Is Deep Learning a Renormalization Group Flow?

ELLEN DE MELLO KOCH[1], ROBERT DE MELLO KOCH [ID][2,3,4,5],
AND LING CHENG[ID][1], **(Senior Member, IEEE)**
[1]School of Electrical and Information Engineering, University of the Witwatersrand, Wits 2050, South Africa
[2]School of Physics and Telecommunication Engineering, South China Normal University, Guangzhou 510006, China
[3]National Institute for Theoretical Physics, University of the Witwatersrand, Wits 2050, South Africa
[4]School of Physics, University of the Witwatersrand, Wits 2050, South Africa
[5]Mandelstam Institute for Theoretical Physics, University of the Witwatersrand, Wits 2050, South Africa

Corresponding author: Robert de Mello Koch (robert@neo.phys.wits.ac.za)

ABSTRACT Although there has been a rapid development of practical applications, theoretical explanations of deep learning are in their infancy. Deep learning performs a sophisticated coarse graining. Since coarse graining is a key ingredient of the renormalization group (RG), RG may provide a useful theoretical framework directly relevant to deep learning. In this study we pursue this possibility. A statistical mechanics model for a magnet, the Ising model, is used to train an unsupervised restricted Boltzmann machine (RBM). The patterns generated by the trained RBM are compared to the configurations generated through an RG treatment of the Ising model. Although we are motivated by the connection between deep learning and RG flow, in this study we focus mainly on comparing a single layer of a deep network to a single step in the RG flow. We argue that correlation functions between hidden and visible neurons are capable of diagnosing RG-like coarse graining. Numerical experiments show the presence of RG-like patterns in correlators computed using the trained RBMs. The observables we consider are also able to exhibit important differences between RG and deep learning.

INDEX TERMS Restricted Boltzmann machines (RBMs), deep learning, deep neural networks, learning theory, renormalization group (RG).

I. INTRODUCTION

The power of machine learning and artificial intelligence is established: these are powerful methods that already outperform humans in specific tasks [1]–[4]. Much of the research carried out in machine learning is of an applied nature. It establishes the practical utility of the method but does not construct an understanding of how deep learning works or even if such an understanding is possible [5]–[10]. Consequently, deep learning remains an impressive but mysterious black box. A possible starting point for a theoretical treatment suggests that deep learning is a form of coarse graining [3], [11], [12]. Since there are more input than output neurons this is almost certainly true. The real question is then if this is a useful observation, one that might shed light on how deep learning works. This is the question we take up in this paper.

We argue that understanding deep learning as a form of coarse graining is a useful observation, and make the case by

adopting and adapting several ideas from theoretical physics. Specifically, in theoretical physics there is a sound framework to carry out coarse graining, known as the renormalization group (RG) [13]. RG provides a systematic way to construct a theory describing large scale features from an underlying microscopic description, which can be understood as recognizing sophisticated emergent patterns, a routine achievement of deep learning. Further, RG is applicable to field theories, that is, to systems with a large number of degrees of freedom so that it seems that RG is well positioned to deal with massive data sets. Finally, the way in which RG works is, in contrast to deep learning, well understood and can be described in precise mathematical language. These features suggest that RG may provide a useful framework in which to describe deep learning and attempts to argue that this is the case have been made in [14]–[17]. We focus on unsupervised learning by a restricted Boltzmann machine (RBM).

Two distinct possible connections to RG have been attempted, both relevant to our study. The first [14] is an attempt to link deep learning to the RG flow. The RG flow

The associate editor coordinating the review of this manuscript and approving it for publication was Corrado Mencar [ID].

IEEE Access

E. de Mello Koch *et al.*: Is Deep Learning a Renormalization Group Flow?

is a smooth process during which degrees of freedom are continuously averaged out, so that we flow from the initial microscopic description to the final macroscopic description. In deep learning one stacks layers of networks to obtain a deep network. The proposed connection of [14] suggests that each layer in the stack performs a small step along the RG flow.[1] We contribute to this discussion by developing quantitative tools with which this proposal can be explored with precision. The basic objects that appear in our analysis are correlation functions between the visible and the hidden neurons. This allows us to decode the mechanics of the RBM's pattern generation and to compare it to what the RG is doing. Although there are important differences, our results indicate remarkable similarities between how the RBM and RG achieve their results. The second approach [15], [16] builds an RBM flow using the weight matrix of the neural network after training is complete. The results of [15], [16] suggest that the RBM flow is closely related to the RG flow. We carry out a critical examination of this conclusion. The central tools we employ are correlation functions defined using the patterns generated by the RBM. We give a detailed and precise argument showing that the largest scale features of RG and RBM patterns are in complete agreement. The correlation functions involved are non-trivial probes of the statistics of the generated pattern[2] so the conclusion we reach is compelling. We also find that if one probes smaller scale features there are important differences between the two patterns. We will comment further on the interpretation of these results in the conclusions.

At this point, a comment is in order. The word "deep" in "deep learning" indicates that many layers are stacked to produce the network. Each layer in our network is an unsupervised RBM. The word "flow" in "RG flow" indicates that many small steps of coarse graining are carried out. Each step performs a local averaging and one basic step is being repeated. In this study we are developing methods that allow a comparison between one step of the renormalization group flow to one layer of the deep network. Thus, although we spend much of our effort on comparing a single RBM layer to a single step in the RG flow, we are interested in understanding learning achieved through the composition of many layers as an RG flow. It is in this sense that although we study a single layer we nevertheless claim we are exploring the problem of deep learning.

The setting for our study is the two dimensional Ising model [19]. This is a simple model of a magnet, built for many individual "spins" each of which should be thought of as a microscopic bar magnet. Each spin can be aligned "up" or "down". Spins align at low temperatures producing a magnet. At high temperatures, spins are aligned randomly and there is no net magnetic field. The spins themselves define a binary pattern (the two states are up or down) and it is

these patterns that the RBM learns. An important motivation for this choice of model is that it is well understood. The theory exhibits a first order phase transition terminating at a critical point. The theory at the critical point enjoys a conformal symmetry so that it can be solved exactly. It exhibits many interesting observables which we use to explore how deep learning is working [20], [21]. For example, if a neural network generates a microstate of the model, we can ask what the corresponding temperature of the microstate is. At the critical point special observables known as primary operators, can be defined. Their correlation functions are power laws with powers that are known. These are the natural variables which encode, completely, the long scale features of the patterns. In this way, the Ising model gives a framework to explore deep learning both through the results of numerical experiments and using the complete understanding of the large scale features of the coarse grained system. To probe whether deep learning is a type of coarse graining, we will see that this knowledge of correlations on large length scales is a valuable tool.

Our study of an Ising magnet may seem rather far removed from more usual (and practical) applications, including for example image recognition and manipulation. However, one might be optimistic that lessons learned from the Ising model are applicable to these more familiar examples. Indeed, the energy function of the Ising model tries to align nearby spins with the result that nearby spins are correlated. This is not at all unlike an image for which the color of nearby pixels is likely to be correlated [22].

