Replacing Markov chain Monte Carlo with generative flow neural networks

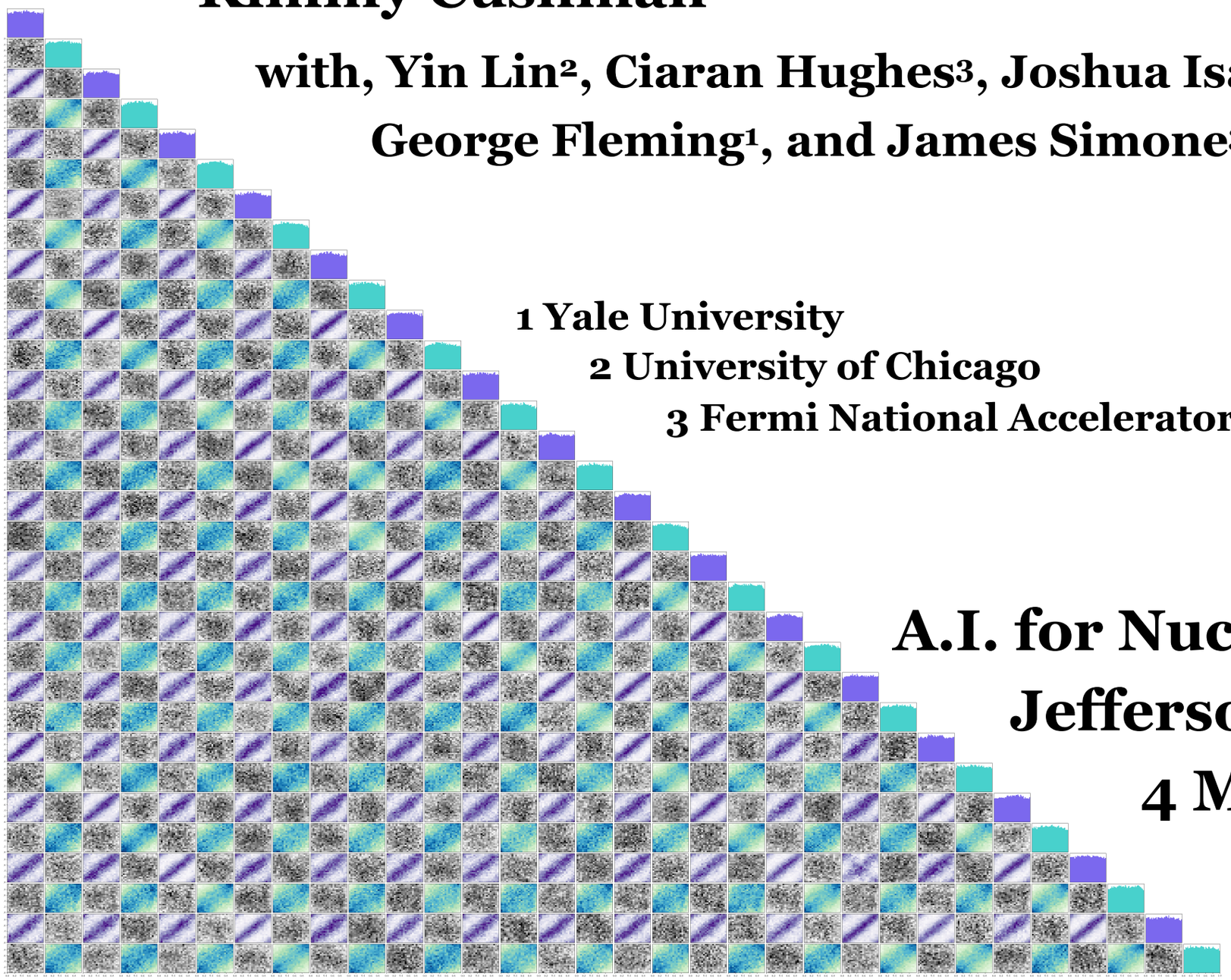
Kimmy Cushman[1]

with, Yin Lin[2], Ciaran Hughes[3], Joshua Isaacson[3], George Fleming[1], and James Simone[3]

1 Yale University

2 University of Chicago

3 Fermi National Accelerator Laboratory



A.I. for Nuclear Physics

Jefferson Lab

4 March 2020

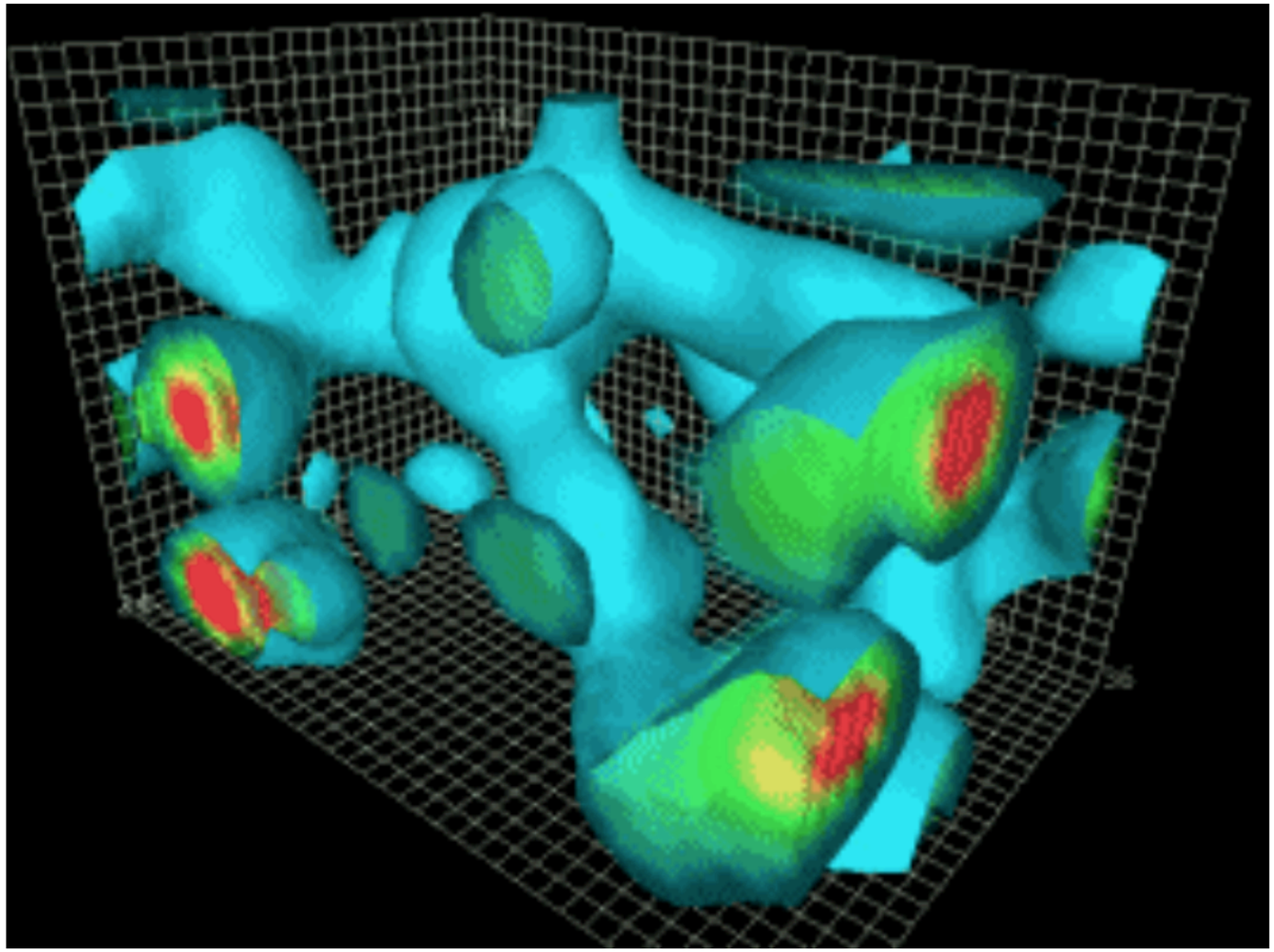
Outline

- **Lattice QCD**
- **MCMC and neural networks**
- **Neural networks for O(3) spin model**
- **How to believe a neural network**
- **Future work**

Lattice QCD

Lattice QCD… what is it?

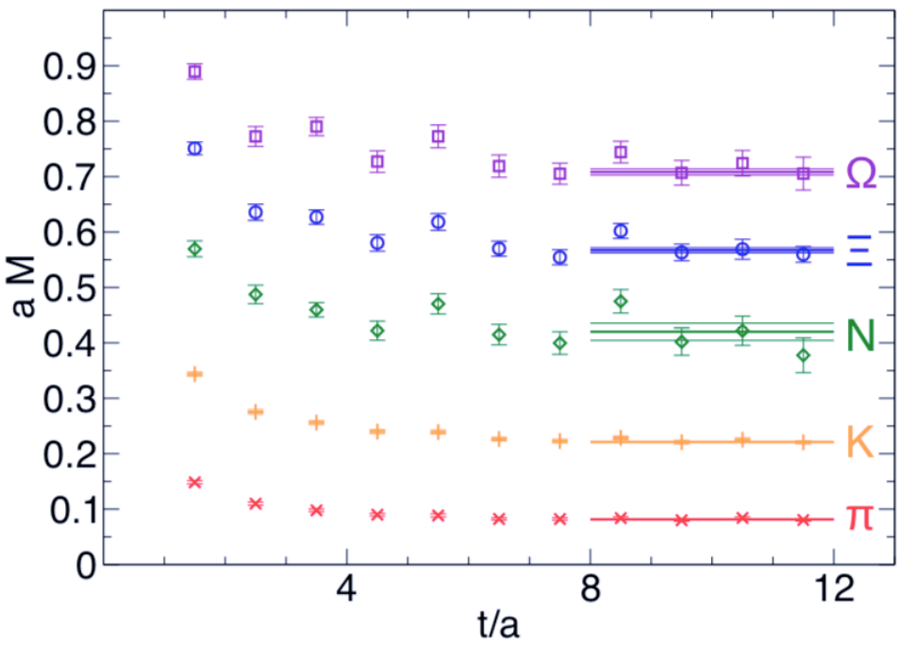
- **Gauge field simulation from first principles**
- **Discretized space-time**
- **Finite volume with periodic conditions**
- **Volume few times size of proton**



http://www.physics.adelaide.edu.au/cssm/lattice/

Motivation for QCD

Mass comes from QCD dynamics



Ab-initio Determination of Light Hadron Masses

S. Dürr[1], Z. Fodor[1,2,3], J. Frison[4], C. Hoelbling[2,3,4],
R. Hoffmann[2], S. D. Katz[2,3], S. Krieg[2], T. Kurth[2],
L. Lellouch[4], T. Lippert[2,5], K.K. Szabo[2], G. Vulvert[4]

[1]NIC, DESY Zeuthen, D-15738 Zeuthen and FZ Jülich, D-52425 Jülich, Germany.
[2]Bergische Universität Wuppertal, Gaussstr. 20, D-42119 Wuppertal, Germany.
[3]Institute for Theoretical Physics, Eötvös University, H-1117 Budapest, Hungary.
[4]Centre de Physique Théorique*, Case 907, Campus de Luminy, F-13288 Marseille Cedex 9, France.
[5]Jülich Supercomputing Centre, FZ Jülich, D-52425 Jülich, Germany.

Budapest-Marseille-Wuppertal Collaboration

Spectroscopy, scattering, PDFs,

CKM matrix elements BSM spectroscopy,

Phase transitions

Image credit: Carlos Jones/ORNL

Importance sampling

$$\langle 0 | \pi(t)\ \pi^{\dagger}(0) | 0 \rangle = C_{\pi}(t) = \langle \mathcal{O} \rangle$$

$$\langle \mathcal{O} \rangle = \frac{\int_D \mathcal{O}[A]\ \mathrm{e}^{iS[A]}}{\int_D \mathrm{e}^{iS[A]}}$$

Importance sampling

$$\langle 0 | \pi(t)\ \pi^{\dagger}(0) | 0 \rangle = C_{\pi}(t) = \langle \mathcal{O} \rangle$$

$$\langle \mathcal{O} \rangle = \frac{\int_D \mathcal{O}[A]\ e^{iS[A]}}{\int_D e^{iS[A]}} \longrightarrow \langle \mathcal{O} \rangle = \frac{1}{Z} \sum_i e^{-S_i} \mathcal{O}_i$$

Importance sampling

$$\langle 0 | \pi(t)\ \pi^{\dagger}(0) | 0 \rangle = C_{\pi}(t) = \langle \mathcal{O} \rangle$$

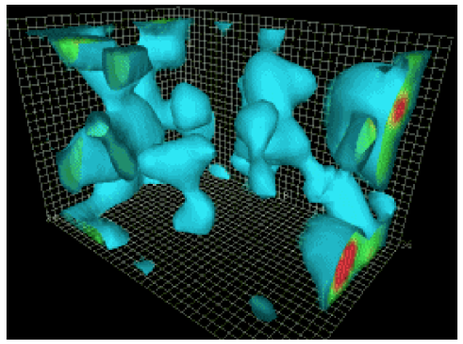
$$\langle \mathcal{O} \rangle = \frac{\int_D \mathcal{O}[A]\ e^{iS[A]}}{\int_D e^{iS[A]}} \longrightarrow \langle \mathcal{O} \rangle = \frac{1}{Z} \sum_i e^{-S_i} \mathcal{O}_i$$

Our only hope!

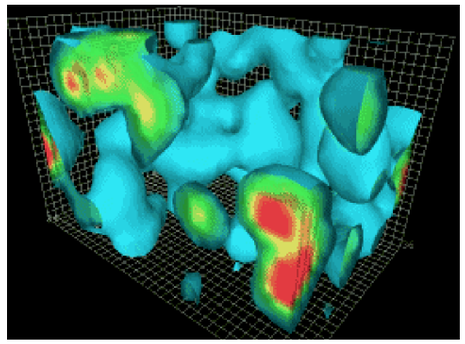
Importance sampling

$$\langle 0 | \pi(t)\ \pi^{\dagger}(0) | 0 \rangle = C_{\pi}(t) = \langle \mathcal{O} \rangle$$

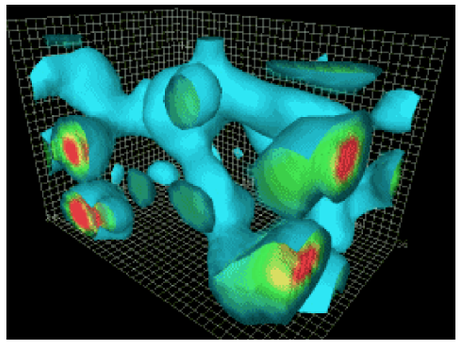
$$\langle \mathcal{O} \rangle = \frac{\int_D \mathcal{O}[A]\ e^{iS[A]}}{\int_D e^{iS[A]}} \longrightarrow \langle \mathcal{O} \rangle = \frac{1}{Z} \sum_i e^{-S_i} \mathcal{O}_i$$



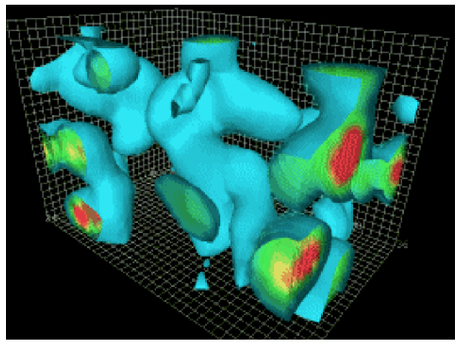
Configuration 1 $\mathcal{O}_1$



Configuration 2 $\mathcal{O}_2$



Configuration 3 $\mathcal{O}_3$



Configuration 4 $\mathcal{O}_4$

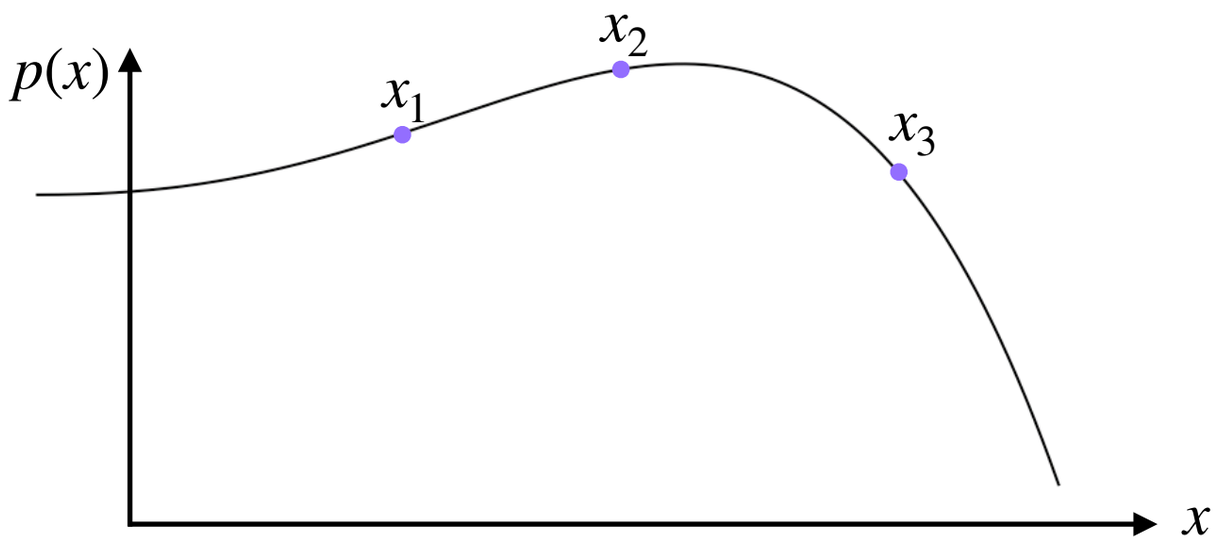
$$\langle \mathcal{O} \rangle = \frac{1}{N} \sum_i^N \mathcal{O}_i$$

MCMC and neural networks

Markov chain Monte Carlo

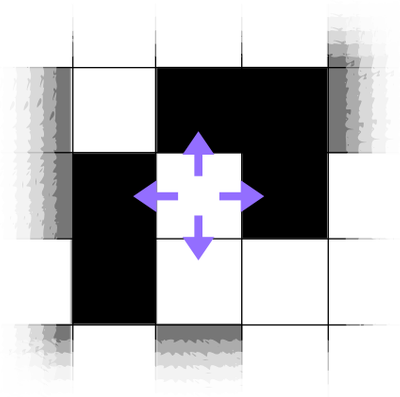
- **Propose possible configuration,** x
- **Propose steps in a chain** $x \to x'$ **accept with**

probability $p = \min\left[1, \dfrac{e^{-S(x')}}{e^{-S(x)}}\right]$



Markov chain Monte Carlo

Drawback: thermalization and autocorrelation



Ising model

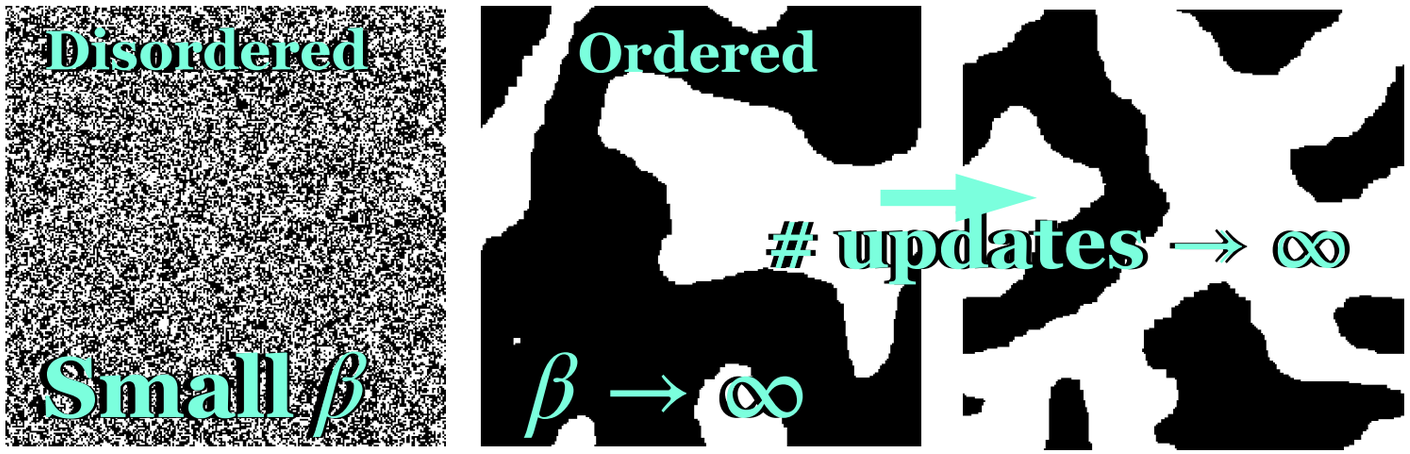
$$S = -\beta \sum_{\langle i,j \rangle} s_i s_j$$

Number *independent* configurations $\ll$ **Number *generated* configurations**

Markov chain Monte Carlo

Drawback: critical slowing down near critical points

Correlation length diverges $\longleftrightarrow$ **Autocorrelation time diverges**



