



Swansea
University

Prifysgol
Abertawe

Quantum field-theoretic machine learning and the inverse renormalization group

Dimitrios Bachtis

Outline

- 1) **Extending machine learning classification capabilities with histogram reweighting/
Interpreting machine learning functions as physical observables**
- 2) **Inverse renormalization group with machine learning**
- 3) **Quantum field-theoretic machine learning**

Each section is \cong 10 mins

Neural Networks as Physical Observables

Supervised machine learning for phase identification

In a supervised framework we can train a **machine learning algorithm** on a set of **training data**, to learn a **function $f(\cdot)$** that separates the **symmetric** and the **broken-symmetry** phases of a system.

We require:

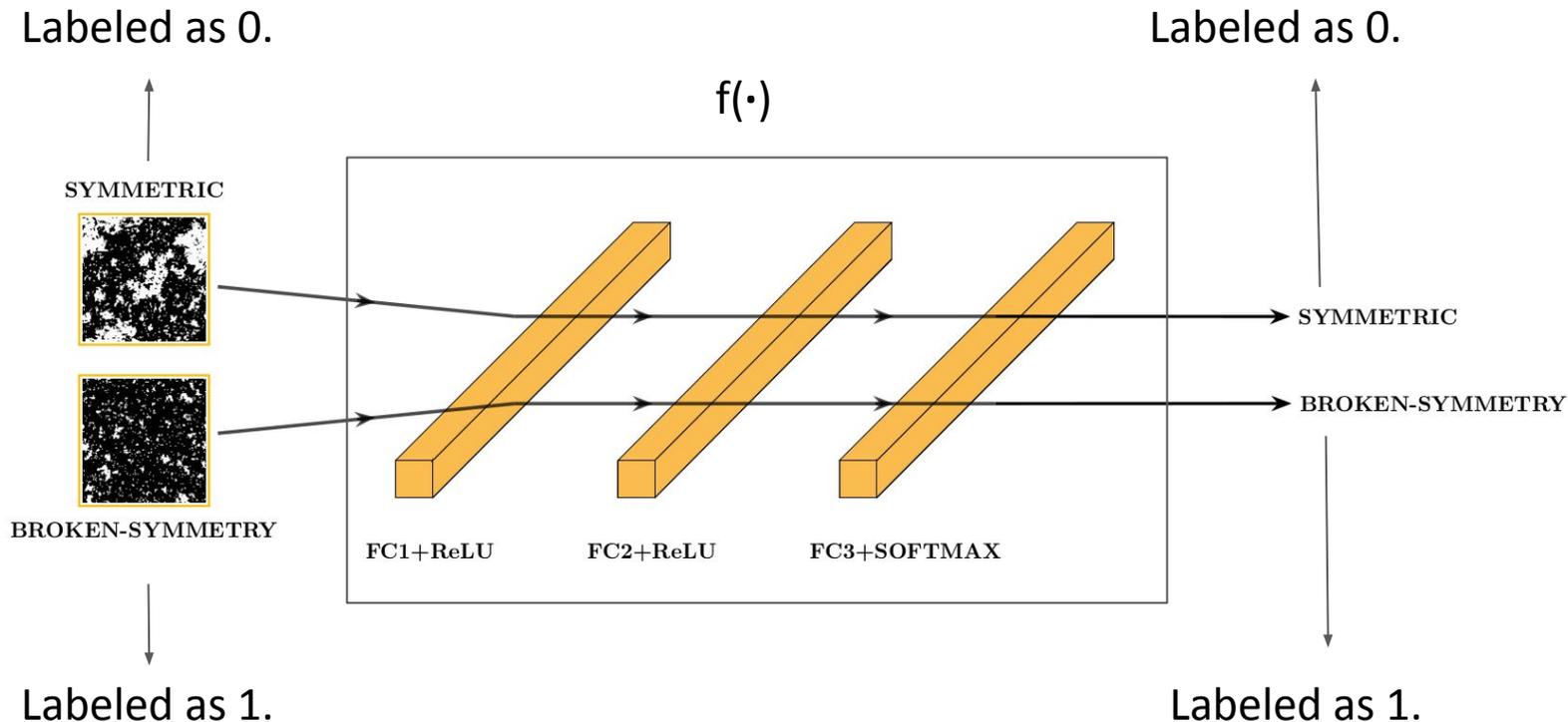
1. A set of configurations from distinct phases. Each configuration has been labeled according to the phase it belongs to.
2. A machine learning algorithm (different algorithms provide different benefits or have different limitations).

[Machine learning phases of matter](#), J. Carrasquilla, R. Melko, Nature Phys 13, 431–434 (2017)

[Machine learning and the physical sciences](#), Carleo et al., Rev. Mod. Phys. 91, 045002 (2019)

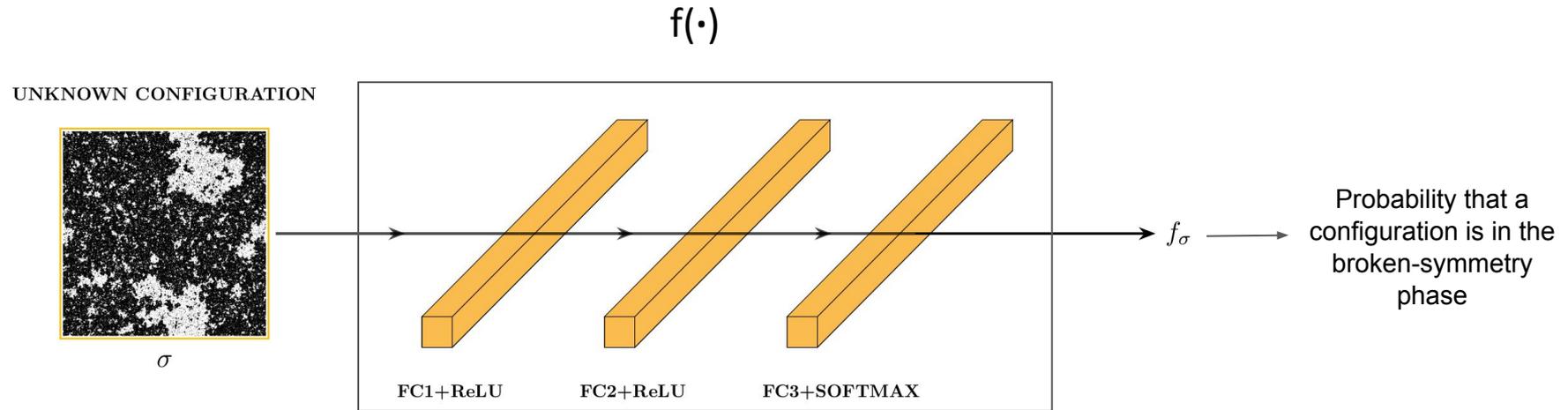
Neural Networks as Physical Observables

Training of a neural network on the Ising model:



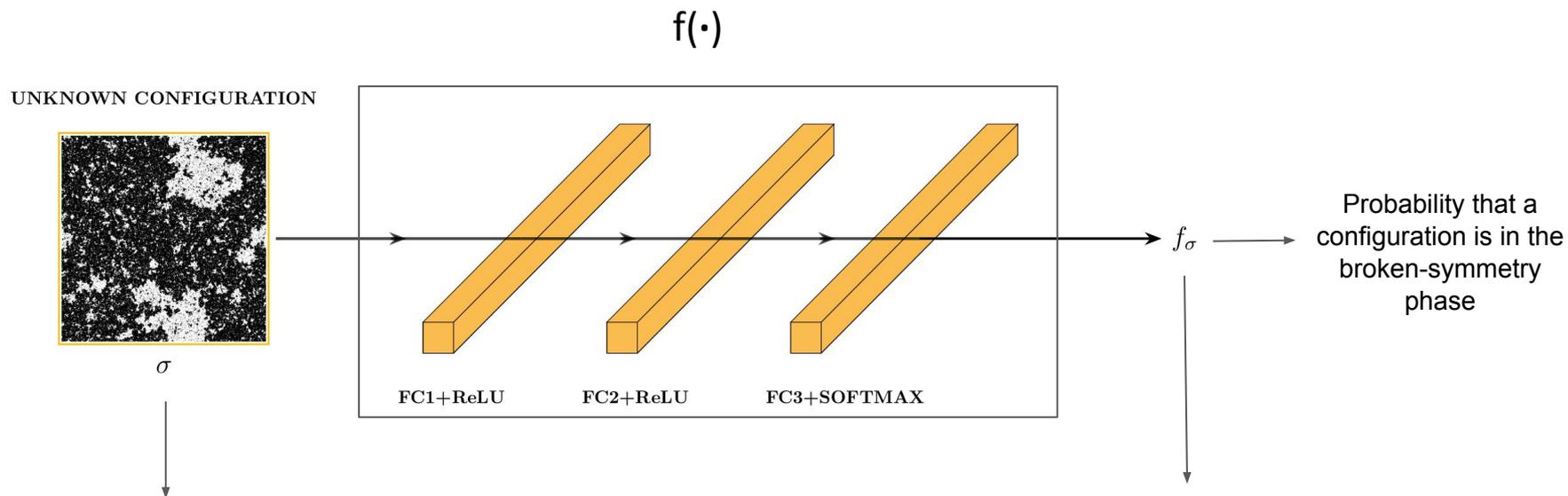
Extending machine learning classification capabilities with histogram reweighting, D. Bachtis, G. Aarts and B. Lucini, Phys. Rev. E **102** (2020).

Neural Networks as Physical Observables



Extending machine learning classification capabilities with histogram reweighting, D. Bachtis, G. Aarts and B. Lucini, Phys. Rev. E **102** (2020).

Neural Networks as Physical Observables



The **configuration** is drawn from an equilibrium distribution and therefore has an **associated Boltzmann weight**.

The **output** is calculated on the configuration so it must have the **same Boltzmann weight**.

Extending machine learning classification capabilities with histogram reweighting, D. Bachtis, G. Aarts and B. Lucini, Phys. Rev. E **102** (2020).

Neural Networks as Physical Observables

The **neural network function f** is an **observable** in the system:

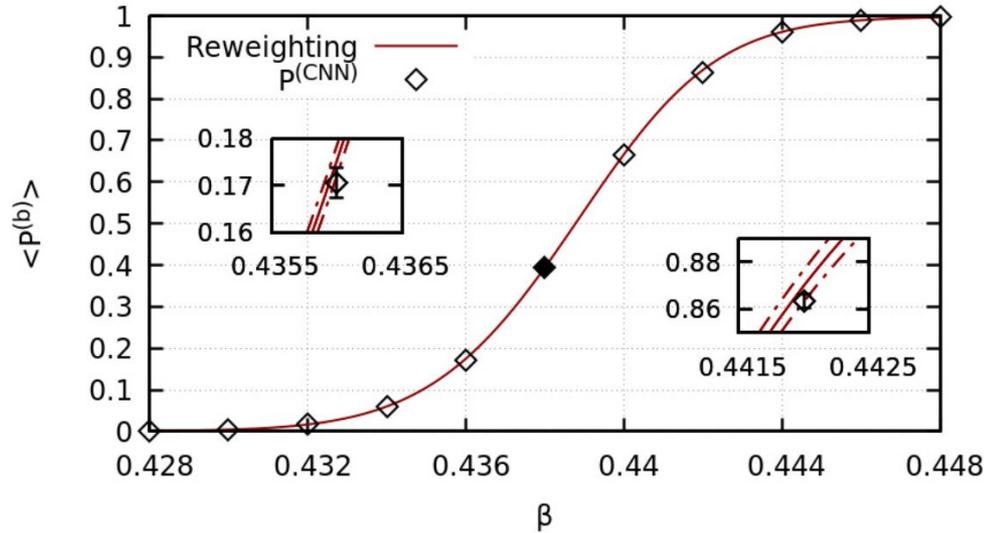
$$\langle f \rangle = \sum_{\sigma} f_{\sigma} p_{\sigma} = \frac{\sum_{\sigma} f_{\sigma} \exp[-\beta E_{\sigma}]}{\sum_{\sigma} \exp[-\beta E_{\sigma}]}$$