A description of deep learning in the RG framework would have important implications. RG explains how macroscopic physics emerges from microscopic physics. This understanding leads to an organization of the microscopic physics into features that are relevant or irrelevant, so that in the end the emergent patterns depend only on a small number of relevant parameters. Carried over to the deep learning context, a similar understanding will strive to explain what features of the data and which weights in the network are important for deep learning. Such an understanding would have implications for what architectures are optimal and how the learning process can be improved and made more efficient.

We now sketch the content of the paper and outline how it is organized. In Section II we give a quick review of RBMs, RG and the Ising model, providing the background needed to follow subsequent arguments. In Section III we consider the RBM flows defined using the matrix of weights learned by the network [15], [16]. By studying correlation functions of primary operators of the Ising conformal field theory, we argue that although the RBM and RG patterns agree remarkably well on the largest scales of the pattern they differ on the short scale structure. In Section IV we examine the possibility that deep learning reconstructs an RG flow, with each layer of the deep network performing one step of the flow. Our discussion begins with a critical look at the argument given in [14] that claims that deep learning is mapped onto the RG flow. The argument shows

---

[1] For a critical discussion of this proposal, see [12], [18].

[2] The studies carried out in [15], [16] used averages of the RBM pattern. Our correlation functions provide more sensitive probes into the structure of the pattern.

E. de Mello Koch *et al.*: Is Deep Learning a Renormalization Group Flow?

IEEE *Access*

a system of equations that is obeyed by both the RBM and a variational realization of the RG flow. Our basic conclusion is that the argument of [14] only shows that aspects of the RBM learning are consistent with the structure of the RG transformation. Indeed, we explicitly construct examples that satisfy the equations derived by [14] that certainly do not perform an RG flow or arise from an RBM. Nevertheless, the arguments of [14] are compelling and we find the possible connection between deep learning and RG fascinating and deserving of further study. Towards this end we couch some of the qualitative observations of [14] as statements about the behavior of well chosen correlation functions. The form of these correlators, puts certain qualitative observations of Section IV.B. of [14] onto a firm quantitative footing. Finally we study the RG flow of the temperature. This turns out to be interesting as it reveals a further difference between the RBM patterns and RG. In the final Section of this paper, we discuss our results and suggest open directions that can be pursued.

II. RBM, RG AND ISING

In this section we introduce the background material used in our study. The first subsection reviews RBMs emphasizing both the structure of the network and its implementation. Following this, the RG is reviewed, with an emphasis on aspects relevant to deep learning. This section concludes with a review of the Ising model, motivating why the model is considered.

A. RESTRICTED BOLTZMANN MACHINES

RBMs perform unsupervised learning to extract features from a given data set [23]–[25]. They have a visible (input) and a hidden (output) layer. The visible layer is made up of visible nodes, v_i with $i = 1, 2 \cdots, N_v$ and the hidden layer is made up of hidden nodes, h_a with $a = 1, 2, \cdots, N_h$ as illustrated in Figure 1.
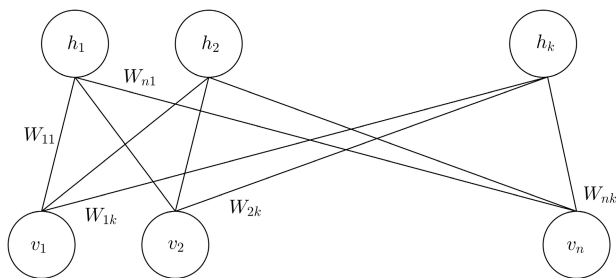

FIGURE 1. An RBM network with visible nodes v_i and hidden nodes h_a where $N_v = n$ and $N_h = k$. Connections between visible and hidden nodes are each associated with a weight W_{ia}.

The visible nodes are set with values of ± 1 and the trained network generates a corresponding pattern by setting the output nodes to ± 1. The values of the hidden neurons are obtained by evaluating a non-linear function on a linear combination of the visible neurons, perhaps offset by a constant bias. The nonlinear function we use here is the hyperbolic tangent which can be seen in equation (12). The

specific linear combination of neurons is represented by connections between nodes, with a weight for each connection. For the RBM there are connections between every visible node and every hidden node, while nodes belonging to the same layer are not connected. The "unrestricted" Boltzmann machines allow connections between any two nodes in the network [26], but this generality comes at a cost: training algorithms are much less efficient [23]–[25]. The connection between visible node v_i and hidden node h_a is assigned a weight, W_{ia}, and visible (hidden) nodes are assigned a bias $b_i^{(v)}$ ($b_a^{(h)}$). Using these ingredients we define a Hamiltonian for the RBM

$$E = -\sum_a b_a^{(h)} h_a - \sum_{i,a} v_i W_{ia} h_a - \sum_i b_i^{(v)} v_i, \quad (1)$$

where $h_a, v_i \in \{-1, 1\}$. The RBM defines the probability distribution for obtaining configurations v and h of visible and hidden vectors by [27]

$$p(v, h) = \frac{1}{\mathcal{Z}} e^{-E}, \quad (2)$$

where $\mathcal{Z}$ is the partition function, obtained by summing over all possible hidden and visible vectors

$$\mathcal{Z} = \sum_{\{v, h\}} e^{-E}. \quad (3)$$

As usual, to determine the marginal distribution of a visible vector, sum over the state space of hidden vectors

$$p(v) = \frac{1}{\mathcal{Z}} \sum_{\{h\}} e^{-E}. \quad (4)$$

Similarly, the marginal distribution of a hidden vector is

$$p(h) = \frac{1}{\mathcal{Z}} \sum_{\{v\}} e^{-E}. \quad (5)$$

The weights, W_{ia} and biases $b_i^{(v)}$, $b_a^{(h)}$ are determined during training. Training strives to match the model distribution $p(v)$ to the distribution $q(v)$ defined by the data and it achieves this by minimizing the Kullback-Leibler (KL) divergence, which is given by

$$\begin{aligned} D_{KL}(q||p) &= \sum_{\{v\}} q(v) \left(\log (q(v)) - \log (p(v)) \right) \\ &= \sum_{\{v\}} q(v) \log \left(\frac{q(v)}{p(v)} \right). \end{aligned} \quad (6)$$

The KL divergence is a measure of how much information is lost when approximating the actual distribution with the model distribution [28]. Training adjusts $\{W_{ia}, b_i^{(v)}, b_a^{(h)}\}$ to minimize the KL divergence. Gradients of the KL divergence used to update the parameters of the RBM are computed as follows

$$\frac{\partial D_{KL}(q||p)}{\partial W_{ia}} = \langle v_i h_a \rangle_{data} - \langle v_i h_a \rangle_{model}, \quad (7)$$