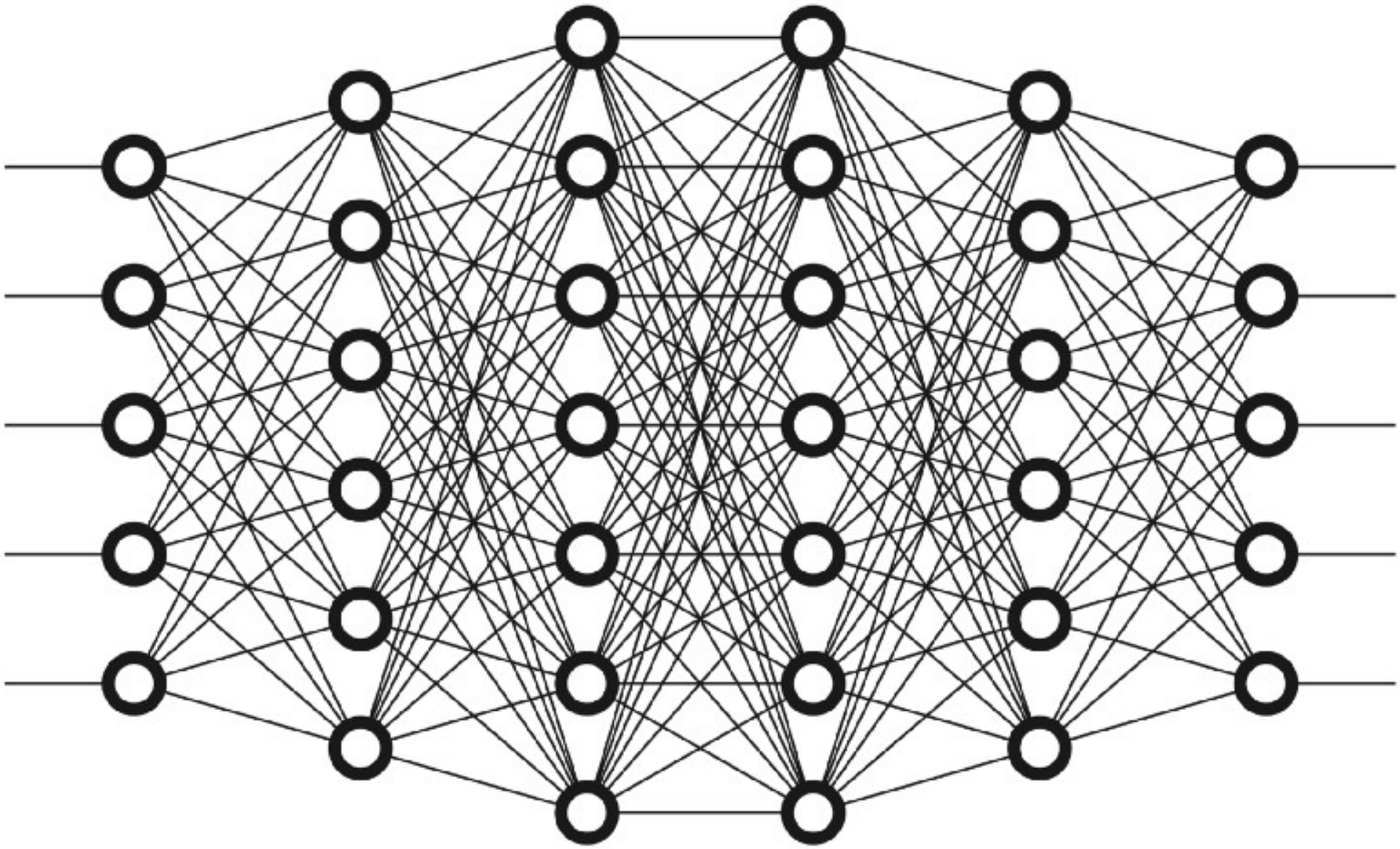
Ising model
ordered/disorder phases

QCD + dark matter
Confined phase/quark gluon plasma

BSM Higgs
Electroweak symmetry breaking transition

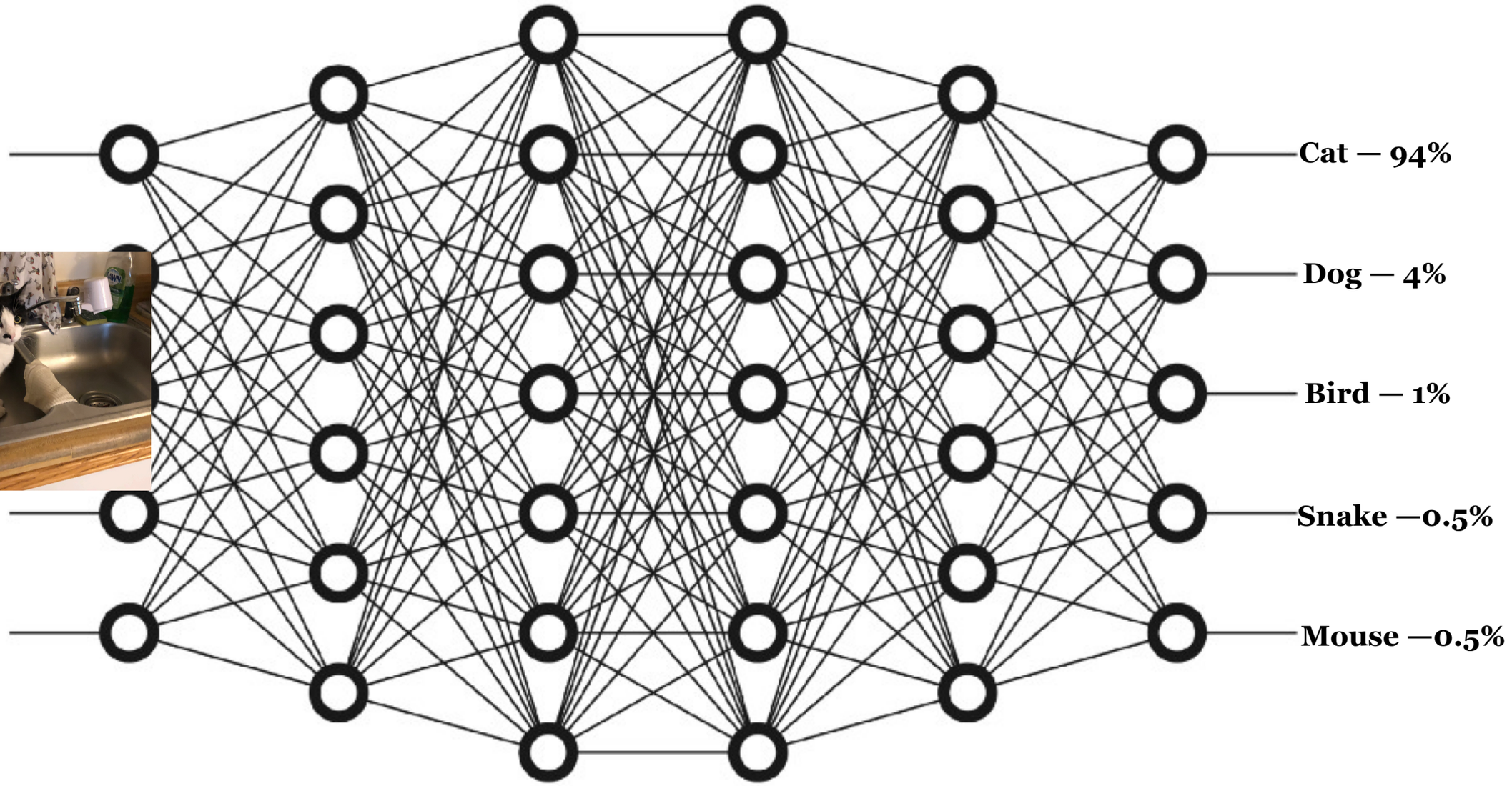
Deep neural networks

Fully connected network with 4 hidden layers



Deep neural networks

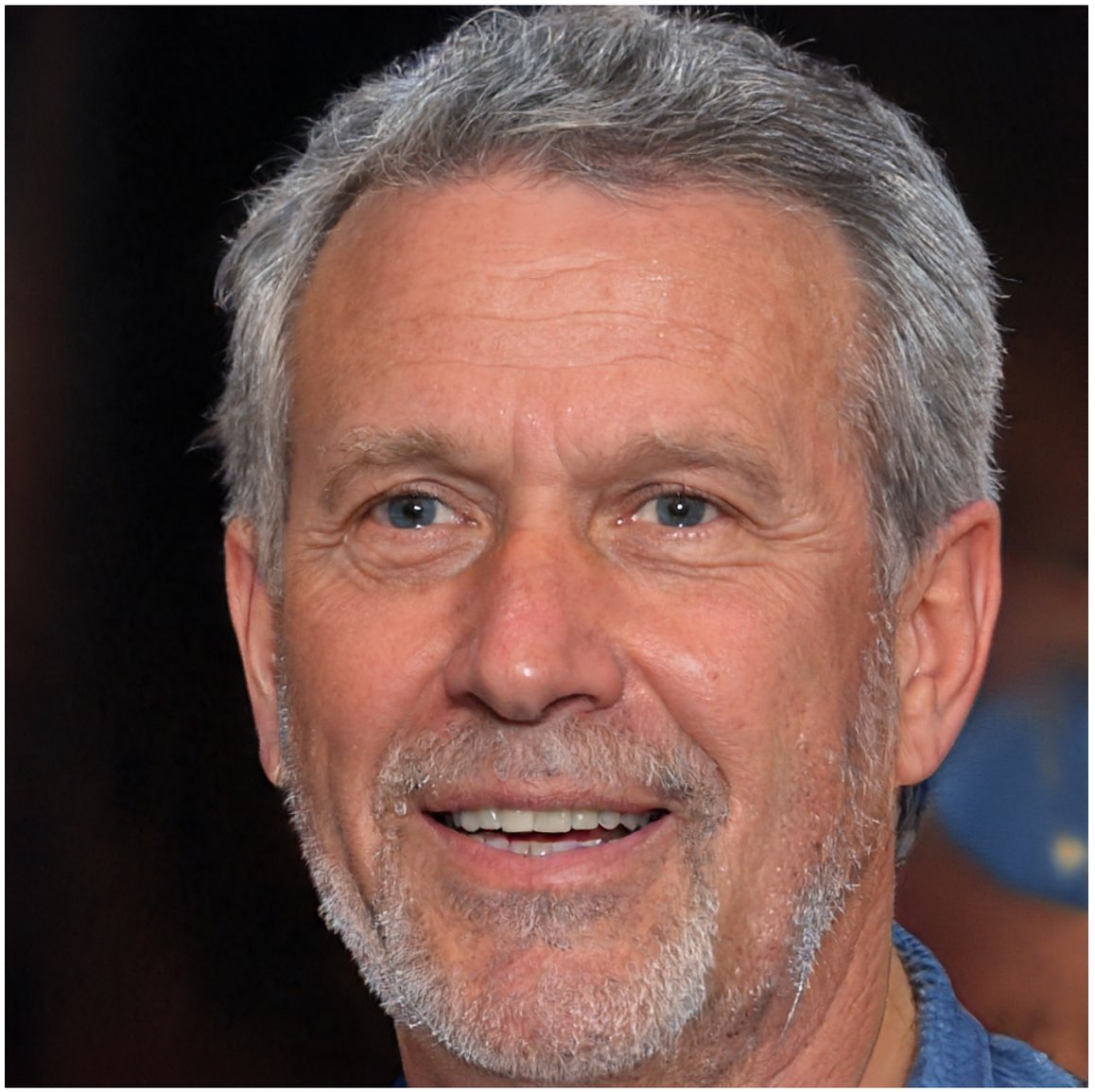
Fully connected network with 4 hidden layers



Generative Neural networks

https://www.thispersondoesnotexist.com

Generative Neural networks

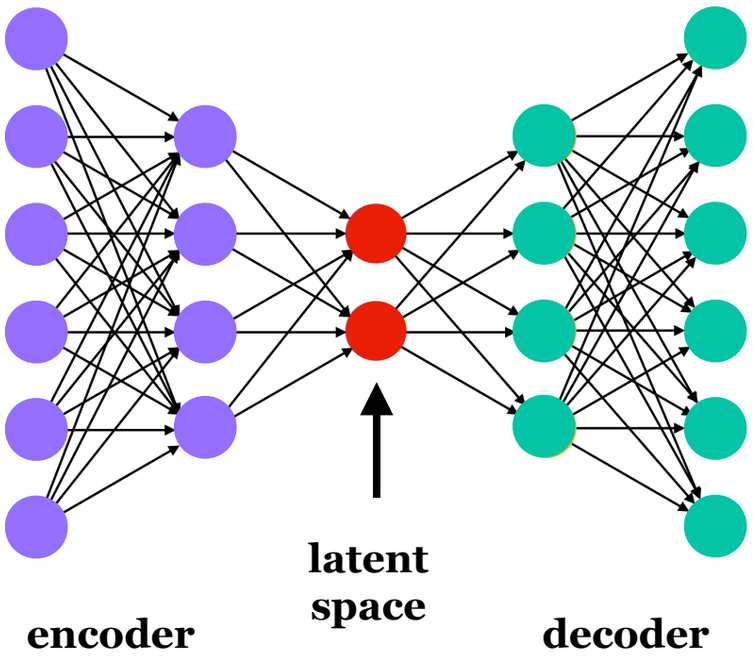


https://www.thispersondoesnotexist.com

Generative Neural networks

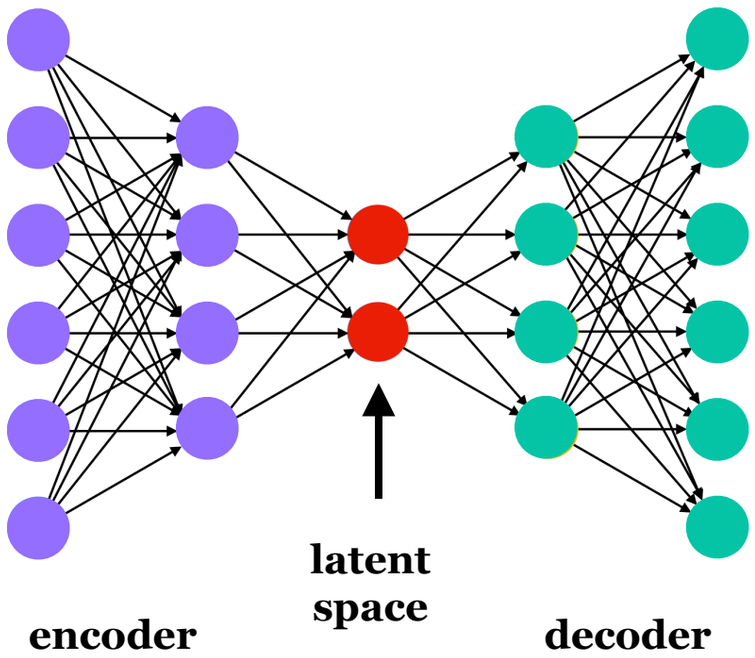
https://www.thispersondoesnotexist.com

Auto-encoders

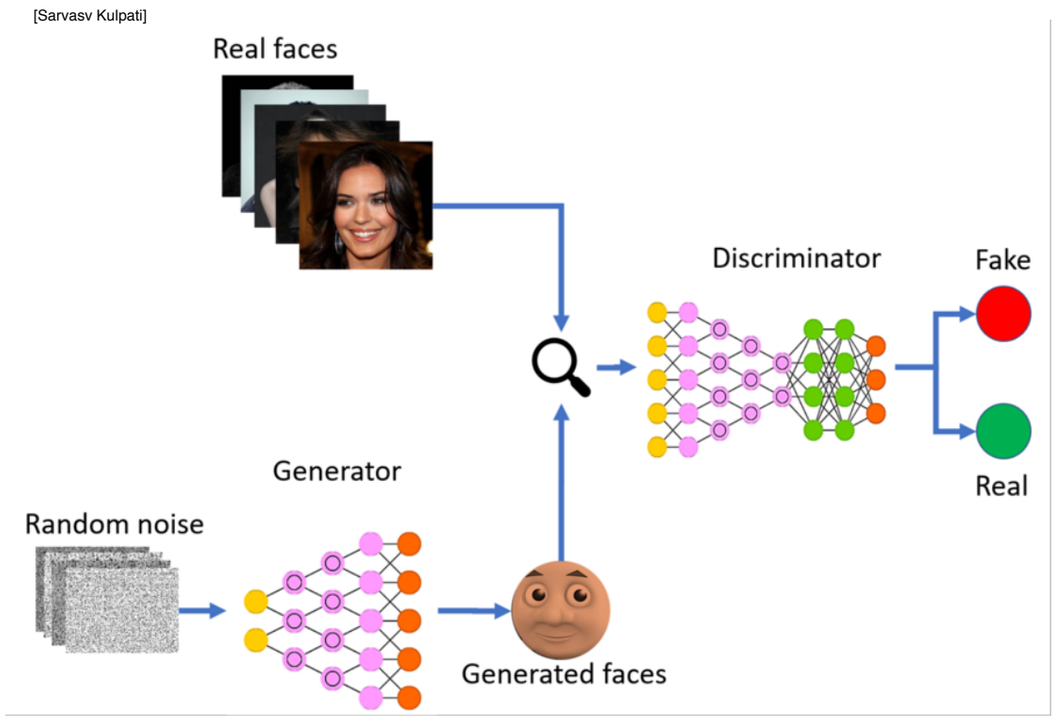


Generative Neural networks

Auto-encoders

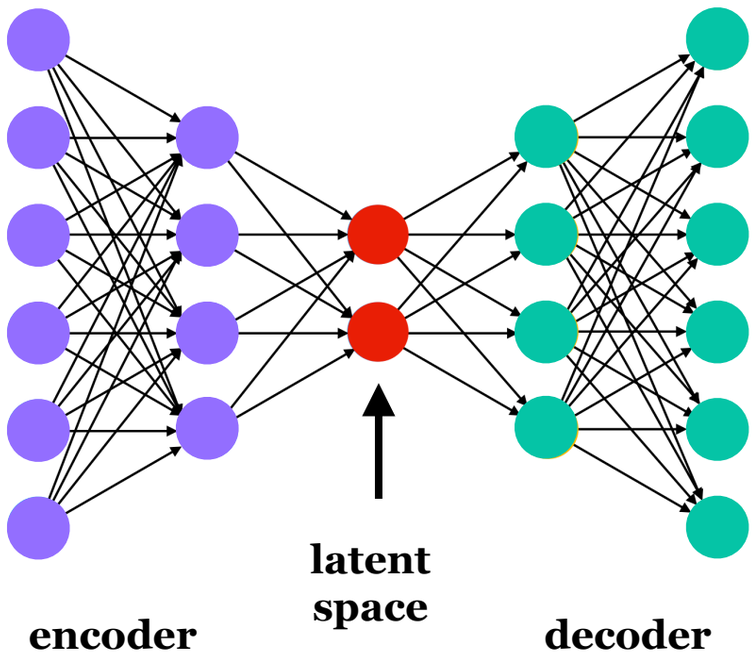


GANs

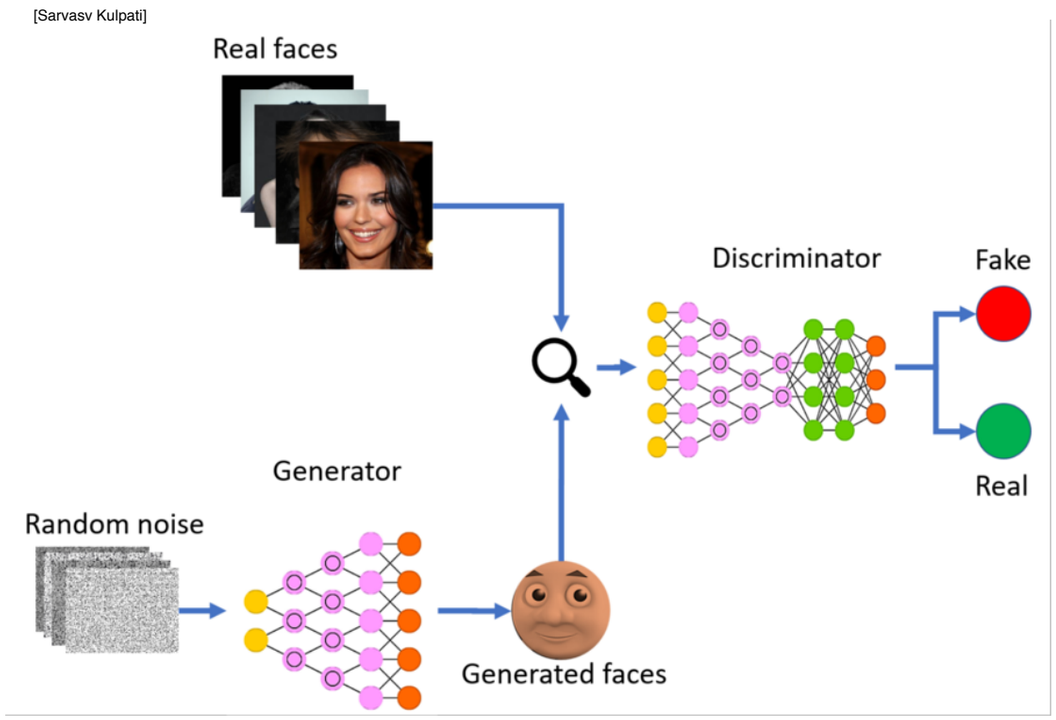


Generative Neural networks

Auto-encoders



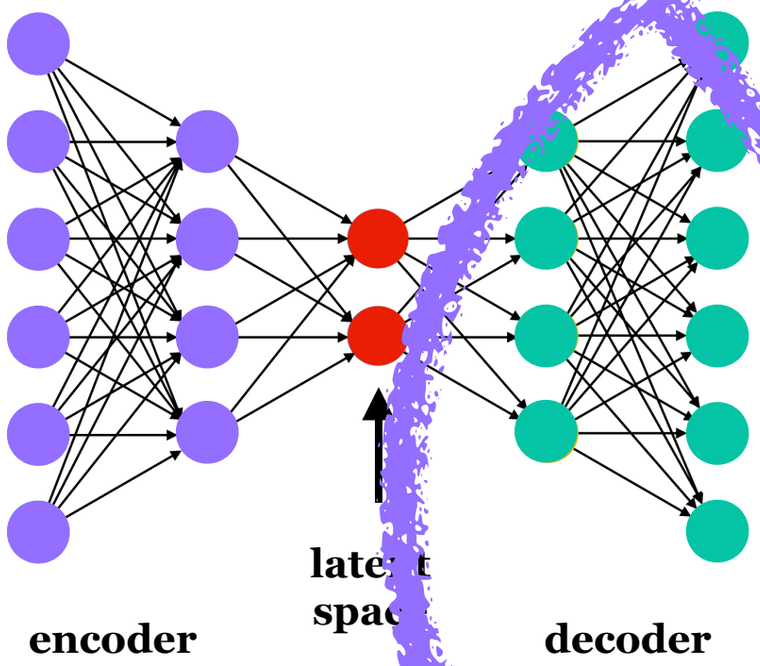
GANs



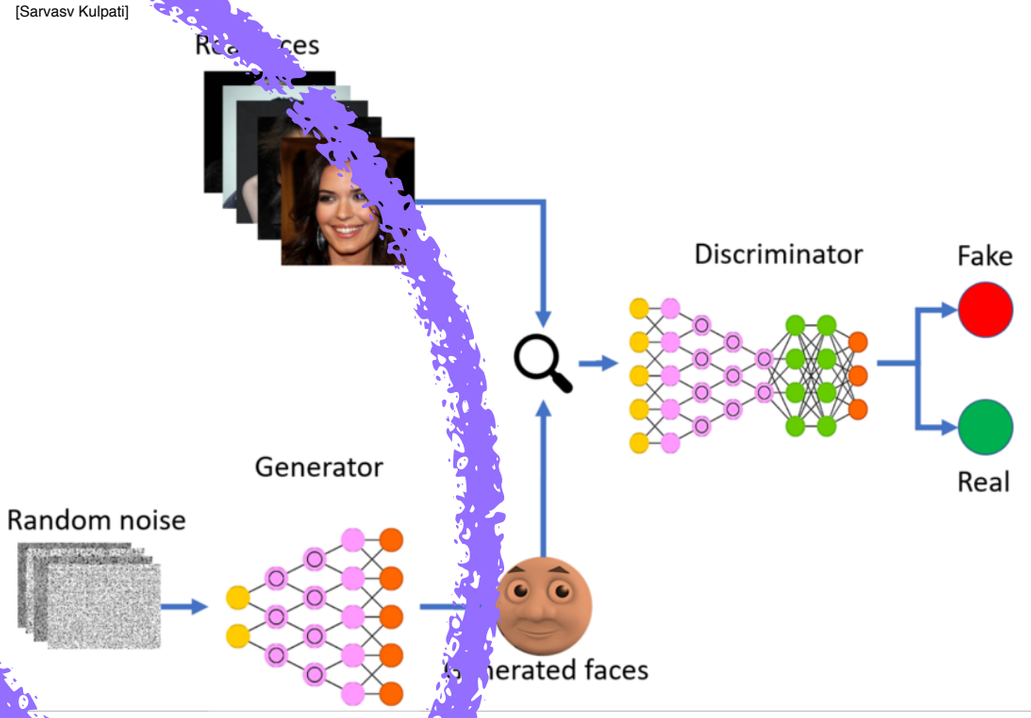
- Require A LOT of training data
- Mode collapse

