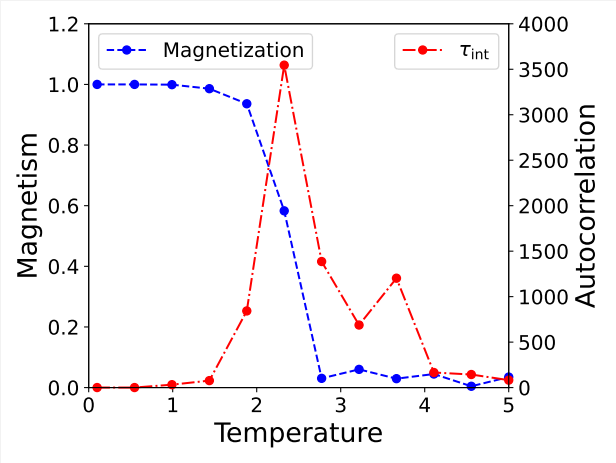


Figure: Autocorrelation increases near the phase transition. Example from 2D Ising model.

$$N_{\text{independent}} = \frac{N}{2\tau_{\text{int}}} \qquad (3)$$

Sign problem

- Sampling efficiency worsens when the probability distribution is complex.

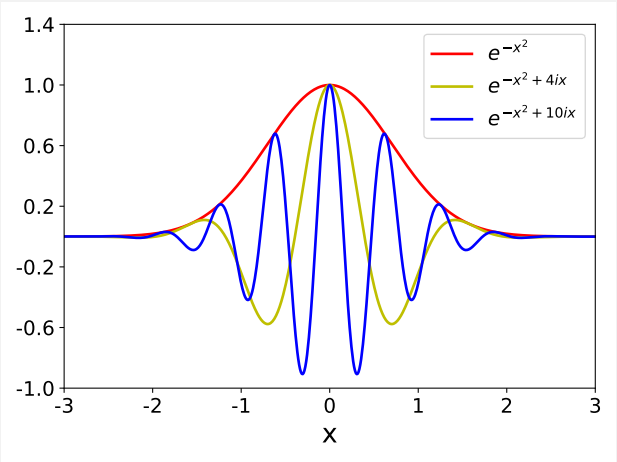


Figure: Real part of the complex probability density function.

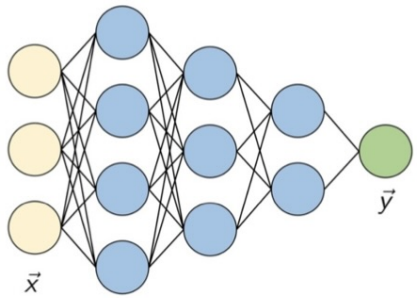
The probability distribution is highly oscillatory.

Generative machine learning models

Generative machine learning model is a machine learning model that maps between input distribution and the target distribution.

$$\vec{x}_0 \sim q(x) \rightarrow p(x_1|x_0) \rightarrow p(x_2|x_1) \cdots p(y|x_n) \rightarrow \vec{y} \sim p(y)$$

Input distribution — Markov process — Target distribution

$$\vec{x}_0 \sim q(x) \rightarrow$$  $$\rightarrow \vec{y} \sim p(y)$$

Input distribution — Generative model — Target distribution

Markov chain can be replaced by a single neural network and can be run in parallel, accelerating the sampling process.

Examples of generative machine learning models

Anonymous
What is generative machine learning model?

ChatGPT
A generative machine learning model is a type of model that is trained to generate new data that is similar to the training data it was exposed to. In other words, it learns the underlying patterns and structures of the data and can then generate new samples that share those characteristics.

Generative models are often contrasted with discriminative models. Discriminative models, in contrast, are trained to distinguish between different classes or categories in the input data.

There are various types of generative models, and one common approach is using Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs).

1. **Generative Adversarial Networks (GANs):** GANs consist of two neural networks, a generator, and a discriminator, that are trained simultaneously through adversarial training. The generator creates new samples, and the discriminator tries to distinguish between real and generated samples. The training process continues until the generator produces samples that are indistinguishable from real data.
2. **Variational Autoencoders (VAEs):** VAEs are a type of autoencoder that learn probabilistic mapping from the input space to a latent space. They generate by sampling from this latent space and then decoding the samples back into data space. VAEs are probabilistic in nature and are trained to generate sam a specific distribution.

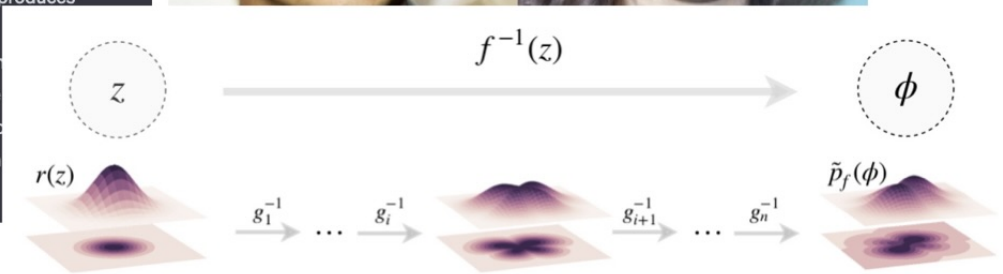
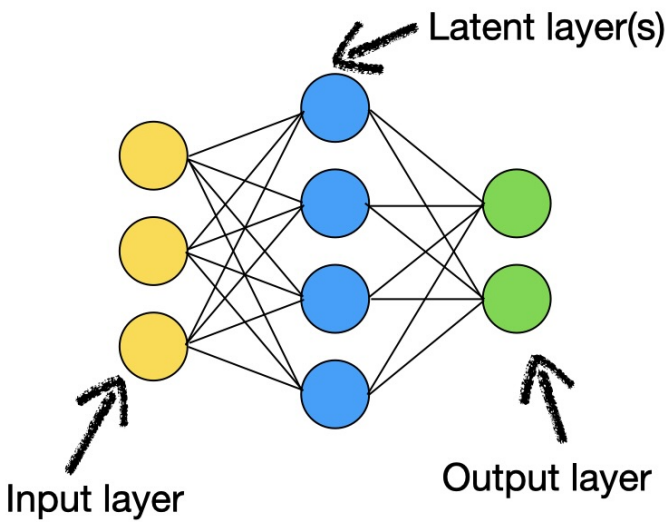