σ : configuration of the system

p_{σ} : Boltzmann probability distribution

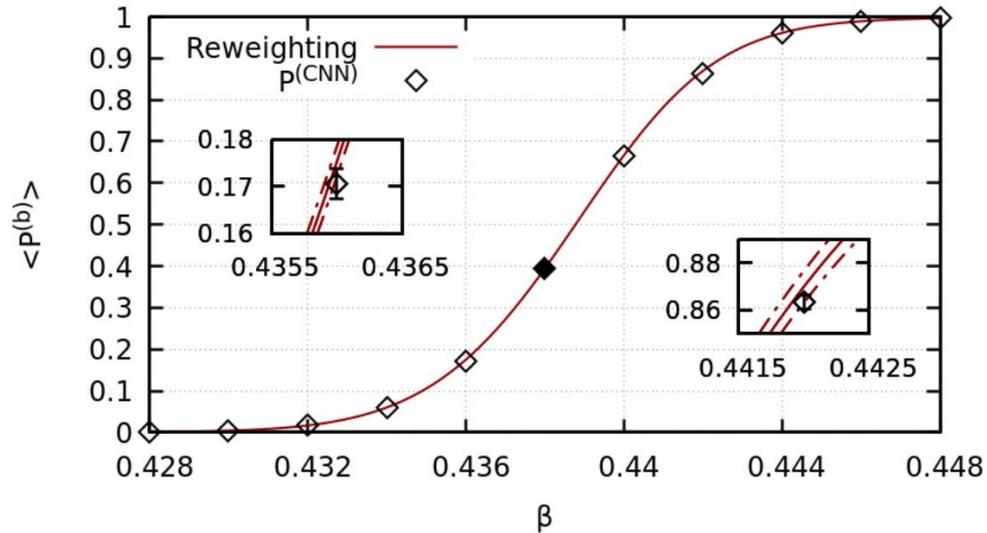
β : inverse temperature

Neural Networks as Physical Observables



Extending machine learning classification capabilities with histogram reweighting, D. Bachtis, G. Aarts and B. Lucini, Phys. Rev. E **102** (2020).

Neural Networks as Physical Observables

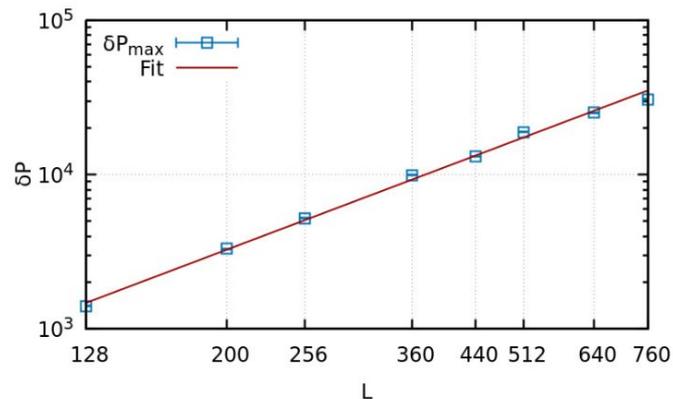
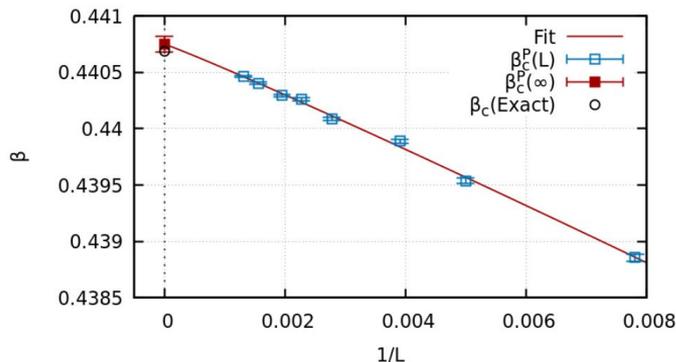


Does it look like an effective order parameter?

Extending machine learning classification capabilities with histogram reweighting, D. Bachtis, G. Aarts and B. Lucini, Phys. Rev. E **102** (2020).

Neural Networks as Physical Observables

Results obtained by quantities derived entirely from the neural network



$$|t| = \left| \frac{\beta_c - \beta_c(L)}{\beta_c} \right| \sim \xi^{-\frac{1}{\nu}} \sim L^{-\frac{1}{\nu}}$$

$$\delta P \sim L^{\frac{\gamma}{\nu}}$$

	β_c	ν	γ/ν
CNN+Reweighting	0.440749(68)	0.95(9)	1.78(4)
Exact	$\ln(1 + \sqrt{2})/2$ ≈ 0.440687	1	7/4 $=1.75$

Extending machine learning classification capabilities with histogram reweighting, D. Bachtis, G. Aarts and B. Lucini, Phys. Rev. E **102** (2020).

Neural Networks as Physical Observables

How can we explain that the neural network function is a statistical-mechanical observable?

Neural Networks as Physical Observables

Parameters, constraints or fields that interact with a system have conjugate variables which represent the response of the system to the perturbation of the corresponding parameter.

Can we make the same statement about the neural network function f ?

Neural Networks as Physical Observables

The (extensive) neural network function Vf can then be included as a term within the Hamiltonian. We consider that Vf couples to an **arbitrary external field Y** and define a modified Hamiltonian for the Ising model:

$$E_Y = E - VfY.$$

If $Y=0$, we have the original Hamiltonian of the Ising model.

Neural Networks as Physical Observables

If we take a derivative of the logarithm of the partition function in terms of the **external field Y** we arrive at the **expectation value** of the **neural network function f** :

$$\langle f \rangle = \frac{1}{\beta V} \frac{\partial \log Z_Y}{\partial Y} = \frac{\sum_{\sigma} f_{\sigma} \exp[-\beta E_{\sigma} + \beta V f_{\sigma} Y]}{\sum_{\sigma} \exp[-\beta E_{\sigma} + \beta V f_{\sigma} Y]}$$

If $Y=0$, we have the original expression of the expectation value.

Neural Networks as Physical Observables

The derivative of the expectation value of the neural network function gives:

$$\chi_f = \frac{\partial \langle f \rangle}{\partial Y} = \beta V(\langle f^2 \rangle - \langle f \rangle^2)$$

χ is a **susceptibility**. It measures the **response** of the **neural network function f** to changes in the **associated external field Y** .

Neural Networks as Physical Observables

The derivative of the expectation value of the neural network function gives:

$$\chi_f = \frac{\partial \langle f \rangle}{\partial Y} = \beta V(\langle f^2 \rangle - \langle f \rangle^2)$$

χ is a **susceptibility**. It measures the **response** of the **neural network function f** to changes in the **associated external field Y** .

What happens if $Y \neq 0$?

Neural Networks as Physical Observables

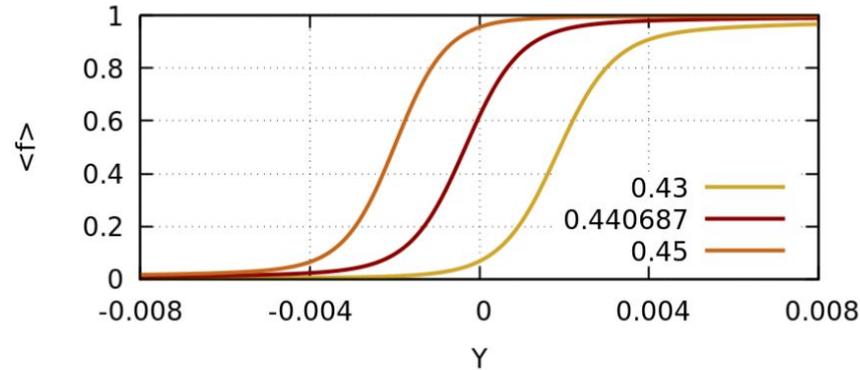


FIG. 2. Mean neural network function $\langle f \rangle$ versus external field Y at inverse temperature $\beta = 0.43, 0.440687, 0.45$ (right to left). The statistical uncertainty is comparable with the width of the lines.

Recall that:

$\beta=0.43 \rightarrow$ symmetric phase

$\beta_c \approx 0.440687 \rightarrow$ inverse critical temperature

$\beta=0.45 \rightarrow$ broken-symmetry phase

Neural Networks as Physical Observables

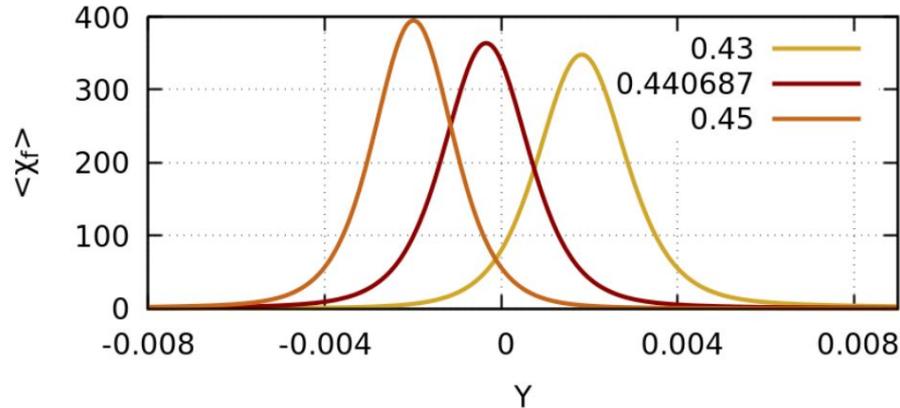


FIG. 3. Mean susceptibility of the neural network function $\langle \chi_f \rangle$ versus external field Y at inverse temperature $\beta = 0.43, 0.440687, 0.45$ (right to left). The statistical uncertainty is comparable with the width of the lines.

Recall that the inverse critical temperature is $\beta_c \approx 0.440687$.

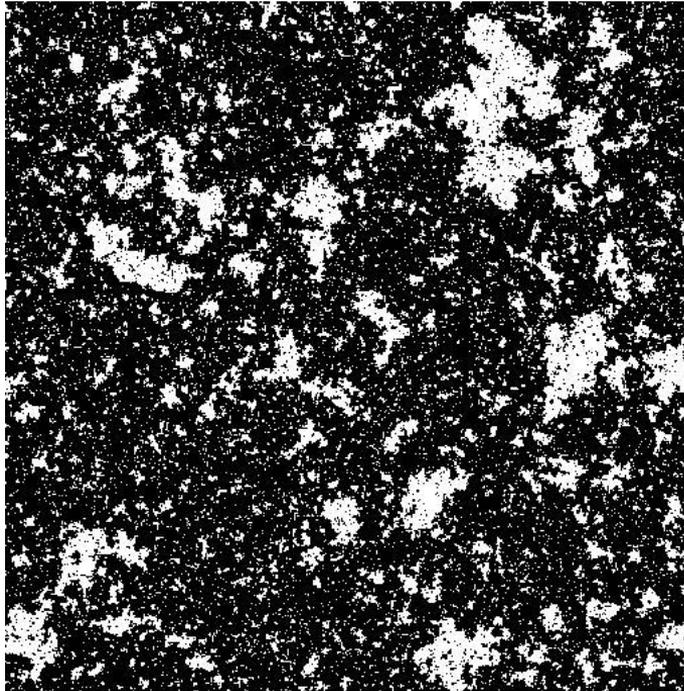
Adding machine learning within Hamiltonians: Renormalization group transformations, symmetry breaking and restoration, D. Bachtis, G. Aarts and B. Lucini, Phys. Rev. Research 3, 013134 (arXiv:2010.00054).

Neural Networks as Physical Observables

Can we study the phase transition induced by the neural network field Y based on a renormalization group approach?