Generative Neural networks

Auto-encoders



GANs

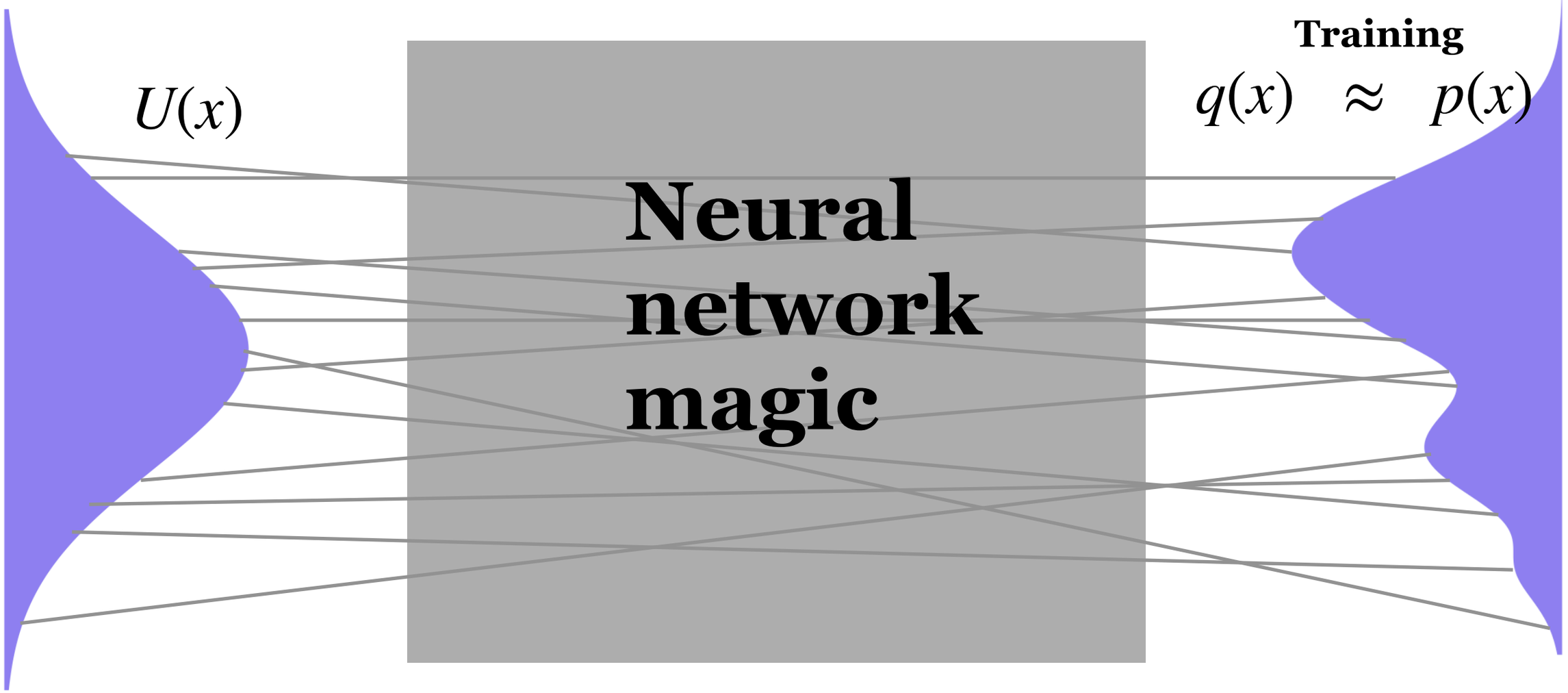


- Require A LOT of training data
- Mode collapse

Generative flow networks for O(3) spin model

Generative flow approach

Goal: **Train neural network to produce configurations by learning *bijection***



See also **Albergo, Kanwar, Shanahan 1904.12072**
Dinh, Sohl-Dickstein, Bengio 1605.08803
Muller, McWilliams, Rousselle, Gross, Novak, 1808.03856

Generative flow approach

Why a bijection?

For generation $U(x) \rightarrow q(x) \ \left(\approx p(x)\right)$

Why $q(x) \rightarrow U(x)$ **?**

Typical importance sampling

$$\langle \mathcal{O} \rangle = \frac{1}{Z} \int dx \ \mathcal{O}(x) \ \mathrm{e}^{-S(x)}$$

$$= \frac{1}{Z} \int dx \ \mathcal{O}(x) \ p(x)$$

$$\approx \frac{1}{N} \sum_i \ \mathcal{O}_i \big|_p$$

Generative flow approach

Why a bijection?

For generation $U(x) \to q(x) \ \left(\approx p(x)\right)$

Why $q(x) \to U(x)$ **?**

Typical importance sampling

$$\langle \mathcal{O} \rangle = \frac{1}{Z} \int dx \ \mathcal{O}(x) \ \mathrm{e}^{-S(x)}$$

$$= \frac{1}{Z} \int dx \ \mathcal{O}(x) \ p(x)$$

$$\approx \frac{1}{N} \sum_i \ \mathcal{O}_i \big|_p$$

Generative flow importance sampling

$$\langle \mathcal{O} \rangle = \frac{1}{Z} \int dx \ \mathcal{O}(x) \ p(x)$$

$$= \frac{1}{Z} \int dx \ \mathcal{O}(x) \ \frac{p(x)}{q(x)} \ q(x)$$

$$\approx \frac{\sum_i \mathcal{O}_i \frac{p_i}{q_i} \big|_q}{\sum_i \frac{p_i}{q_i} \big|_q}$$

Generative flow approach

Why a bijection?

For generation $U(x) \to q(x) \ \left(\approx p(x)\right)$

Why $q(x) \to U(x)$ **?**

Typical importance sampling

$$\langle \mathcal{O} \rangle = \frac{1}{Z} \int dx \ \mathcal{O}(x) \ \mathrm{e}^{-S(x)}$$

$$= \frac{1}{Z} \int dx \ \mathcal{O}(x) \ p(x)$$

$$\approx \frac{1}{N} \sum_i \ \mathcal{O}_i \big|_p$$

Sampled according to $p(x)$
Hard to sample from (MCMC)

Generative flow importance sampling

$$\langle \mathcal{O} \rangle = \frac{1}{Z} \int dx \ \mathcal{O}(x) \ p(x)$$

$$= \frac{1}{Z} \int dx \ \mathcal{O}(x) \ \frac{p(x)}{q(x)} \ q(x)$$

$$\approx \frac{\sum_i \mathcal{O}_i \frac{p_i}{q_i} \big|_q}{\sum_i \frac{p_i}{q_i} \big|_q}$$

Sampled according to $q(x)$
Easy to sample from ($U(x)$ **is easy)**

Neural network architecture

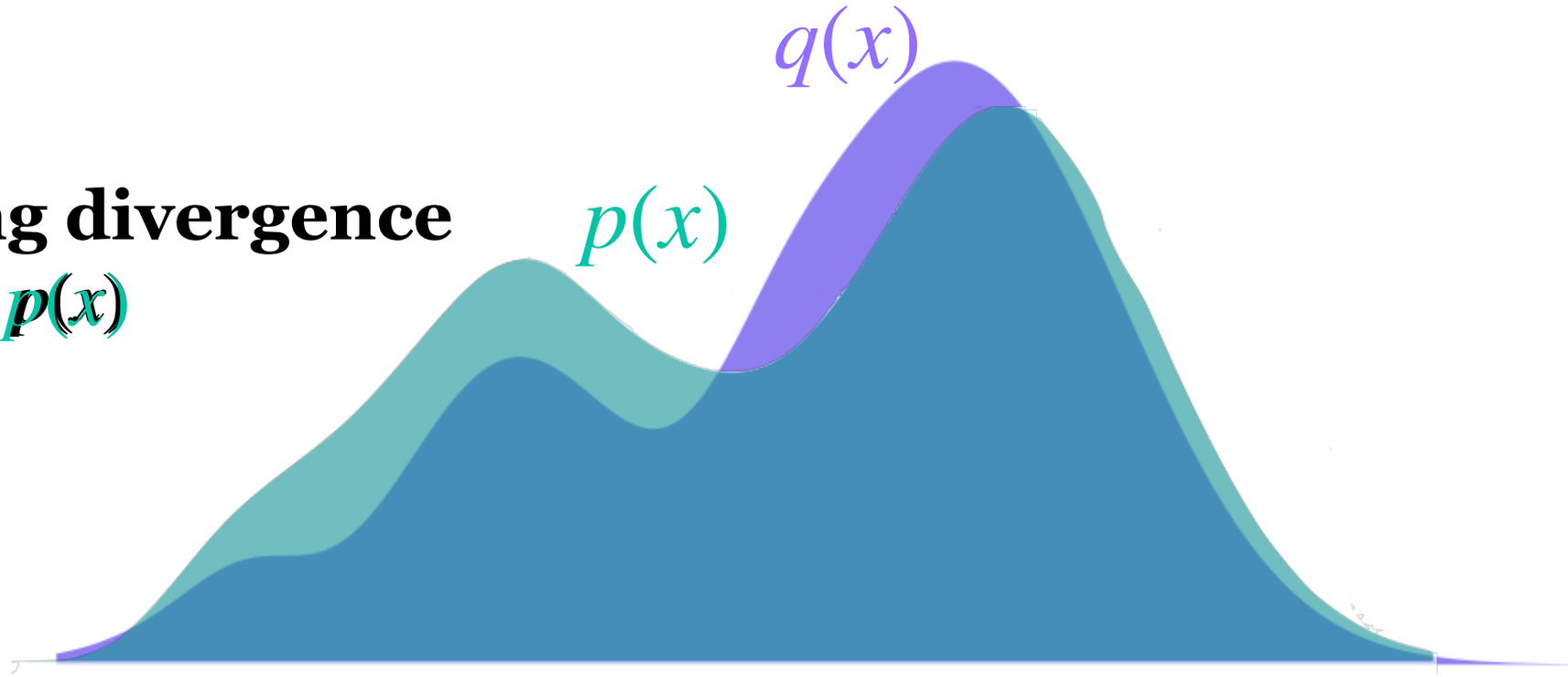
Composition of coupling layers

$$H = h_L \circ \cdots \circ h_2 \circ h_1 \qquad H(U(x)) = q(x)$$

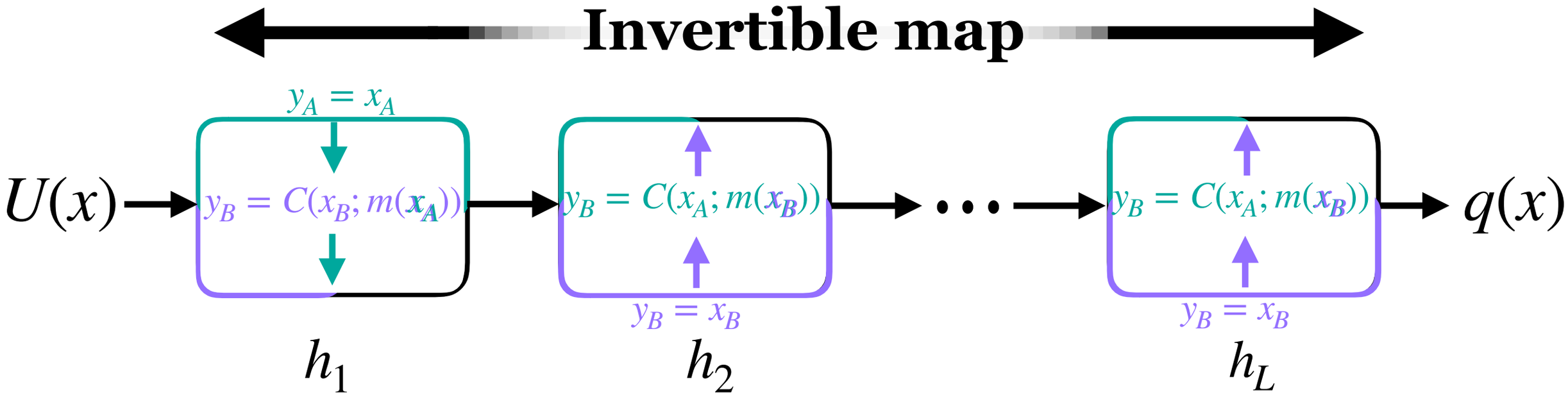
Learned Jacobian of variable transformation
$U \rightarrow q$

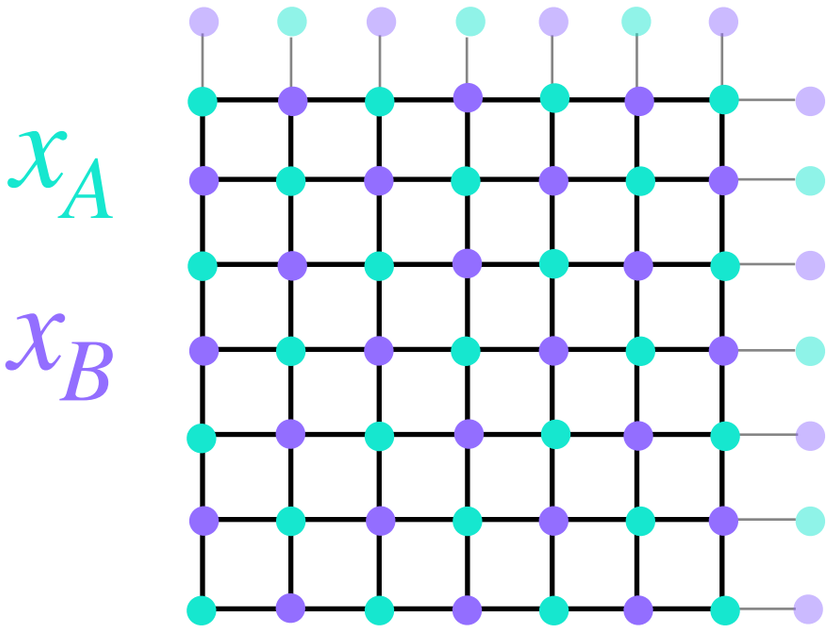
$$q(x_i) = U(x_i) \left| \frac{\partial H(x;\theta)}{\partial x_i} \right|^{-1}$$

Train by minimizing divergence between $q(x)$ and $p(x)$

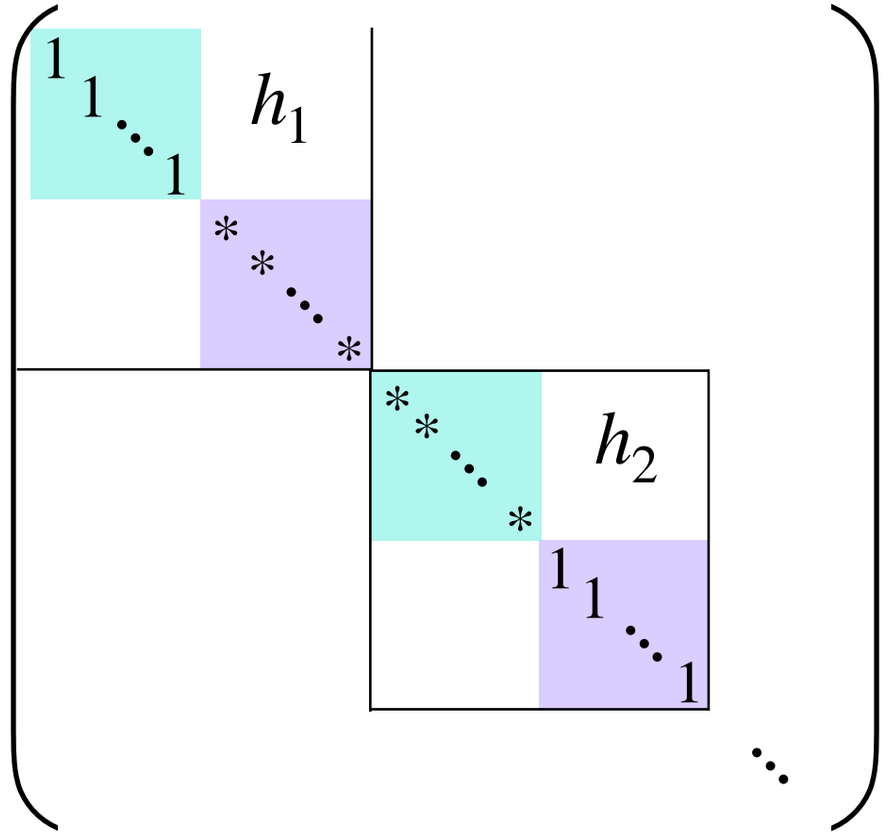


Neural network architecture





$$\frac{\partial H}{\partial x_i} =$$



Neural network architecture

Choose coupling layers function $C(x_A, m(x_B))$ *easily invertible*

Ex: $C(x_A; s, t) = x_A \odot \mathrm{e}^s + t$

$\vec{s}$ **and** $\vec{t}$ **learned parameters**

$$y_B = x_B$$

$$y_A = C\big(x_B; m(x_A)\big)$$

Neural network architecture

Choose coupling layers function $C(x_A, m(x_B))$ *easily invertible*

Ex: $C(x_A; s, t) = x_A \odot \mathrm{e}^s + t$

$\vec{s}$ **and** $\vec{t}$ **learned parameters**

$$y_B = x_B$$

$$y_A = C\left(x_B; m(x_A)\right)$$

$$\frac{\partial h_1}{\partial x_B} = 1 \quad \frac{\partial h_1}{\partial x_i^A} = \mathrm{e}^{s_i}$$

$$\frac{\partial H}{\partial x_i} = \begin{pmatrix} 1 & & & & & & & \\ & 1 & & & & & & \\ & & \ddots & & & & & \\ & & & 1 & & & & \\ & & & & \mathrm{e}^{s_1} & & & \\ & & & & & \mathrm{e}^{s_2} & & \\ & & & & & & \ddots & \\ & & & & & & & \mathrm{e}^{s_8} \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \ddots \end{pmatrix} \begin{matrix} h_1 & \\ & \\ & \\ & \\ & \\ & \\ & \\ & \\ & h_2 \\ & \\ & \\ & \\ & \\ & \\ & \\ & \\ & \ddots \end{matrix}$$

Neural network architecture

Choose coupling layers function $C(x_A, m(x_B))$ *easily invertible*

Ex: $C(x_A; s, t) = x_A \odot \mathrm{e}^{s} + t$

$\vec{s}$ and $\vec{t}$ learned parameters

$$y_B = x_B$$

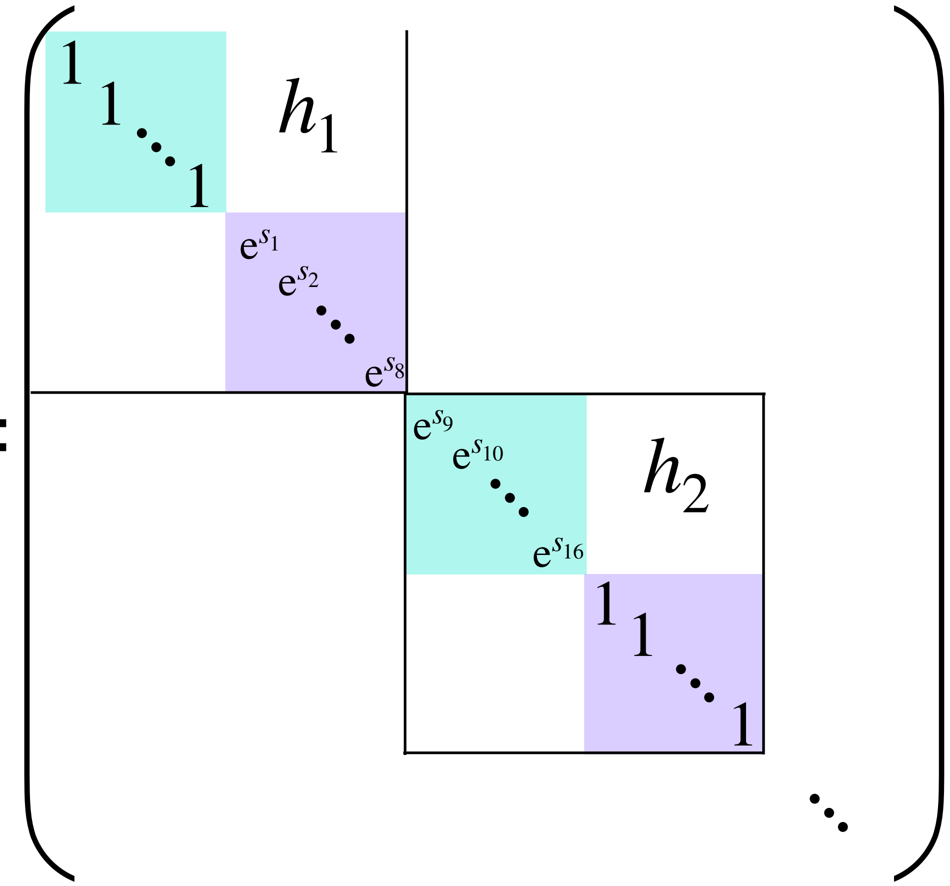
$$y_A = C\big(x_B; m(x_A)\big)$$

$$\frac{\partial h_1}{\partial x_B} = 1 \quad \frac{\partial h_1}{\partial x_i^A} = \mathrm{e}^{s_i}$$

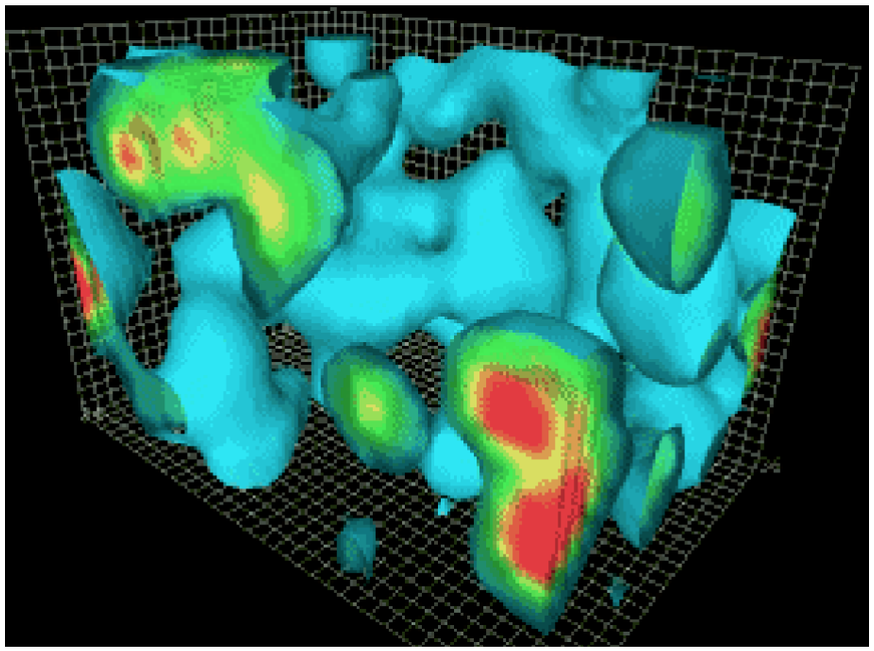
$$q(x_i) = U(x_i) \left| \frac{\partial H(x;\theta)}{\partial x_i} \right|^{-1}$$