Figure: Chat-GPT, GAN, Normalizing flow. arXiv:2006.11239, arXiv:2101.08176

Typical structure of neural network



- Output $\vec{y}$ of the network

  $\vec{y} = f(W\vec{x})$, fitting parameters W

- Minimize the objective function

  $\mathcal{L} = \mathcal{L}(\hat{y}, \vec{y}) \rightarrow$ find optimal W

List of hyperparameters

1. Number of layers/width of each layer $\Rightarrow$ Volume of your model
2. Activation function "f" (ReLU, tanh, sigmoid, etc)
3. Objective function "$\mathcal{L}$" (MSE, cross entropy, etc)
4. Optimization algorithm (SGD, Adam, Newton-Rapson, Runge-Kutta, etc)

   ⋮

ML $\Rightarrow$ Highly non-linear optimization problem

Requirements from Physics/ML

However, physical systems require some properties from sampling algorithms that have to be satisfied to be considered reliable.

Physics Nat. Rev. Phys, 5, 526-535	**Machine Learning**
• Respect symmetries	• Type of activation function arXiv:2305.02334
• Exactness	• Choice of structure arXiv:2309.10688
• Scalability	• Volume of the model arXiv:2209.04882
• Tractable likelihood	• Initialization of weights

Restricted Boltzmann Machine (RBM)

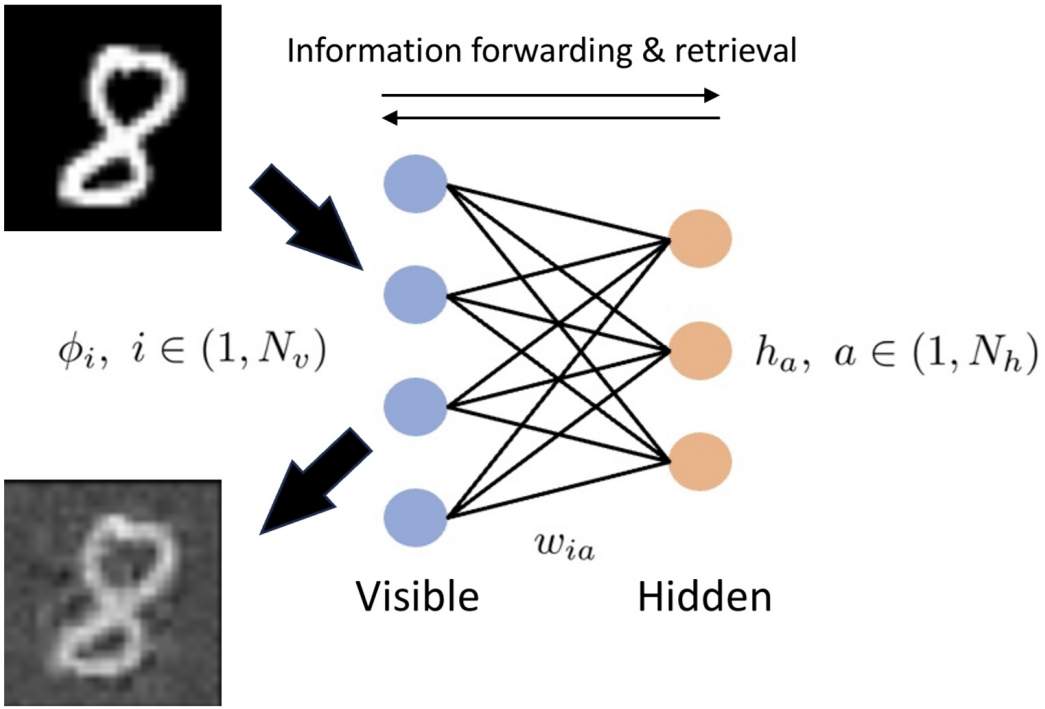


Figure: Structure of a Restricted Boltzmann Machine.

- Restricted Boltzmann Machine (RBM) is a generative machine learning model consisting of two layers of nodes.
- The probability distribution of the output layer is given by combination of training parameters.

$$p(\phi) = \frac{1}{Z} e^{-\frac{1}{2}\phi^T K \phi} \qquad (4)$$

where $K = \mu^2 \mathbb{1} - \sigma^2 W W^T$ is called **RBM kernel**.

Training dynamics

Model parameters are updated by gradient assent to maximize the log-likelihood function. For given M number of data from the target distribution, the log-likelihood function is given by

$$\mathcal{L}(\phi|W) = \frac{1}{M}\sum_{d}^{M} \log p(\phi^{(d)}; W), \qquad W_{ia}^{(t+1)} = W_{ia}^{(t)} + \eta \frac{\partial \mathcal{L}}{\partial W_{ia}} \tag{5}$$

The gradient of the log-likelihood function is given by

$$\frac{\partial \mathcal{L}}{\partial W_{ia}} = \sigma_h^2 \sum_j \left[\langle \phi_i \phi_j \rangle_{\text{target}} - \langle \phi_i \phi_j \rangle_{\text{model}}\right] W_{aj} \tag{6}$$

$$= \sigma_h^2 \sum_j \left[K^{-1}_{\text{target},ij} - K^{-1}_{ij}\right] W_{aj} \tag{7}$$