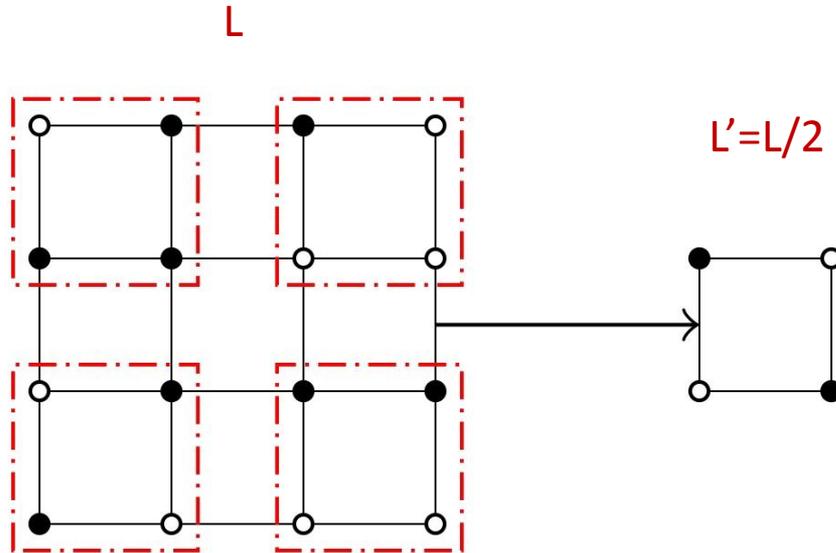
Neural Networks as Physical Observables

L



Adding machine learning within Hamiltonians: Renormalization group transformations, symmetry breaking and restoration, D. Bachtis, G. Aarts and B. Lucini, Phys. Rev. Research 3, 013134 (arXiv:2010.00054).

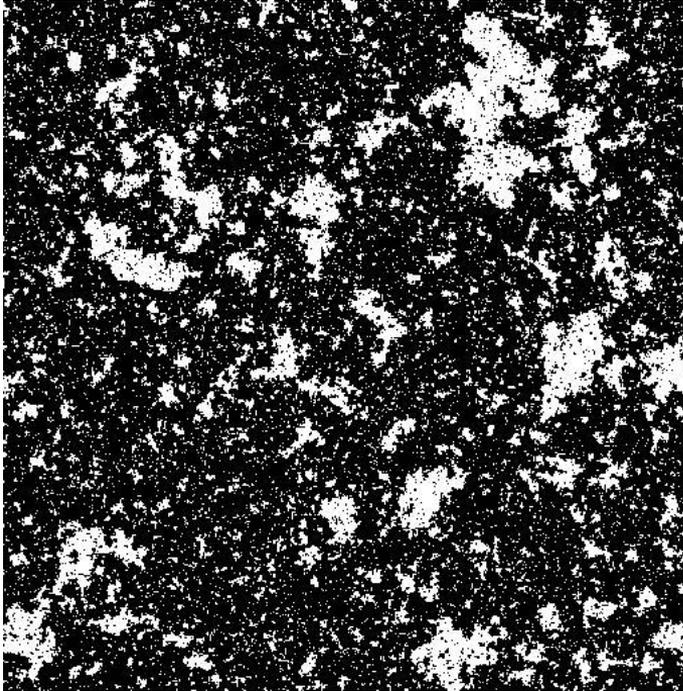
Neural Networks as Physical Observables



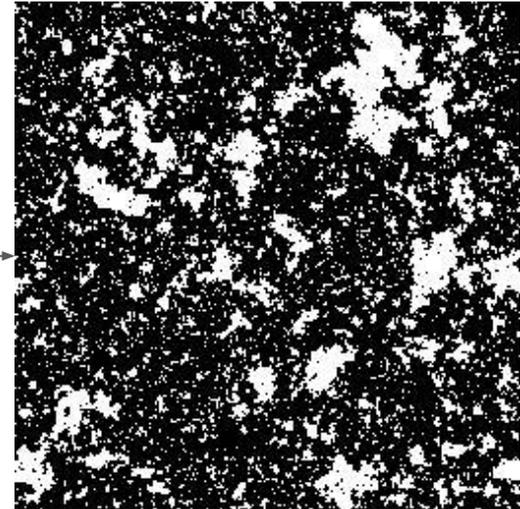
Spin blocking transformation with a **rescaling factor** of $b=2$ and the **majority rule**

Neural Networks as Physical Observables

L



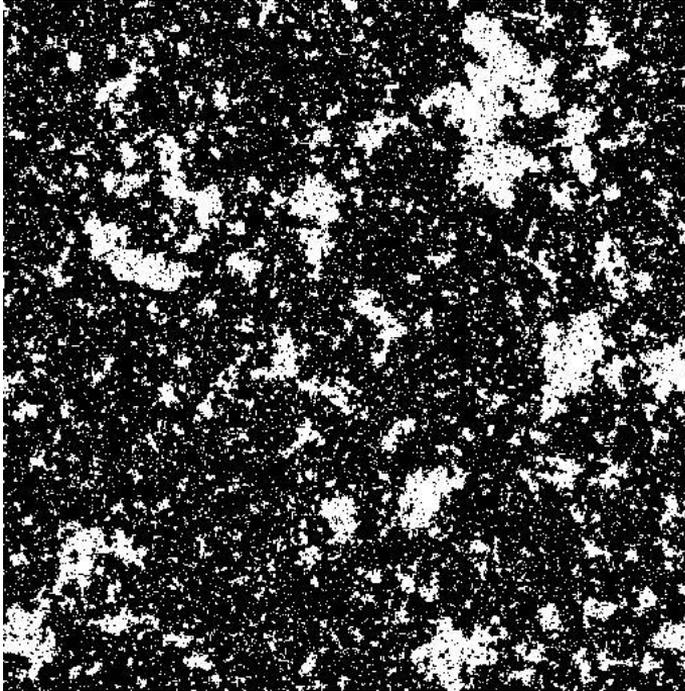
$L'=L/2$



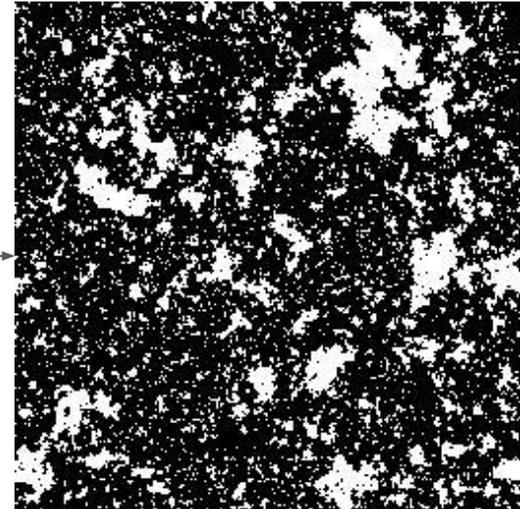
Adding machine learning within Hamiltonians: Renormalization group transformations, symmetry breaking and restoration, D. Bachtis, G. Aarts and B. Lucini, Phys. Rev. Research 3, 013134 (arXiv:2010.00054).

Neural Networks as Physical Observables

L, ξ



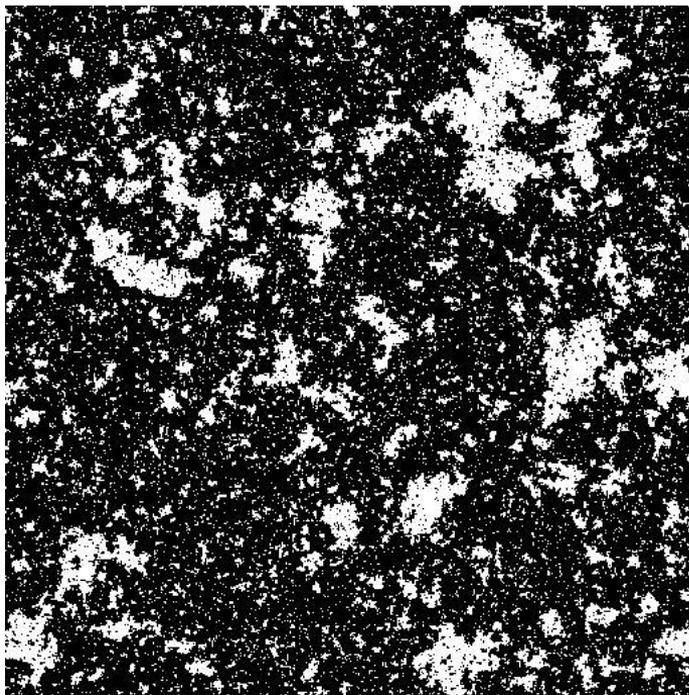
$L'=L/2, \xi'=\xi/2$



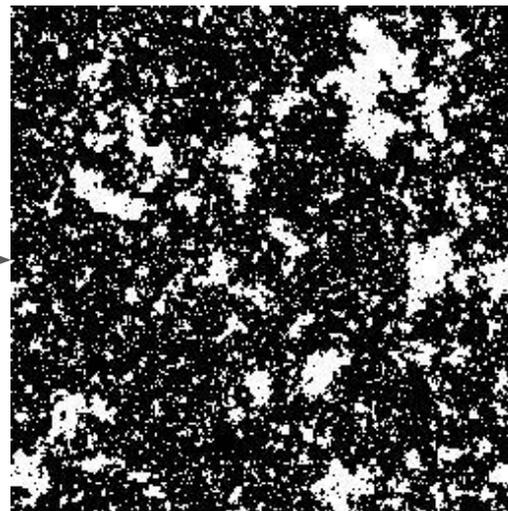
Adding machine learning within Hamiltonians: Renormalization group transformations, symmetry breaking and restoration, D. Bachtis, G. Aarts and B. Lucini, Phys. Rev. Research 3, 013134 (arXiv:2010.00054).

Neural Networks as Physical Observables

L, ξ, β

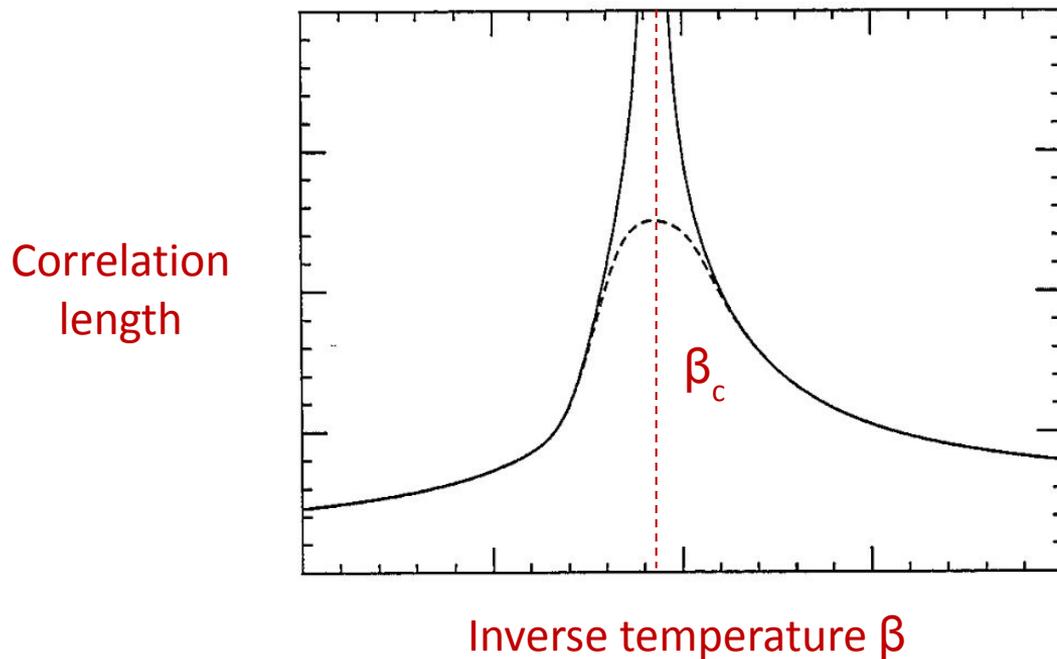


$L'=L/2, \xi'=\xi/2, \beta'$



Adding machine learning within Hamiltonians: Renormalization group transformations, symmetry breaking and restoration, D. Bachtis, G. Aarts and B. Lucini, Phys. Rev. Research 3, 013134 (arXiv:2010.00054).