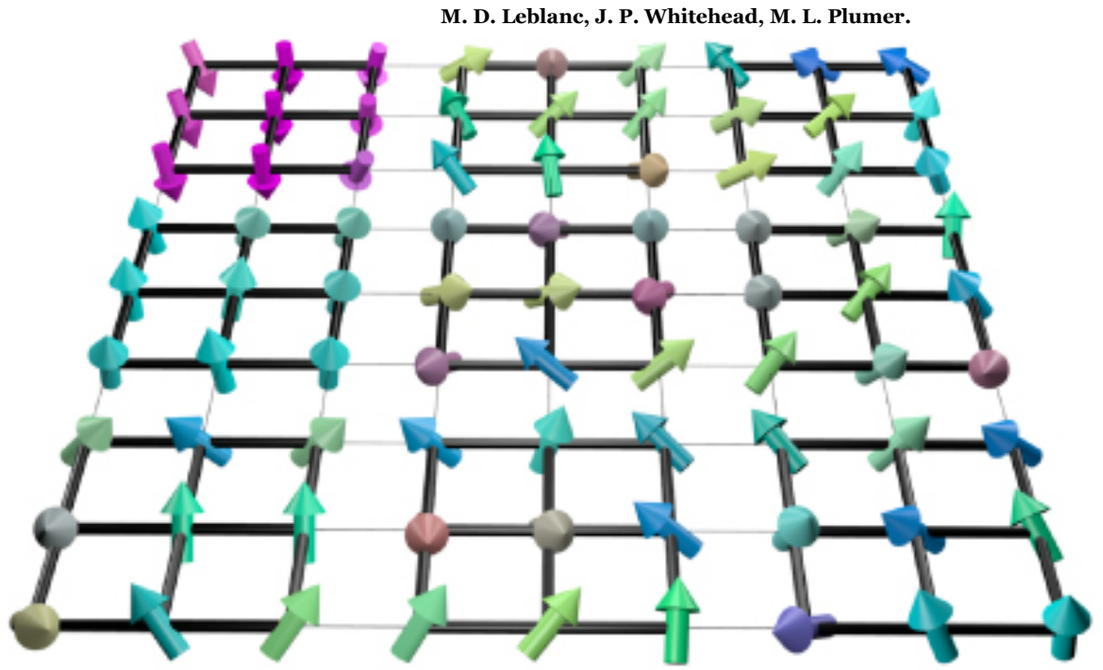
$$= U(x_i)\ \mathrm{e}^{-\sum s_i}$$

$$\frac{\partial H}{\partial x_i} =$$



QCD vs O(3) model





Lattice QCD (pure gauge)

$$S = \frac{\beta}{3} \sum_{n} \sum_{\mu<\nu} \mathrm{Re\ tr} \left[1 - U_{\mu\nu} \right]$$

$8 \times 4\, N^4$ **degrees of freedom**

$SU(3)$ **local gauge symmetry**

O(3) spin model

$$S = -\beta \sum_{\langle i,j \rangle} \vec{s}_i \cdot \vec{s}_j$$

$2\, N^2$ **degrees of freedom**

$O(3)$ **symmetry**

O(3) spin model training

1) Choose system size

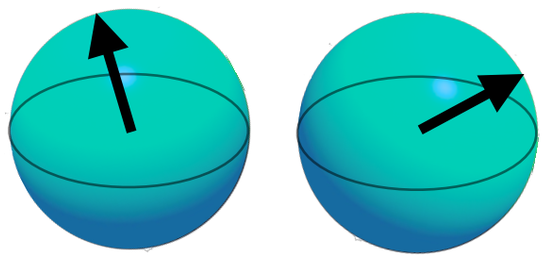
2) Choose hyper parameters

3) Train - loss function

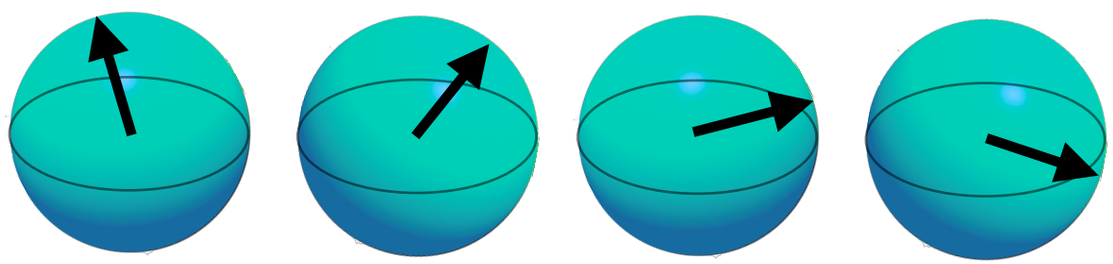
4) Validation

Choose system

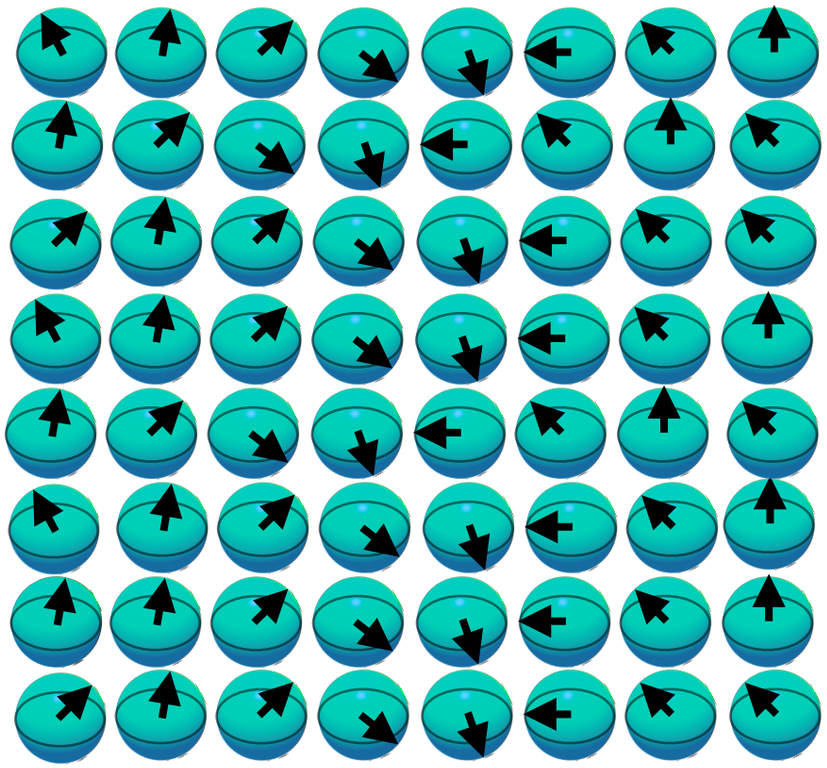
2x1 spin chain $S = -2\beta\ s_1 \cdot s_2$



4x1 spin chain $S = -\frac{\beta}{2} \sum_x\ s_x \cdot s_{x\pm1}$



8x8 spin lattice $S = -\frac{\beta}{2} \sum_x\ s_x \cdot s_{x\pm\hat{i}} + s_x \cdot s_{x\pm\hat{j}}$

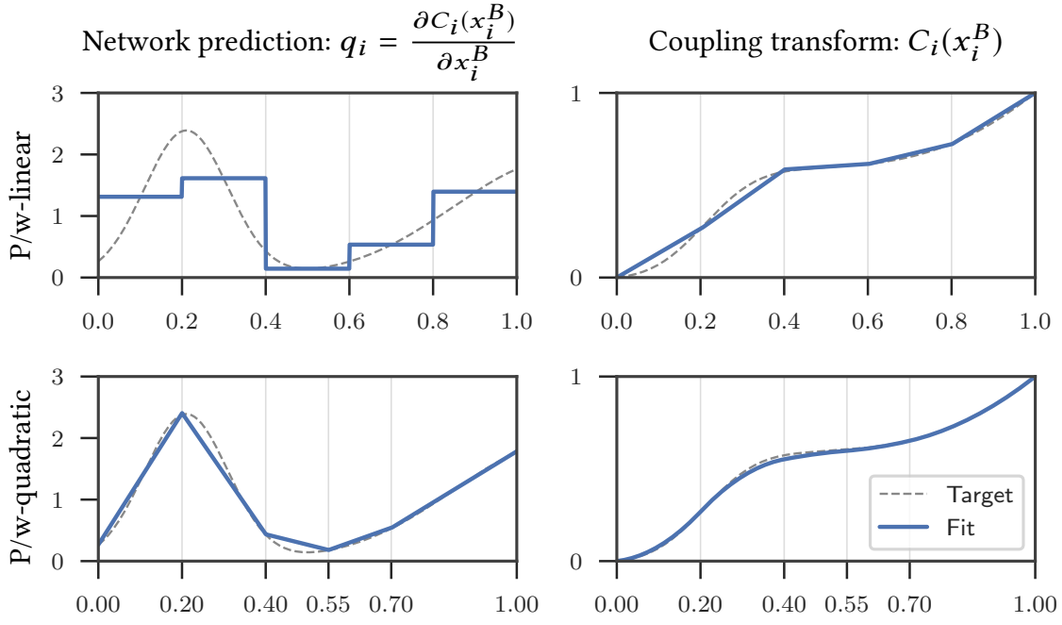


Choose hyper parameters

1) Number of samples - 5000

2) Number of networks - 5

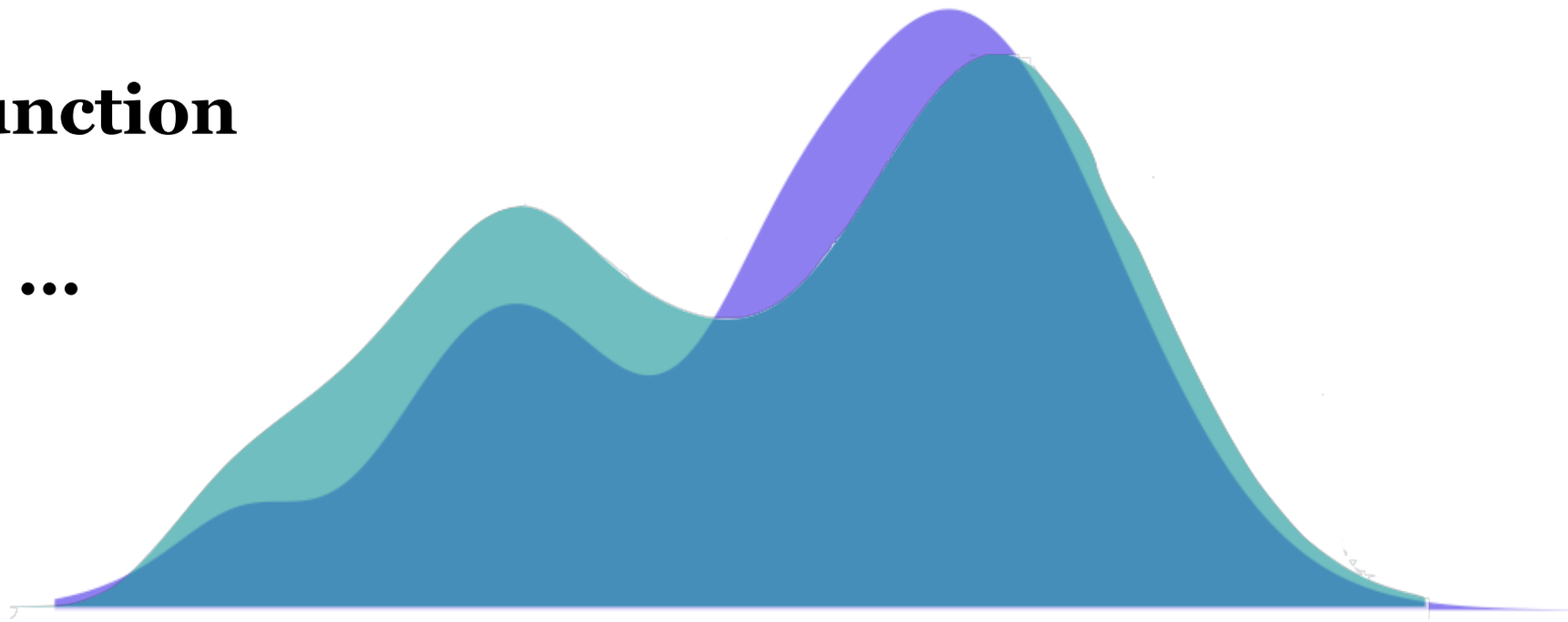
3) Number of layers - 6



Disney Research *Neural Important Sampling*

4) Coupling transform - piecewise cumulative distribution function

5) Learning rate, ...



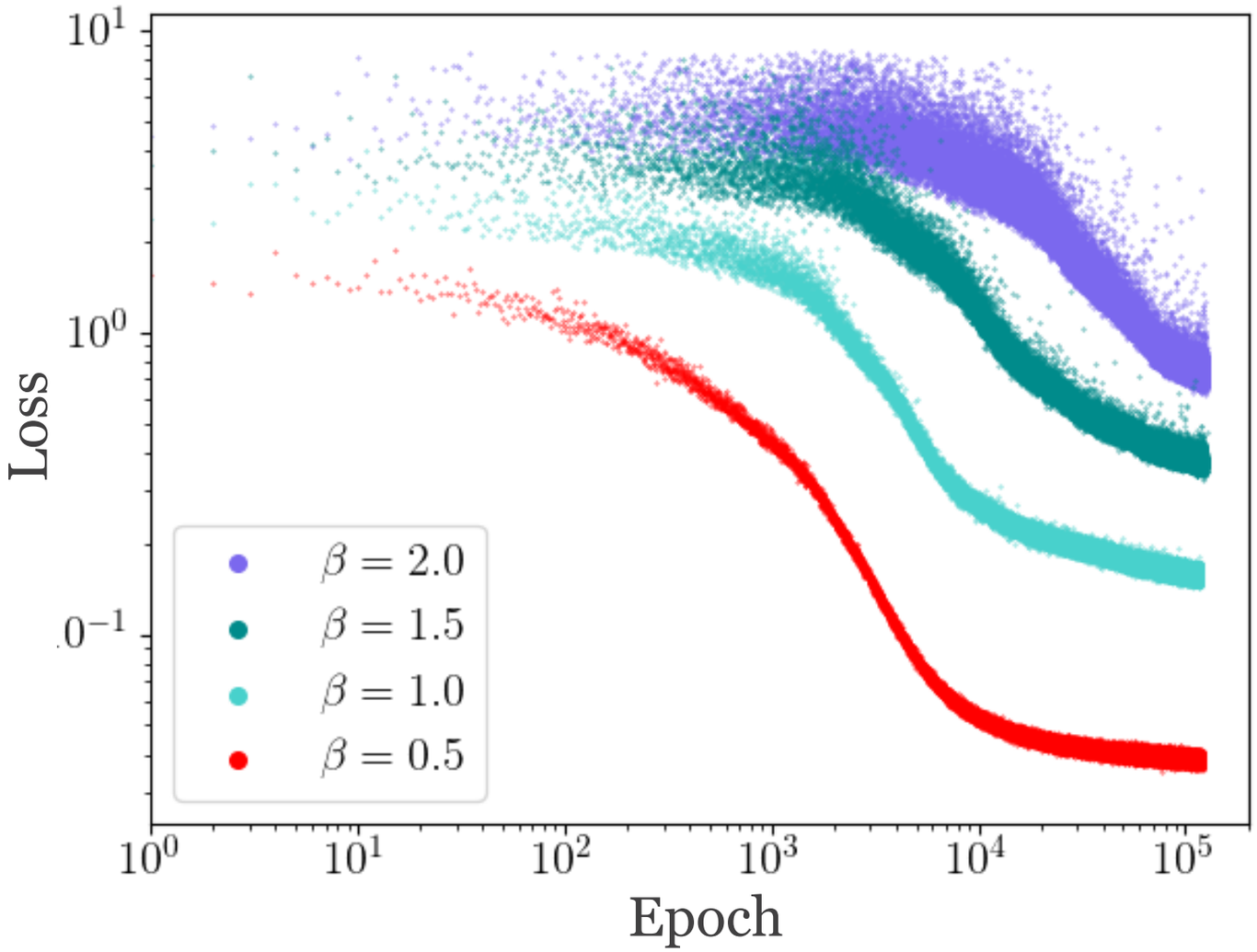
Training against loss

Loss functions

KL $$L(x,\theta) = \sum \frac{p_i}{q_i} \log\left(\frac{p_i}{q_i}\right)$$

Exp $$L(x,\theta) = \sum \frac{p_i}{q_i} \log\left(\frac{p_i}{q_i}\right)^2$$

Chi² $$L(x,\theta) = \sum \frac{(p_i - q_i)^2}{q_i^2}$$

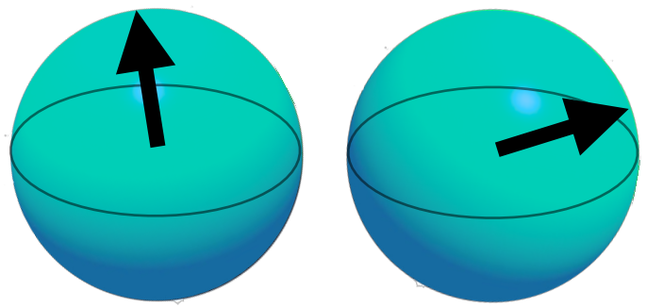


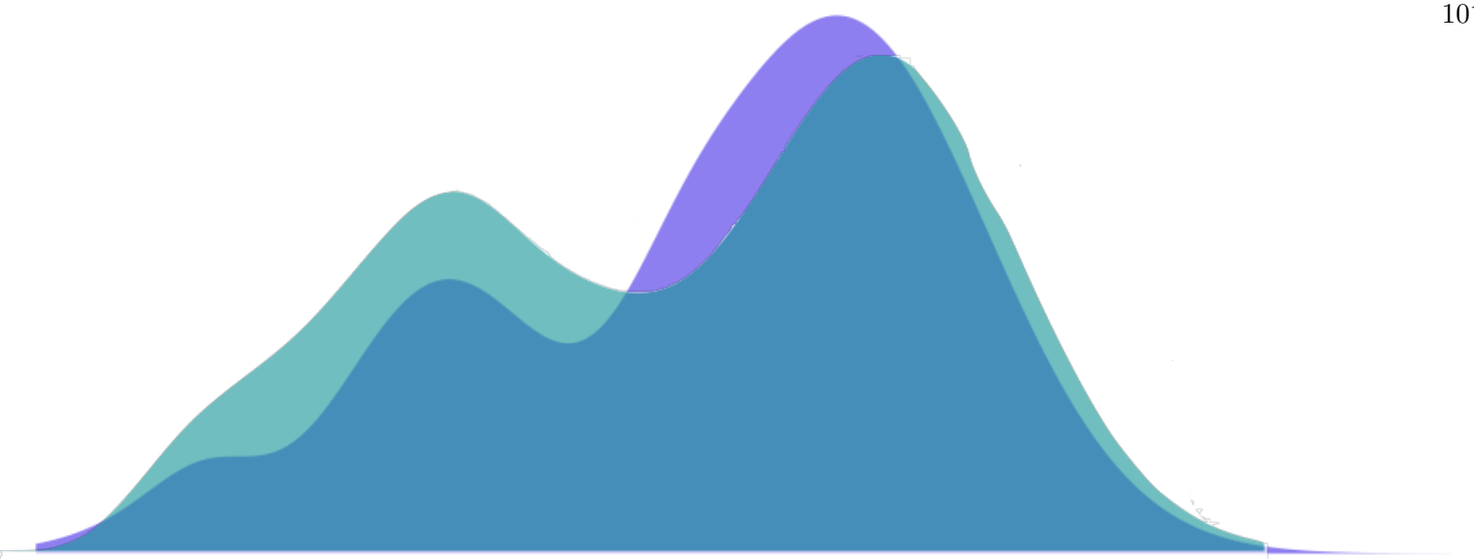
How to believe a neural network: Validation

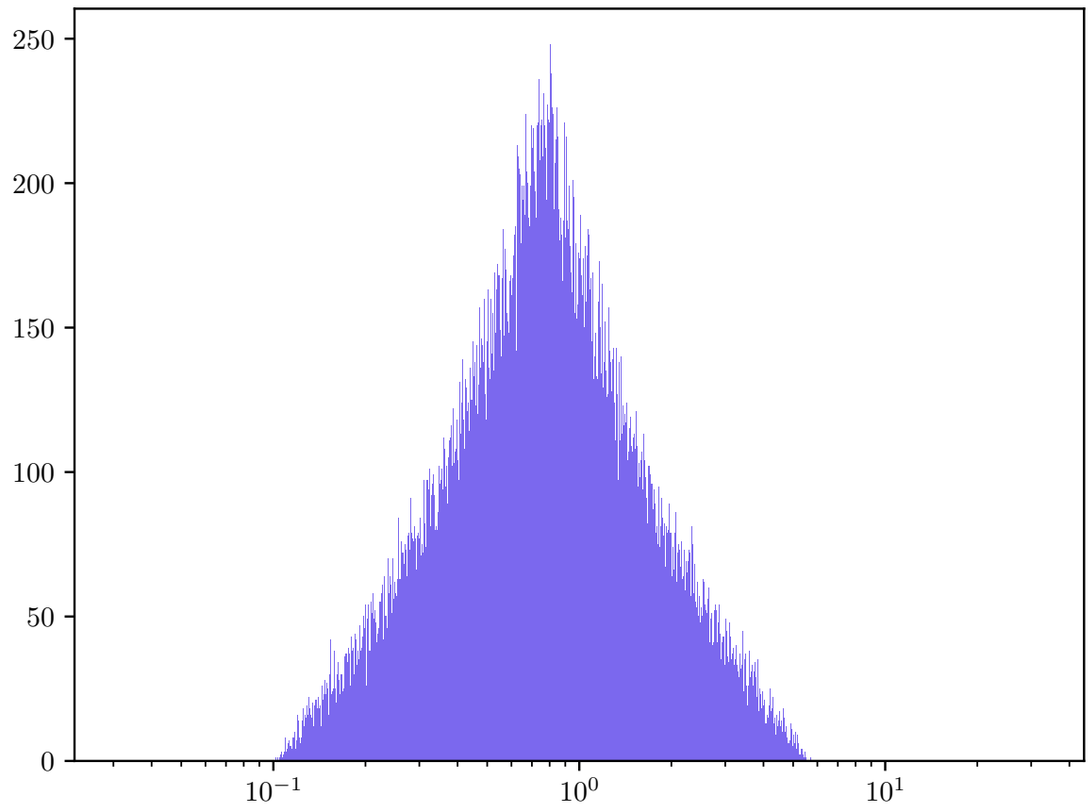
Check p and q

Learning:

$$q(x) \rightarrow p(x)$$

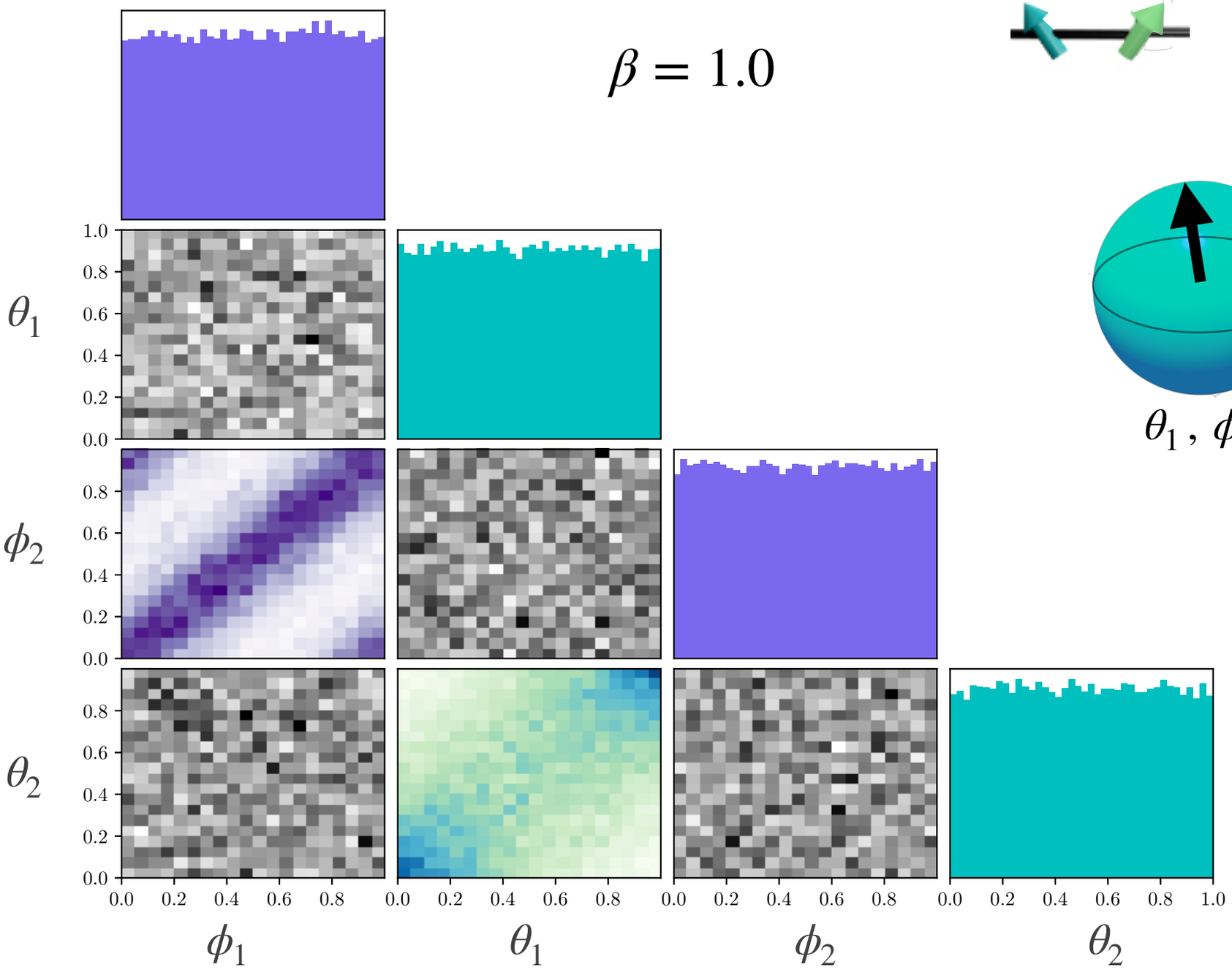




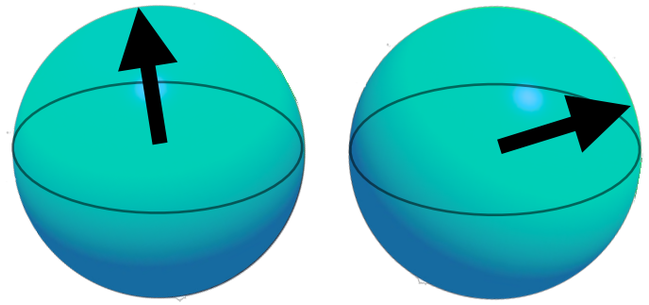


$$\frac{p_i}{q_i}$$

Qualitative correlations



4 parameters

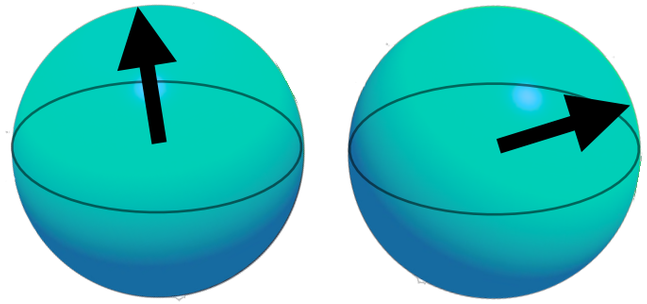


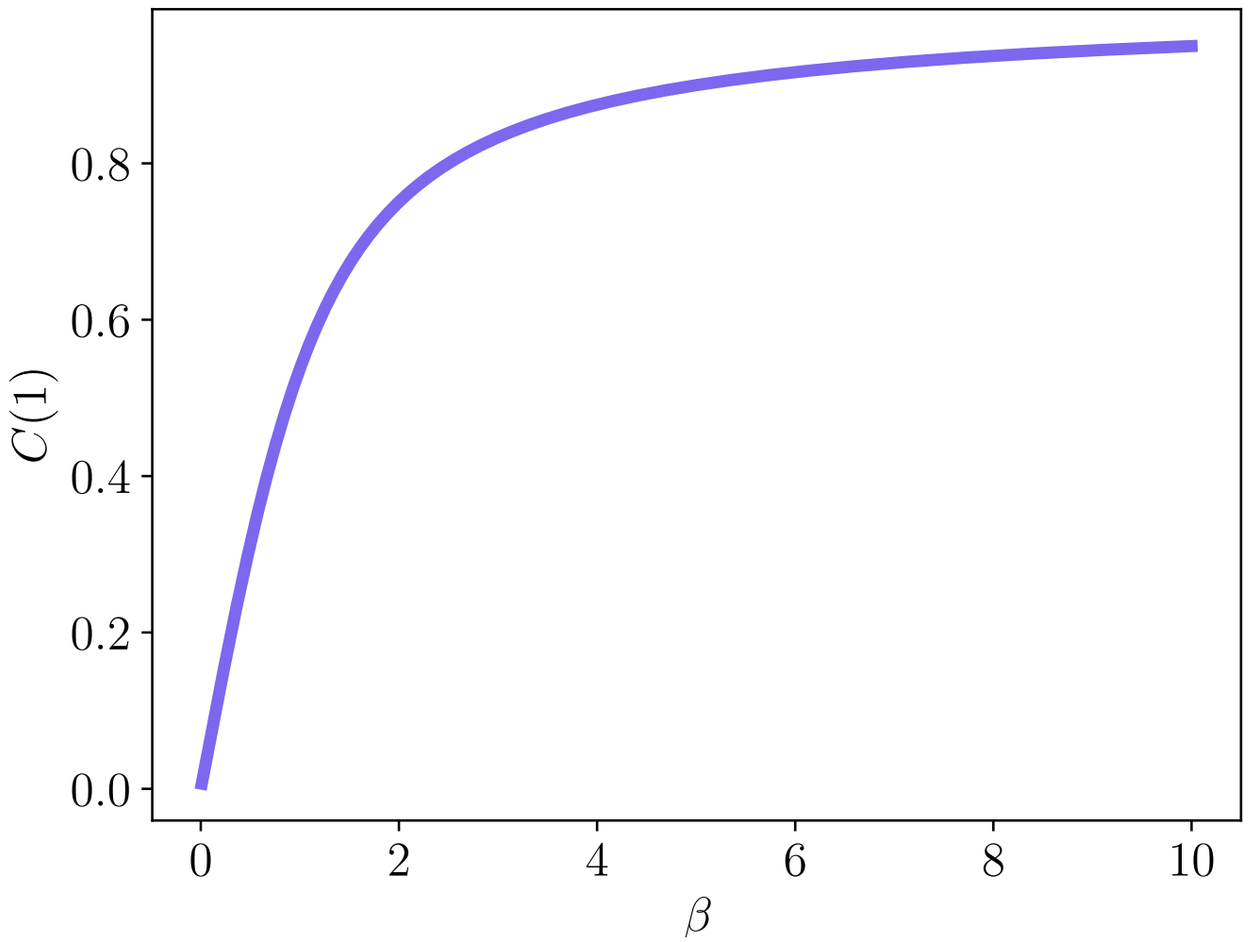
Quantitative correlations

$$S = -2\beta\ s_1 \cdot s_2 = -2\beta \cos\theta$$

$$C(1) = \frac{\int d\cos\theta\ \ \cos\theta\ \mathrm{e}^{-S(\theta)}}{\int d\cos\theta\ \mathrm{e}^{-S(\theta)}}$$

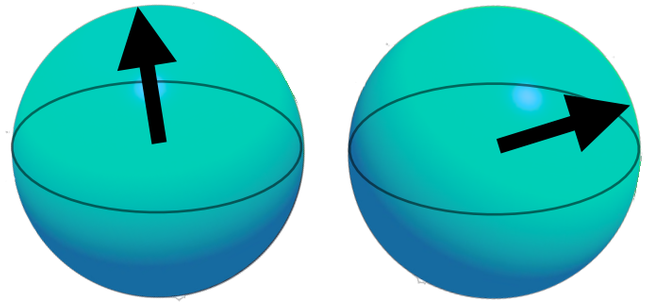
$$= \coth(2\beta) - \frac{1}{2\beta}$$

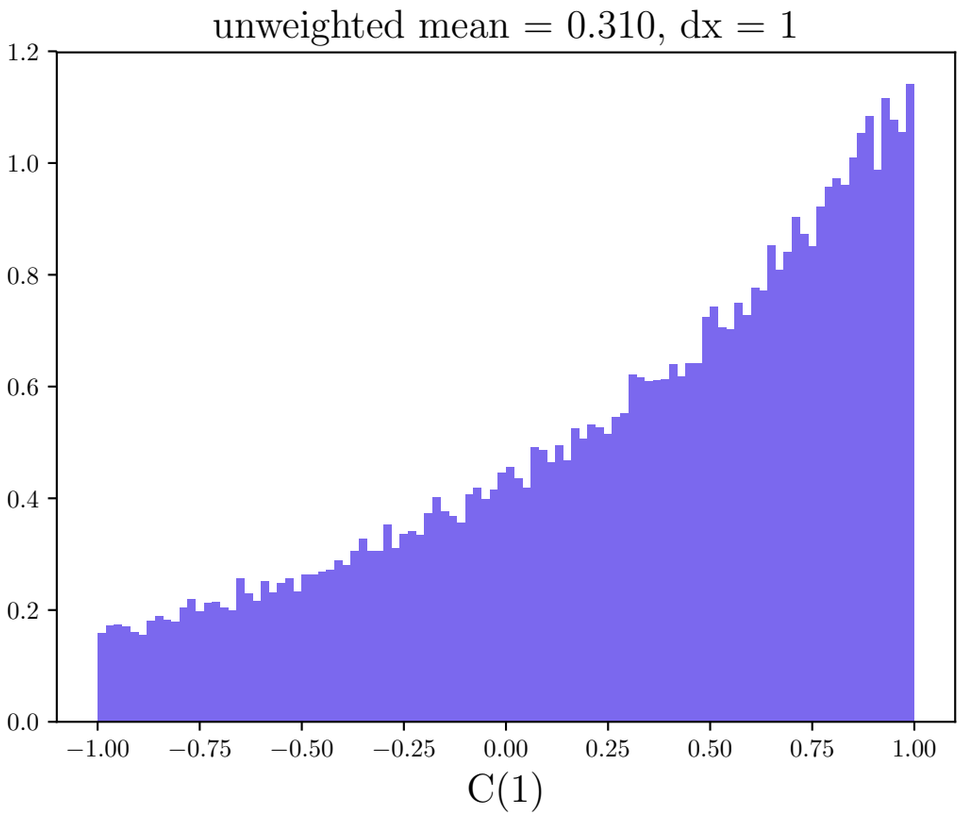


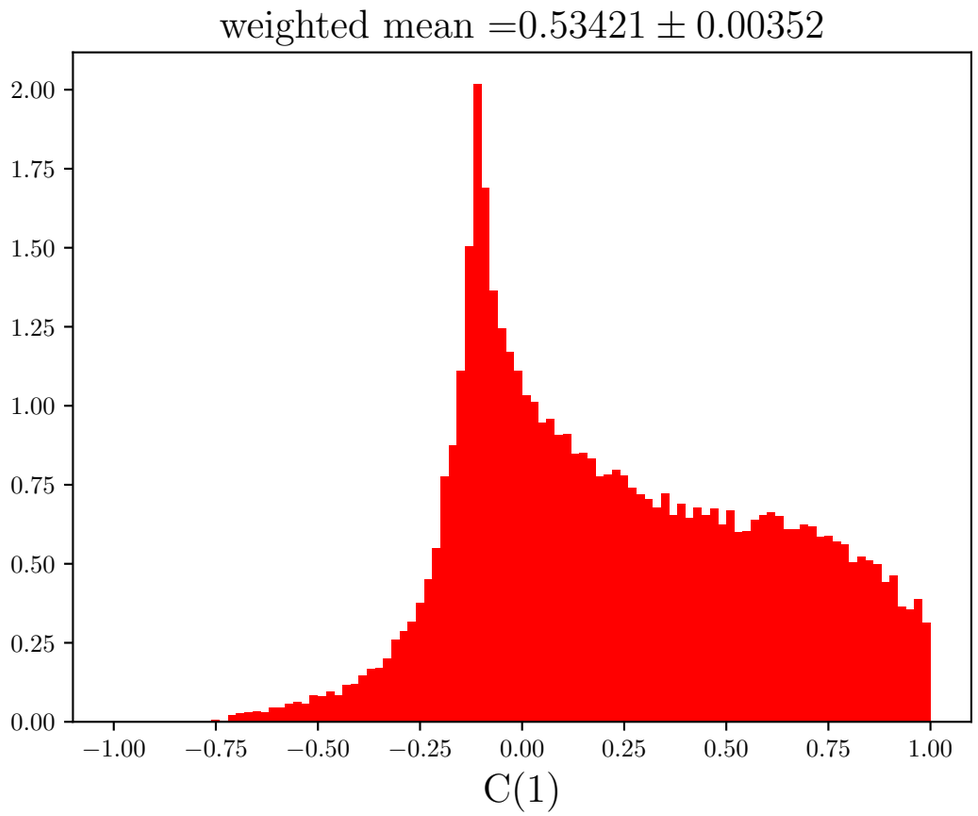


Quantitative correlations

$$C(1,\beta = 1) = 0.53731 \qquad \langle \mathcal{O} \rangle = \sum_i \mathcal{O}_i \frac{p_i}{q_i} |_q$$

