Swansea University
Prifysgol Abertawe

Quantum field-theoretic machine learning

Dimitrios Bachtis

Joint work with Profs. Gert Aarts and Biagio Lucini.

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

And why?

Probability distribution

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

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

Probability distribution

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

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

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

Probability distribution

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

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

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

2. By controlling the factors we are able to control the form of the probability distribution.

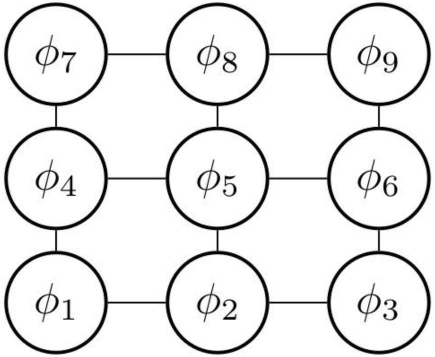
Representation

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 $\mathcal{G}(\Lambda, e)$ where e is the set of edges in $\mathcal{G}$.

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.

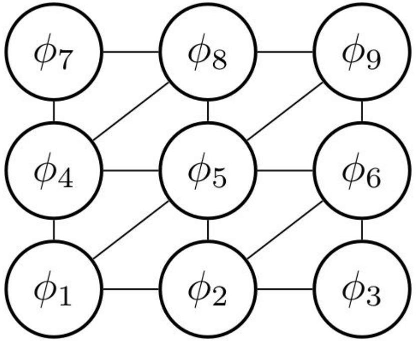
Representation

On the square lattice a maximal clique is an edge.



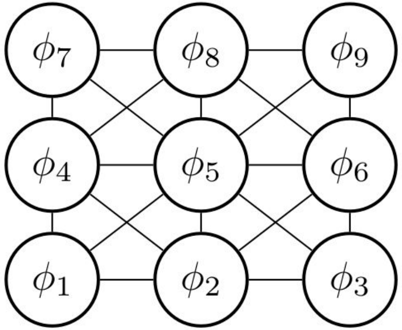
(a)

On a triangular lattice a maximal clique is a triangle.

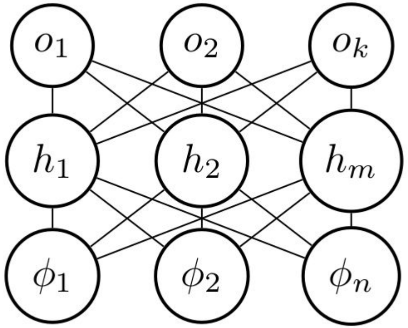


(b)

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



(c)



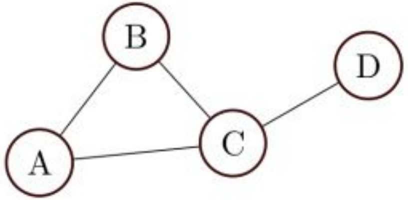
(d)

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

Representation

Given a graph $\mathcal{G}(\Lambda, e)$, the random variables φ_i at each point i define a Markov random field if they fulfill the local Markov property with respect to $\mathcal{G}$.

The local Markov property denotes that a random variable φ_i depends only on its neighbors and it is conditionally independent of all other random variables in the set:



$$p(\phi_i | (\phi_j)_{j \in \Lambda - \phi_i}) = p(\phi_i | (\phi_j)_{j \in n_i}).$$

Representation

Hammersley-Clifford theorem

A strictly positive distribution p satisfies the local Markov property of an undirected graph $\mathcal{G}$, if and only if p can be represented as a product of strictly positive potential functions ψ_c over $\mathcal{G}$, one per maximal clique c, i.e.

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

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

Representation

There are two different directions to pursue:

1. We can devise potential functions that satisfy the Hammersley-Clifford theorem to construct a Markov random field.

Representation

There are two different directions to pursue:

1. We can devise potential functions that satisfy the Hammersley-Clifford theorem to construct a Markov random field.
2. We can evaluate if known physical systems can be recast within this mathematical framework by verifying instead if they satisfy the theorem.

We will pursue the second direction.