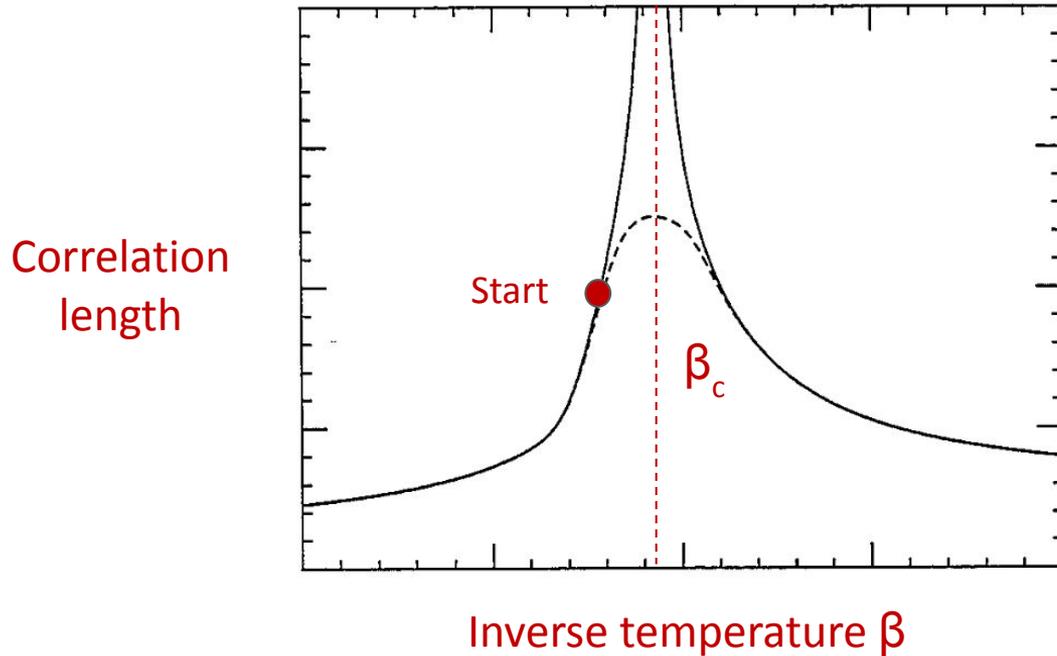
Neural Networks as Physical Observables



Altered figure from (Newman, Barkema) book (Fig 4.1)

Adding machine learning within Hamiltonians: Renormalization group transformations, symmetry breaking and restoration, D. Bachtis, G. Aarts and B. Lucini, Phys. Rev. Research 3, 013134 (arXiv:2010.00054).

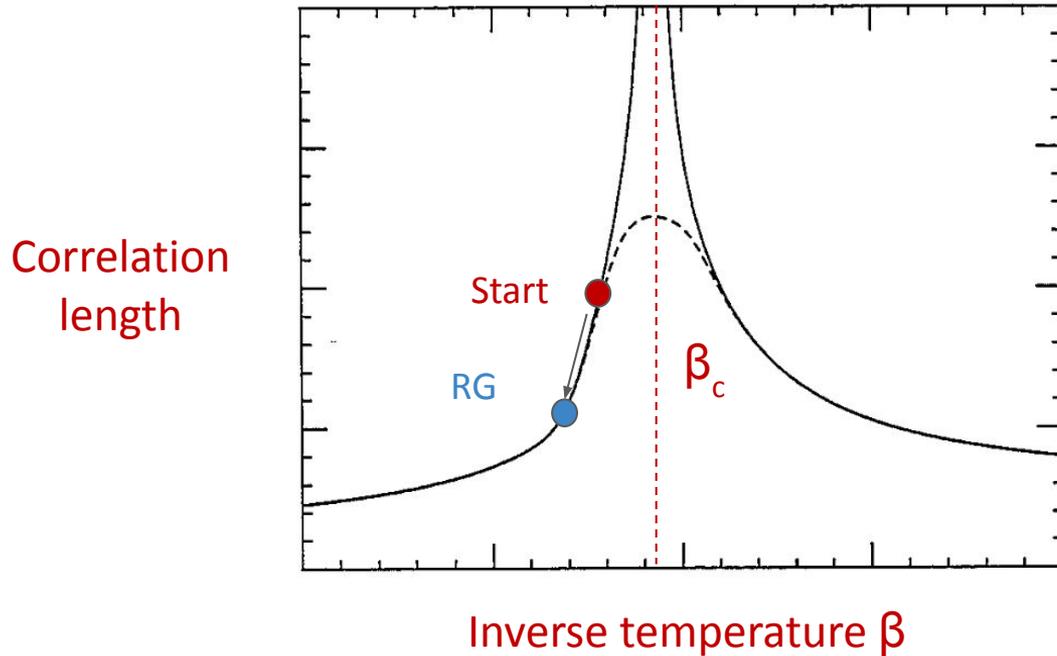
Neural Networks as Physical Observables



Altered figure from (Newman, Barkema) book (Fig 4.1)

Adding machine learning within Hamiltonians: Renormalization group transformations, symmetry breaking and restoration, D. Bachtis, G. Aarts and B. Lucini, Phys. Rev. Research 3, 013134 (arXiv:2010.00054).

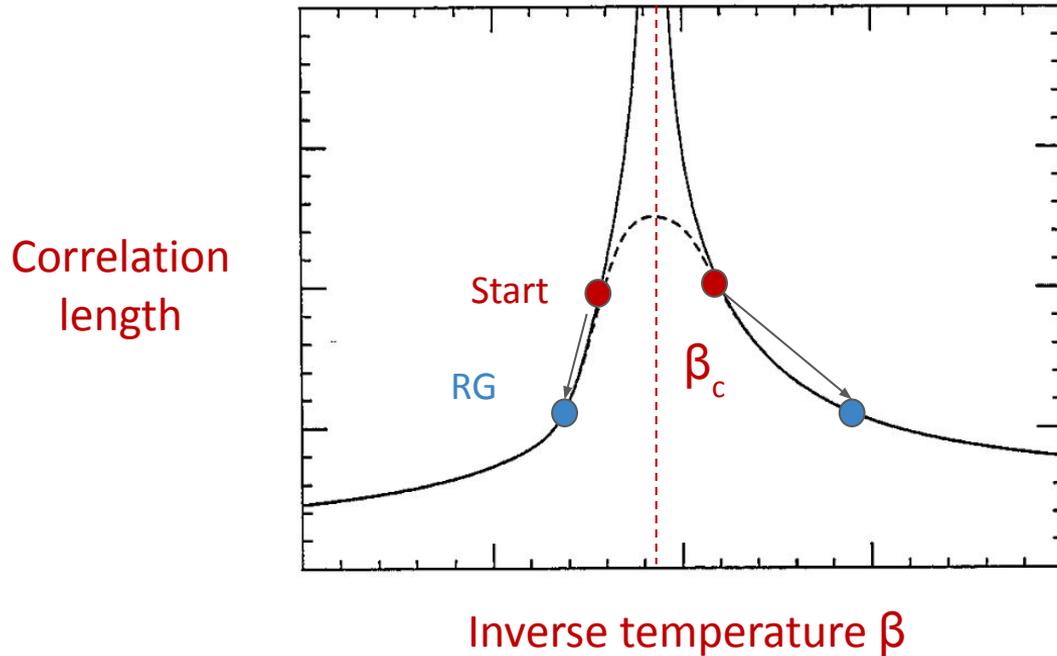
Neural Networks as Physical Observables



Altered figure from (Newman, Barkema) book (Fig 4.1)

Adding machine learning within Hamiltonians: Renormalization group transformations, symmetry breaking and restoration, D. Bachtis, G. Aarts and B. Lucini, Phys. Rev. Research 3, 013134 (arXiv:2010.00054).

Neural Networks as Physical Observables



Altered figure from (Newman, Barkema) book (Fig 4.1)

Adding machine learning within Hamiltonians: Renormalization group transformations, symmetry breaking and restoration, D. Bachtis, G. Aarts and B. Lucini, Phys. Rev. Research 3, 013134 (arXiv:2010.00054).

Neural Networks as Physical Observables

There is one inverse temperature where the original and the rescaled systems have the same correlation length: **the inverse critical temperature $\beta_c=0.440687$** .

At the inverse critical temperature β_c the correlation length diverges, it becomes infinite, and **intensive observable quantities** of the two systems will become equal.

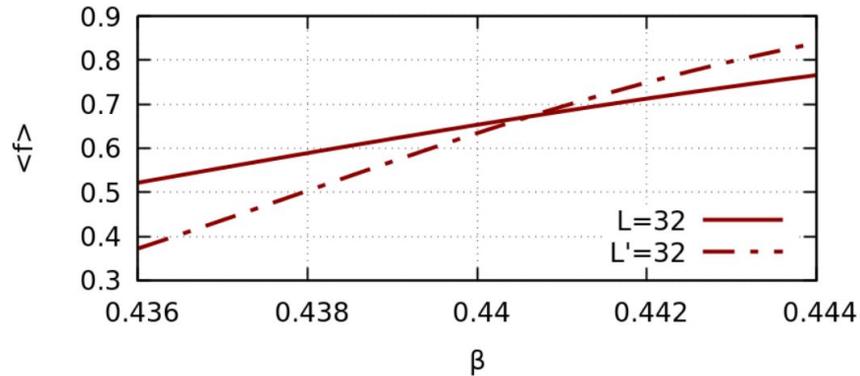
Neural Networks as Physical Observables

There is one inverse temperature where the original and the rescaled systems have the same correlation length: **the inverse critical temperature $\beta_c=0.440687$.**

At the inverse critical temperature β_c the correlation length diverges, it becomes infinite, and **intensive observable quantities** of the two systems will become equal.

We can use the neural network function f as an observable to locate the critical point.

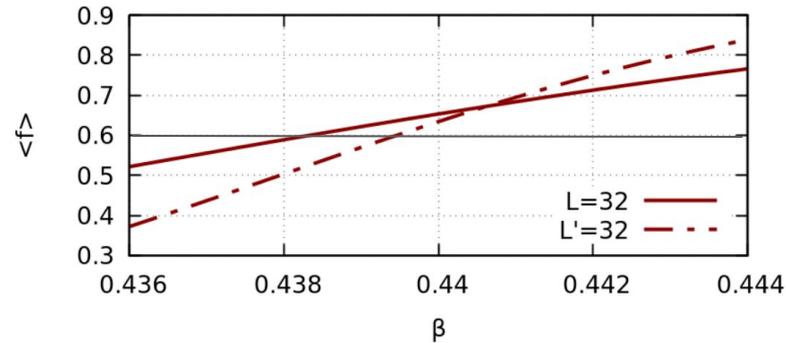
Neural Networks as Physical Observables



At the intersection point:

$$f(\beta_c) = f'(\beta_c)$$

Neural Networks as Physical Observables



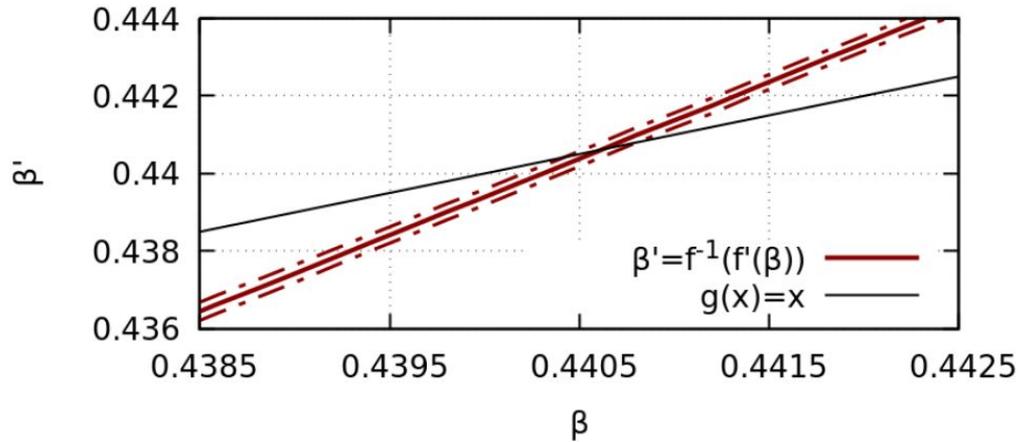
More generally:

$$f(\beta') = f'(\beta)$$

We can form a **mapping** between the rescaled and the original inverse temperature:

$$\beta' = f^{-1}(f'(\beta))$$

Neural Networks as Physical Observables



$$\beta_c = 0.44063(21)$$

Neural Networks as Physical Observables

The original and the rescaled systems **have a different distance from the critical point.**

This distance can be measured by defining the **reduced inverse temperature** for the original and the rescaled system:

$$t = \frac{\beta_c - \beta}{\beta_c}$$

Original

$$t' = \frac{\beta_c - \beta'}{\beta_c}$$

Rescaled

Neural Networks as Physical Observables

The original and the rescaled systems have **different correlation lengths**.