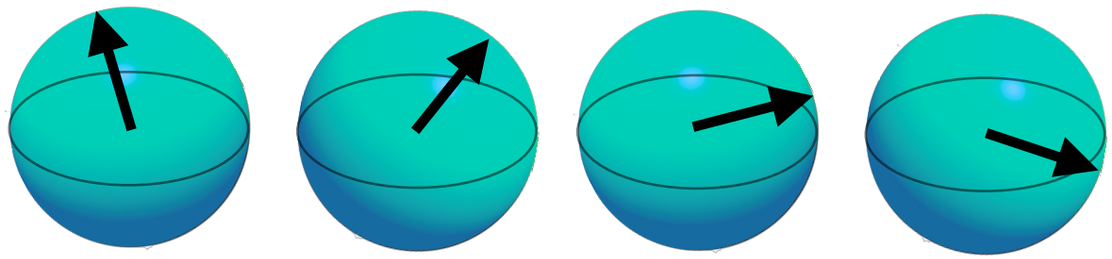


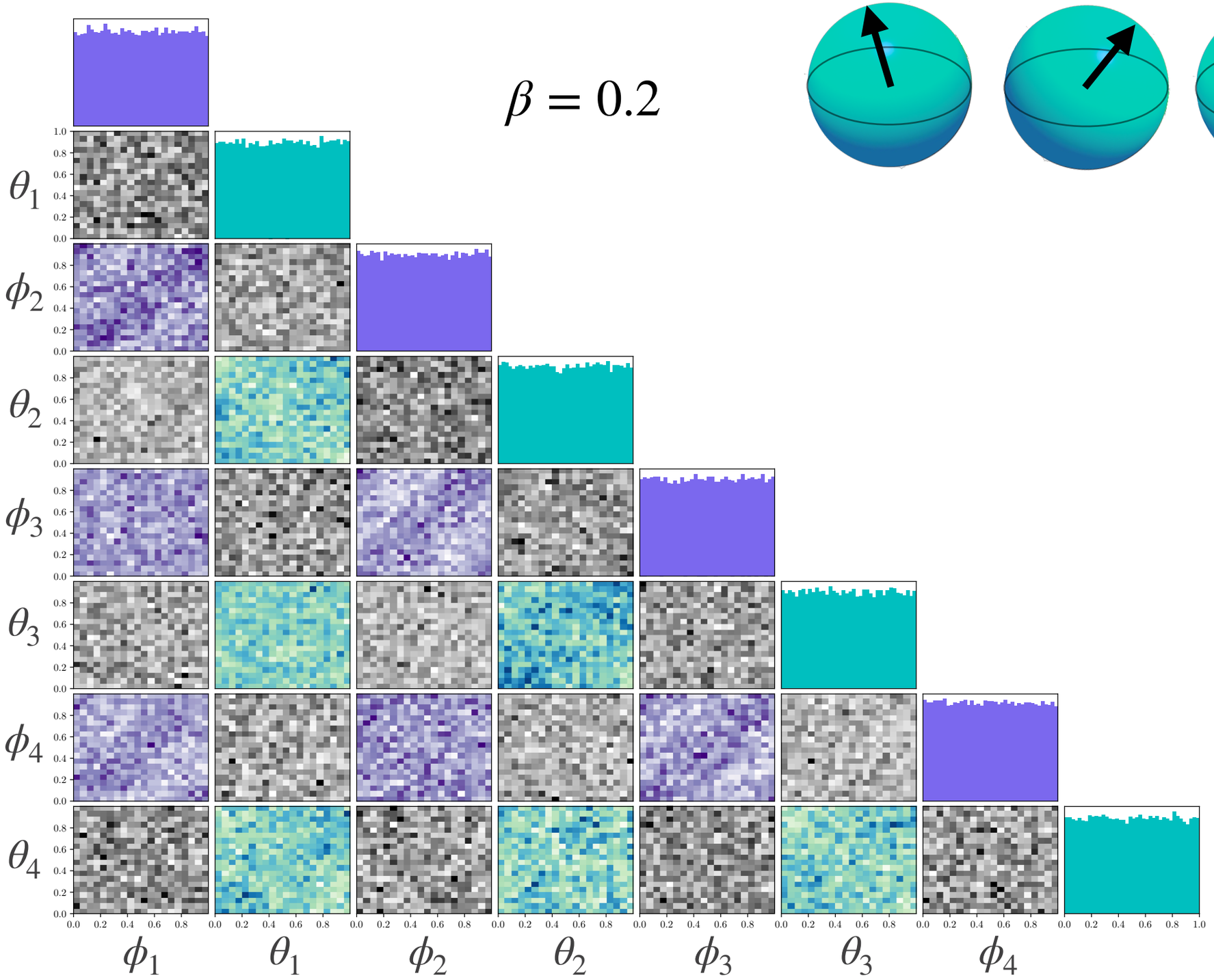




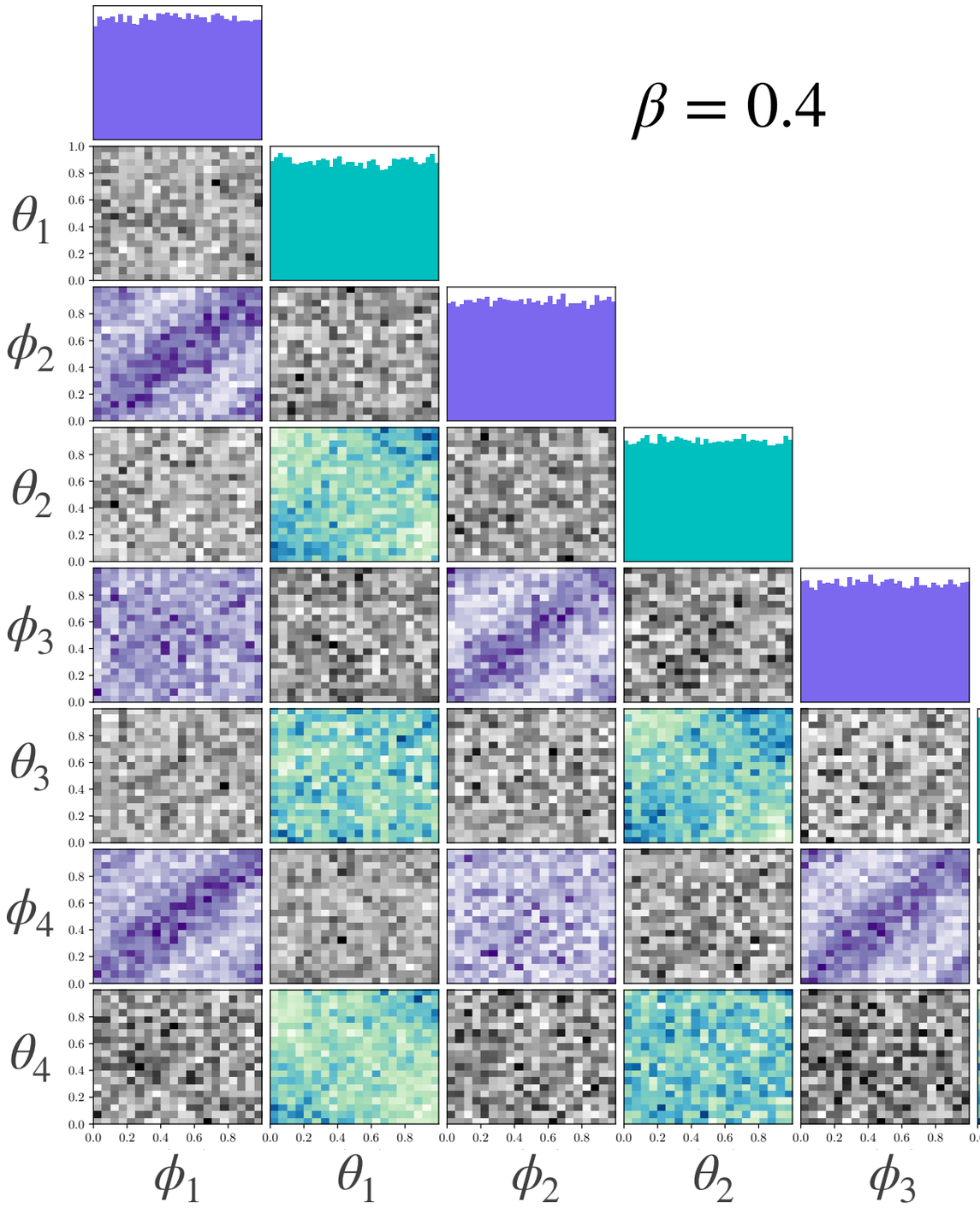
Quantitative correlations

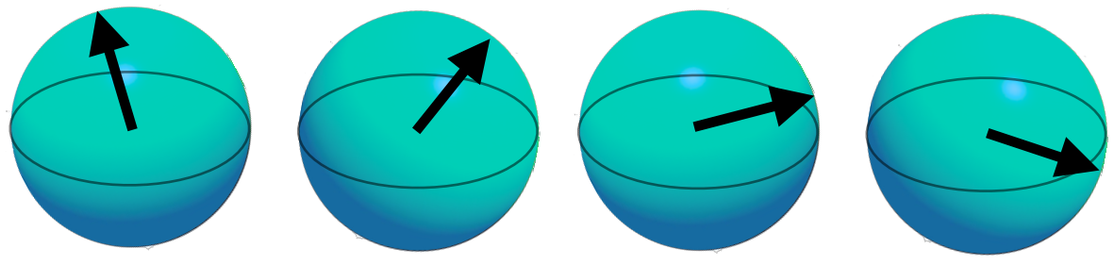
$\beta = 0.2$





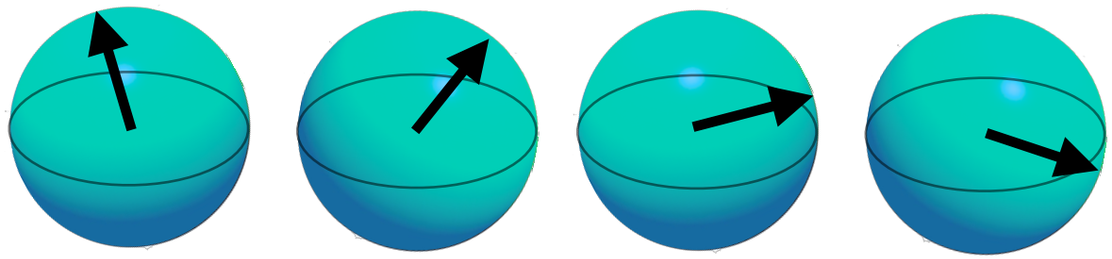
Quantitative correlations

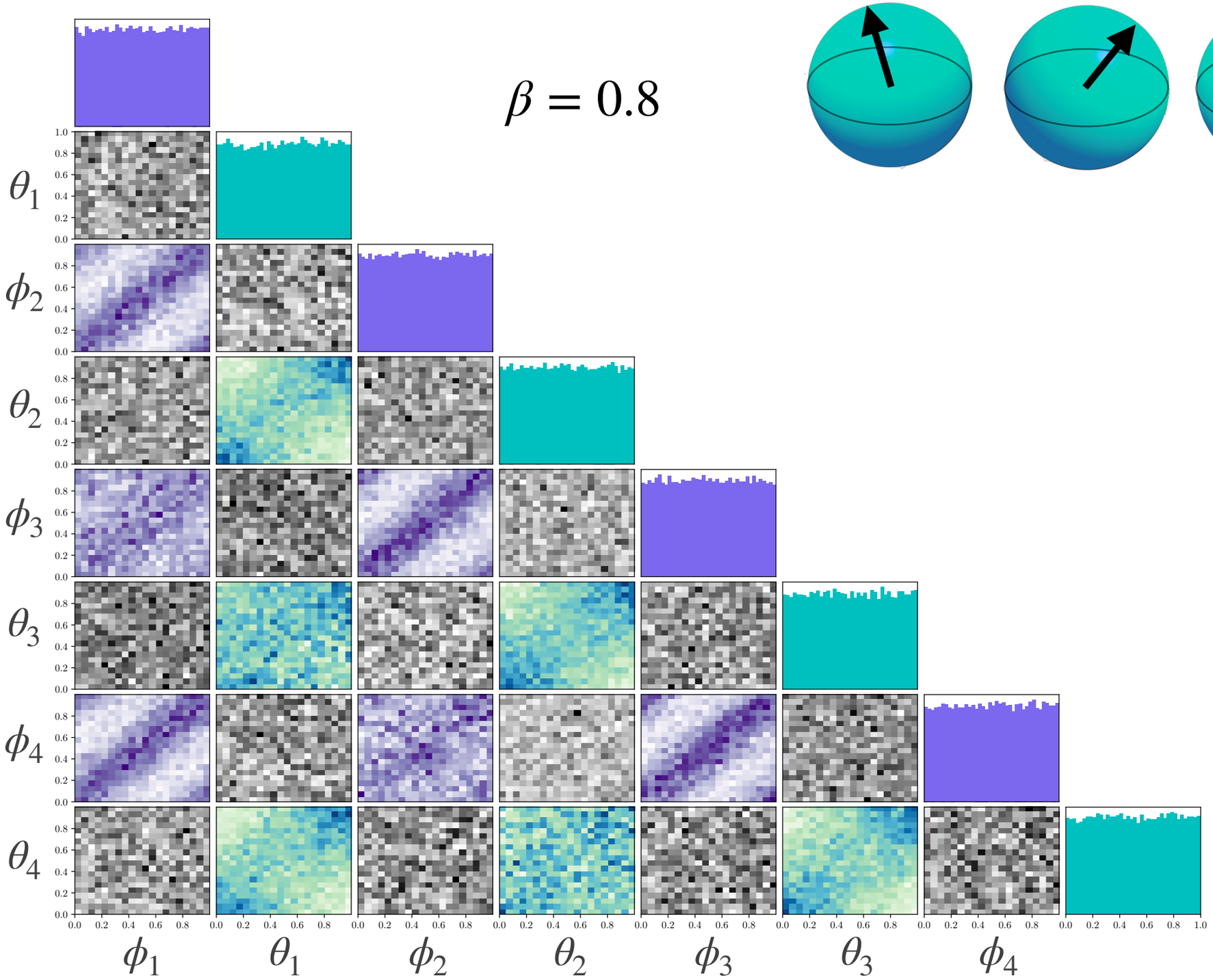




Quantitative correlations

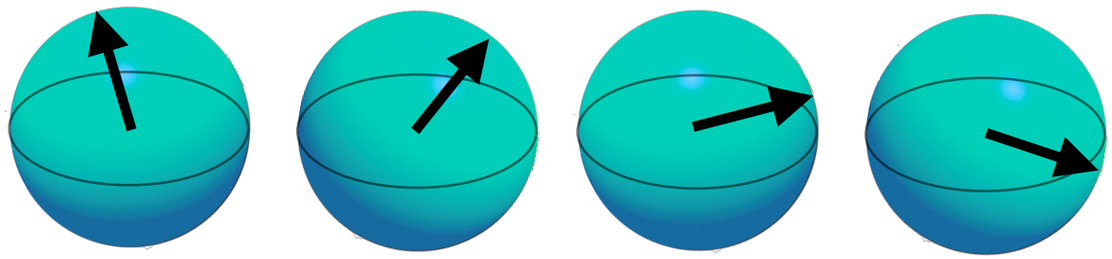
$\beta = 0.8$

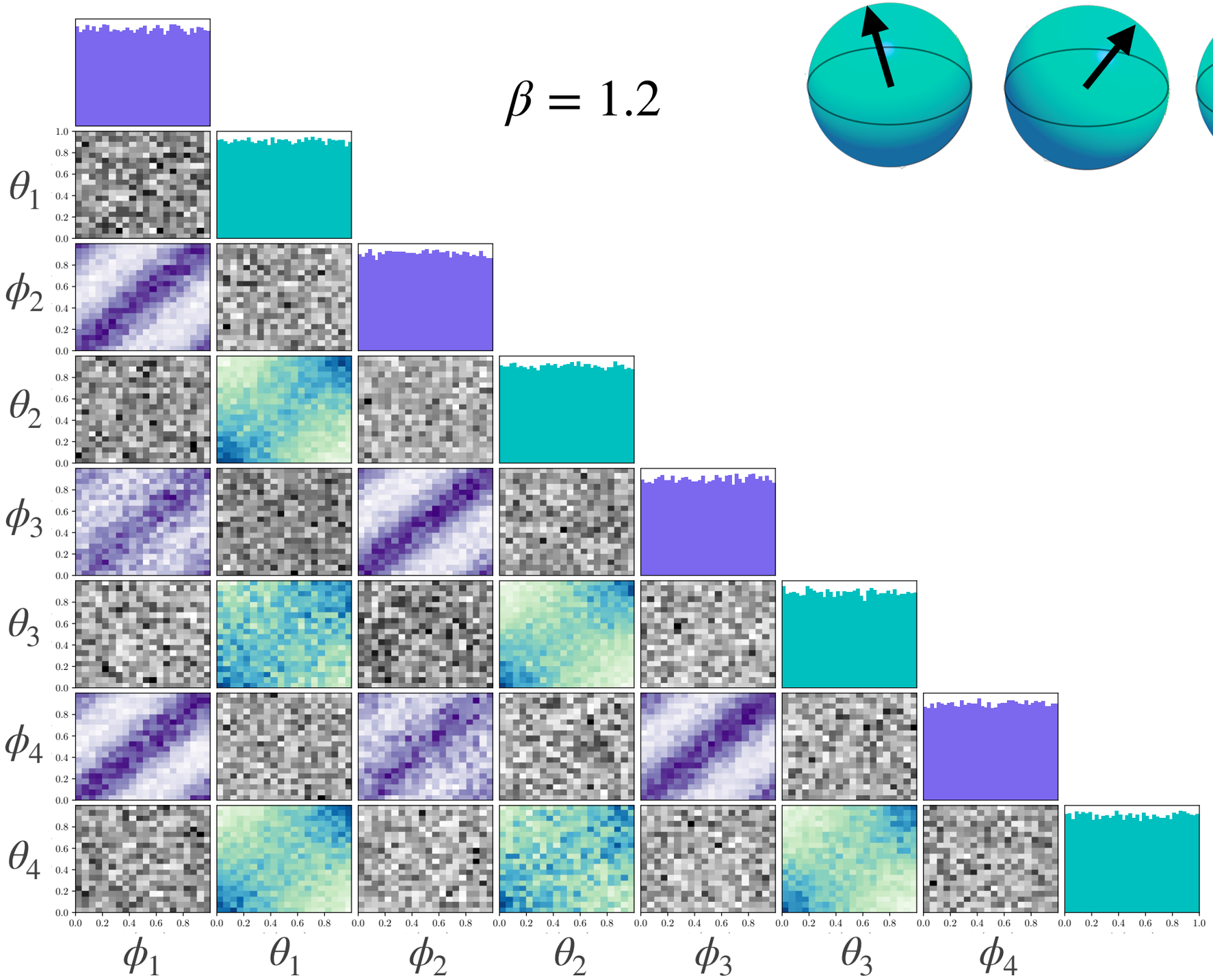




Quantitative correlations

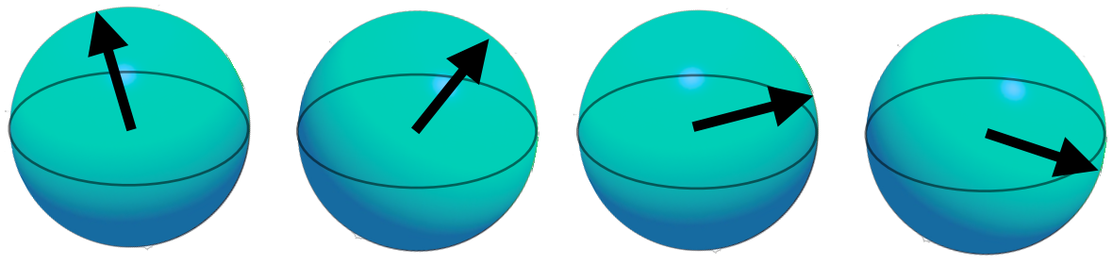
$\beta = 1.2$

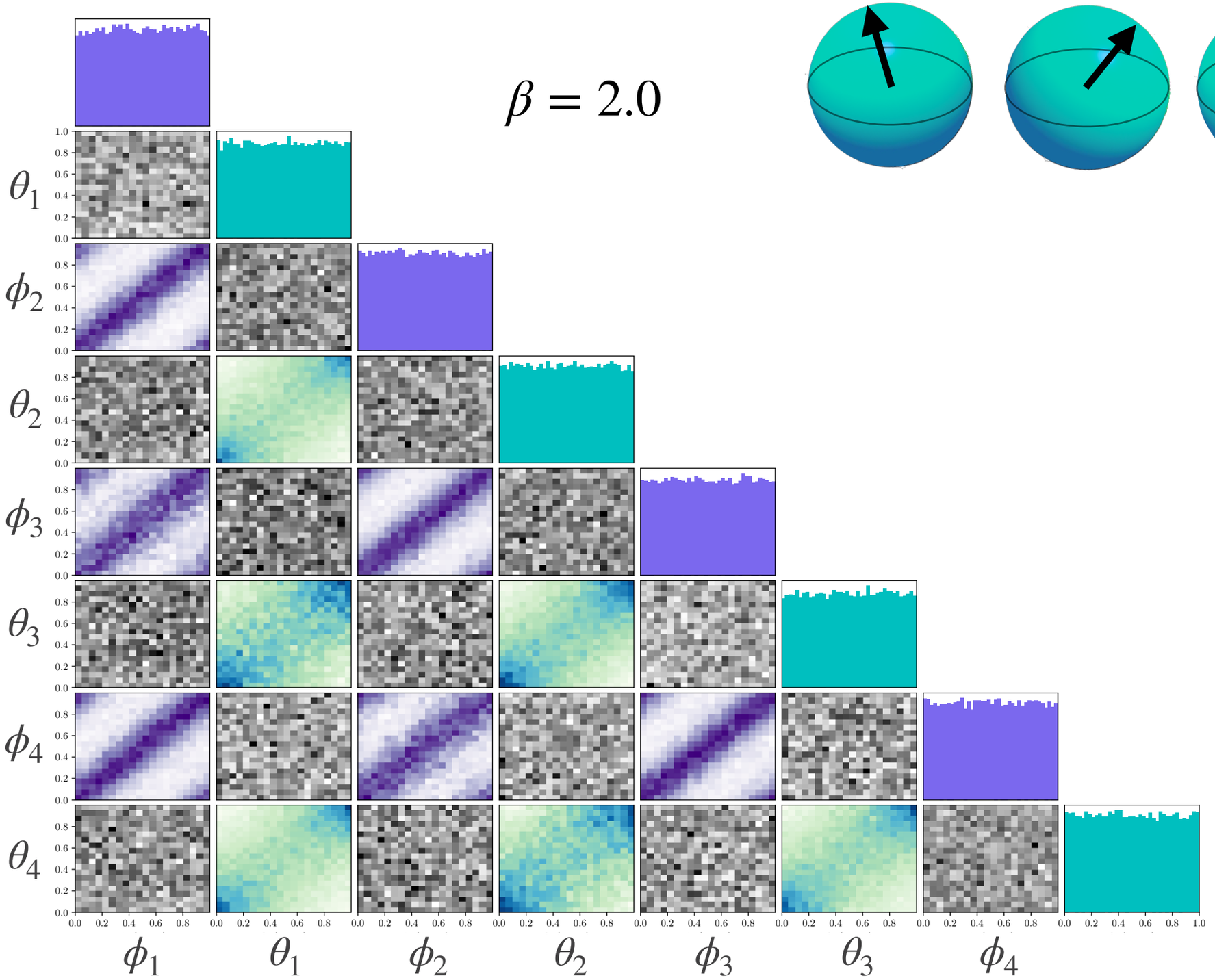




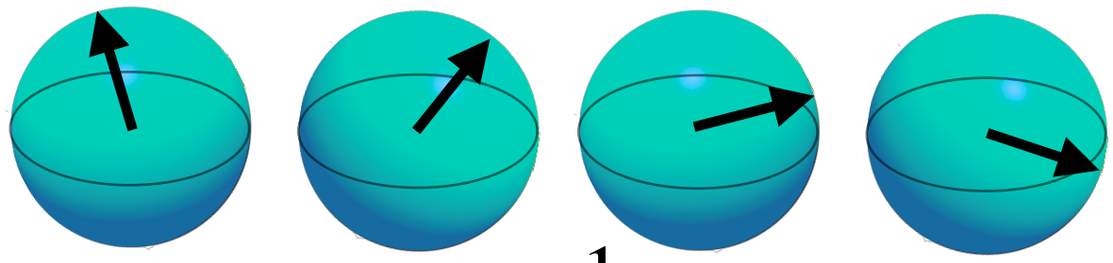
Quantitative correlations

$\beta = 2.0$





Quantitative correlations



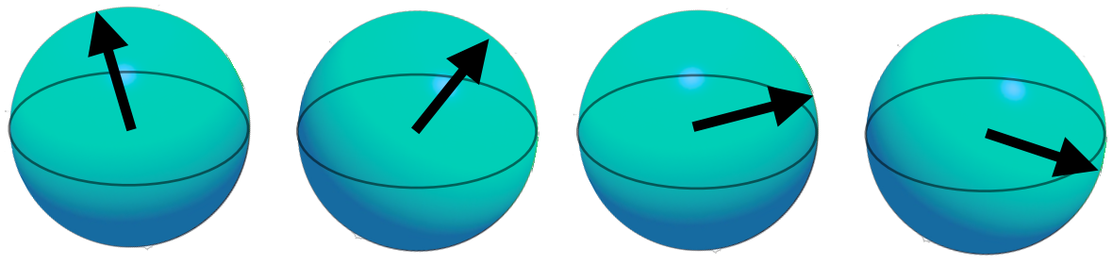
3 methods:

1) **Traditional Metropolis MCMC** $\langle \mathcal{O} \rangle = \frac{1}{N} \sum \mathcal{O}_i |_p$

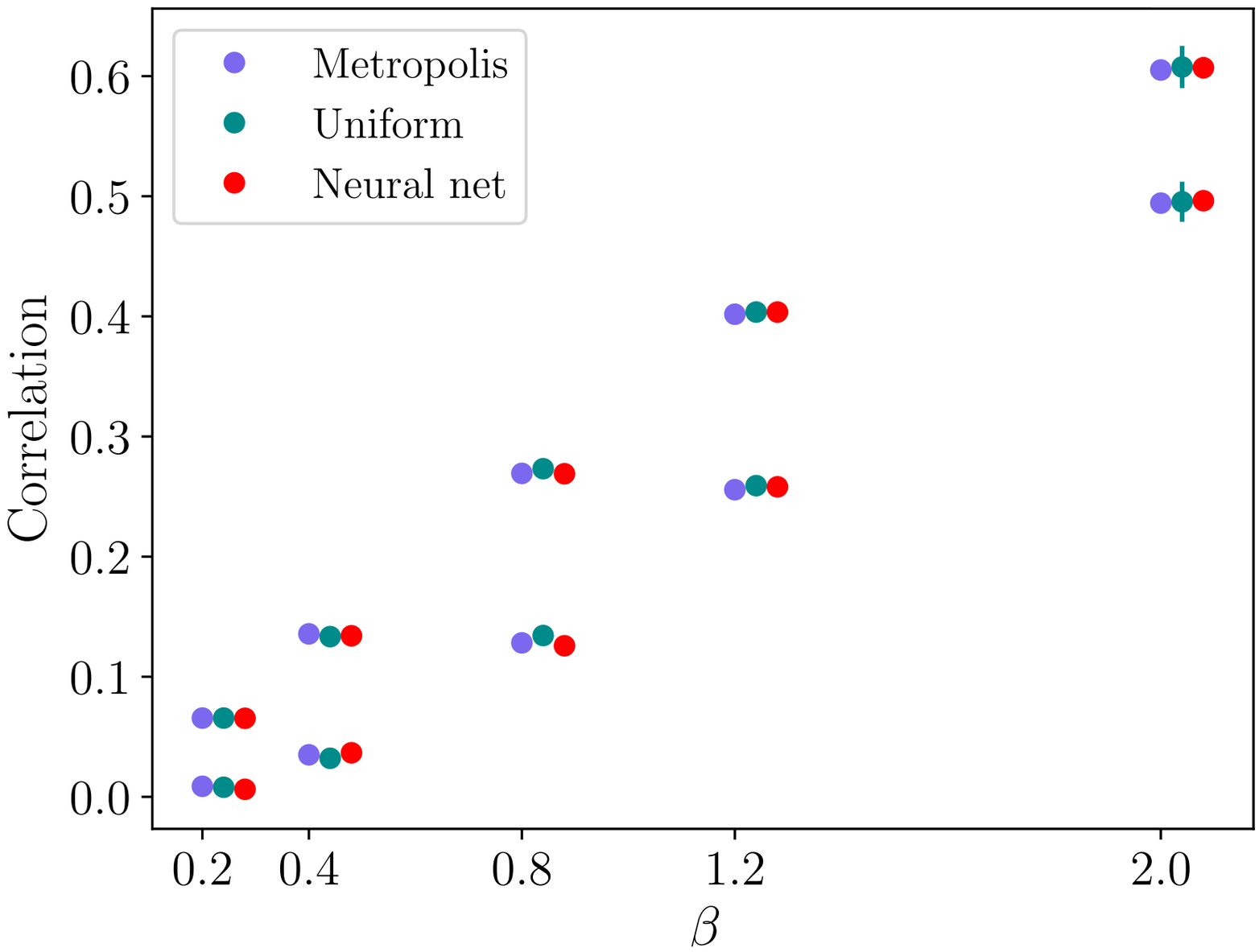
2) **Random (uniform)** $\langle \mathcal{O} \rangle = \frac{\sum \mathcal{O}_i \, p_i}{\sum p_i}$

3) **Generative flows training** $\langle \mathcal{O} \rangle = \frac{\sum \mathcal{O}_i \, \frac{p_i}{q_i} |_q}{\sum \frac{p_i}{q_i}}$

Quantitative correlations



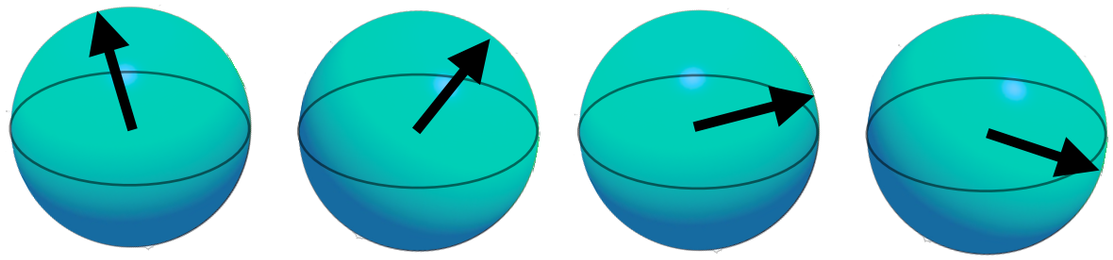
50,000 samples from each



How to believe a neural network: Tests

Tests of invariant physics

Similar actions



Long-distance physics should not depend on short distance details of action

$$S = -\sum \beta \; s_x \cdot s_{x\pm1}$$

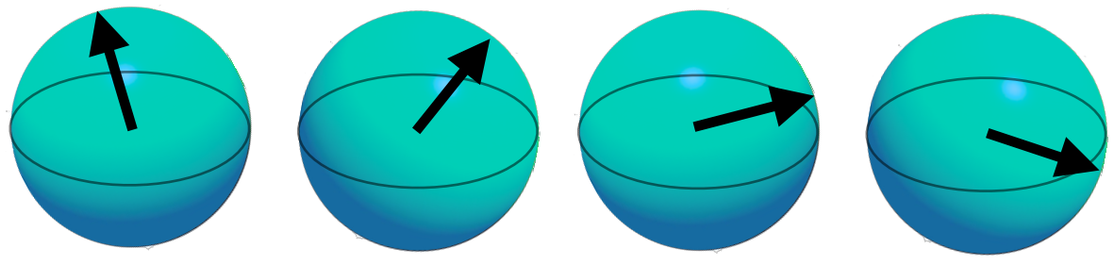
$$S = -\sum \beta_1 \; s_x \cdot s_{x\pm1} + \beta_2 \; s_x \cdot s_{x\pm2}$$

$$S = -\sum \beta_1 \; s_x \cdot s_{x\pm1} + \beta_2 \; s_x \cdot s_{x\pm2} + \beta_3 \; (s_x \cdot s_{x\pm1})^2$$

$$S = -\sum \beta_\alpha \; S_\alpha$$

Tests of invariant physics

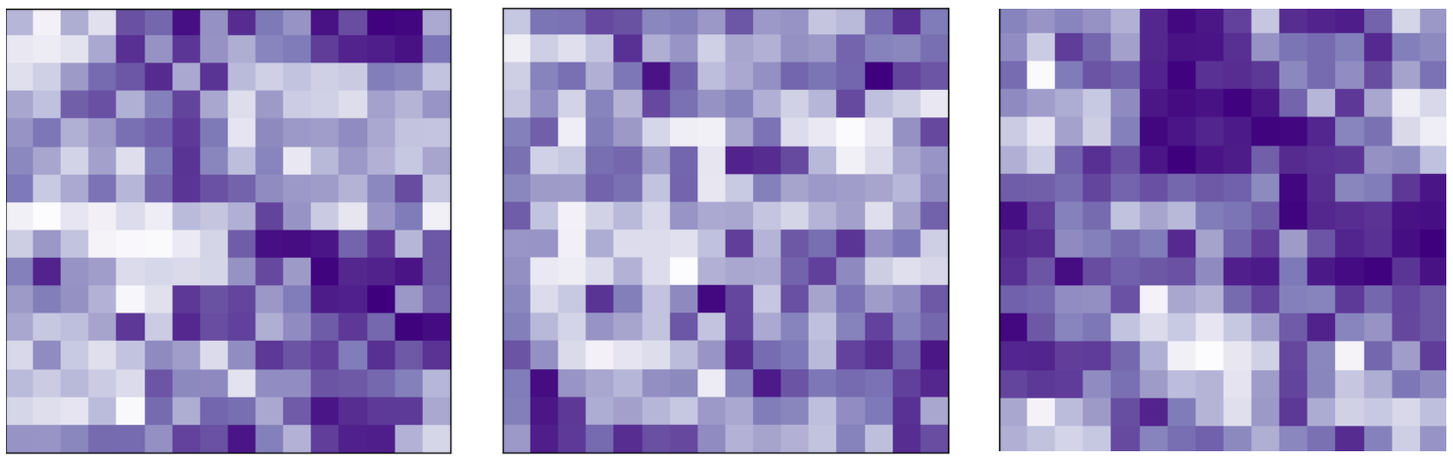
Similar actions



1) Given 1-term action, predict β from ensemble

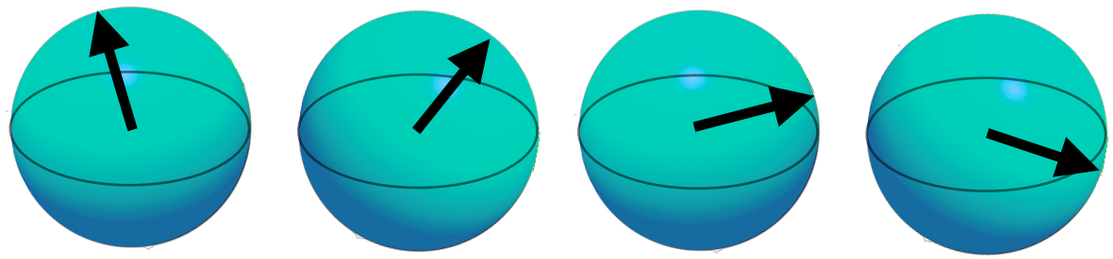
$$S_1 = -\sum \beta \; s_x \cdot s_{x\pm 1}$$

What is β ?