IEEE *Access*

E. de Mello Koch *et al.*: Is Deep Learning a Renormalization Group Flow?

$$\frac{\partial D_{KL}(q\|p)}{\partial b_i^{(v)}} = \langle v_i \rangle_{data} - \langle v_i \rangle_{model}, \tag{8}$$

$$\frac{\partial D_{KL}(q\|p)}{\partial b_a^{(h)}} = \langle h_a \rangle_{data} - \langle h_a \rangle_{model}, \tag{9}$$

where the expectation values appearing above are easily derived using (2). They are given explicitly in Appendix A. The data set contains an enormous number N_s of samples implying that the method just outlined is numerically intractable: the sum over the whole state space of visible and hidden vectors is too expensive. In this study we consider networks with N_v in the range of 100 to 1024 nodes and N_h in the range of 81 to 256 nodes. The number of samples N_s we sum over lies in the range of 2^{81} and 2^{1024}. To make progress, we approximate the KL divergence by the contrastive divergence [25]. Rather than summing over the entire state space of visible and hidden vectors, one simply sets the states of the visible units to the training data [27]. This is an enormous simplification. Given a set of visible vectors, the hidden vectors are sampled by setting each h_a to 1 with probability

$$p(h_a = 1|v) = \frac{1}{2}\left(1 + \tanh\left(\sum_i W_{ia}v_i + b_a^{(h)}\right)\right). \tag{10}$$

Likewise, given a set of h_a, we are able to sample visible vectors by setting each v_i to 1 with probability

$$p(v_i = 1|h) = \frac{1}{2}\left(1 + \tanh\left(\sum_a W_{ia}h_a + b_i^v\right)\right). \tag{11}$$

Expectation values for the data are computed using $\hat{h}$, generated using (10) and $\hat{v}$, provided by the training data set.

To determine model expectation values, determine a sample of visible vectors $\{\tilde{v}\}$ and a sample of the hidden vectors $\{\tilde{h}\}$, using the equations (11) and (10) as we now explain. The set $\{\tilde{v}\}$, is calculated using $\{\hat{h}\}$ and equation (11). We then determine $\{\tilde{h}\}$, using $\{\tilde{v}\}$ and equation (10). The equations for the Ath vectors in the sets $\{\hat{h}\}$, $\{\tilde{v}\}$ and $\{\tilde{h}\}$ are thus

$$\hat{h}_a^{(A)} = \tanh\left(\sum_i W_{ia}\hat{v}_i^{(A)} + b_a^{(h)}\right), \tag{12}$$

$$\tilde{v}_i^{(A)} = \tanh\left(\sum_a W_{ia}\hat{h}_a^{(A)} + b_i^{(v)}\right), \tag{13}$$

$$\tilde{h}_a^{(A)} = \tanh\left(\sum_i W_{ia}\tilde{v}_i^{(A)} + b_a^{(h)}\right), \tag{14}$$

Expectation values of the model are approximated using these sets. Again, summing these much smaller sets (and not the complete space of hidden and visible vectors) is an enormous simplification.

Using this approximation the expressions used to train the RBM are

$$\langle v_i h_a \rangle_{data} = \frac{1}{N_s}\sum_A \hat{v}_i^{(A)}\hat{h}_a^{(A)}, \tag{15}$$

$$\langle v_i h_a \rangle_{model} = \frac{1}{N_s}\sum_A \tilde{v}_i^{(A)}\tilde{h}_a^{(A)}, \tag{16}$$

$$\langle v_i \rangle_{data} = \frac{1}{N_s}\sum_A \hat{v}_i^{(A)}, \tag{17}$$

$$\langle v_i \rangle_{model} = \frac{1}{N_s}\sum_A \tilde{v}_i^{(A)}, \tag{18}$$

$$\langle h_a \rangle_{data} = \frac{1}{N_s}\sum_A \hat{h}_a^{(A)}, \tag{19}$$

$$\langle h_a \rangle_{model} = \frac{1}{N_s}\sum_A \tilde{h}_a^{(A)}, \tag{20}$$

where $A = 1, 2, 3, \ldots, N_s$ denotes samples in the training data set made up of N_s samples. These approximations achieve a dramatic speed up in training. Although this method performs well in practice [1], [29], [30], it is difficult to understand when and why the approximations work [27], [31]. This approximation does not follow the gradient of any function [32].

B. RG
RG is a tool used routinely in quantum field theory and statistical mechanics [13]. RG coarse grains by first organizing the theory according to length scales and then averaging over the short distance degrees of freedom. The result is an effective theory for the long distance degrees of freedom. RG thus gives a systematic procedure to determine the dynamical laws governing macroscopic physics of a system with given microscopic laws, and it achieves this by employing coarse graining. The analogy to deep learning should be evident: deep learning also extracts regularities from massive data sets.

At this point it is helpful to make a comment on what a field is. A field is a type of observable. A very simple example of a field could be the temperature inside a room. The measured value of the temperature depends on exactly where in the room you make the measurement[3] and when you make the measurement.[4] Anything that can be measured everywhere and/or everywhen is an example of a field.

To illustrate RG consider the example provided by quantum field theory. To have a concrete example in mind, which is relevant to the discussion that follows, we might study a field $\phi(x)$ describing the magnetization inside a magnet. The value of the field $\phi(x)$ gives the value of the magnetization at position x. $\phi(x)$ can be manipulated as we would normally manipulate a function of x. In particular, we can take derivatives of $\phi(x)$ with respect to x and we can take its Fourier transform to obtain $\phi(k)$. Observables $\mathcal{O}$ are functions (usually polynomials) of the field and its derivatives. Examples of observables are the energy or momentum of the field. To calculate the expected value $\langle\mathcal{O}\rangle$ of observable $\mathcal{O}$,

---

[3]For example, there maybe an air conditioner in the room. Points closer to the air conditioner will be cooler.
[4]Temperature measurements at midnight in the middle of winter will typically be lower than measurements at midday in the middle of summer.

integrate (i.e. average) over all possible field configurations

$$\langle \mathcal{O} \rangle = \int [d\phi] e^{-S} \mathcal{O}. \qquad (21)$$

To make sense of this integral one can work on a lattice. Here we use the term "lattice" to denote an ordered array of points, and we imagine replacing the continuum of space with this discrete structure, so that the set of all possible positions in space is now a discrete set. The integral over all possible field configurations then becomes a product of ordinary integrals, with one integral over the allowed range of the field, at each lattice site. The range of the field is usually taken to be the real number field. The factor e^{-S}, which defines a probability measure on the space of fields, depends on the theory considered. S is called the action of the theory and is also a polynomial in the field and its derivatives, with the coefficients of the polynomial providing the parameters of the theory, things like couplings and masses. A theory is defined by specifying S.