They should therefore diverge to the thermodynamic limit according to different relations:

$$\xi \sim |t|^{-\nu}$$

Original

$$\xi' \sim |t'|^{-\nu}$$

Rescaled

The correlation length exponent is the same because both the original and the rescaled systems are Ising models.

Neural Networks as Physical Observables

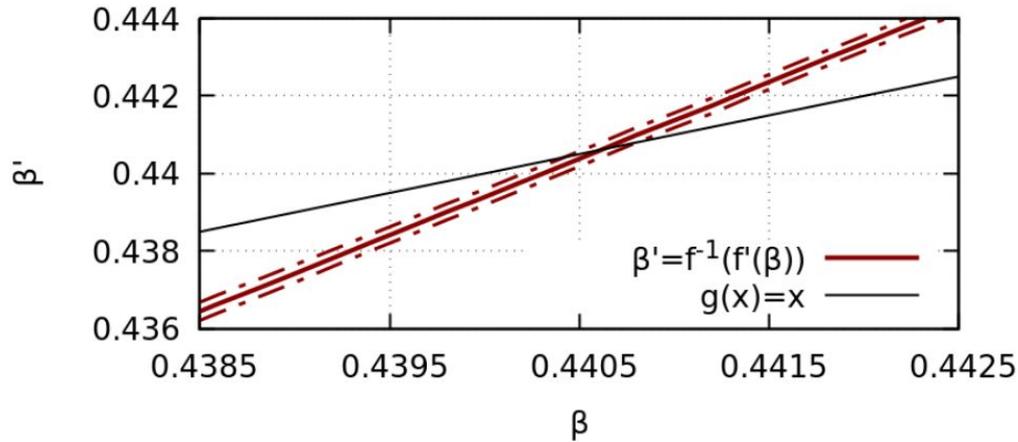
By dividing the two relations of the correlation lengths we obtain:

$$\left(\frac{t}{t'}\right)^{-\nu} = b.$$

We then substitute and **linearize the renormalization group transformation** based on a Taylor expansion to leading order, to obtain:

$$\nu = \frac{\log b}{\log \left. \frac{d\beta'}{d\beta} \right|_{\beta_c}}$$

Neural Networks as Physical Observables



$$\nu = \frac{\log b}{\log \left. \frac{d\beta'}{d\beta} \right|_{\beta_c}}$$

$$\beta_c = 0.44063(21)$$

$$\nu = 1.01(2)$$

Neural Networks as Physical Observables

The neural network field Y induces a phase transition.

Then Y affects the correlation length. Another exponent can be defined that governs the divergence of the correlation length precisely at the critical point:

$$\xi \sim |Y|^{-\theta_Y}$$

Neural Networks as Physical Observables

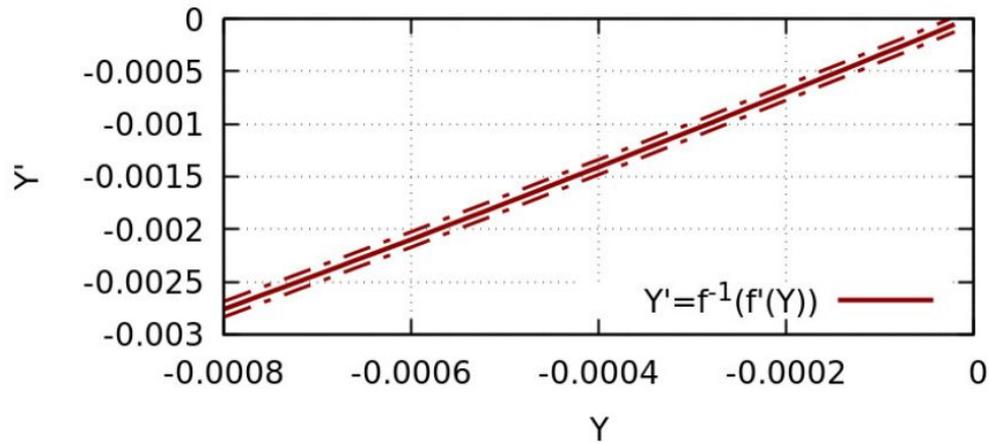
Similarly to the inverse temperatures a mapping can be formed that relates **the original and the rescaled neural network field**:

$$Y' = f^{-1}(f'(Y))$$

A new expression can be obtained that allows numerical calculation of the exponent θ_y at the vicinity of the phase transition:

$$\theta_Y = \frac{\log b}{\log \left. \frac{dY'}{dY} \right|_{Y=0}}$$

Neural Networks as Physical Observables



$$\theta_Y = 0.534(3)$$

Inverse renormalization group

Can we devise an **inverse renormalization group** approach that can be applied for an arbitrary number of steps to iteratively increase the lattice size of the system?

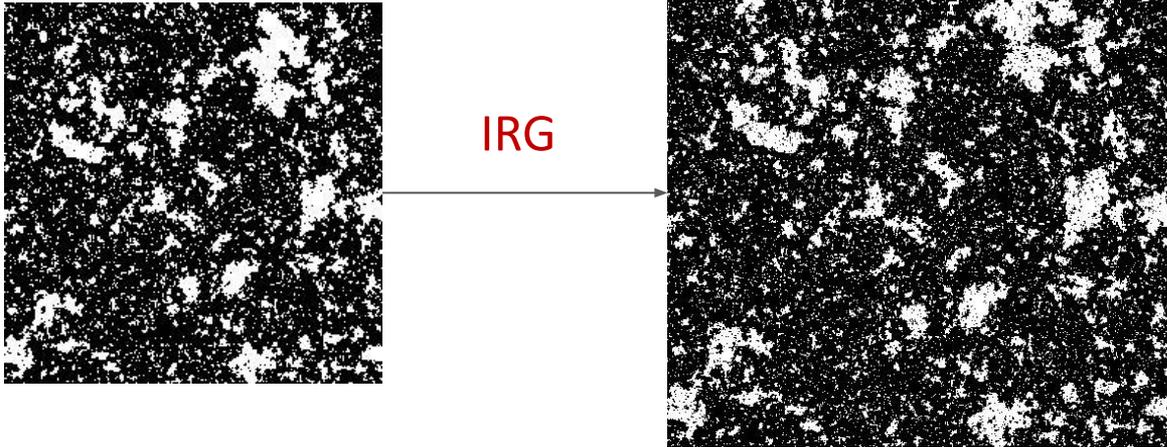
Inverse renormalization group

Can we devise an **inverse renormalization group** approach that can be applied for an arbitrary number of steps to iteratively increase the lattice size of the system?

If yes, then we can obtain configurations of systems with larger lattice size without simulating them, hence evading the critical slowing down effect.

Inverse renormalization group

In the inverse renormalization group new degrees of freedom will be introduced within the system.

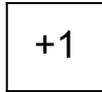


Inverse Renormalization Group in Quantum Field Theory, D. Bachtis, G. Aarts, F. Di Renzo, B. Lucini, Phys. Rev. Lett. 128, 081603 (2022)

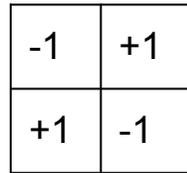
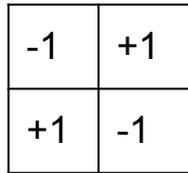
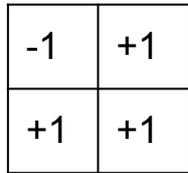
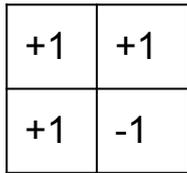
Inverse renormalization group

Inversion of a majority rule in the **Ising** model

Original degree of freedom



Possible rescaled degrees of freedom



...

For the inverse renormalization group in the Ising model, see:

[Inverse Monte Carlo Renormalization Group Transformations for Critical Phenomena](#), D. Ron, R. Swendsen, A. Brandt, Phys. Rev. Lett. 89, 275701 (2002)

Inverse renormalization group

Inversion of a summation in the ϕ^4 model

Original degree of freedom

0.40

Possible rescaled degrees of freedom

0.01	0.36
0.02	0.01

-421.1	90.1
0.5	330.9

...

Inverse renormalization group

We can learn a set of transformations that can mimic the inversion of a standard renormalization group transformation.

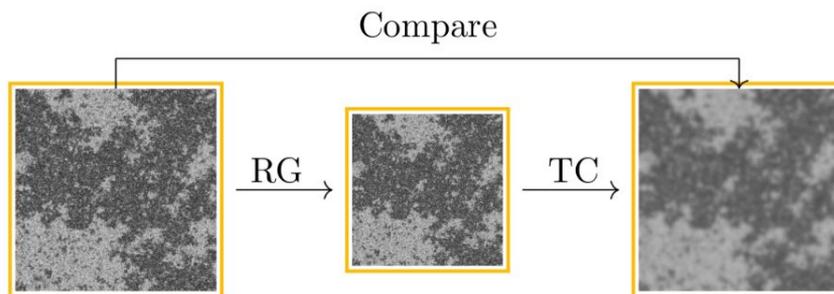


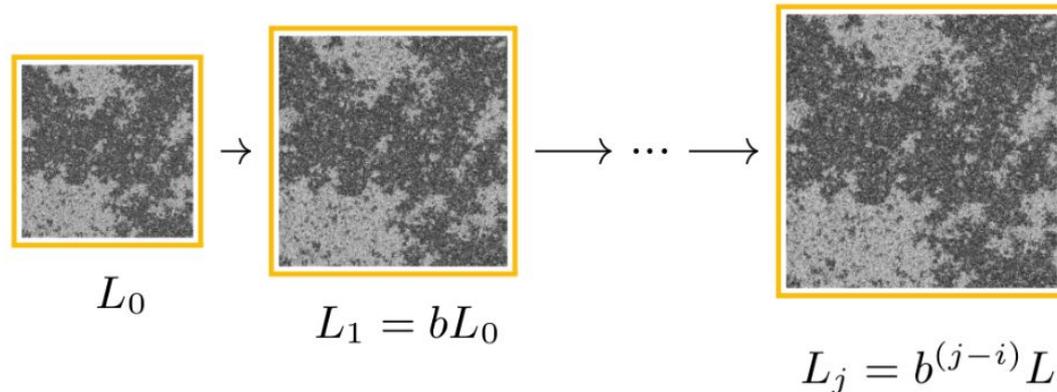
FIG. 3. Illustration of the optimization approach. Transposed convolutions (TC) are applied on configurations produced with the renormalization group (RG) to construct a set of configuration which is compared with the original.

Inverse Renormalization Group in Quantum Field Theory, D. Bachtis, G. Aarts, F. Di Renzo, B. Lucini, *Phys. Rev. Lett.* 128, 081603 (2022)

Inverse renormalization group

The benefit:

Once learned, we can apply this set of inverse transformations iteratively to arbitrarily increase the size of the system.