Representation

2d φ^4 theory:

$$\mathcal{L}_E = \frac{\kappa}{2}(\nabla\phi)^2 + \frac{\mu_0^2}{2}\phi^2 + \frac{\lambda}{4}\phi^4,$$

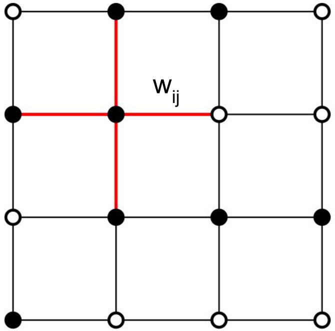
$$S_E = -\kappa_L \sum_{\langle ij\rangle} \phi_i\phi_j + \frac{(\mu_L^2 + 4\kappa_L)}{2}\sum_i \phi_i^2 + \frac{\lambda_L}{4}\sum_i \phi_i^4.$$

k_L, μ_L, λ_L dimensionless parameters

$$w = \kappa_L,\ a = (\mu_L^2 + 4\kappa_L)/2,\ b = \lambda_L/4$$

Inhomogeneous φ^4 theory:

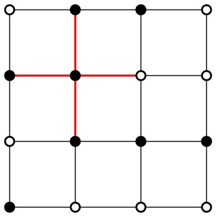
$$S(\phi;\theta) = -\sum_{\langle ij\rangle} w_{ij}\phi_i\phi_j + \sum_i a_i\phi_i^2 + \sum_i b_i\phi_i^4,$$



Representation

The φ^4 theory is formulated on a square lattice which is equivalent to a graph $\mathcal{G}(\Lambda,e)$ where Λ is the set of lattice sites and e the edges. A non-unique choice of potential function per each maximal clique is:

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



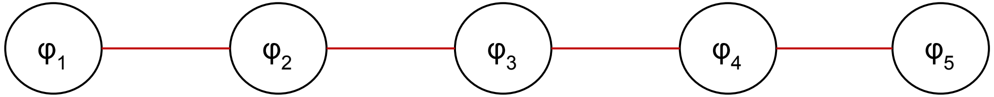
The probability distribution is expressed as a product of strictly positive potential functions ψ, over each maximal clique:

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

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

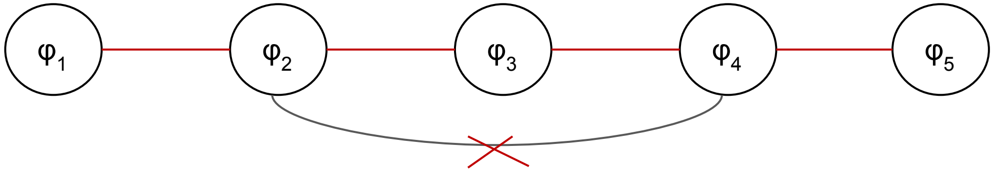
Representation

The Markov property in a Markov chain



Representation

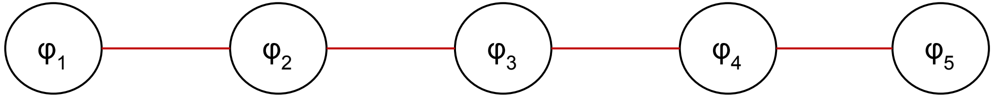
The Markov property in a Markov chain



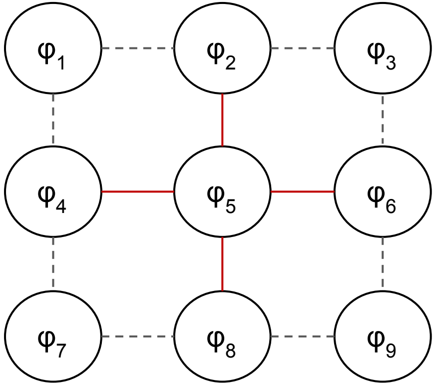
Not allowed!!

Representation

The Markov property in a Markov chain



A Markov random field satisfies the Markov property in high-dimensions



Learning

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

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...

Learning

The φ^4 theory has a probability distribution p(φ;θ) with action S(φ;θ):

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

We now consider a different statistical system or quantum field theory with action or Hamiltonian A and a target probability distribution q(φ):

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

Learning

We can then define an asymmetric distance between the probability distributions p(φ;θ) and q(φ), which is called the Kullback-Leibler divergence:

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

Learning

We can then define an asymmetric distance between the probability distributions p(φ;θ) and q(φ), 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(φ;θ) to draw samples from the target distribution q(φ).