To coarse grain, express the position space field in terms of momentum space components

$$\phi(x) = \int dk \, e^{ik \cdot x} \phi(k). \qquad (22)$$

$e^{ik \cdot x}$ oscillates in position space with wavelength $\frac{2\pi}{k}$. High momentum (big k) components have small wavelengths and encode small distance structure. Low momentum components have huge wavelengths and describe large distance structure. Declare there is a smallest possible structure, implemented by cutting off the momentum modes at a large momentum Λ as follows

$$[d\phi] = \prod_{k < \Lambda} d\phi(k). \qquad (23)$$

RG breaks the integration measure into high and low momentum components $[d\phi] = [d\phi_<][d\phi_>]$ where

$$[d\phi_<] = \prod_{k < (1-\epsilon)\Lambda} d\phi(k)$$
$$[d\phi_>] = \prod_{(1-\epsilon)\Lambda < k < \Lambda} d\phi(k). \qquad (24)$$

The dimensionless parameter ϵ defines the split between the two sets of components. In the end we imagine taking $\epsilon \to 0$ as explained below. RG considers observables that depend only on large scale structure of the theory, i.e. observables that depend only on $\phi_<$. In this case, when computing the expected value of $\mathcal{O}$ we can pull $\mathcal{O}$ out of the integral over $\phi_>$ and integrate over the high momentum components

$$\langle \mathcal{O}(\phi_<) \rangle = \int [d\phi_<] \int [d\phi_>] \, e^{-S} \mathcal{O}(\phi_<)$$
$$= \int [d\phi_<] \, e^{-S_{eff}} \mathcal{O}(\phi_<). \qquad (25)$$

This procedure of splitting momentum components into two sets and integrating over the large momenta defines a new action S_{eff}. Repeating the procedure many times defines the RG flow under which S_{eff} changes continuously. To obtain a continuous flow we should take the limit $\epsilon \to 0$, so that the procedure needs to be repeated an infinite number of times to flow to low momentum. The parameter ϵ should be thought of as a step size in a discrete flow. It is not a physical parameter and must be taken small enough that the results of computations are independent of ϵ. After the flow, one is left with an integral over the very long wavelength modes. This completes the coarse graining: we have a new theory defined by S_{eff}. The new theory uses only long wavelength components of the field and correctly reproduces the expected value of any observable depending only on long wavelength components. Values of the parameters of the theory, which appear in S_{eff}, change under this transformation. In general, many possible terms are generated and appear in S_{eff}. Each possible term defines a coupling of the theory. Each coupling can be classified as marginal (the size of the coupling is unchanged by the RG flow), relevant (the coupling grows under the flow) or irrelevant (the coupling goes to zero under the flow). It is a dramatic insight of Wilson that almost all couplings in any given quantum field theory are irrelevant and so the low energy theory is characterized by a handful of parameters. This is a dramatic (experimentally verified) simplicity hidden in the rather complicated quantum field theory. This simplicity explains why "simple large scale patterns" can emerge from "complicated short distance data". The possibility that the same simplicity is at work in deep learning is a key motivation of this paper.

Although the equation (25) defines the relationship between S and S_{eff}, the connection is rather abstract and a few clarifying remarks are in order. Consider the situation in which we started with an action S and we have flowed to obtain some effective low energy dynamics S_{eff}. The action S is the original action of the theory. In the case of a magnet, this would describe the dynamics of atomic spins, where the relevant length scale is 10^{-10}m. The effective action would describe the dynamics of a classical magnet, where the relevant length scale may be 10^{-3}m or even larger. The renormalization group is the coarse graining that constructs the classical macroscopic physics from the microscopic physics.

Conceptually, the coarse graining performed by RG is well defined. Computationally, it is almost impossible to carry out. To develop a useful calculation scheme, partition the momentum components into tiny sets (i.e. follow (24) with ϵ infinitesimal) and ask what happens when we average over a single tiny set. Two things happen: couplings g_i change $\delta g_i = \beta_i \epsilon$ and the strength of the field changes $\delta \phi = \gamma \epsilon \phi$. One can prove that all observables built using n fields will obey the Callan-Symanzik equation [33]

$$\left(\mu \frac{\partial}{\partial \mu} + \sum_i \beta_i \frac{\partial}{\partial g_i} + n\gamma \right) \langle \mathcal{O} \rangle = 0. \qquad (26)$$

The parameter μ here defines the scale of the effective theory: the smallest wavelength in the effective theory is $\frac{2\pi}{\mu}$. This equation provides a remarkable and simple description of the RG coarse graining that captures the essential

IEEE Access

E. de Mello Koch *et al.*: Is Deep Learning a Renormalization Group Flow?

features of the long distance effective theory. In practice correlation functions are computed and then inserted into the Callan-Symanzik equation. The β_i and γ functions are then read from the resulting equations.

If RG (or a variant of it) is relevant to understanding deep learning, it makes concrete suggestions for the resulting theory. For example, is there an analogue of the Callan-Symanzik equation? One might assign beta functions $\beta_{ia}, \beta_i^{(v)}, \beta_a^{(h)}$ to the weights W_{ia} and biases $b_i^{(v)}, b_a^{(h)}$. These would determine which parameters of the RBM are relevant, irrelevant or marginal.

The RG flow halts at a fixed point, described by a conformal field theory. This field theory enjoys additional symmetries including scale invariance. It is interesting to note that the possibility that scale invariance plays a role in deep learning has been raised in [12], [14]–[17].

Although we have focused mainly on a physical model in this article, we should point out that there are many other applications of the renormalization group formalism. For example, RG has been used to understand the spread of forest fires [34], in the modeling of the spread of infectious diseases in epidemiology [35], for the prediction of earthquakes [35] and more generally, to any system with self organized criticality [36].

C. ISING MODEL

The Ising model is a model for a magnet. The two dimensional model has a discrete variable, called a spin, $\sigma_{\vec{k}} = \pm 1$ on each site of a rectangular lattice. The sites are labeled by a two dimensional vector $\vec{k}$, which has integer components. A state of the system is given by specifying a collection of spins, $\{\sigma_{\vec{k}}\}$, one for each site in the lattice. To refer to a specific state of the spin system we use the notation $\sigma = \{\sigma_{\vec{k}}\}$. Spins on adjacent sites i and j interact with strength J_{ij}. Each spin will also interact with an external magnetic field h_j, with strength μ. The energy of a given configuration $\{\sigma_{\vec{k}}\}$ of spins is determined by the Hamiltonian

$$H = -\sum_{\langle ij \rangle} J_{ij}\sigma_i\sigma_j - \mu \sum_j h_j\sigma_j, \qquad (27)$$

where the first sum is over adjacent pairs, indicated by $\langle ij \rangle$. We simplify the model by setting the external field to zero $h_j = 0$, and by choosing the couplings in the most symmetric possible way $J_{ij} = J$. Since J is an energy we can set it to 1 by choosing units appropriately. The probability of configuration $\sigma = \{\sigma_{\vec{k}}\}$ of spins is given by the Boltzmann distribution, with inverse temperature $\beta \geq 0$

$$P_\beta(\sigma) = \frac{e^{-\beta H(\{\sigma_{\vec{k}}\})}}{Z_\beta}, \qquad (28)$$

where the constant Z_β, the partition function, is given by

$$Z_\beta = \sum_{\{\sigma_{\vec{k}}\}} e^{-\beta H(\sigma)}. \qquad (29)$$

Averages of physical observables are defined by

$$\langle f \rangle_\beta = \sum_\sigma f(\sigma)P_\beta(\sigma). \qquad (30)$$

We study unsupervised learning of the Ising model by an RBM. The visible data that is used to train the network is generated using the probability measure (28). The lattices have a total of $L_v \times L_v$ sites, with each site indexed by a position vector $\vec{k}$. We rearrange this array of spins σ into an $N_v = L_v \times L_v$ dimensional vector by concatenating the rows of the given array. These components of these vectors are the training data input to the visible nodes of the RBM.