Inverse Renormalization Group in Quantum Field Theory, D. Bachtis, G. Aarts, F. Di Renzo, B. Lucini, Phys. Rev. Lett. 128, 081603 (2022)

Inverse renormalization group

The set of transformations can be applied iteratively to arbitrarily increase the lattice size:

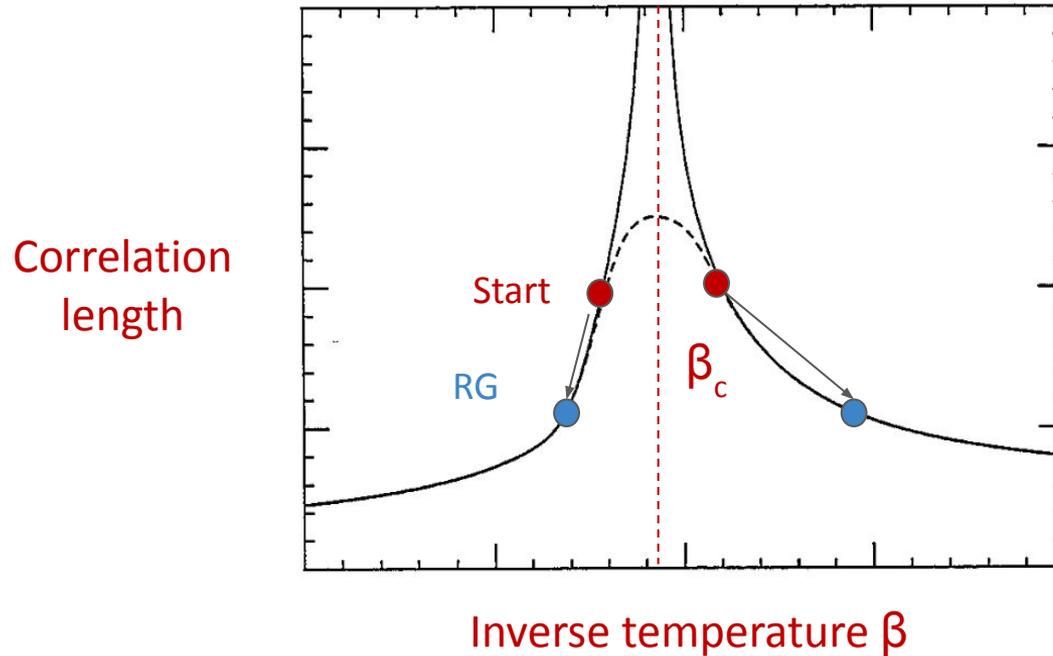
$$L_j = b^{(j-i)} L_i \quad j > i \geq 0, \text{ and } L_0 = L$$

However the increase in the lattice size will induce an analogous increase in the correlation length of the system:

$$\xi_j = b^{(j-i)} \xi_i$$

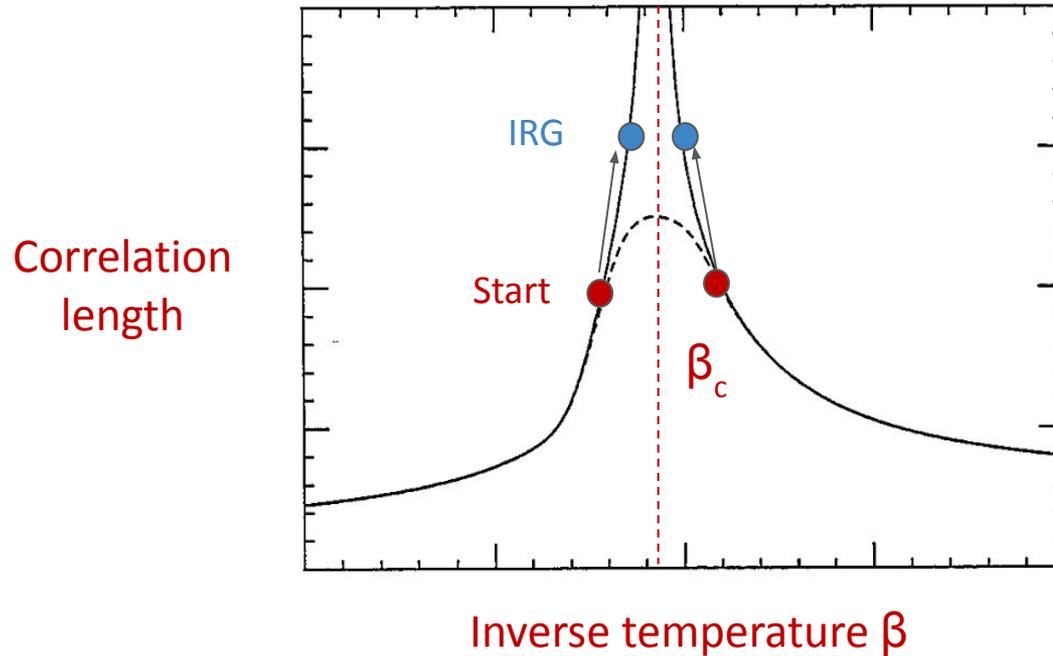
What are the implications?

Inverse renormalization group



Inverse Renormalization Group in Quantum Field Theory, D. Bachtis, G. Aarts, F. Di Renzo, B. Lucini, Phys. Rev. Lett. 128, 081603 (2022)

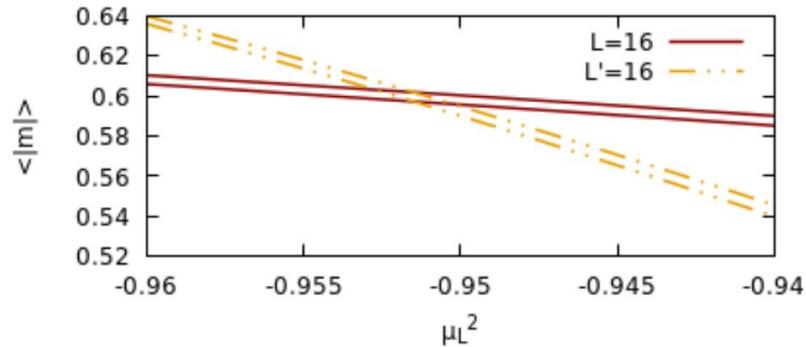
Inverse renormalization group



Inverse Renormalization Group in Quantum Field Theory, D. Bachtis, G. Aarts, F. Di Renzo, B. Lucini, Phys. Rev. Lett. 128, 081603 (2022)

Inverse renormalization group

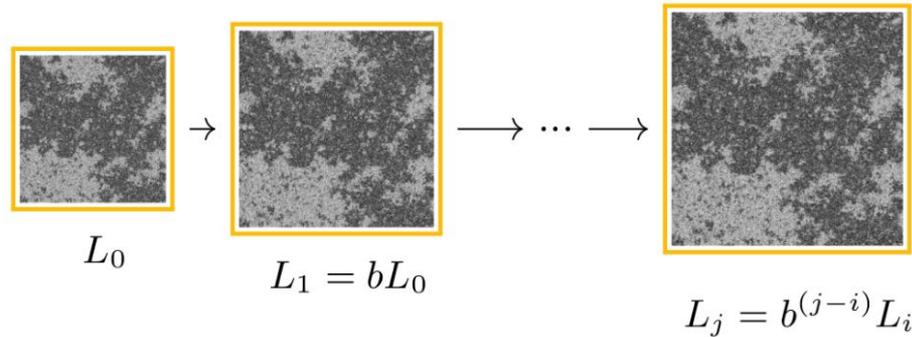
First, we verify that the **standard MC renormalization group** method works in the ϕ^4 theory:



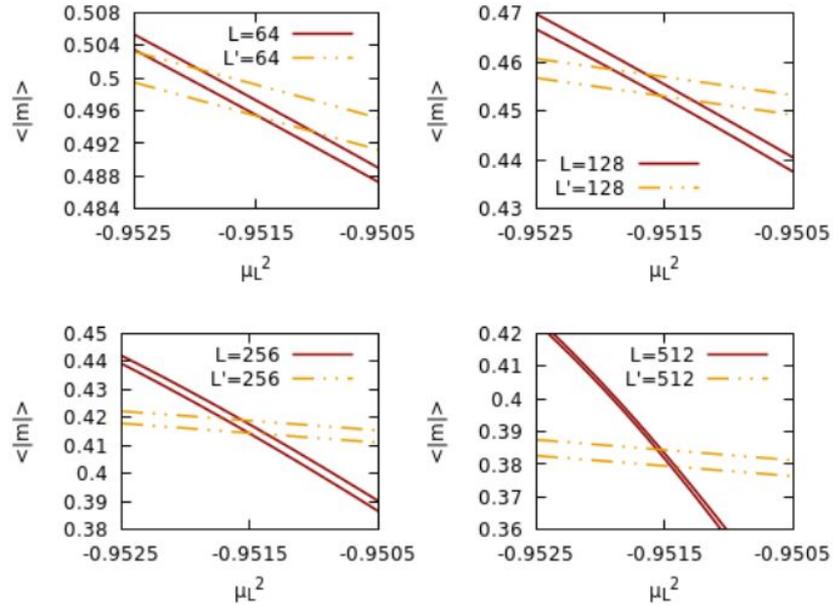
Then we invert the standard transformation that we verified as being successful.

Inverse renormalization group

Now, we start from a lattice size $L_0=32$ in each dimension and apply the inverse transformations to obtain systems of lattice sizes $L_1=64$, $L_2=128$, $L_3=256$, $L_4=512$.



Inverse renormalization group



Inverse Renormalization Group in Quantum Field Theory, D. Bachtis, G. Aarts, F. Di Renzo, B. Lucini, Phys. Rev. Lett. 128, 081603 (2022)

Inverse renormalization group

Can we now use the inverse renormalization group approach to calculate critical exponents?

The relations that govern the critical behaviour of the magnetization for an original (i) and a rescaled (j) system are

$$m_i \sim |t_i|^\beta \qquad m_j \sim |t_j|^\beta$$

They can be equivalently expressed in terms of the correlation length as

$$m_i \sim \xi_i^{-\beta/\nu} \qquad m_j \sim \xi_j^{-\beta/\nu}$$

where ν is the correlation length exponent

Inverse renormalization group

By dividing the magnetizations (or magnetic susceptibilities), taking the natural logarithm, and applying L'Hôpital's rule, we obtain

$$\frac{\beta}{\nu} = -\frac{\ln \left. \frac{dm_j}{dm_i} \right|_{K_c}}{\ln \frac{\xi_j}{\xi_i}} = -\frac{\ln \left. \frac{dm_j}{dm_i} \right|_{K_c}}{(j-i) \ln b}, \quad \frac{\gamma}{\nu} = \frac{\ln \left. \frac{d\chi_j}{d\chi_i} \right|_{K_c}}{\ln \frac{\xi_j}{\xi_i}} = \frac{\ln \left. \frac{d\chi_j}{d\chi_i} \right|_{K_c}}{(j-i) \ln b}.$$

We can use the expressions above to calculate the critical exponents without ever experiencing a critical slowing down effect.

Inverse renormalization group

TABLE I. Values of the critical exponents γ/ν and β/ν . The original system has lattice size $L = 32$ in each dimension and its action has coupling constants $\mu_L^2 = -0.9515$, $\lambda_L = 0.7$, $\kappa_L = 1$. The rescaled systems are obtained through inverse renormalization group transformations.

L_i/L_j	32/64	32/128	32/256	32/512	64/128	64/256	64/512	128/256	128/512	256/512
γ/ν	1.735(5)	1.738(5)	1.741(5)	1.742(5)	1.742(5)	1.744(5)	1.744(5)	1.745(5)	1.745(5)	1.746(5)
β/ν	0.132(2)	0.130(2)	0.128(2)	0.128(2)	0.128(2)	0.127(2)	0.127(2)	0.126(2)	0.126(2)	0.126(2)

TABLE II. Values of the critical exponents γ/ν and β/ν . The original system has lattice size $L = 8$ in each dimension and its action has coupling constants $\mu_L^2 = -1.2723$, $\lambda_L = 1$, $\kappa_L = 1$. The rescaled systems are obtained through inverse renormalization group transformations.