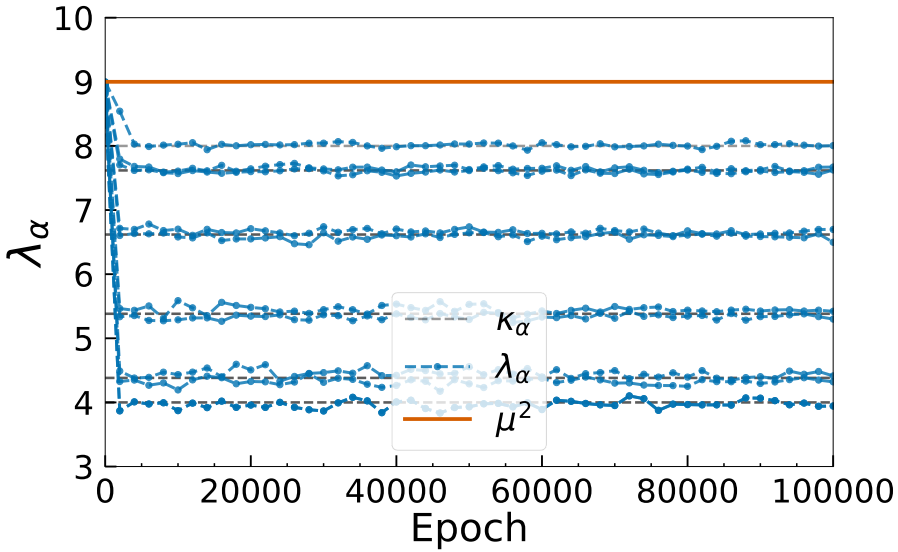
where, $K^{-1}_{\text{target},ij} = \langle \phi_i \phi_j \rangle_{\text{target}} \equiv \frac{1}{M}\sum_d^M \phi_i^{(d)} \phi_j^{(d)}$ and $K^{-1}_{ij} = \langle \phi_i \phi_j \rangle_{\text{model}}$ are the two point correlation function of the training dataset and RBM respectively.

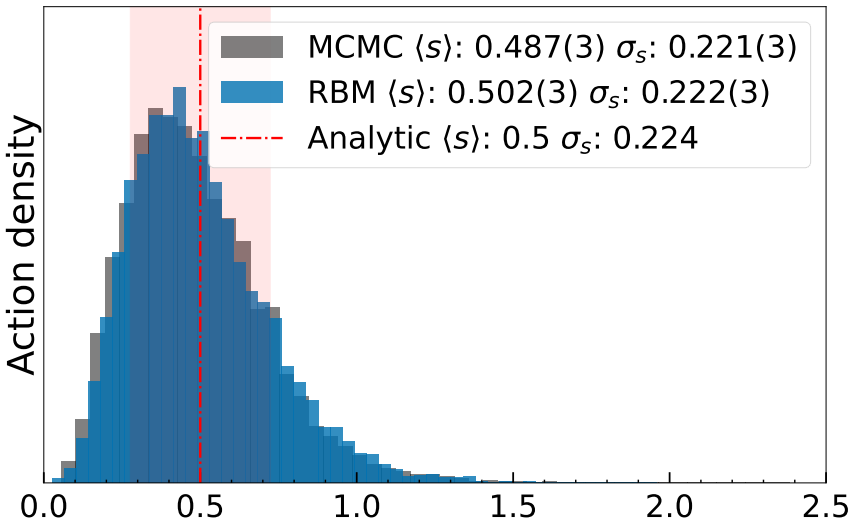
A. Decelle and C. Furtlehner, arXiv:2011.11307, (2021)

G. Hinton, UTML TR 2010-003 lecture note, (2010)

Numerical results: Scalar field theory

Target distribution: $p(\phi) = \frac{1}{Z} e^{-\frac{1}{2}\phi(\partial^2 - m^2)\phi}$





$\lambda_\alpha = \mu^2 \mathbb{1} - \sigma_h^2 \xi_\alpha^2$: Eigenvalue of K

$\kappa_\alpha = p^2 + m^2 = (2 + m^2) + 2\cos\left(\frac{2\pi}{N}\alpha\right)$: Eigenvalue of K_{target}

Restricted Boltzmann Machine (RBM)

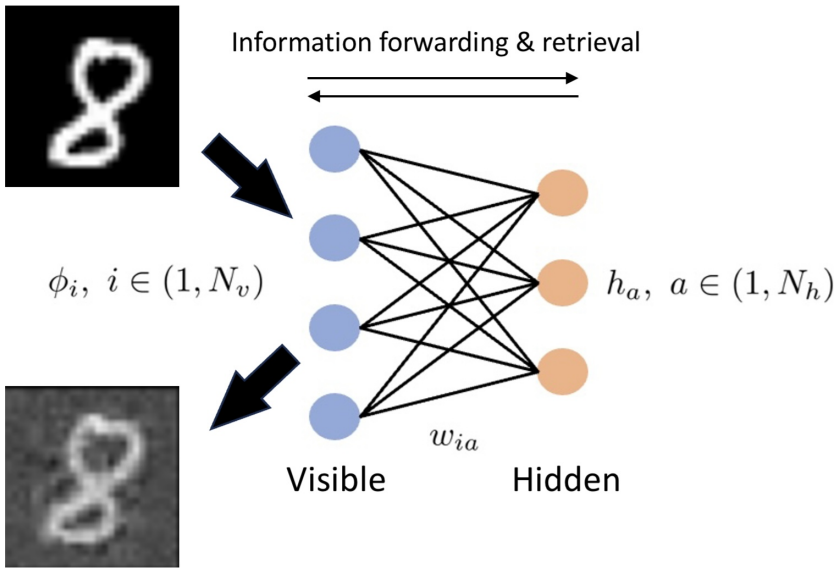


Figure: Structure of a Restricted Boltzmann Machine.

The probability distribution of the visible layer is given by

$$p(\phi) = \frac{1}{Z} e^{-\frac{1}{2}\phi^T K \phi}$$

where $K = \mu^2 \mathbb{1} - \sigma^2 W W^T$.

- What does it mean to have larger/smaller μ^2?

$$K > 0 \text{ model to converge } \Rightarrow \text{ sets maximum of } W^2$$

- What does it mean to have a larger/smaller number of hidden nodes?

$$W_{ij} \Rightarrow \text{ Rectangular matrix of size } (N_v, N_h)$$

Training dynamics in SVD basis

We can rewrite the gradient equation in terms of singular values of the coupling matrix W.

$$W = U\Xi V^T, \quad UU^T = \mathbb{1}, \quad VV^T = \mathbb{1} \tag{8}$$
$$\Xi = \text{diag}(\xi_1, \xi_2, \cdots) : \text{ singular values of } W,$$

Then,

$$K = \mu^2\mathbb{1} - \sigma_h^2 U\Xi\Xi^T U^T \equiv UD_K U^T, \quad K_{\text{target}} = O_\phi D_\phi O_\phi^T \tag{9}$$

where

$$D_K = \text{diag}(\mu^2 - \sigma_h^2\xi_1^2, \mu^2 - \sigma_h^2\xi_2^2, \cdots), \quad D_\phi = \text{diag}(\kappa_1, \kappa_2, \cdots)$$

$$\frac{\partial \mathcal{L}}{\partial W} = U\sigma_h^2 \left[U^T O_\phi D_\phi^{-1} O_\phi^T U - D_K^{-1}\right] \Xi V^T$$

From the gradient equation written in the singular value basis, we can find some interesting properties of training the RBM.