There are good reasons to focus on the Ising model. The model has a fixed point in its RG flow. The fixed point is described by a well known conformal field theory (CFT) [37]. This fixed point is an unstable fixed point meaning that generic flows move away from the fixed point. We must tune things carefully if we are to terminate on the fixed point. This tuning is necessary because there is a relevant operator present in the spectrum of the conformal field theory and it tends to push us away from the fixed point. We need to tune the temperature. If the temperature is slightly above the critical temperature, thermal fluctuations destroy the long range correlations that are forming, whilst if we are slightly below the critical temperature, the tendency of spins to align dominates and we find a state with all spins aligned and no fluctuations at all. It is only exactly at the critical temperature that the system exhibits the interesting long range correlations that are described by the CFT. The papers [15], [16] argue that the RBM flow always flows to the fixed point. This challenges conventional wisdom and it suggests a different kind of coarse graining to that employed by RG, is at work. A distinct proposal [14] claims that the RG flow arises by stacking RBMs to produce a classic deep learning scenario. Each layer of the deep network performs a step in the flow.

At the Ising model fixed point, detailed checks of both proposals are possible. There are CFT observables, known as primary operators, whose correlators are power laws of distances on the lattice. The powers entering these power laws are known, so that we have a rich and detailed data set that the RBM must reproduce if it is indeed performing an RG coarse graining. This is a compelling motivation for the model. Another advantage of the model is simplicity: it is a model of spins which take the values ± 1 so it defines a simple model with discrete variables, well suited to numerical study and naturally accommodated in the RBM framework. Finally, the Ising model is not that far removed from real world applications: the Ising Hamiltonian favors configurations with aligned neighboring spins. Thus, at low enough temperatures "smooth" slowly varying configurations of spins are favored. This is similar to data defining images for example, where neighboring pixels are likely to have the same color. In slightly poetic language one could say that at low temperatures the Ising model favors pictures and not speckle.

We end this section with a summary of the most relevant features of the Ising model fixed point. At the critical

E. de Mello Koch *et al.*: Is Deep Learning a Renormalization Group Flow?

IEEE *Access*

temperature

$$T_c = \frac{2J}{k \ln(1 + \sqrt{2})}, \tag{31}$$

where J is the interaction strength and k is the Boltzmann constant, the Ising model undergoes a second order phase transition. The critical temperature is given by $T_c \approx 2.269$ when $J = 1$. There are two competing phases: an ordered (low temperature) phase in which spins align producing a macroscopic magnetization, and a disordered (high temperature) phase in which spins fluctuate randomly and the magnetization averages to zero. At the critical point the Ising model develops a full conformal invariance and one can use the full power of conformal symmetry to tackle the problem. The field which takes values ± 1 in the Ising model is a primary field, of dimension $\Delta = \frac{1}{8}$. The two and three point correlation functions of primary fields are determined by conformal invariance to be

$$\langle \phi(\vec{x}_1)\phi(\vec{x}_2) \rangle = \frac{B_1}{|\vec{x}_1 - \vec{x}_2|^{2\Delta}}, \tag{32}$$

$$\langle \phi(\vec{x}_1)\phi(\vec{x}_2)\phi(\vec{x}_3) \rangle = \frac{B_2}{|\vec{x}_1 - \vec{x}_2|^{\Delta}|\vec{x}_1 - \vec{x}_3|^{\Delta}|\vec{x}_2 - \vec{x}_3|^{\Delta}}, \tag{33}$$

where B_1 and B_2 are constants. Since we study the Ising model on a lattice, the positions $\vec{x}_1, \vec{x}_2$ and $\vec{x}_3$ appearing in the above correlation functions refer to sites in a lattice. There is also a primary operator in the Ising model (which we describe below) with a dimension $\Delta = 1$. These correlation functions must be reproduced by the RBM if it is indeed flowing to the critical point of the Ising model.

III. FLOWS DERIVED FROM LEARNED WEIGHTS

In this section we consider the RBM flows introduced in [15], [16]. These flows use the weight matrix W_{ia}, and bias vectors $b_i^{(v)}$ and $b_a^{(h)}$, obtained by training, to define a continuous flow from an initial spin configuration to a final spin configuration. The flow appears to exhibit a fascinating behavior: given any initial snapshot, the RBM flows towards the critical point of the Ising model. This is in contrast to the RG which flows away from the fixed point. In addition, the number of spins in the configuration is a constant along the RBM flow. In contrast to this, the number of spins in the configuration decreases along the RG flow, as high energy modes are averaged over to produce the coarse grained description. Despite these differences, the flow of [15], [16] appears to produce configurations ever closer to the critical temperature and these configurations yield impressively accurate predictions for the critical exponents of the Ising magnet. Our goal in this section is to further test if the RBM flow produces configurations at the critical point of the Ising model. We explore the spatial dependence of spin correlations in configurations produced by the flow. Our results prove that on large scales the Ising critical point configurations are correctly reproduced. However, we are also able to prove that as one starts to probe smaller scales there are definite quantifiable departures from the Ising predictions.

A. RBM FLOW

RBM flows [15], [16] are generated using equation (13) together with the trained weight matrix, W_{ia}, and bias vectors, $b_i^{(v)}$ and $b_a^{(h)}$. Our data set $\hat{v}^{(A)}$ is labeled by an index A. For each value of A, $\hat{v}^{(A)}$ is a collection of spin values, one for each lattice site. The RBM flow is generated through a series of discrete steps, with each step producing a new data set of the same size as the original. Denote the data set produced after k steps of flow by $\tilde{v}^{(A,k)}$, and by convention we identify $\tilde{v}^{(A,0)}$ with the original training set. Apply equation (13) to $\hat{v}^{(A,k)}$ and then apply (14) to carry out a single step of the RBM flow, with the result $\tilde{v}^{(A,k+1)}$. The flow proceeds by repeatedly applying equations (13) and (14). Concretely, for a flow of length n, we have

$$\tilde{v}_i^{(A,1)} = \tanh\left(\sum_a W_{ia}\hat{h}_a^{(A)} + b_i^{(v)}\right),$$

$$\tilde{v}_i^{(A,2)} = \tanh\left(\sum_a W_{ia}\tilde{h}_a^{(A,1)} + b_i^{(v)}\right),$$

$$\vdots$$

$$\tilde{v}_i^{(A,n)} = \tanh\left(\sum_a W_{ia}\tilde{h}_a^{(A,n-1)} + b_i^{(v)}\right). \tag{34}$$

where

$$\tilde{h}_a^{(A)} = \tanh\left(\sum_i W_{ia}\tilde{v}_i^{(A)} + b_a^{(h)}\right),$$

$$\tilde{h}_a^{(A,1)} = \tanh\left(\sum_i W_{ia}\tilde{v}_i^{(A,1)} + b_a^{(h)}\right),$$

$$\tilde{h}_a^{(A,2)} = \tanh\left(\sum_i W_{ia}\tilde{v}_i^{(A,2)} + b_a^{(h)}\right),$$

$$\vdots$$

$$\tilde{h}_a^{(A,n)} = \tanh\left(\sum_i W_{ia}\tilde{v}_i^{(A,n)} + b_a^{(h)}\right). \tag{35}$$

Note that the length of the vector $\tilde{v}^{(A,k)}$ is a constant of the flow and consequently there is not obviously any coarse graining implemented.