We achieve the minimization with a gradient-based approach.

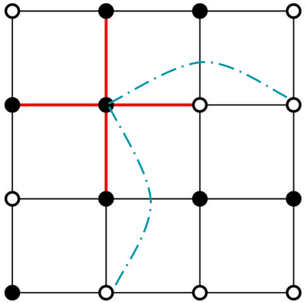
Learning

A proof-of-principle demonstration is to use the inhomogeneous action S:

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

to learn an action that includes longer-range interactions:

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



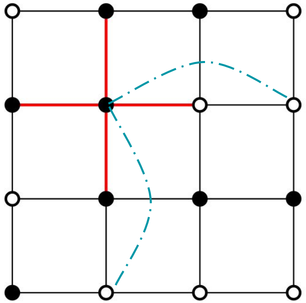
Learning

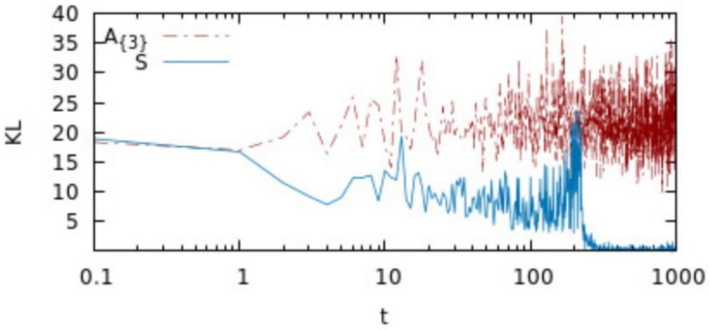
A proof-of-principle demonstration is to use the inhomogeneous action S:

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

to learn an action that includes longer-range interactions:

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





Learning

Three reweighting (simultaneous) steps: Make the (already trained) inhomogeneous action S:

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

Equal to the target action A (acts as a correction step):

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

Learning

Three reweighting (simultaneous) steps: Make the (already trained) inhomogeneous action S:

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

Equal to the target action A (acts as a correction step):

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

Extrapolate in the parameter space along the trajectory of a coupling constant g’ of A

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

Learning

Three reweighting (simultaneous) steps: Make the (already trained) inhomogeneous action S:

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

Equal to the target action A (acts as a correction step):

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

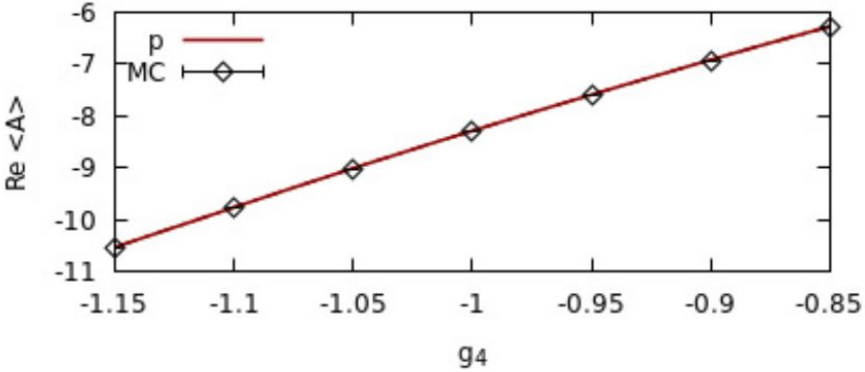
Extrapolate in the parameter space along the trajectory of a coupling constant g’ of A

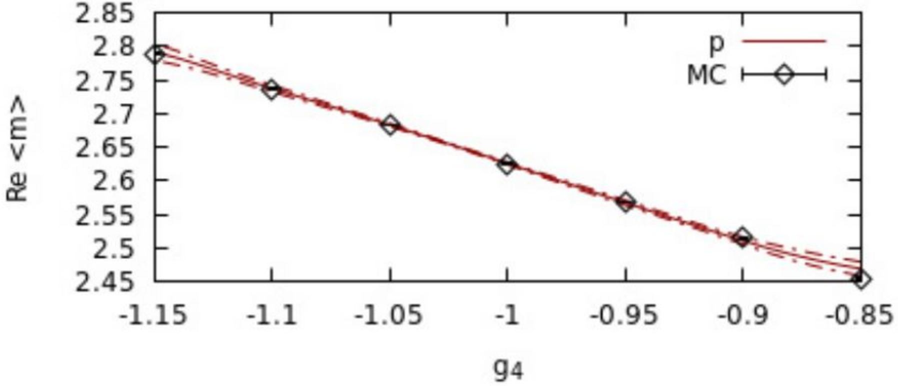
$$\mathcal{A}_{\{4\}}(\phi) = -\sum_{\langle ij\rangle}\phi_i\phi_j + 1.52425\sum_i\phi_i^2 + 0.175\sum_i\phi_i^4 - \sum_{\langle ij\rangle_{nnn}} g'\phi_i\phi_j$$