UV regularization by RBM mass parameter μ^2

We can rewrite the gradient equation in terms of singular values of the coupling matrix W.

$$\frac{\partial \mathcal{L}}{\partial \xi_\alpha} = \sigma_h^2 \left[\frac{1}{\kappa_\alpha} - \frac{1}{\mu^2 - \sigma_h^2 \xi_\alpha^2} \right] \xi_\alpha \tag{10}$$

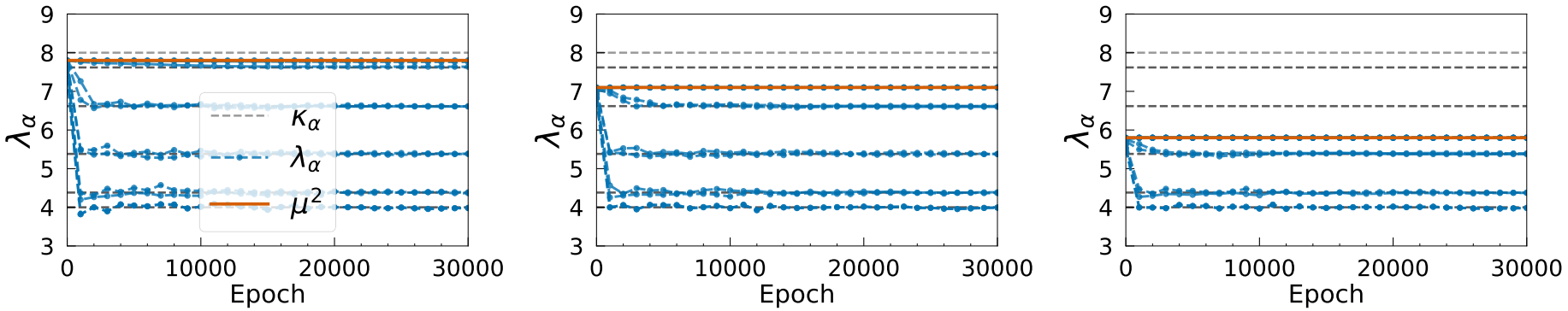
ξ_α: Singular value of W $\quad$ κ_α: Eigenvalue of K_{target}

There are two fixed points of ξ_α

$$\sigma_h^2 \xi_\alpha^2 = \begin{cases} \mu^2 - \kappa_\alpha & \cdots \quad \mu^2 > \kappa_\alpha \\ 0 & \cdots \quad \mu^2 \leq \kappa_\alpha \end{cases} \Rightarrow \lambda_\alpha = \begin{cases} \mu^2 - \sigma_h^2 \xi_\alpha^2 \sim \kappa_\alpha \\ \mu^2 \end{cases} \tag{11}$$

The fixed point equation indicates that the singular value ξ_α of the coupling matrix W dies off, if the corresponding eigenmode κ_α of the target kernel is larger than the RBM mass parameter μ^2.

Numerical results: Scalar field theory



(a) $\mu^2 = 7.8$ **(b)** $\mu^2 = 7.1$ **(c)** $\mu^2 = 5.8$

λ_α is a eigenvalue of the RBM kernel K.

$$\lambda_\alpha = \mu^2 \mathbb{1} - \sigma_h^2 \xi_\alpha^2 \tag{12}$$

The eigenvalues of the RBM kernel is trained to match the eigenvalues of the target kernel K_ϕ, but the high momentum modes larger than the RBM mass parameter are cut off.

UV regularization by reducing degrees of freedom

In case of $N_h < N_v$, the number of degrees of freedom of the RBM is restricted by the number of hidden nodes.

$$\mathrm{rank}(WW^T) \leq \min(N_v, N_h) \quad \Rightarrow \quad \mathrm{rank}(K) \leq \min(N_v, N_h) \tag{13}$$

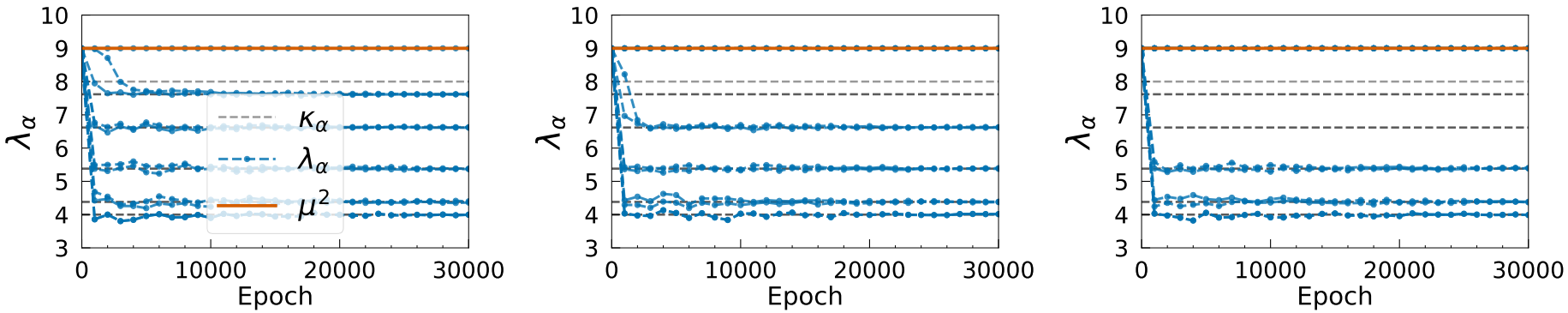
$$\mathrm{eig}(WW^T) = (\underbrace{\xi_1^2, \xi_2^2, \cdots, \xi_{N_h}^2}_{N_h}, \underbrace{0, \cdots, 0}_{N_v - N_h}) \tag{14}$$

Then $N_v - N_h$ number of modes of the RBM kernel are fixed to the RBM mass parameter μ^2 and the model possesses N_h degrees of fredom.

$$\lambda_\alpha = \begin{cases} \mu^2 - \sigma_h^2 \xi_\alpha^2 & \cdots \quad \alpha \leq N_h \\ \mu^2 & \cdots \quad \alpha > N_h \end{cases} \tag{15}$$

By removing the degrees of freedom in the RBM, we are keeping N_h number of lowest modes of the target kernel.