B. NUMERICAL RESULTS

This section considers statistical properties of configurations produced by the RBM flow. At the Ising critical point, the theory enjoys a conformal invariance. Using this symmetry a special class of operators with a definite scaling dimension Δ can be identified. The utility of these operators is that their spatial two point correlation functions drop off as a known power of the distance between the two operators, as reviewed above in equation (32). These two point functions can be evaluated using the RBM flow configurations and, if these configurations are critical Ising states, they must reproduce the known correlation functions. This is one of the checks performed and it detects discrepancies with the
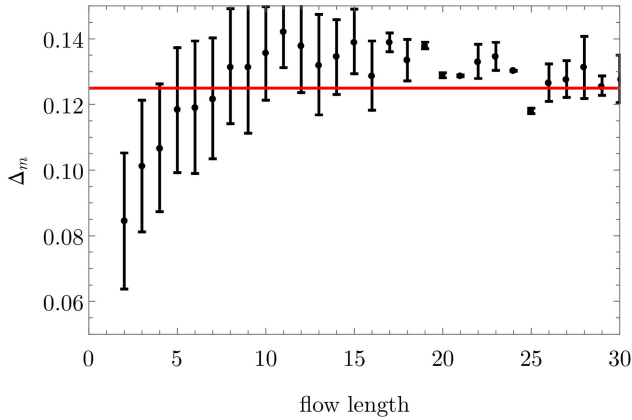
IEEE *Access*

E. de Mello Koch *et al.*: Is Deep Learning a Renormalization Group Flow?

FIGURE 2. Estimates of the scaling dimension Δ_m versus flow length, obtained using the average magnetization of flows at temperatures $T = 2.1$, 2.2 and 2.3. The red line indicates the value of $\Delta = 0.125$ at the critical point. After approximately 8 flows, Δ_m converges to this critical value. The error bars are determined using Mathematica's NonlinearModelFit function. Mathematica uses the Student's t-distribution to calculate a confidence interval for the given parameters with a 90% confidence level.

Ising model predictions. There are two primary operators we consider. This first is the basic spin variable minus its average value $s_{ij} = \sigma_{ij} - \bar{\sigma}$. The prediction for the two point function is (32) with $\Delta_s = \frac{1}{8}$. This correlator falls off rather slowly, so that this two point function probes the large scale features of the RBM flow configurations. The RBM flow nicely reproduces this correlator and in fact, this is enough to reproduce the critical exponent for the Ising model consistent with the results of [16]. One should note however that our computation and those of [16] could very well have disagreed, since they probe different things. The critical exponent evaluated in [16] uses the magnetization computed from different flows generated by the RBM, at temperatures around the critical temperature. Magnetization measures the average of the spin in the lattice. It is blind to the spatial location of each spin. On the other hand, the two-point correlation function is entirely determined by the spatial location of spins in a single flow configuration. Thus, the two point correlation function uses data at a single temperature, but uses detailed spatial dependence of the lattice state. We also consider a second primary operator

$$\epsilon_{ij} = s_{ij} \cdot (s_{i+1,j} + s_{i-1,j} + s_{i,j+1} + s_{i,j-1}) - \bar{\epsilon}, \quad (36)$$

which has $\Delta_\epsilon = 1$. We have again subtracted off the average value of the operator. This correlation function falls off much faster and is consequently a probe of shorter scale features of the RBM configurations. The RBM flow fails to reproduce this correlation function, indicating that the RBM configurations differ from those of the critical Ising model. They have the same long distance features, but differ on shorter length scales.

Consider an RBM network with 100 visible nodes and 81 hidden nodes. This corresponds to an input lattice of size 10×10 and an output lattice of size 9×9. The number of

visible and hidden nodes is chosen to match [16], so that we can compare our results to existing literature. We would like to study a lattice that is as small as possible but large enough to detect the power law fall off of the correlation functions we study. The power law behavior is given in the scaling dimensions Δ_s and Δ_ϵ. We demonstrate that when studying a lattice of size 9×9 or larger we correctly determine the values for Δ_s and Δ_ϵ using configurations generated from MC simulations. These results are shown in Figures 6a and 5b.

The network trains on data generated by Monte Carlo simulations which use the Boltzmann distribution given in equation (28) [38]. The training data set includes 20000 samples at each temperature, ranging from 0 to 5.9 in increments of 0.1. This gives a total of 1200000 configurations. Training uses 10000 iterations of contrastive divergence, performed with the update equations (12) to (20) which are derived from equations (7), (8) and (9) [39].

Once the flow configurations are generated, following [15], [16], a supervised network is used to measure the temperature of each flow. The supervised network allows us to measure discrete temperatures of $T = 0, 0.1, \ldots, 5.9$.

We train a network which consists of three layers, an input layer with 100 nodes, a hidden layer with 80 nodes and an output layer with 60 nodes which correspond to the 60 temperatures we want to measure. All nodes in the input layer are connected to all nodes in the hidden layer, and all nodes in the hidden layer are connected to all nodes in the output layer. No connections between nodes within the same layer exist.

The Ath sample of input data, $Z_A^{(1)}$ is transformed by the hidden layer using the hyperbolic tangent function $f(x) = \tanh(x)$ as follows

$$Z_{Aa}^{(2)} = f(\sum_i Z_{Ai}^{(1)} W_{ia}^{(1)} + b_a^{(1)}). \quad (37)$$

The output layer then transforms $Z_A^{(2)}$ into an output probability using the softmax function $g(x)$

$$Z_{A\mu}^{(3)} = g(\sum_a Z_{Aa}^{(2)} W_{a\mu}^{(2)} + b_\mu^{(2)}) =: g(U_{A\mu}^{(2)}), \quad (38)$$

where the softmax function normalizes the output to sum to 1 as follows

$$g(U_{A\mu}^{(2)}) = \frac{\exp(U_{A\mu}^{(2)})}{\sum_\nu \exp(U_{A\nu}^{(2)})}. \quad (39)$$

The output of the trained network can thus be interpreted as probabilities for the temperatures we measure. The highest probability in the output is taken as the temperature for the given input.

We make use of the Keras library [40] to train this network using the back-propagation algorithm [41]. The cost function used to train the network is the KL divergence as is used in [16]. We choose a learning rate of $\epsilon = 0.1$ and train the network for 3000 epochs. In Figure 3 we show the validation and training loss versus the number of training epochs. The supervised network is trained using the same data set used to

E. de Mello Koch *et al.*: Is Deep Learning a Renormalization Group Flow?

IEEE *Access*

train the RBM as well as corresponding labels for each vector. These labels are one hot encoded vectors which correspond to the temperatures $T = 0, 0.1, \ldots, 5.9$. Here we use a split of 40 % of the input data for validation and 60 % of the input data for training. Figure 3 shows that the supervised network has converged to a loss very close to zero after 3000 epochs.
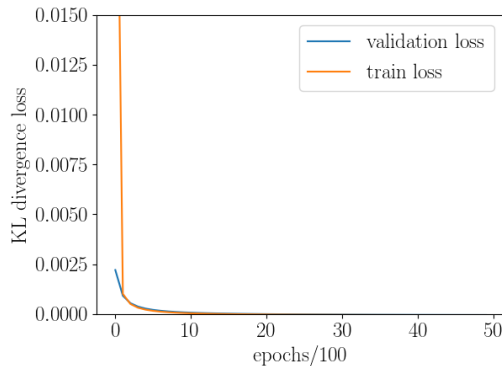

FIGURE 3. Plot showing the KL divergence loss function of the supervised temperature measuring network versus the number of epochs divided by 100. This network consists of an input layer with 100 nodes, a hidden layer with 80 nodes and and output layer of 60 nodes.