L_i/L_j	8/16	8/32	8/64	8/128	8/256	8/512	16/32	16/64	16/128	16/256	16/512
γ/ν	1.694(6)	1.708(6)	1.717(6)	1.723(6)	1.727(6)	1.730(6)	1.721(6)	1.728(6)	1.732(6)	1.735(6)	1.737(6)
β/ν	0.154(2)	0.147(2)	0.142(2)	0.139(2)	0.137(2)	0.135(2)	0.140(2)	0.136(2)	0.134(2)	0.132(2)	0.131(2)

L_i/L_j	32/64	32/128	32/256	32/512	64/128	64/256	64/512	128/256	128/512	256/512
γ/ν	1.735(6)	1.738(6)	1.740(6)	1.740(6)	1.741(6)	1.742(6)	1.742(7)	1.743(6)	1.743(7)	1.743(7)
β/ν	0.133(2)	0.131(2)	0.130(2)	0.129(2)	0.129(2)	0.129(2)	0.128(2)	0.128(2)	0.127(2)	0.127(2)

Ising universality class: $\gamma/\nu=1.75$, $\beta/\nu=0.125$.

Inverse Renormalization Group in Quantum Field Theory, D. Bachtis, G. Aarts, F. Di Renzo, B. Lucini, Phys. Rev. Lett. 128, 081603 (2022)

Can we view machine learning as part of
quantum field theory?

And why?

Construction of quantum fields from Markoff fields, E. Nelson, J. Funct. Anal. 12, 97 (1973)

Quantum field-theoretic machine learning, D. Bachtis, G. Aarts and B. Lucini, Phys. Rev. D 103, 074510, (arXiv:2102.09449).

Quantum field-theoretic machine learning

A probability distribution is a product of **strictly positive** and appropriately normalized **factors** (or **potential functions**) ψ :

$$p(\phi) = \frac{\prod_{c \in C} \psi_c(\phi)}{\int_{\phi} \prod_{c \in C} \psi_c(\phi) d\phi},$$

Quantum field-theoretic machine learning

A probability distribution is a product of **strictly positive** and appropriately normalized **factors** (or **potential functions**) ψ :

$$p(\phi) = \frac{\prod_{c \in C} \psi_c(\phi)}{\int_{\phi} \prod_{c \in C} \psi_c(\phi) d\phi},$$

1. **Factors are the fundamental building blocks of probability distributions.**

Quantum field-theoretic machine learning

A probability distribution is a product of **strictly positive** and appropriately normalized **factors** (or **potential functions**) ψ :

$$p(\phi) = \frac{\prod_{c \in C} \psi_c(\phi)}{\int_{\phi} \prod_{c \in C} \psi_c(\phi) d\phi},$$

1. **Factors are the fundamental building blocks of probability distributions.**
2. **By controlling the factors we are able to control the probability distribution.**

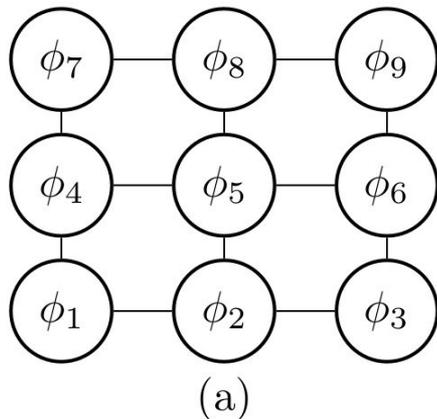
Quantum field-theoretic machine learning

We require some form of **representation** to construct the probability distribution. We are going to use a finite set Λ that we express as a **graph** $G(\Lambda, e)$ where e is the set of edges in G .

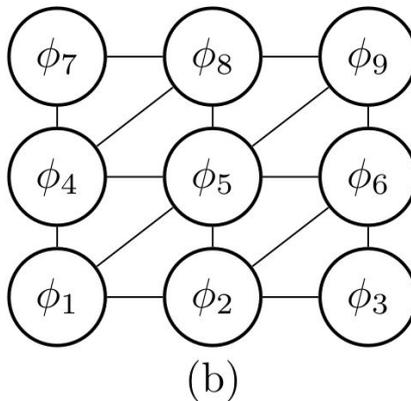
A **clique** c is a subset of Λ where the points are pairwise connected. A **maximal clique** is a clique where we cannot add another point that is pairwise connected with **all** the points in the subset.

Quantum field-theoretic machine learning

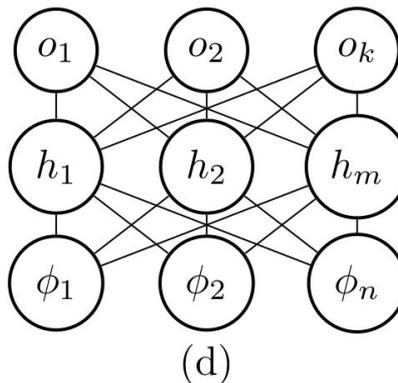
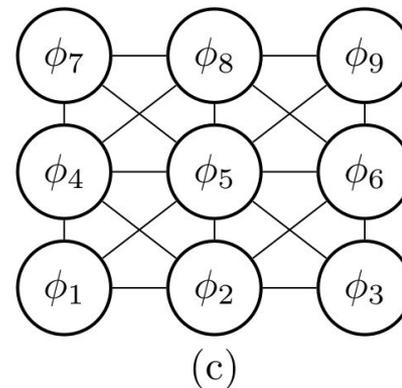
On the **square lattice** a **maximal clique** is an **edge**.



On a **triangular lattice** a **maximal clique** is a **triangle**.



On the **square lattice with both diagonals** a **maximal clique** is a **square**.



On the **bipartite graph**, which represents standard neural network architectures a **maximal clique** is an **edge**.

Quantum field-theoretic machine learning

Hammersley-Clifford theorem

A strictly positive distribution p satisfies the local Markov property of an undirected graph G :

$$p(\phi_i | (\phi_j)_{j \in \Lambda - i}) = p(\phi_i | (\phi_j)_{j \in \mathcal{N}_i})$$

if and only if p can be represented as a product of strictly positive potential functions ψ_c over G , one per maximal clique c , i.e.

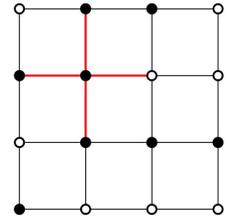
$$p(\phi) = \frac{1}{Z} \prod_{c \in C} \psi_c(\phi), \quad Z = \int_{\phi} \prod_{c \in C} \psi_c(\phi) d\phi$$

where Z is the partition function and ϕ are all possible states of the system.

Quantum field-theoretic machine learning

The ϕ^4 lattice field theory is, by definition, formulated on a square lattice which is equivalent to a graph $G(\Lambda, e)$. A non-unique choice of potential function per each maximal clique is:

$$\psi_c = \exp \left[-w_{ij} \phi_i \phi_j + \frac{1}{4} (a_i \phi_i^2 + a_j \phi_j^2 + b_i \phi_i^4 + b_j \phi_j^4) \right],$$



The probability distribution is expressed as a product of strictly positive potential functions ψ , over each maximal clique:

$$p(\phi; \theta) = \frac{\exp \left[\sum_{c \in C} \ln \psi_c(\phi) \right]}{\int_{\phi} \exp \left[\sum_{c \in C} \ln \psi_c(\phi) \right] d\phi} = \frac{1}{Z} \prod_{c \in C} \psi_c(\phi).$$

The ϕ^4 theory satisfies Markov properties and it is therefore a Markov random field.

Quantum field-theoretic machine learning

Having established that certain physical systems are Markov random fields, how do we use them for machine learning?

Quantum field-theoretic machine learning

Having established that certain physical systems are Markov random fields, how do we use them for machine learning?

Exactly in the same way as any other machine learning algorithm...

Quantum field-theoretic machine learning

The ϕ^4 theory has a **probability distribution** $p(\phi; \theta)$ with action $S(\phi; \theta)$:

$$p(\phi; \theta) = \frac{\exp[-S(\phi; \theta)]}{\int_{\phi} \exp[-S(\phi, \theta)] d\phi}.$$

We now consider a quantum field theory with action \mathcal{A} and a **target probability distribution** $q(\phi)$:

$$q(\phi) = \exp[-\mathcal{A}] / Z_{\mathcal{A}}$$

Quantum field-theoretic machine learning

We can then define an asymmetric distance between the probability distributions $p(\phi; \theta)$ and $q(\phi)$, which is called the **Kullback-Leibler divergence**:

$$KL(p||q) = \int_{-\infty}^{\infty} p(\phi; \theta) \ln \frac{p(\phi; \theta)}{q(\phi)} d\phi \geq 0.$$

Quantum field-theoretic machine learning

We can then define an asymmetric distance between the probability distributions $p(\phi; \theta)$ and $q(\phi)$, which is called the **Kullback-Leibler divergence**:

$$KL(p||q) = \int_{-\infty}^{\infty} p(\phi; \theta) \ln \frac{p(\phi; \theta)}{q(\phi)} d\phi \geq 0.$$

We want to minimize the Kullback-Leibler divergence.

By minimizing it we would make the two probability distributions equal. **We can then use the probability distribution $p(\phi; \theta)$ of the ϕ^4 theory to draw samples from the target distribution $q(\phi)$ of action A.**

Quantum field-theoretic machine learning

We substitute the two probability distributions in the Kullback-Leibler divergence to obtain:

$$F_{\mathcal{A}} \leq \langle \mathcal{A} - S \rangle_{p(\phi; \theta)} + F \equiv \mathcal{F},$$

Bogoliubov Inequality

$\langle \rangle$ denotes expectation value

There are two important observations on the above equation:

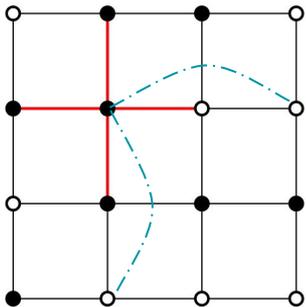
1. It sets a rigorous upper bound to the calculation of the free energy of the system with action \mathcal{A} .
2. The bound is dependent entirely on samples drawn from the distribution $p(\phi; \theta)$ of the ϕ^4 theory.

Quantum field-theoretic machine learning

We have conducted a variety of proof-of-principle applications to demonstrate that the inhomogeneous action

$$S(\phi; \theta) = - \sum_{\langle ij \rangle} w_{ij} \phi_i \phi_j + \sum_i a_i \phi_i^2 + \sum_i b_i \phi_i^4,$$

is able to represent more intricate actions, such as actions that include longer range interactions



$$\mathcal{A}_{\{4\}}(\phi) = - \sum_{\langle ij \rangle} \phi_i \phi_j + 1.52425 \sum_i \phi_i^2 + 0.175 \sum_i \phi_i^4 - \sum_{\langle ij \rangle_{nnn}} \phi_i \phi_j$$

See Ref.

Quantum field-theoretic machine learning

What if the target probability distribution $q(\varphi)$ is unknown?

Quantum field-theoretic machine learning

Earlier we defined the Kullback-Leibler divergence as:

$$KL(p||q) = \int_{-\infty}^{\infty} p(\phi; \theta) \ln \frac{p(\phi; \theta)}{q(\phi)} d\phi \geq 0.$$

We will now consider the **opposite divergence**:

$$KL(q||p) = \int_{-\infty}^{\infty} q(\phi) \ln \frac{q(\phi)}{p(\phi; \theta)} d\phi.$$

Quantum field-theoretic machine learning

We are searching for the optimal values of the coupling constants in the ϕ^4 action that are able to reproduce the data as configurations in the equilibrium distribution.

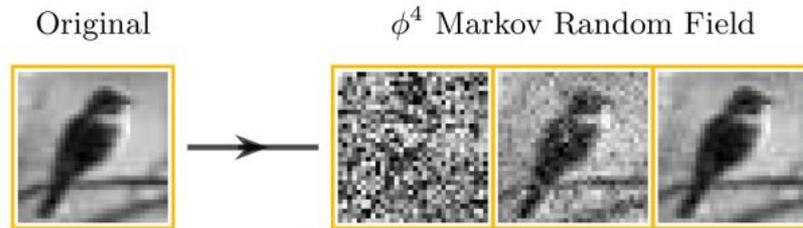
$$S(\phi; \theta) = - \sum_{\langle ij \rangle} w_{ij} \phi_i \phi_j + \sum_i a_i \phi_i^2 + \sum_i b_i \phi_i^4,$$

Quantum field-theoretic machine learning

We are searching for the optimal values of the coupling constants in the ϕ^4 action that are able to reproduce the data as configurations in the equilibrium distribution.