Numerical results: Scalar field theory



(a) $N_h = 9$ **(b)** $N_h = 7$ **(c)** $N_h = 5$

Figure: High momentum cut-off by removing degrees of freedom

The N_h number of modes are trained to match the eigenvalues of the target kernel from the lowest value.

Numerical results: MNIST data

Eigenvalue spectrum of the correlation matrix $\langle \phi_i \phi_j \rangle_{\mathrm{MNIST}}$ contains modes that are very close to 0.

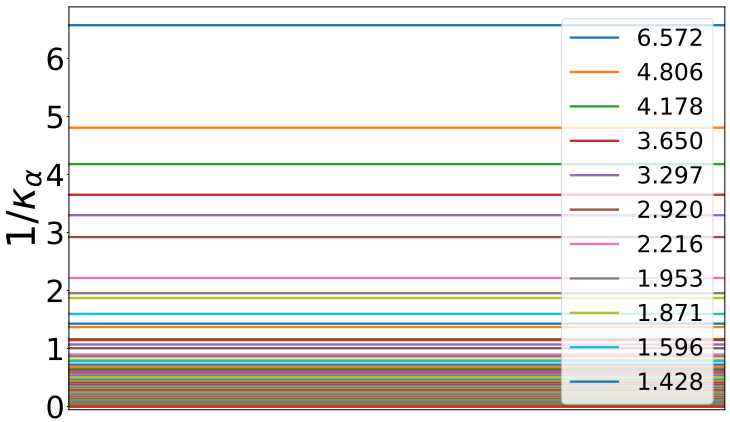


Figure: Eigenvalue spectrum of the MNIST correlation matrix $\langle \phi_i \phi_j \rangle_{\mathrm{MNIST}}$.

$$K_{\mathrm{MNIST}} = \langle \phi_i \phi_j \rangle^{-1}_{\mathrm{MNIST}} \qquad (16)$$

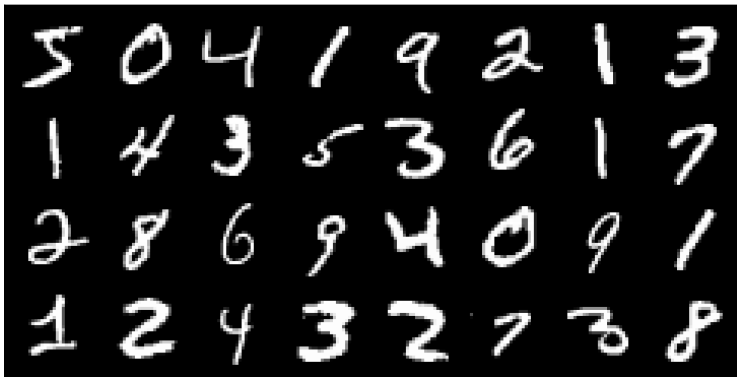
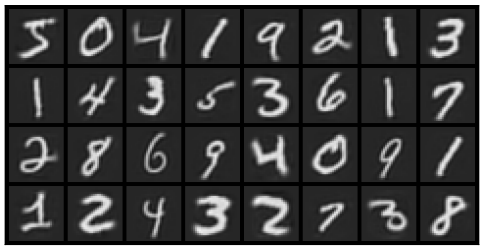


Figure: The MNIST dataset.

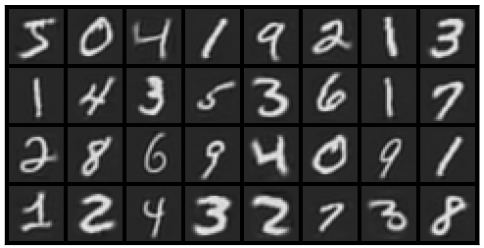
- The correlation matrix of the dataset contains 784 eigenvalues and some of the eigenvalues are very small.
- This indicates that the MNIST dataset is a UV divergent dataset.

Regeneration quality: MNIST

As more modes are cut off, the regeneration quality of the RBM decreases.



(a) $N_h = 784$

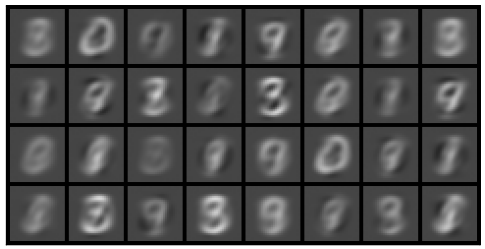


(b) $N_h = 225$

(c) $N_h = 64$

(d) $N_h = 36$

(e) $N_h = 16$



(f) $N_h = 4$

Figure: Quality of the regenerated image with different numbers of hidden nodes.

Langevin dynamics and Fokker Planck equation

In continuous time limit $\eta \to 0$, we can describe the training dynamics of the machine learning using Langevin equation.

$$\partial_t \xi^2 = \partial_{\xi^2}\mathcal{L} + r, \qquad \partial_{\xi^2}\mathcal{L} = \left[\frac{1}{\kappa} - \frac{1}{\mu^2 - \xi^2}\right]\xi^2 \tag{17}$$

where r is a Gaussian random noise. The corresponding Fokker-Planck equation is

$$\partial_t P(\xi^2, t) = \partial_{\xi^2}\left[\eta\partial_{\xi^2} - \partial_{\xi^2}\mathcal{L}\right] P(\xi^2, t) \tag{18}$$

Solving the equation, we can obtain leading order behavior of the relaxation time.

$$P(\xi^2, t) \propto c_0|0\rangle + \sum_{n>0} c_n|n\rangle e^{-\lambda_n t} \quad \Rightarrow \quad \lambda_1 \sim \frac{\mu^2 - \kappa}{\kappa^2} \tag{19}$$