To estimate Δ using magnetization the study [16] selects flows at temperatures close to T_c, where the average magnetization m depends on temperature as

$$m \approx \frac{A|T - T_c|^{\Delta_m}}{T_c}. \tag{40}$$

In [16] the magnetization is expanded about the critical point to give values for $A = 1.22$ and $\Delta_m = 1/8 = 0.125$

$$m \approx 1.222 \frac{|T_c - T|^{1/8}}{T_c}. \tag{41}$$

We denote Δ obtained by this fitting as Δ_m. The fit also determines A. The value of $T_c = 2.269$ is a known theoretical value for the 2D Ising model with coupling strength $J = 1$. The fit uses the magnetization computed at temperatures $T = 2.1, 2.2$ and 2.3. A plot of Δ_m versus flow length is given in Figure 5. The error bars shown in Figure 5 (and all subsequent plots) are determined using Mathematica's NonlinearModelFit function. The error bars show the standard error obtained from the regression. Mathematica uses the Student's t-distribution to calculate a confidence interval for the given parameters with a 90% confidence level.

Our results indicate that we converge to the correct critical value $\Delta_m = 0.125$ for flows of length 26. It is evident from Figure 5 that the flow converges to the theoretical value depicted by the red horizontal line. The convergence of the flow is also reflected as a decrease in the size of the error bars as the flow proceeds. In [16] the value for A/T_c is found to be 0.931. The value of A/T_c which we find after convergence is 0.942 with a 90% confidence interval within ± 0.0168794 of this value. Although the values we obtain for A and Δ_m are consistent with the results of [16], they have used a flow of
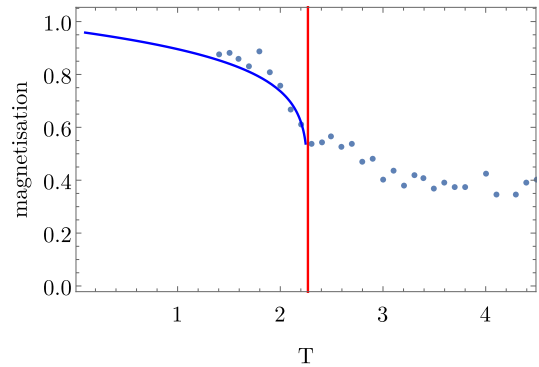

FIGURE 4. Plot showing the fit for equation (40). The blue line shows the function $m = 0.942(T - 2.269)^{0.126}$. This function is fitted using the dots which show the average magnetization for flows of length 26 at various temperatures measured using the supervised network. The vertical red line shows the critical temperature of $T_c \approx 2.269$. Equation (40) is fitted using data points for temperatures near T_c.

length 9, at which point Δ_m is correctly determined. As just described, we need longer flows for convergence. The fitting of Δ_m and A, for a flow length of 26, is shown in Figure 4.

We now shift our focus to consider spatial two point correlation functions computed using the configurations generated by the RBM flows. The correlators are calculated using the flow configuration and the result is then fitted to the function in equation (32) to estimate Δ. We denote this estimate by Δ_s as it is determined using spatial information. For RBM flows at the critical temperature, the prediction which is determined by theory is $\Delta_s = 0.125$ at $T = 2.269$, as explained above. We expect that for temperatures below the critical temperature Δ_s will be less than the theoretical value of 0.125. For temperatures that are above the critical temperature we expect that Δ_s will be greater than 0.125. With a lattice of 10 by 10 spins, we find $\Delta_s = 0.1263$ using Monte Carlo Ising model configurations. A plot showing this estimate can be seen in Figure 5b. The point of this exercise is to demonstrate that a lattice of size 10 by 10 is large enough to estimate the scaling dimension of interest and to verify the integrity of the data set used in the numerical simulations.

Figure 5a shows the scaling dimension Δ_s versus the flow length, for RBM flows at temperatures of 2.2 (in gray) and 2.3 (in black). The red horizontal line indicates the scaling dimension at the critical point. The results are intuitively appealing. The gray points in Figure 5a show estimates of Δ_s from flows slightly below the critical temperature, where the scaling dimension is slightly underestimated. Below the critical temperature spins are more likely to align and so the correlator should fall off more slowly than at the critical temperature. This is what our results show. The black points in Figure 5a show Δ_s estimated using flows slightly above the critical temperature. The scaling dimension is over estimated, again as expected. Selecting flows at T_c would determine the scaling dimension in between the values shown iin Figure 5a. This gives a value very close to the theoretical value of $\Delta_s = 0.125$.
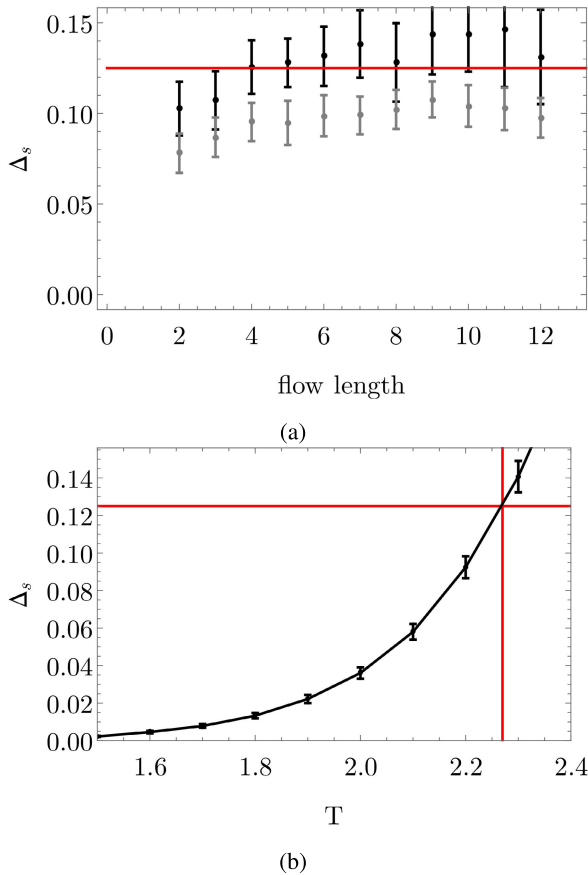
IEEE *Access*

E. de Mello Koch *et al.*: Is Deep Learning a Renormalization Group Flow?

(a)

(b)

FIGURE 5. Plot showing the estimated scaling dimension Δ_s versus flow length using the two-point correlation function for (a) flows at $T = 2.2$ (in gray) and flows at $T = 2.3$ (in black). Plot (b) shows the estimated scaling dimension versus temperature for the Ising model data used for training. The error bars are determined using Mathematica's NonlinearModelFit function. Mathematica uses the Student's t-distribution to calculate a confidence interval for the given parameters with a 90% confidence level.