Extrapolate to an imaginary term

$$\mathcal{A}_{\{5\}}(\phi) = -\sum_{\langle ij\rangle}\phi_i\phi_j + 1.52425\sum_i\phi_i^2 + 0.175\sum_i\phi_i^4 - \sum_{\langle ij\rangle_{nnn}} g'\phi_i\phi_j + i0.15\sum_i\phi_i^2$$

Learning


-6
-7
-8
-9
-10
-11
p
MC
Re <A>
-1.15
-1.1
-1.05
-1
-0.95
-0.9
-0.85
g_4


2.85
2.8
2.75
2.7
2.65
2.6
2.55
2.5
2.45
p
MC
Re <m>
-1.15
-1.1
-1.05
-1
-0.95
-0.9
-0.85
g_4

Learning

What have we achieved?

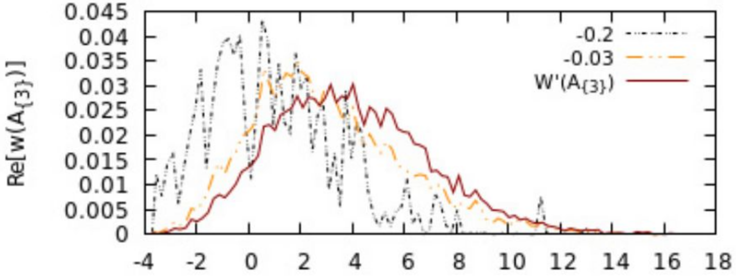
Learning

It is impossible to reweight from the standard action:

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

To this action:

$$\mathcal{A}_{\{5\}}(\phi) = -\sum_{\langle ij\rangle}\phi_i\phi_j + 1.52425\sum_i \phi_i^2 + 0.175\sum_i \phi_i^4 - \sum_{\langle ij\rangle_{nnn}}\phi_i\phi_j + i0.15\sum_i \phi_i^2$$



$$\mathcal{W}(S) = \frac{\sum_{\Re[\mathcal{A}'],\Im[\mathcal{A}']} h(S,\Re[\mathcal{A}'],\Im[\mathcal{A}'])\exp[S-\Re[\mathcal{A}']-i\Im[\mathcal{A}']]}{\sum_{S,\Re[\mathcal{A}'],\Im[\mathcal{A}']} h(S,\Re[\mathcal{A}'],\Im[\mathcal{A}'])\exp[S-\Re[\mathcal{A}']-i\Im[\mathcal{A}']]},$$

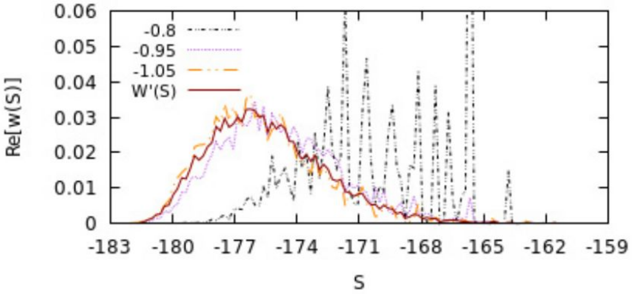
Learning

However it is possible to reweight from the inhomogeneous action:

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

To this action:

$$\mathcal{A}_{\{5\}}(\phi) = -\sum_{\langle ij\rangle}\phi_i\phi_j + 1.52425\sum_i\phi_i^2 + 0.175\sum_i\phi_i^4 - \sum_{\langle ij\rangle_{nnn}}\phi_i\phi_j + i0.15\sum_i\phi_i^2$$



$$\mathcal{W}(S) = \frac{\sum_{\Re[\mathcal{A}'],\Im[\mathcal{A}']} h(S,\Re[\mathcal{A}'],\Im[\mathcal{A}'])\exp[S-\Re[\mathcal{A}']-i\Im[\mathcal{A}']]}{\sum_{S,\Re[\mathcal{A}'],\Im[\mathcal{A}']} h(S,\Re[\mathcal{A}'],\Im[\mathcal{A}'])\exp[S-\Re[\mathcal{A}']-i\Im[\mathcal{A}']]},$$

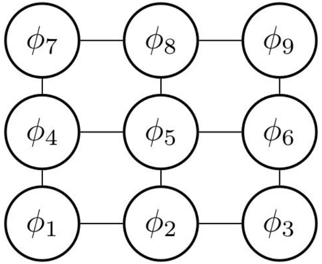
Learning

Conclusion:

Inhomogeneous actions have increased representational capacity compared to homogeneous actions

Neural Networks

φ⁴ Markov random field

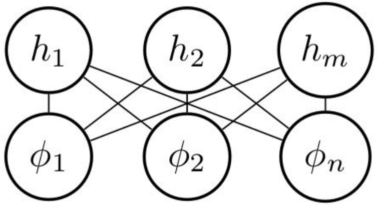


$$S(\phi;\theta) = -\sum_{\langle ij\rangle} w_{ij}\phi_i\phi_j + \sum_i a_i\phi_i^2 + \sum_i b_i\phi_i^4,$$

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

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

φ⁴ neural network



Hidden layer

Visible layer

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

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

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

Neural Networks

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 binary

Bernoulli-Bernoulli restricted Boltzmann machine:

$b_i = n_j = m_j = a_i = 0$
φ_{i}, h_j binary

φ^4-Bernoulli restricted Boltzmann machine:

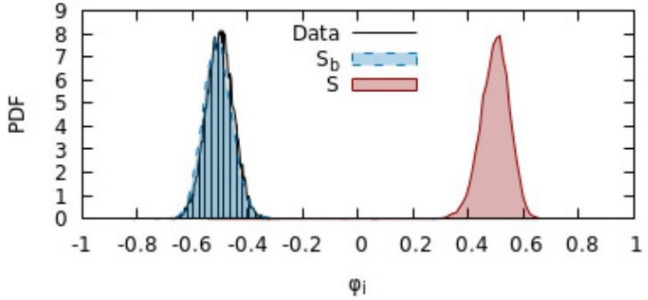
$m_j = n_j = 0$
h_j binary

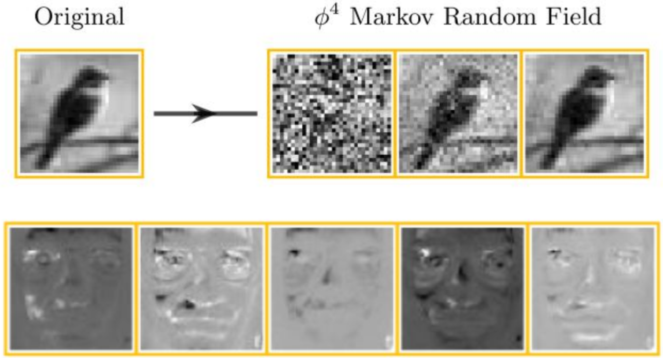
φ^4 equivalence with the Ising model (under an appropriate limit)

Learning

The same approach can be used even when the probability distribution is not known, but we have the probability distribution encoded in some available data. The data can be anything: images or experimental data or a set of Monte Carlo configurations.

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





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

Conclusions

1. Lattice field theories emerge as natural machine learning algorithms and we can investigate machine learning as a physical concept within quantum field theory: e.g. what are the phase transitions of the φ^4 machine learning algorithms?
2. Mathematical foundations of quantum field theory: Quantum fields in Minkowski space can be constructed by Markov fields in Euclidean space (the Nelson-Symanzik perspective of quantum field theory).
3. Lattice field theory is inherently a computational research field: easy to implement quantum-field theoretic machine learning algorithms and to study them computationally from the perspective of quenched disorder.

Thank you for your attention!

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

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

EuroPLEx
European network for Particle physics,
Lattice field theory and Extreme
computing