Relaxation time and learning rate

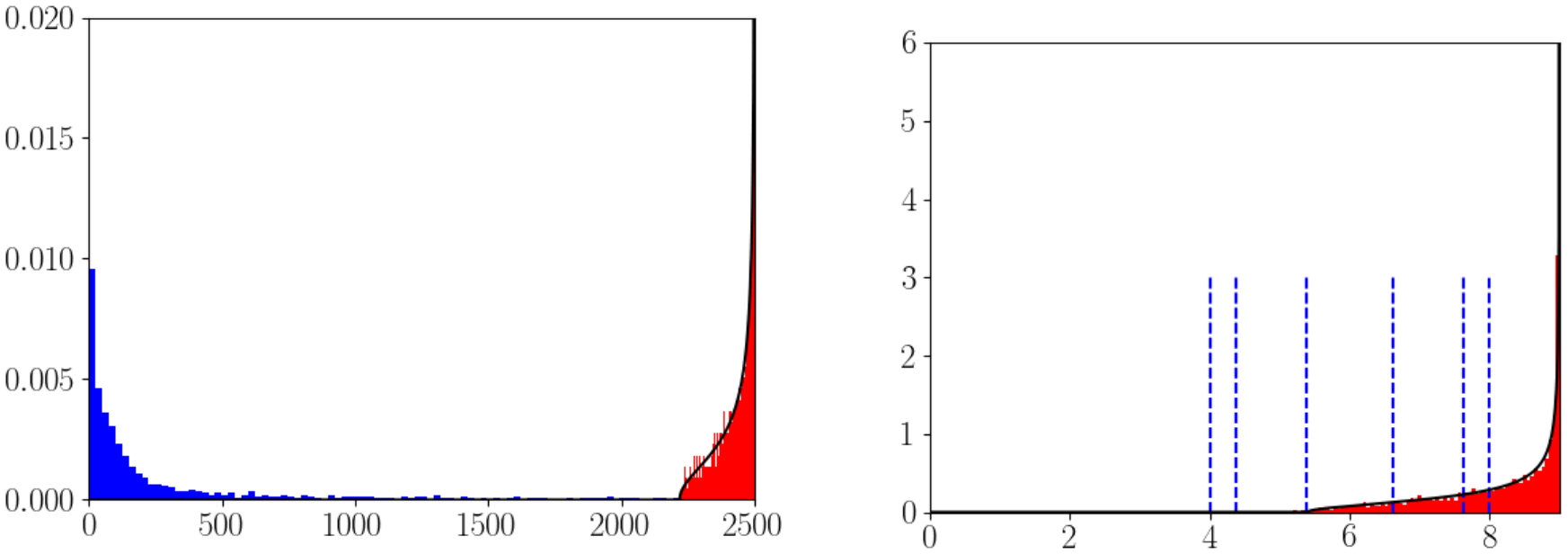


Figure: The relaxation of ξ^2 towards the equilibrium distribution. (Left) MNIST data (Right) Scalar field

Phase transitions in the Machine learning models

- Ferromagnetic phase transition in Binary models

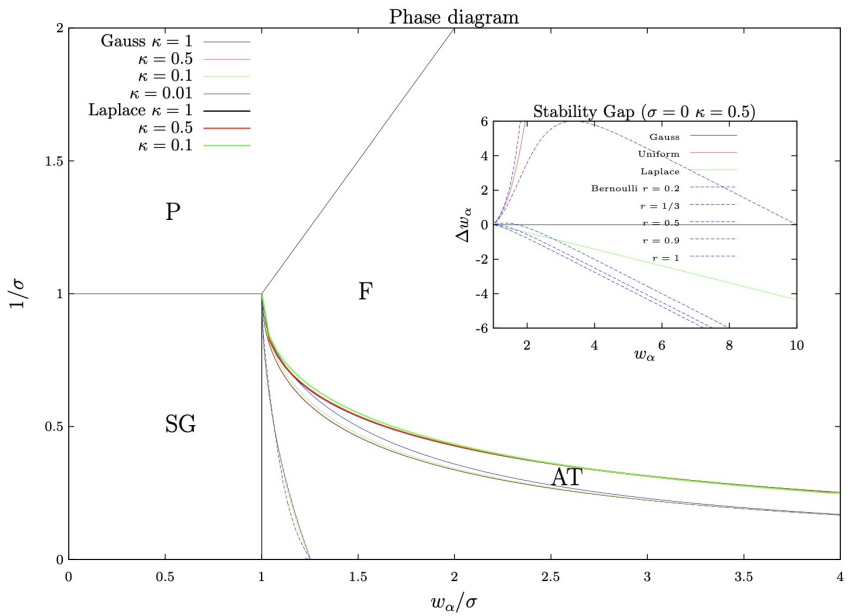


Figure: Phase diagram of Binary RBMs, Decelle et al. (2018)

- Grokking phase transition

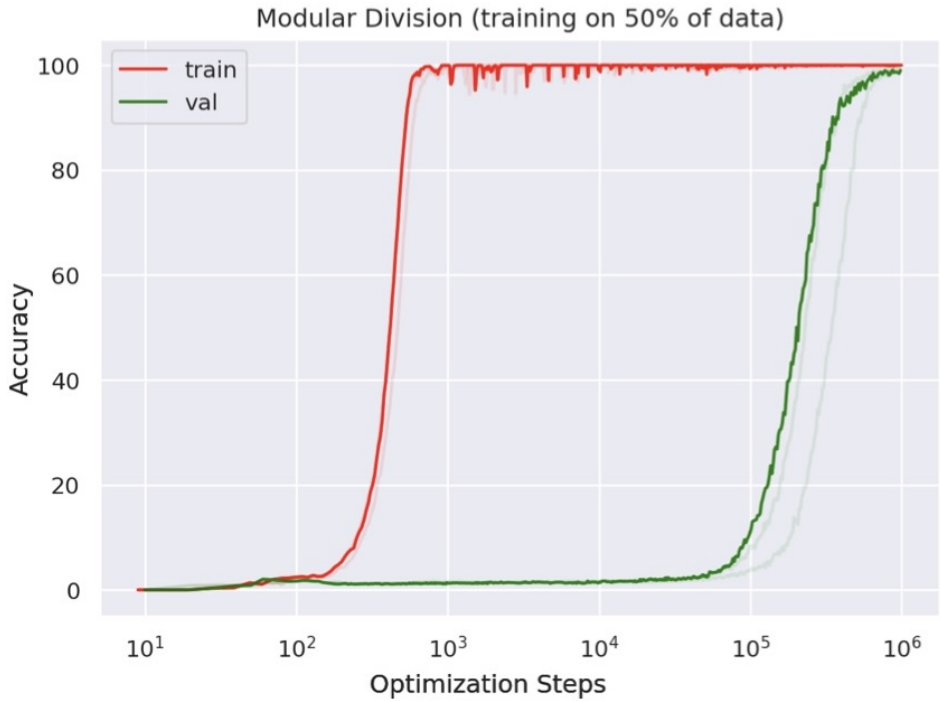


Figure: Grokking transition in overparameterized neural networks, Power et al. (2022)