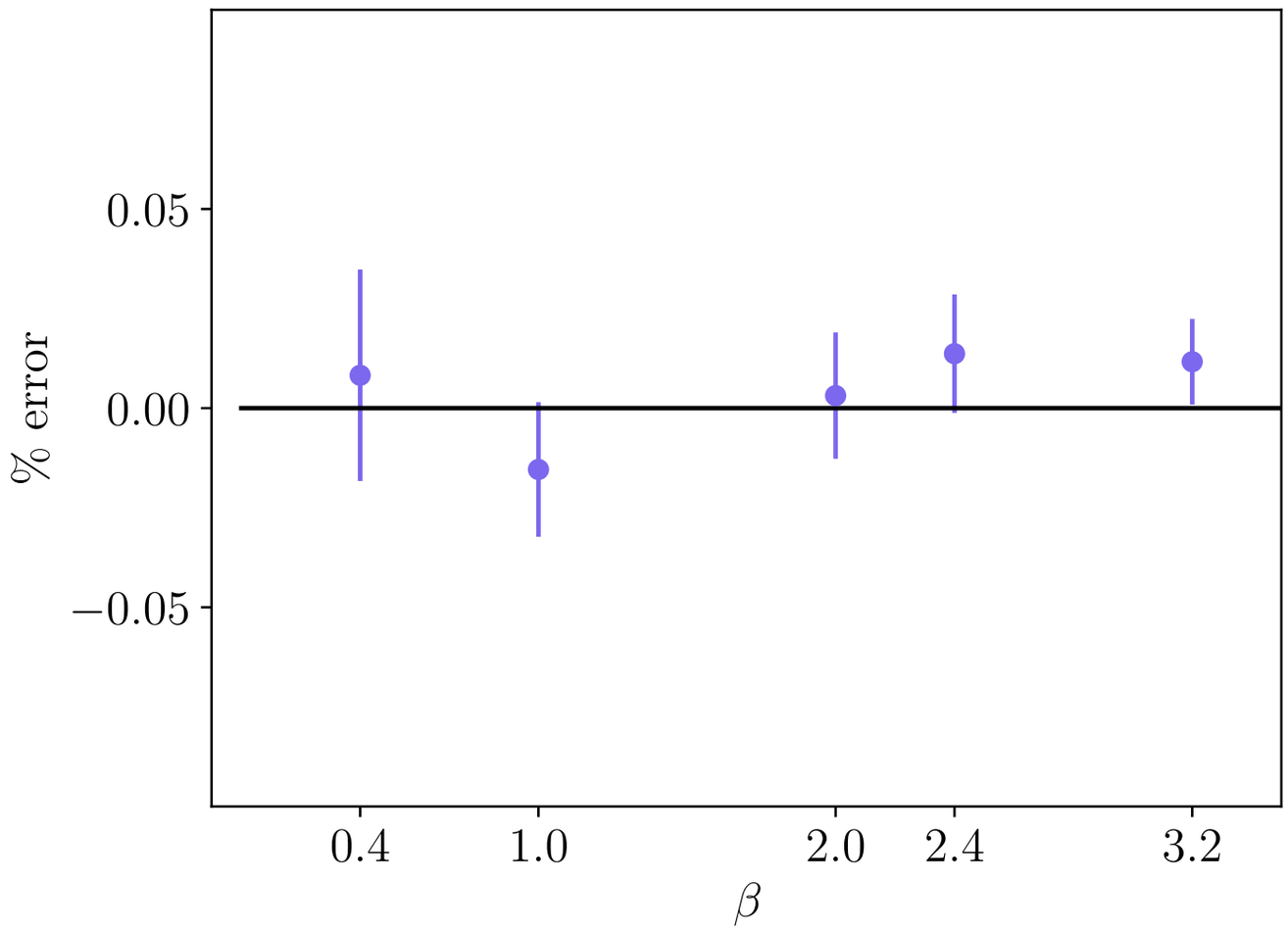
Tests of invariant physics

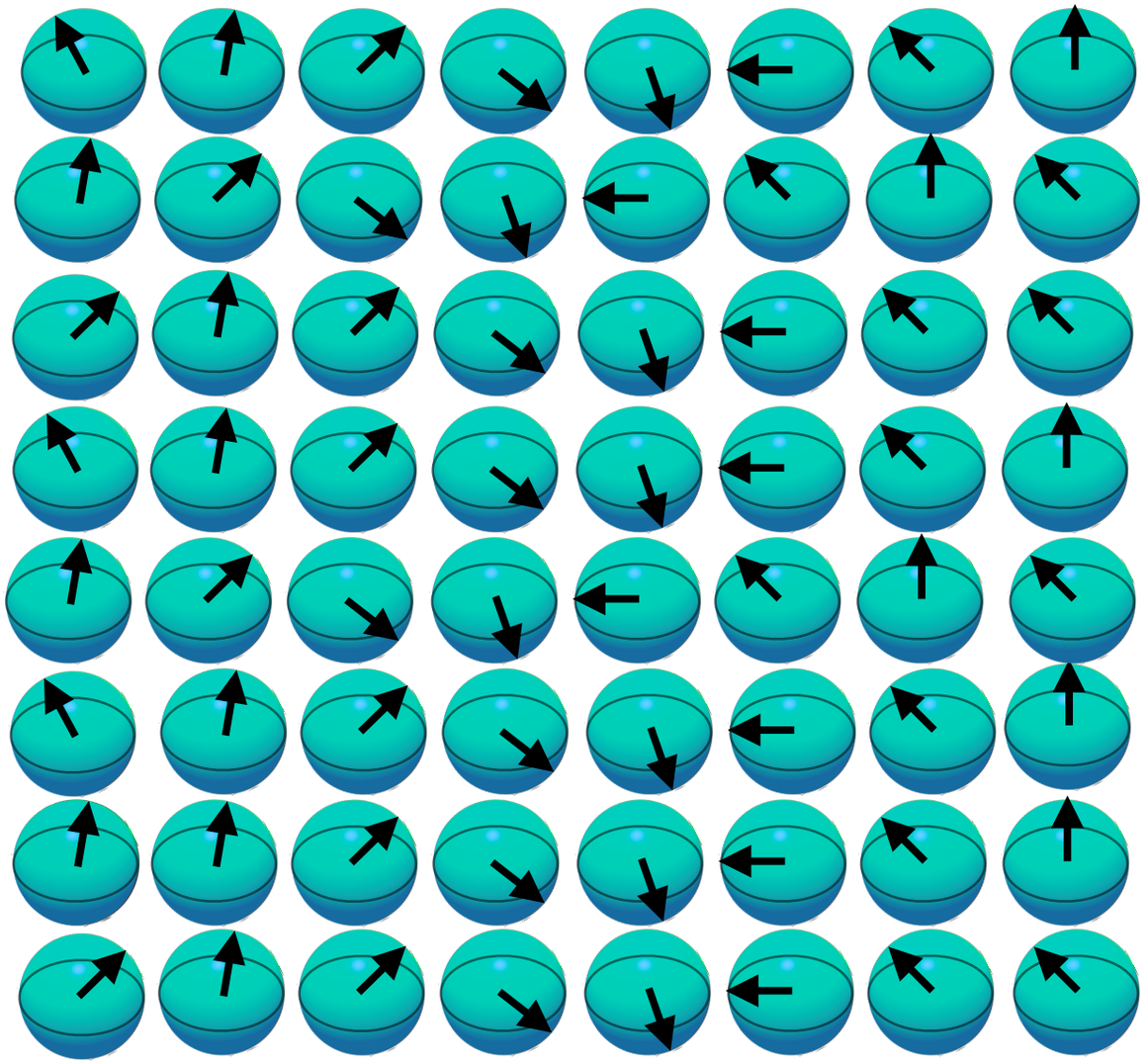
Similar actions



M. Hasenbusch et al., Phys Lett B 338 (1994) 308-312

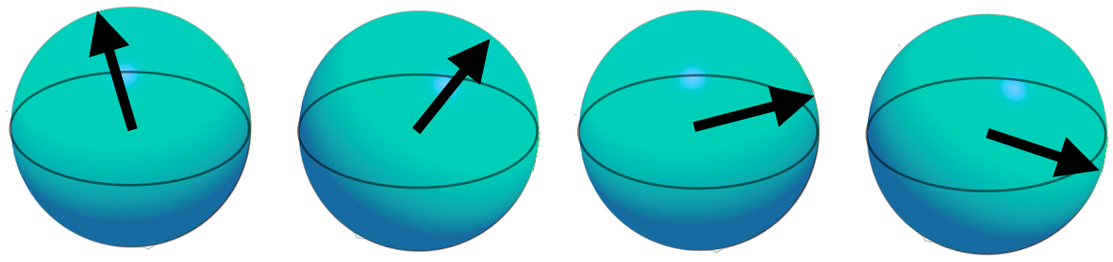
1) Given 1-term action, predict β from ensemble





Tests of invariant physics

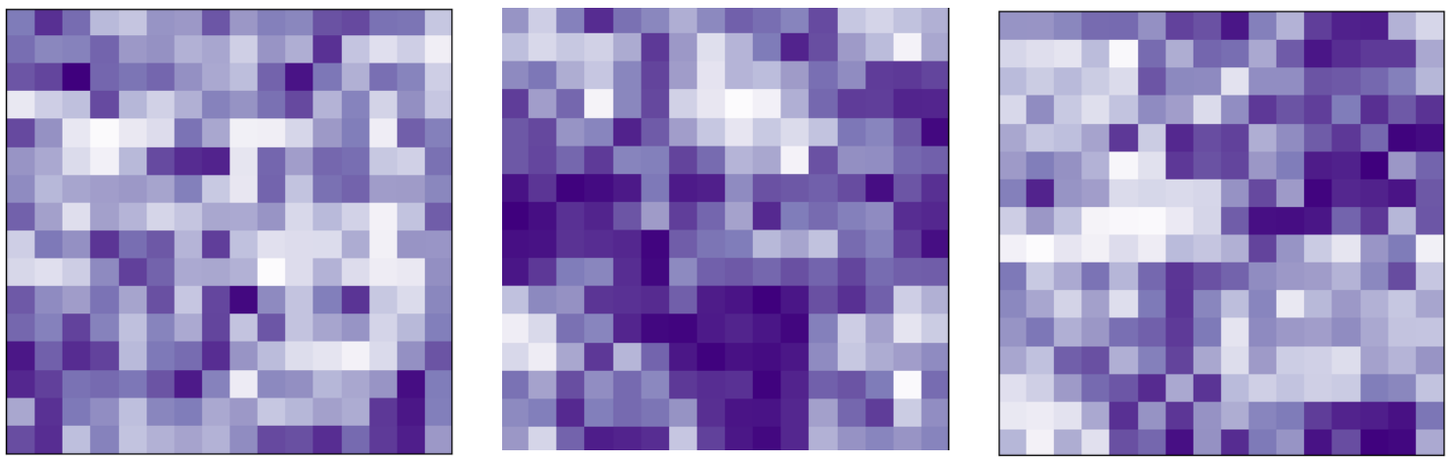
Similar actions



1) Given 1-term action, predict β from ensemble

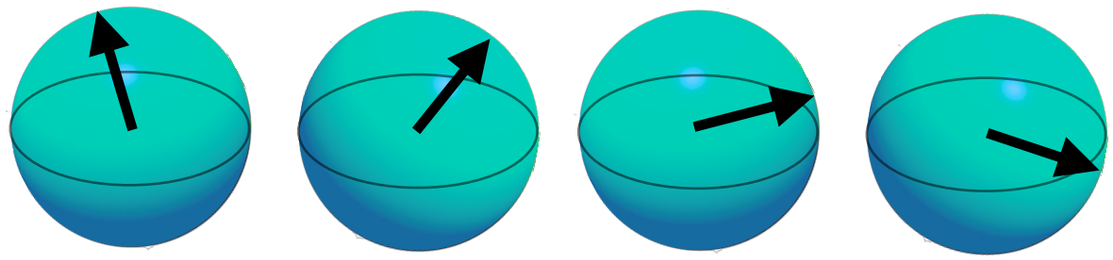
2) Given 2-term action, predict β_1 , β_2 from ensemble

$$S_2 = -\sum \beta_1 \; s_x \cdot s_{x\pm1} + \beta_2 \; s_x \cdot s_{x\pm2}$$ **What are β_1 , β_2 ?**



Tests of invariant physics

Similar actions



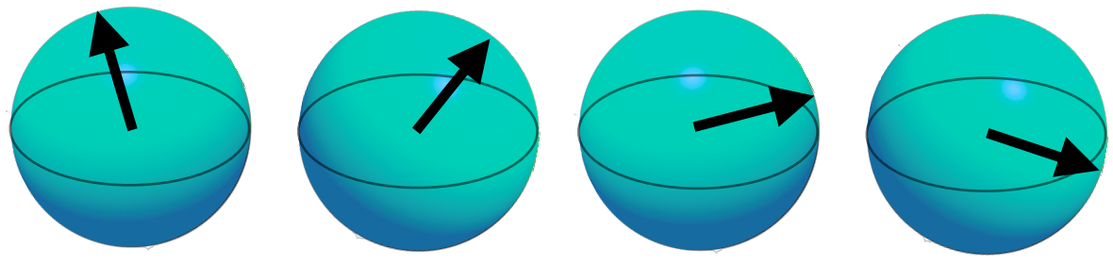
1) Given 1-term action, predict β from ensemble

2) Given 2-term action, predict β_1 , β_2 from ensemble

3) Compute corresponding "truncated" actions: $\{\beta_1, \beta_2\} \rightarrow \{\beta_1, 0\}$

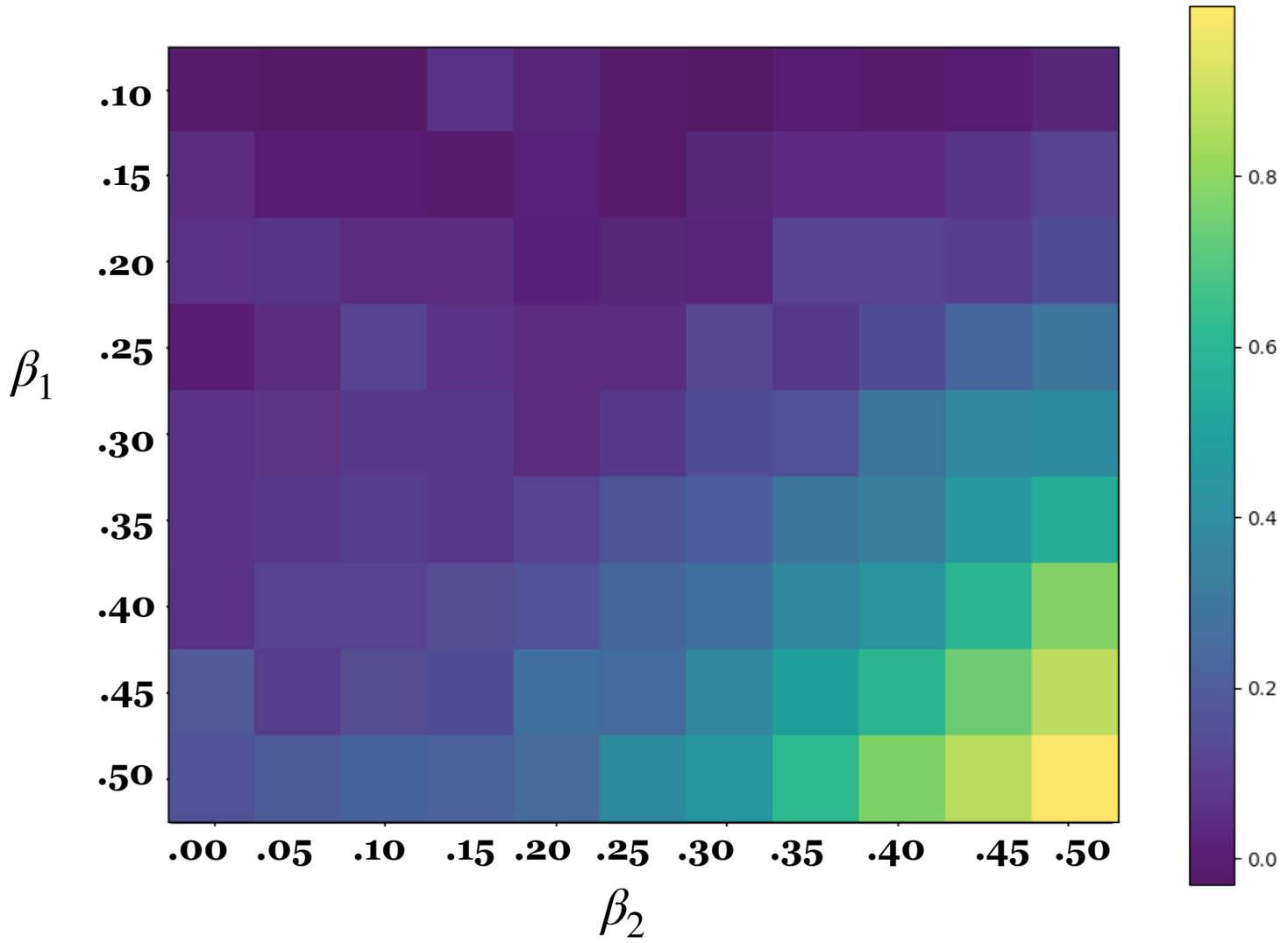
Tests of invariant physics

Similar actions



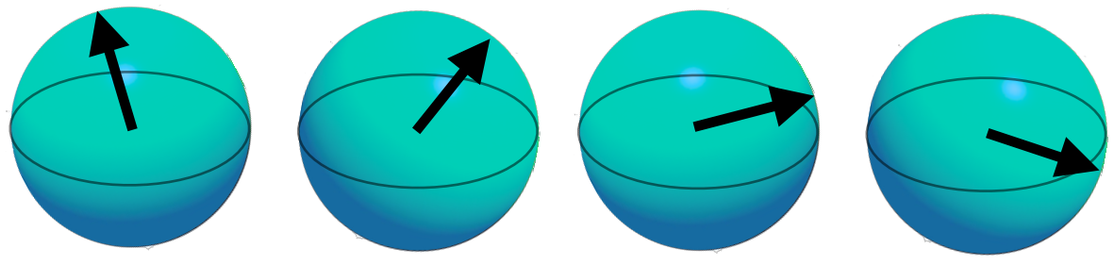
3) Compute corresponding "truncated" actions: $\{\beta_1, \beta_2\} \rightarrow \{\beta_1, 0\}$

$$\frac{C(3\sqrt{2})}{C(4\sqrt{2})}$$



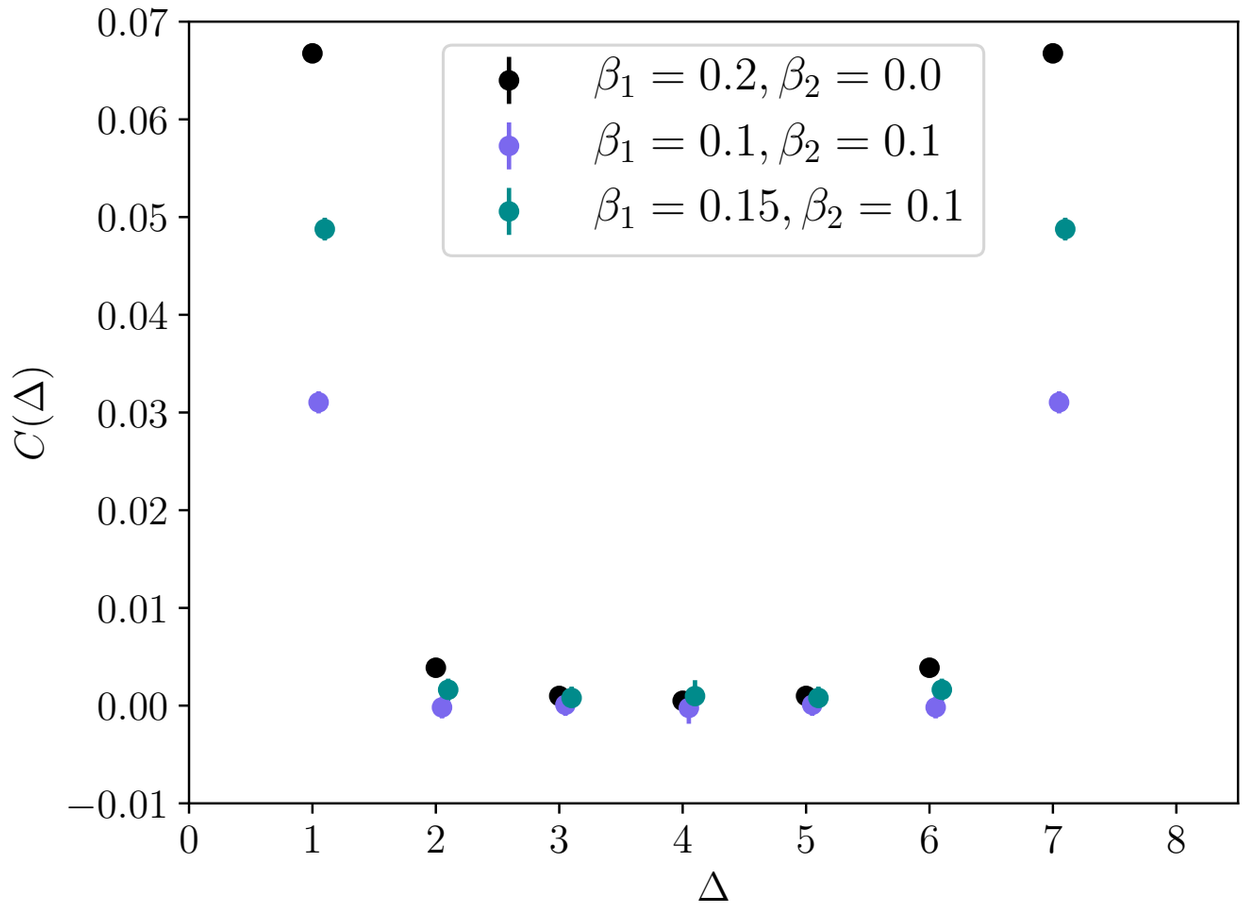
Tests of invariant physics

Similar actions



3) Compute corresponding "truncated" actions: $\{\beta_1, \beta_2\} \rightarrow \{\beta_1, 0\}$

4) Test neural networks



Future work

Future work

- **Complete tests of invariant physics**
- **Train larger 2D lattices**
- **Attempt gauge theory - Schwinger model**
- **Lattice QCD (small lattices)**
- **Cost-benefit analysis**

Thank you!