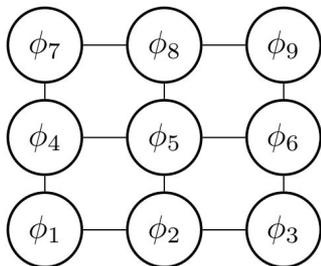
$$S(\phi; \theta) = - \sum_{\langle ij \rangle} w_{ij} \phi_i \phi_j + \sum_i a_i \phi_i^2 + \sum_i b_i \phi_i^4,$$

Case of an image:



Quantum field-theoretic machine learning

ϕ^4 Markov random field

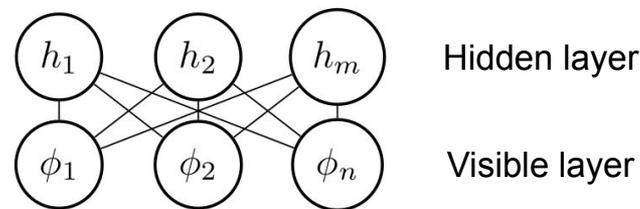


$$S(\phi; \theta) = - \sum_{\langle ij \rangle} w_{ij} \phi_i \phi_j + \sum_i a_i \phi_i^2 + \sum_i b_i \phi_i^4,$$

$$\theta = \{w_{ij}, a_i, b_i\}$$

$$p(\phi; \theta) = \frac{\exp[-S(\phi; \theta)]}{\int_{\phi} \exp[-S(\phi, \theta)] d\phi}.$$

ϕ^4 neural network



$$S(\phi, h; \theta) = - \sum_{i,j} w_{ij} \phi_i h_j + \sum_i r_i \phi_i + \sum_i a_i \phi_i^2$$

$$+ \sum_i b_i \phi_i^4 + \sum_j s_j h_j + \sum_j m_j h_j^2 + \sum_j n_j h_j^4,$$

$$\theta = \{w_{ij}, r_i, a_i, b_i, s_j, m_j, n_j\}.$$

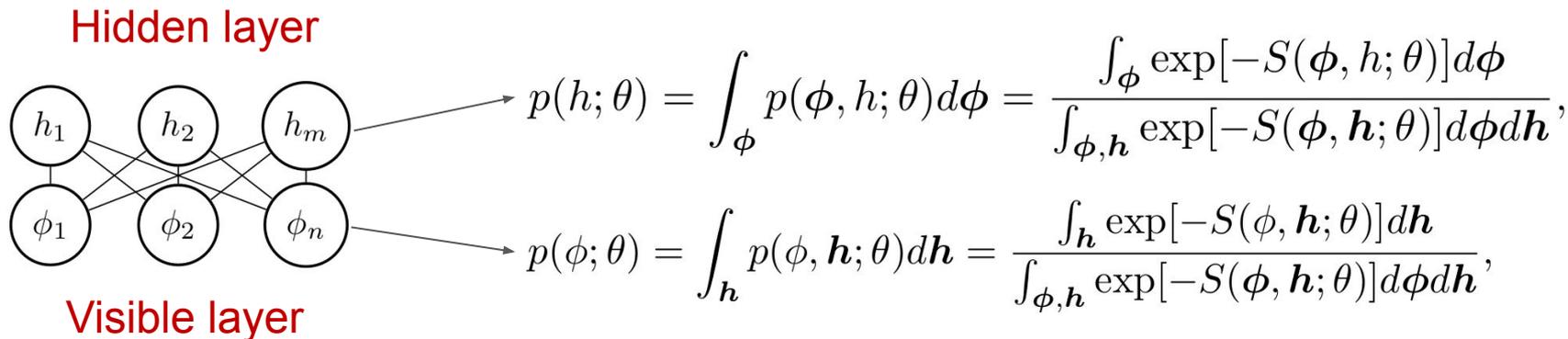
$$p(\phi, h; \theta) = \frac{\exp[-S(\phi, h; \theta)]}{\int_{\phi, \mathbf{h}} \exp[-S(\phi, \mathbf{h}; \theta)] d\phi d\mathbf{h}}.$$

Quantum field-theoretic machine learning

From the **joint probability distribution** of the ϕ^4 neural network

$$p(\phi, \mathbf{h}; \theta) = \frac{\exp[-S(\phi, \mathbf{h}; \theta)]}{\int_{\phi, \mathbf{h}} \exp[-S(\phi, \mathbf{h}; \theta)] d\phi d\mathbf{h}}.$$

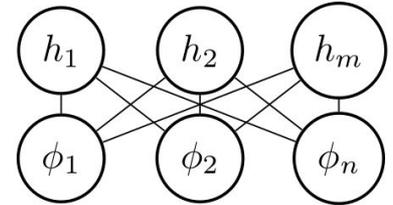
We are able to marginalize out variables and derive **marginal probability distributions** $p(\phi; \theta)$ and $p(\mathbf{h}; \theta)$:



Quantum field-theoretic machine learning

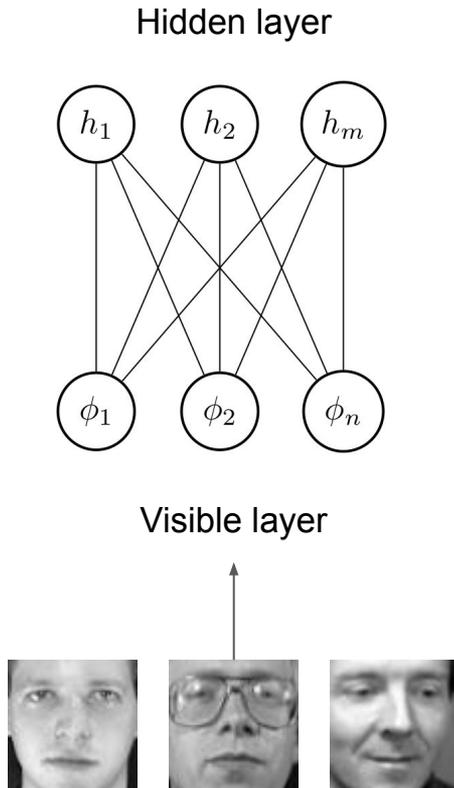
We now want to minimize the asymmetric distance between the **empirical probability distribution $q(\phi)$** and the **marginal probability distribution $p(\phi; \theta)$** :

$$KL(q||p) = \int_{-\infty}^{\infty} q(\phi) \ln \frac{q(\phi)}{p(\phi; \theta)} d\phi.$$

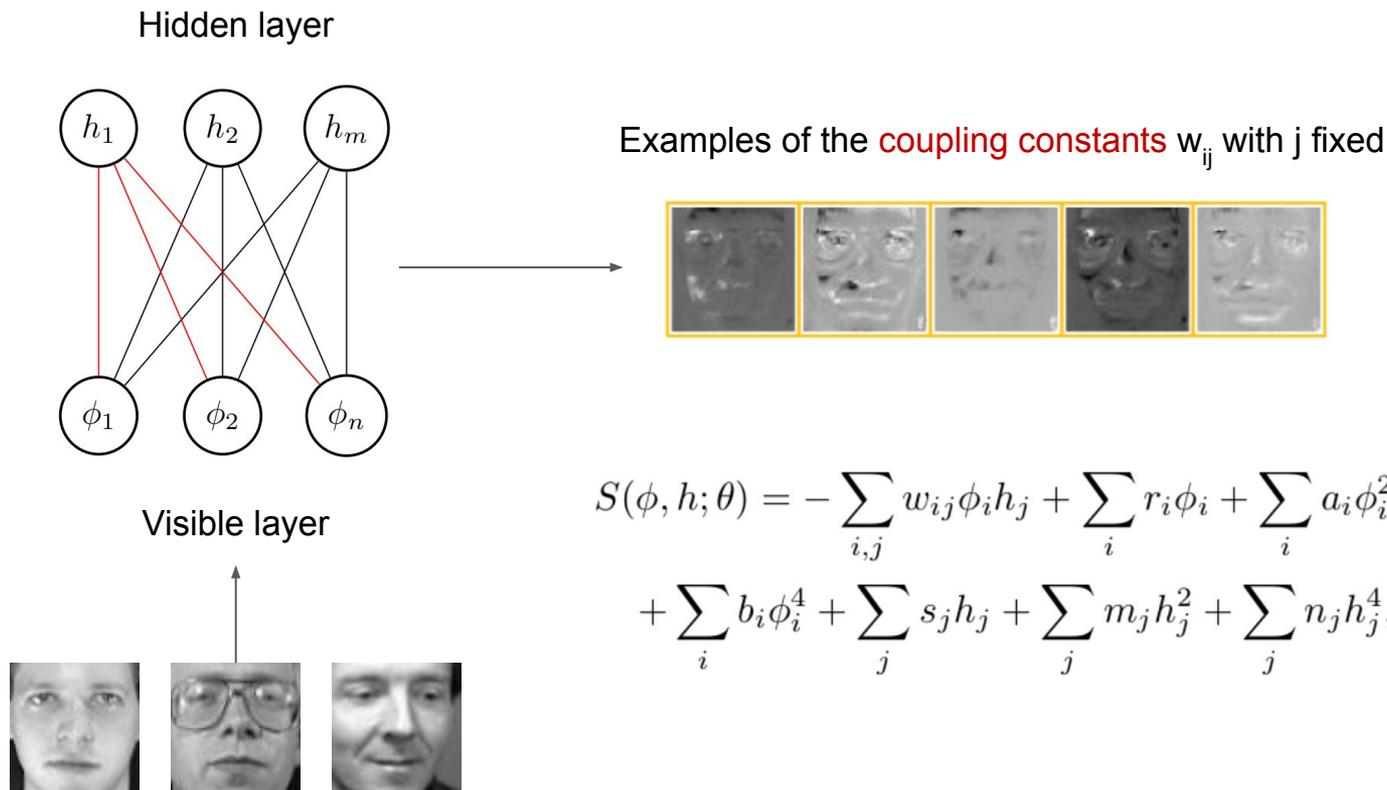


In other words, we want to reproduce the dataset in the visible layer. The hidden layer will then uncover dependencies on the data.

Quantum field-theoretic machine learning



Quantum field-theoretic machine learning



Quantum field-theoretic machine learning

The ϕ^4 neural network:

$$S(\phi, h; \theta) = - \sum_{i,j} w_{ij} \phi_i h_j + \sum_i r_i \phi_i + \sum_i a_i \phi_i^2 \\ + \sum_i b_i \phi_i^4 + \sum_j s_j h_j + \sum_j m_j h_j^2 + \sum_j n_j h_j^4,$$

is a generalization of other neural network architectures:

Gaussian-Gaussian
restricted Boltzmann
machine:

$$b_i = n_j = 0$$

Gaussian-Bernoulli
restricted Boltzmann
machine:

$$b_i = n_j = m_j = 0 \\ h_j \text{ binary}$$

Bernoulli-Bernoulli
restricted Boltzmann
machine:

$$b_i = n_j = m_j = a_i = 0 \\ \phi_i, h_j \text{ binary}$$

ϕ^4 -Bernoulli restricted
Boltzmann machine:

$$m_j = n_j = 0 \\ h_j \text{ binary}$$

ϕ^4 equivalence with the Ising model (under an appropriate limit)

Summary

- 1) **Interpretation of machine learning functions as physical observables:**
 - a) How to construct effective order parameters with machine learning.
 - b) How to reweight machine learning functions in parameter space.
 - c) How to include machine learning functions within Hamiltonians to induce phase transitions.
 - d) How to utilize the renormalization group to obtain critical exponents using machine learning functions.
- 2) **Inverse renormalization group with machine learning:**
 - a) How to generate configurations of systems with larger lattice size without having to simulate these systems and without critical slowing down effect.
 - b) How do inverse renormalization group flows emerge.
 - c) How to calculate multiple critical exponents with the inverse renormalization group.
- 3) **Quantum field-theoretic machine learning:**
 - a) How to derive machine learning algorithms and neural networks from quantum field theories



This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 813942

Thank you for your attention!