Conclusion

- Explaining the dynamics of the machine learning models using the language of physics is beneficial to both ML and physics.
- Using tools of physics, we can solve explanability and scalability of the machine learning models.
- Machine learning models contain rich physical structures such as non-equilibrium stochastic process and phase transitions.
- It is multi-disciplinary research area, so any collaboration is welcome!

Thank you!

Gibbs sampling and information retrieval

Step 1 A sample from the target distribution is passed to the visible layer.

Step 2 The value of the hidden layer is sampled from the conditional probability distribution with the given value of the visible layer.

$$p(h|\phi) = \frac{p(\phi, h)}{\int \mathcal{D}h\, p(\phi, h)} = \frac{q_h e^{\phi^T W h}}{\int \mathcal{D}h\, q_h e^{\phi^T W h}}$$

Step 3 The value of visible layer is sampled from the conditional probability distribution with the given value of the hidden layer.

$$p(\phi|h) = \frac{p(\phi, h)}{\int \mathcal{D}\phi\, p(\phi, h)} = \frac{q_v e^{\phi^T W h}}{\int \mathcal{D}\phi\, q_v e^{\phi^T W h}}$$

Step 4 The samples retrieved from the model follows the probability distribution of the visible layer.

$$p(\phi) = \int \mathcal{D}h\, p(\phi, h) \tag{20}$$

Parameterization of the Gaussian RBM

Gaussian RBM is a RBM with Gaussian distribution as a prior for both visible and hidden layer.

$$q_v(\phi) = \exp\left(-\frac{\mu^2}{2}\phi^T\phi\right), \quad q_h(h) = \exp\left(-\frac{1}{2\sigma_h^2}(h-\eta)^T(h-\eta)\right) \tag{21}$$

With Gaussian prior of above parameterization, the joint probability distribution of the model is given by

$$p(\phi, h) = \frac{1}{Z}\exp\left(\phi^T W h - \frac{\mu^2}{2}\phi^T\phi - \frac{1}{2\sigma_h^2}(h-\eta)^T(h-\eta)\right) \tag{22}$$

The conditional probability is given by

$$p(\phi|h) = \prod_i \mathcal{N}\left(\phi_i; \frac{1}{\mu^2}\sum_a W_{ia}h_a, \frac{1}{\mu^2}\right), \quad p(h|\phi) = \prod_a \mathcal{N}\left(h_a; \eta_a + \sigma_h^2\sum_i \phi_i W_{ia}, \sigma_h^2\right) \tag{23}$$

Gaussian RBM as a Scalar Field RBM

Integrating out the hidden layer, we obtain the probability distribution of the visible layer, which is a probability distribution of the retrieved samples.

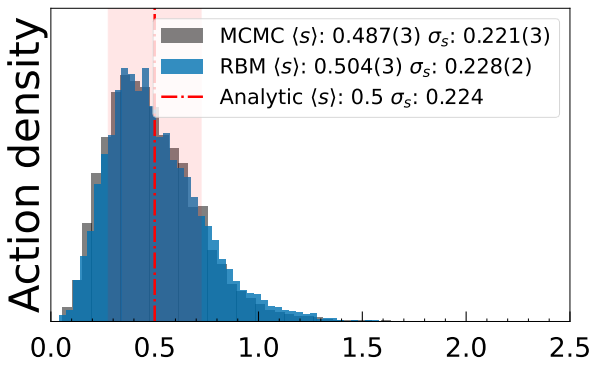
$$p(\phi) = \int \mathcal{D}h\, p(\phi, h) = \frac{1}{Z} \exp\left(-\frac{1}{2}\phi^T K \phi + \phi^T J\right) \tag{24}$$

with RBM kernel K, symmetry breaking external source J, and the partition function Z.

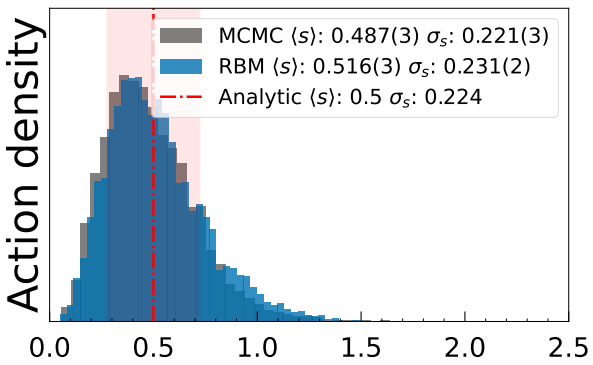
$$K \equiv \mu^2 \mathbb{1} - \sigma_h^2 W W^T, \quad J = W\eta, \quad Z = \int \mathcal{D}\phi \exp\left(-\frac{1}{2}\phi^T K \phi + \phi^T J\right) \tag{25}$$

From the form of the probability distribution of the visible layer, we can find that **the Gaussian RBM can be seen as a free scalar field theory with all to all coupling.**

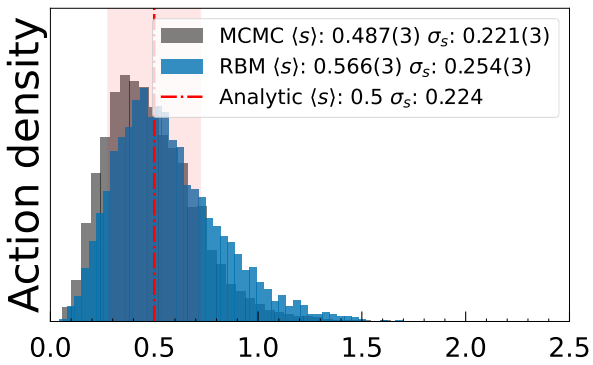
Regeneration quality: Scalar field theory