Thank you!

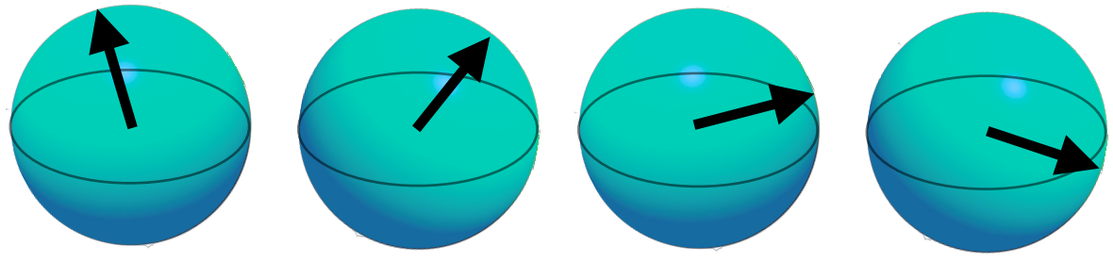
Tests of invariant physics

Monte Carlo renormalization group

Canonical demon algorithm

$$S = -\sum \beta_\alpha \, S_\alpha$$

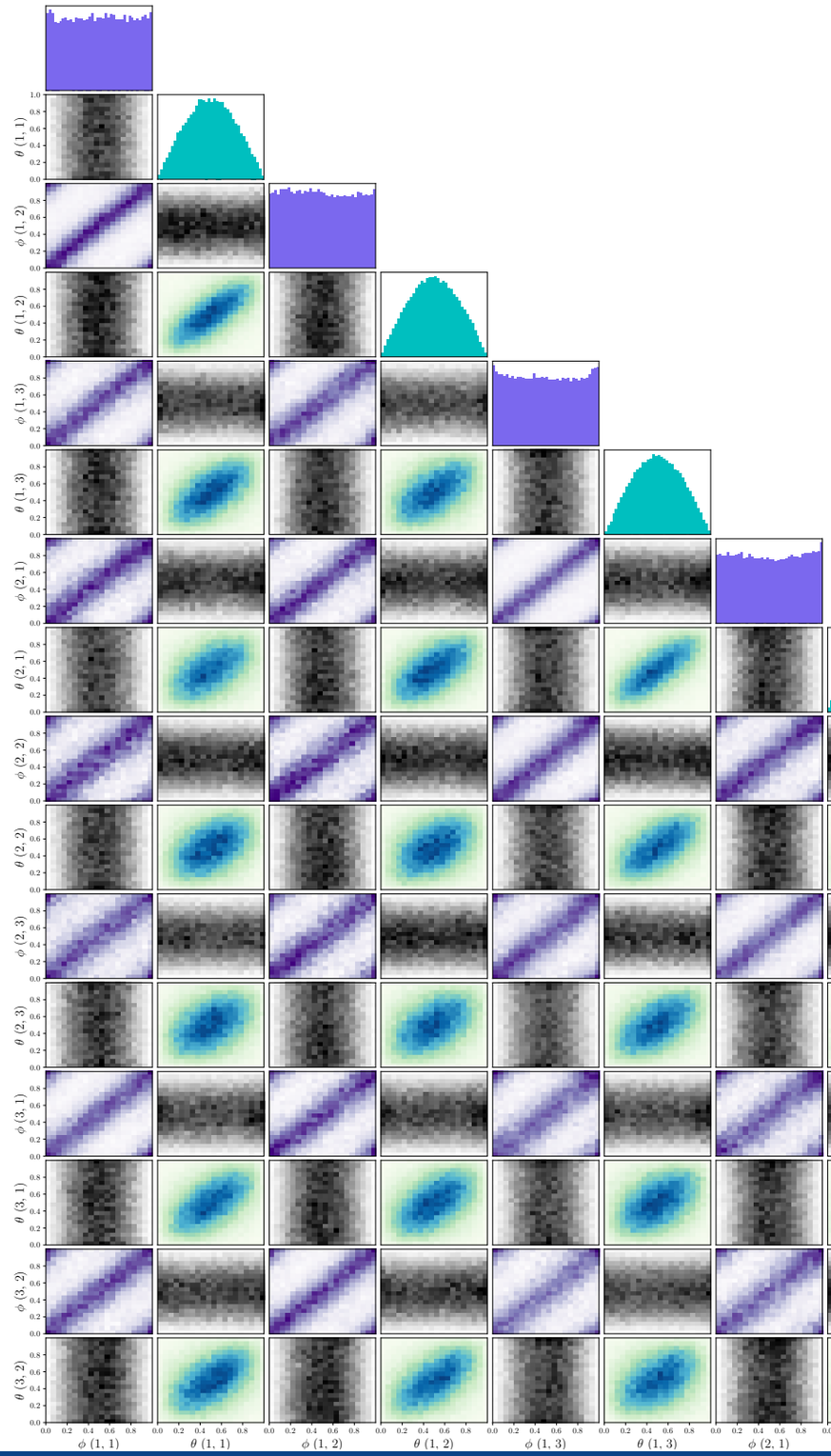
$$S_D = +\sum \beta_\alpha \, d_\alpha \quad d \in [0, d_{\max}]$$



M. Hasenbusch et al., Phys Lett B 338 (1994) 308-312

Micro-canonical MCMC with $S_{\text{tot}} = S + S_D \;:\; \Delta S_{\text{tot}} = 0$

$$Z = \left(\Pi_\alpha \int_0^{d_{\max}} \mathrm{d}d_\alpha\right) \int \mathrm{d}\phi \; \mathrm{e}^{-(S+S_D)} \qquad \Rightarrow \qquad \langle d_\alpha \rangle = \frac{1}{\beta_\alpha}\left(1 - \frac{\beta_\alpha d_{\max}}{\mathrm{e}^{\beta_\alpha d_{\max}} - 1}\right)$$

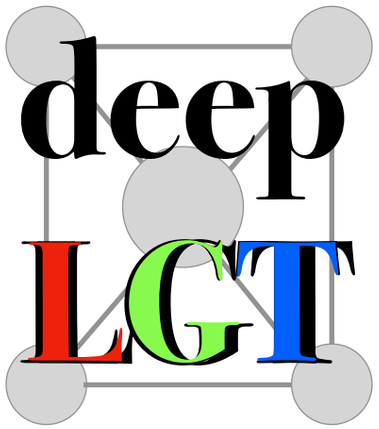


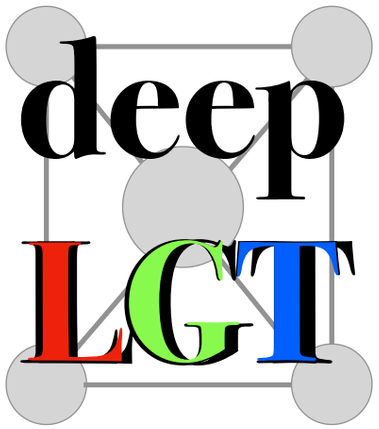
3x3 configuration

$\beta = 1.0$

$\phi \in 2\pi \times [0,1] = [0,2\pi]$

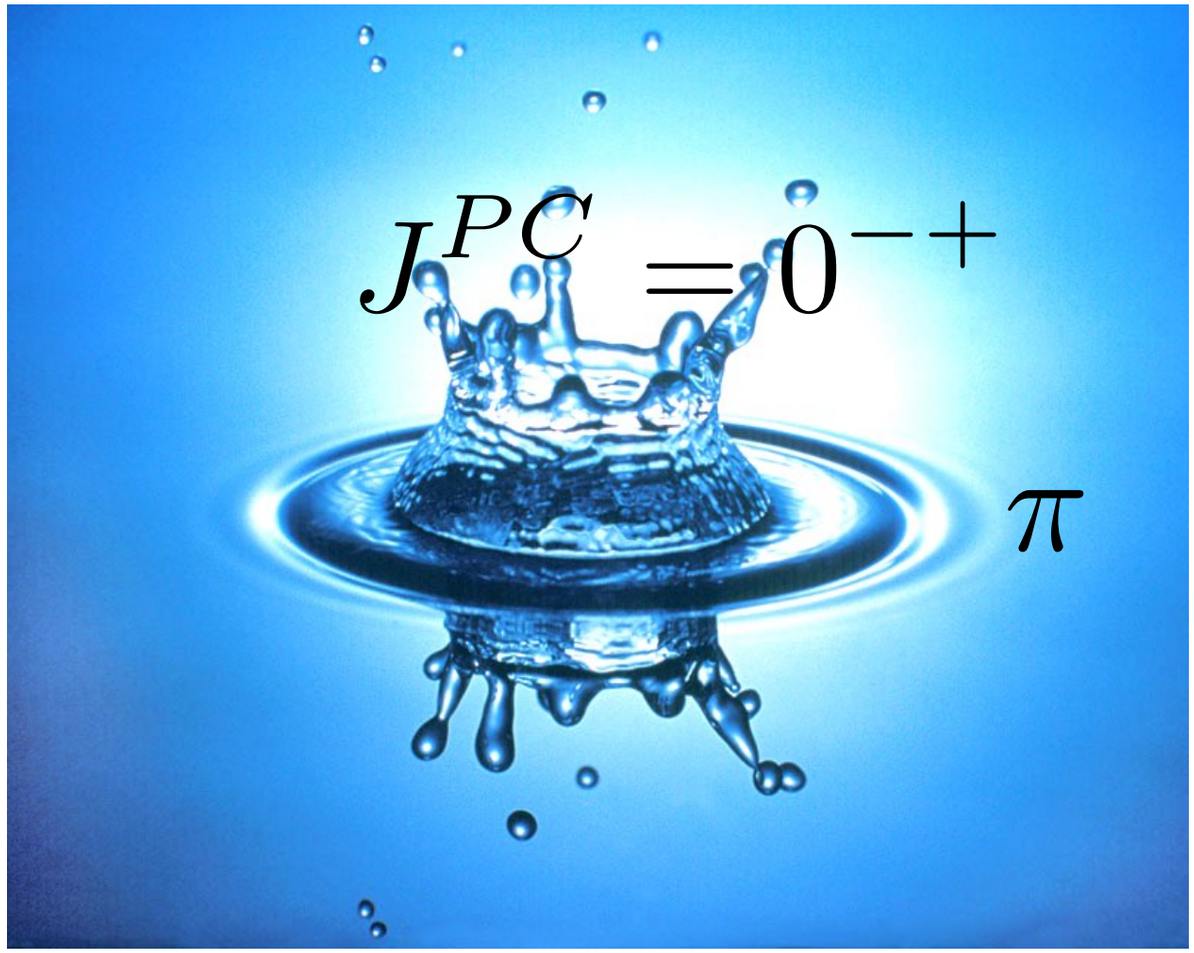
$\theta \in \cos^{-1}(2 \times [0,1] - 1) = [0,\pi]$


deep
LGT


deep
LGT

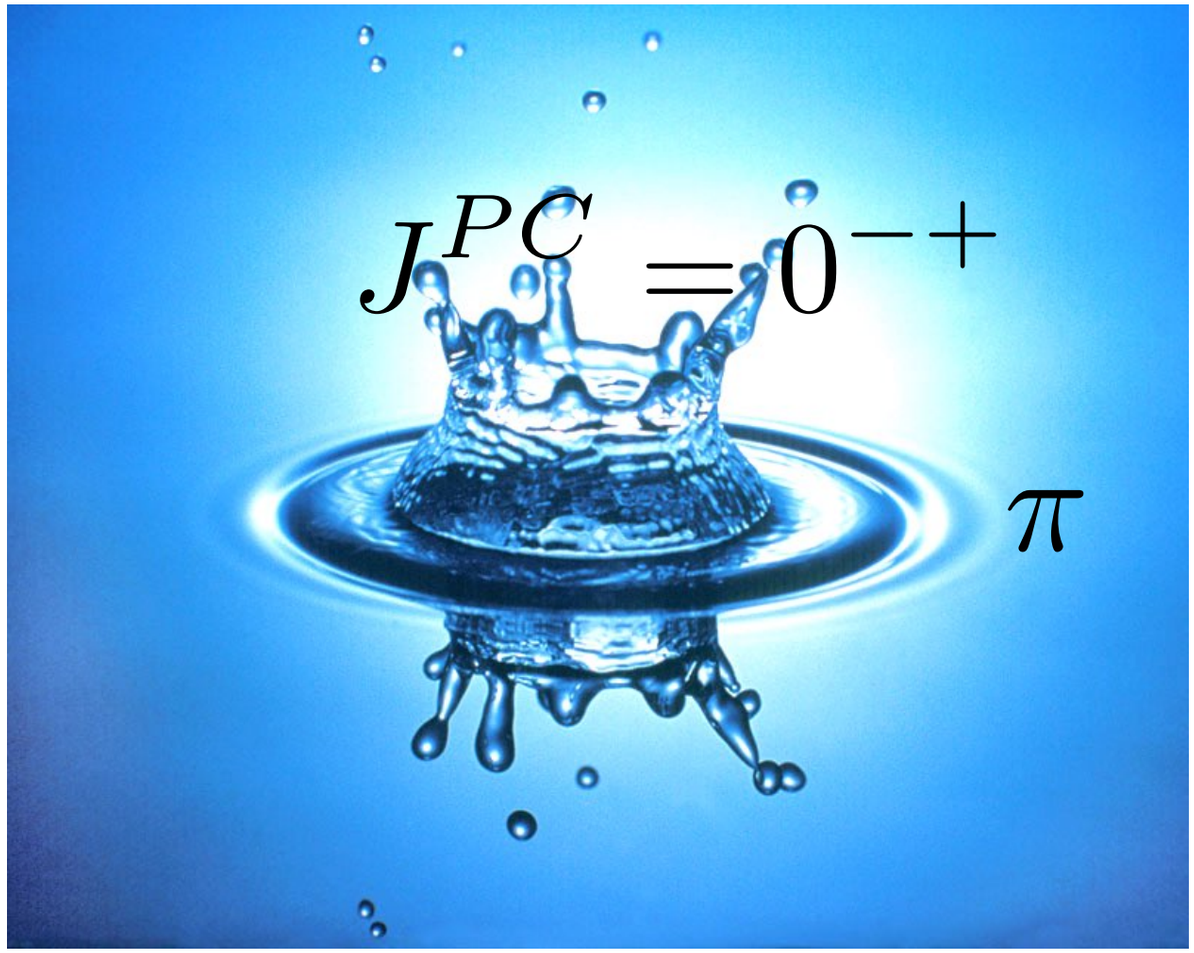
Correlation functions

$$C(t) = \langle 0|\bar{\pi}(t)\pi(0)|0\rangle$$



Correlation functions

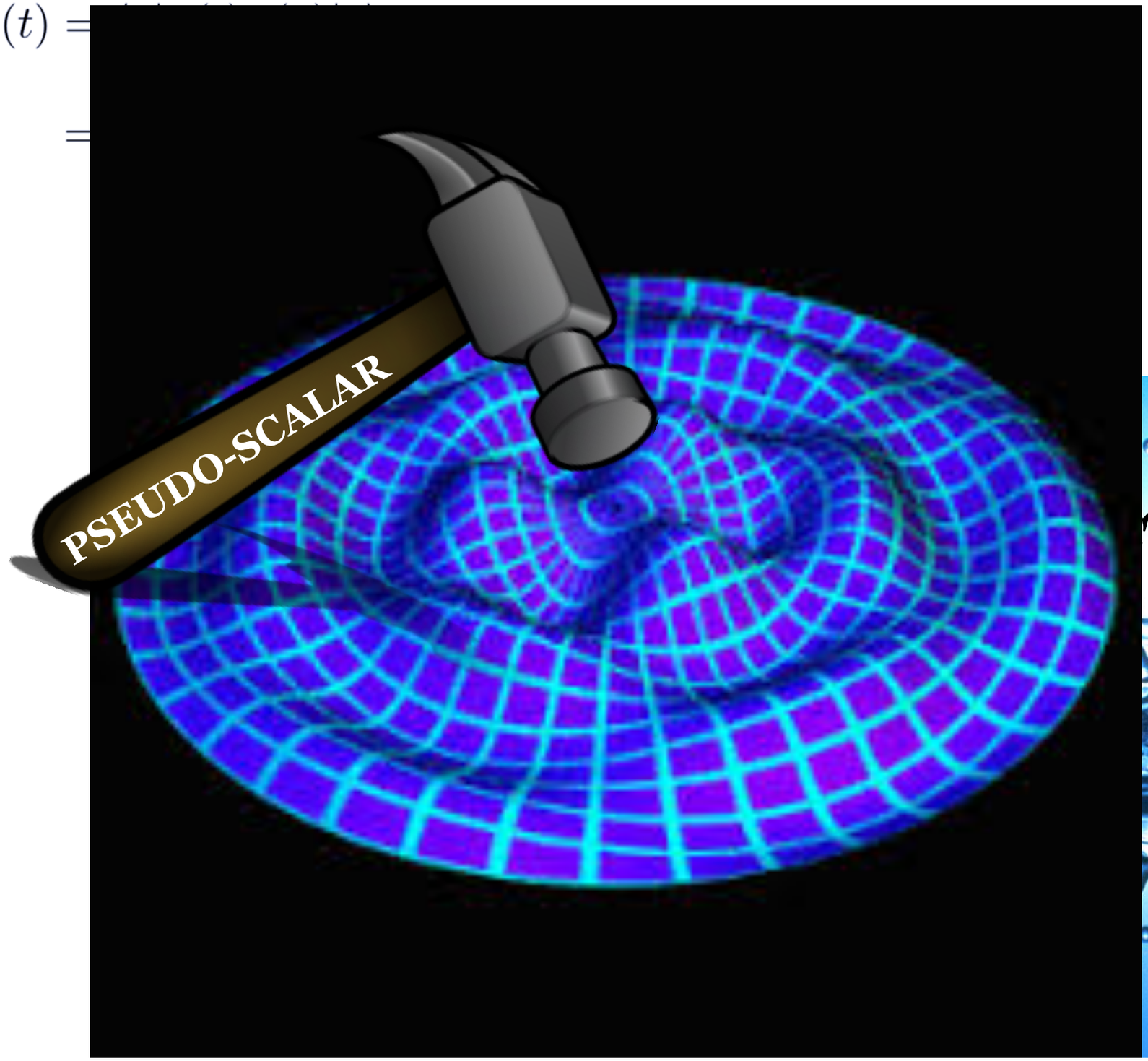
$$C(t) = \langle 0|\bar{\pi}(t)\pi(0)|0\rangle$$
$$= \sum_{m}^{\infty} \langle 0|\pi(0)|E_m\rangle \mathrm{e}^{-E_m t} \langle E_m|\pi(0)|0\rangle$$

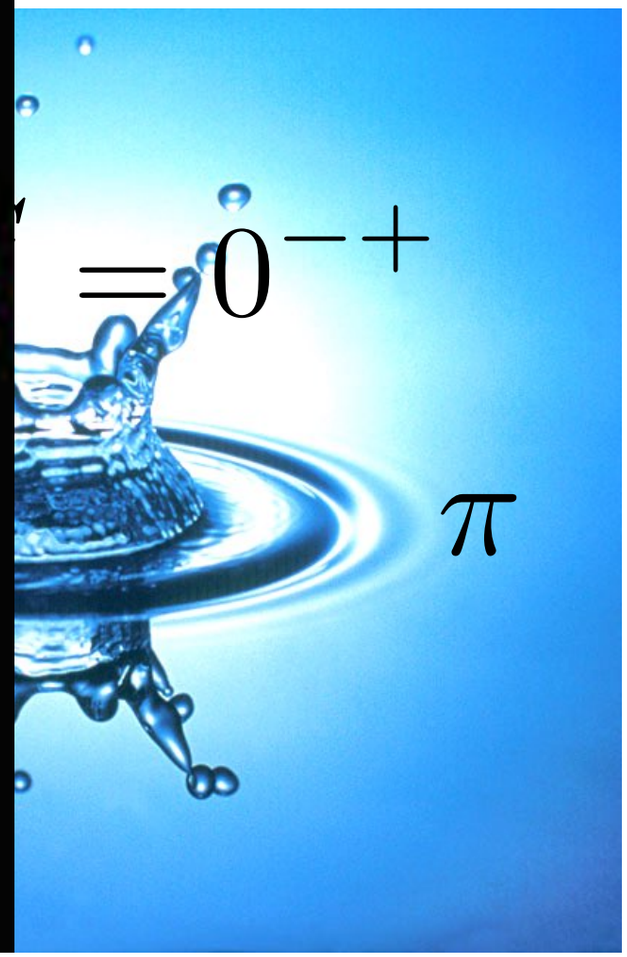


Correlation functions

$$C(t) =$$

$$=$$





http://watersoundimage.yolasite.com/what-is-a-w-s-image.php

Correlation functions

$$C(t) = \langle 0|\bar{\pi}(t)\pi(0)|0\rangle$$

$$= \sum_{m}^{\infty} \langle 0|\pi(0)|E_m\rangle \mathrm{e}^{-E_m t} \langle E_m|\pi(0)|0\rangle$$

$$\boxed{\Rightarrow C(t) = \sum_{}^{\infty} a_m \, \mathrm{e}^{-E_m t}}$$