(a) $\mu^2 = 7.8$

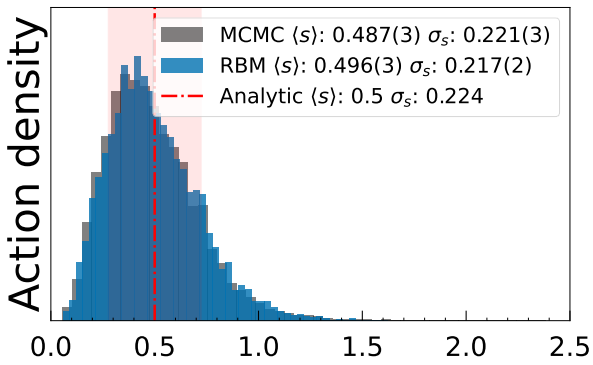


(b) $\mu^2 = 7.1$

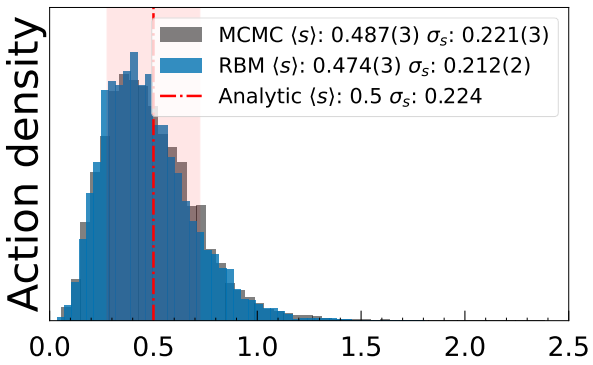


(c) $\mu^2 = 5.8$

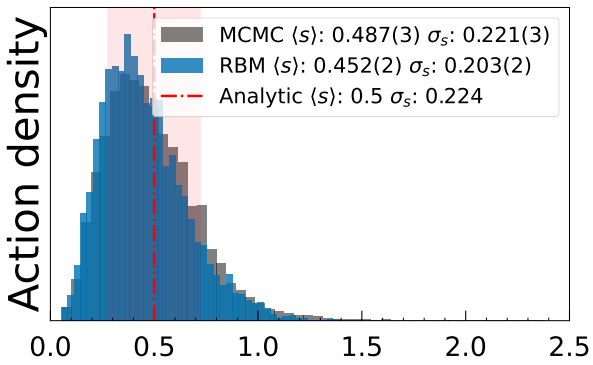
Figure: Regenerated action density with different μ^2.



(a) $N_h = 9$



(b) $N_h = 7$



(c) $N_h = 5$

Figure: Regenerated action density with different number of hidden nodes.

Unsupervised learning

- From the gradient equation,

$$\frac{\partial \mathcal{L}}{\partial W_{ia}} = \sigma_h^2 \sum_j \left[\langle \phi_i \phi_j \rangle_{\text{target}} - \langle \phi_i \phi_j \rangle_{\text{model}} \right] W_{aj} \tag{26}$$

$$= \sigma_h^2 \sum_j \left[K^{-1}_{\text{target},ij} - K^{-1}_{ij} \right] W_{aj} \tag{27}$$

- Predefined data is only needen when estimating the two-point correlation function of the target distribution.
- This can be avoided if the two-point correlation function of the target distribution is analyticaly obtainable.
- For example, for 1D free scalar field theory with periodic boundary condition,

$$\langle \phi_i \phi_j \rangle_{\text{target}} = K^{-1}_{\phi,ij} \tag{28}$$

with,

$$K_{\phi,ij} = (m^2 + 2)\delta_{ij} - \delta_{i,j+1} - \delta_{i,j-1} \tag{29}$$

Rotational part of the gradient

The orthogonal matrix that diagonalizes the target kernel and the model kernel is not necessarily alligned (or orthogonal) to each other before the training.

$$K_\phi = O_\phi D_\phi O_\phi^T, \quad K = U D_K U^T \tag{30}$$

with

$$O_\phi O_\phi^T = \mathbb{1}, \quad U U^T = \mathbb{1} \tag{31}$$

are the orthogonal matrices that diagonalize the target kernel and the model kernel. The full gradeint equation considering the rotation is,

$$\sigma_h^2 \left[K_\phi^{-1} - K^{-1} \right] = U \sigma_h^2 \left[U^T O_\phi D_\phi^{-1} O_\phi^T - D_K^{-1} \right] \Xi V^T \tag{32}$$

where Ξ is a rectangular singular value matrix and V is the right singular matrix of W.

$$W = U \Xi V^T \tag{33}$$

$N_v = 2$ $N_h = 1$ case

- In case of $N_v = 2$ and $N_h = 1$, the training equation is directly traiceable.
- The orthogonal matrix O_ϕ and U are 2×2 rotation matrices with angle θ_ϕ and θ_U respectively.
- The combination $O_\phi^T U$ is a rotation over an angle $\Delta\theta = \theta_U - \theta_\phi$.
- Training equation for the angle $\Delta\theta$ is written as,

$$\frac{d\Delta\theta}{dt} = \sigma_h^2 \left(\frac{1}{\kappa_2} - \frac{1}{\kappa_1} \right) \cos\Delta\theta \sin\Delta\theta \tag{34}$$

which gives two limits of $\Delta\theta$,

$$\kappa_2 > \kappa_1 \quad \Leftrightarrow \quad \tilde{\rho} < 0 \quad \Leftrightarrow \quad \lim_{t\to\infty} \Delta\theta(t) = 0, \quad \Leftrightarrow \quad \xi_1 \to \kappa_1,\ \xi_2 = 0 \tag{35}$$

$$\kappa_2 < \kappa_1 \quad \Leftrightarrow \quad \tilde{\rho} > 0 \quad \Leftrightarrow \quad \lim_{t\to\infty} \Delta\theta(t) = \frac{\pi}{2} \quad \Leftrightarrow \quad \xi_1 \to \kappa_2,\ \xi_2 = 0. \tag{36}$$