(a)

(b)

FIGURE 6. Plots showing Δ_ϵ calculated using (a) Monte Carlo Ising model data on a 10 by 10 lattice (in black) and a 9 by 9 lattice (grey), (b) RBM flows at a temperature of $T = 2.2$, 2.3 and 2.4. Error bars in plot (a) indicate a 90% confidence interval. No error bars are shown in (b) as the error bars are larger than the y range. The error bars shown in (a) are determined using Mathematica's NonlinearModelFit function. Mathematica uses the Student's t-distribution to calculate a confidence interval for the given parameters with a 90% confidence level.

The two point correlation functions for the spin variable establish that the critical Ising states and the states produced by the RBM flow share the same large scale spatial features. We will now consider the two point correlation function of the ϵ_{ij} field, which probes spatial features on a smaller scale. Using critical Ising data generated using Monte Carlo, on a lattice of size 10 by 10 and 9 by 9, we estimate Δ_ϵ at various temperatures as shown in Figure 6a. We can estimate the error for the point we are interested in, at $\Delta_\epsilon = 1$ and $T = 2.269$ which is depicted by the red vertical and horizontal lines shown in Figure 6a. The error bar of the estimate for the grey line (9 × 9 lattice) is ±0.059 for an estimated value of Δ_ϵ being 1.013. For the black line (10 × 10 lattice) we have an error bar of ±0.044 on an estimated value of 1.023 for Δ_ϵ. This is determined using the average of the values as well as the errors on either side of the red vertical line at $T = 2.269$. Figure 6a, demonstrates that a lattice of size 9 × 9 is large enough to correctly determine the ϵ scaling dimension Δ_ϵ.

The intersection of the red horizontal and vertical lines cross the critical temperature and prediction $\Delta_\epsilon = 1$. Interpolating the Ising data with a continuous curve, we would pass
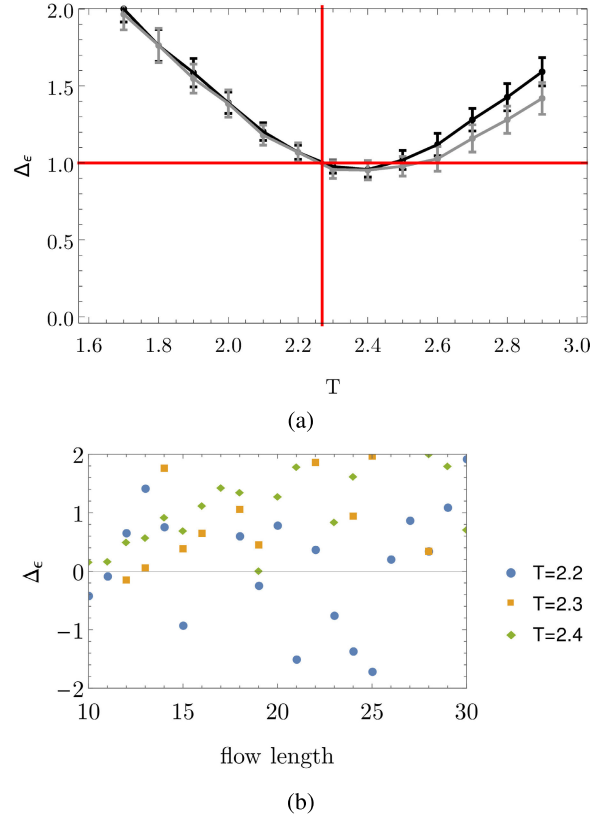
through the intersection point, as predicted. These numerical results again demonstrate that a lattice of size 10 by 10 is large enough for the questions we consider. The RBM flows are unable to confirm this prediction. Indeed, the RBM flows near T_c are summarized in Figure 6b. None of the three temperatures shown have a value of Δ_ϵ that converges with flow length.

The fact that the RBM produces configurations that correctly reproduce the correlation function of the spin field s_{ij} but not of the ϵ_{ij} implies that although the spatial correlations encoded into the RBM flow configurations agree with those of the critical Ising configurations at long length scales, the two start to differ on smaller length scales. This conclusion agrees with [16] which also finds differences between the RBM flow and RG. Reference [16] considers $h \neq 0$ and uses different arguments to reach the conclusion.

IV. FLOWS DERIVED FROM DEEP LEARNING
The RBM flows of the previous section provide one possible link to RG. An independent line reasoning, developed in [14], claims a mapping between deep learning and RG. The idea is not that there is an analogy between deep learning and RG,

E. de Mello Koch *et al.*: Is Deep Learning a Renormalization Group Flow?

IEEE *Access*

but rather, that the two are to be identified. The argument for this identity starts from the energy function of the RBM, which is

$$E(\{v_i, h_a\}) = b_a h_a + v_i W_{ia} h_a + c_i v_i. \tag{42}$$

This energy determines the probability of obtaining configuration $\{v_i, h_a\}$ as

$$p_\lambda(\{v_i, h_a\}) = \frac{e^{-E(\{v_i, h_a\})}}{\mathcal{Z}}, \tag{43}$$

where $\lambda = \{b_a, W_{ia}, c_i\}$ are the parameters of the RBM model which are tuned during training. Marginal distributions for hidden and visible spins are defined as follows

$$p_\lambda(\{h_a\}) = \sum_{\{v_i\}} p_\lambda(\{v_i, h_a\}) = \text{tr}_{v_i} \, p_\lambda(\{v_i, h_a\}),$$
$$p_\lambda(\{v_i\}) = \sum_{\{h_a\}} p_\lambda(\{v_i, h_a\}) = \text{tr}_{h_a} \, p_\lambda(\{v_i, h_a\}). \tag{44}$$

The equations (44) are key equations of the RBM and [14] essentially uses these to characterize the RBM. The comparison to RG is made by employing a version of RG known as variational RG. This is an approximate method that can be used to perform the renormalization group transformation in practice. As explained in Appendix B, the variational RG uses an operator $T(\{v_i, h_a\})$ defined as follows

$$\frac{e^{H_\lambda^{RG}(\{h_a\})}}{\mathcal{Z}} = \text{tr}_{v_i} \frac{e^{T(\{v_i, h_a\}) - H(\{v_i\})}}{\mathcal{Z}}. \tag{45}$$

In this formula, $H(\{v_i\})$ is the microscopic Hamiltonian describing the dynamics of the visible spins and $H_\lambda^{RG}(\{h_a\})$ is the coarse grained Hamiltonian describing the hidden spins where here λ defines the parameters of the variational RG. Block spin averaging is discussed in more detail in Appendix B-B. The operator $T(\{v_i, h_a\})$ is required to obey (see in equation (69))

$$\text{tr}_{h_a} \, e^{T(\{v_i, h_a\})} = 1, \tag{46}$$

which obviously implies that

$$\text{tr}_{h_a} \, e^{T(\{v_i, h_a\}) - H(\{v_i\})} = e^{-H(\{v_i\})}. \tag{47}$$

Notice that (45) and (47) exactly match (44) as long as we identify

$$T(\{v_i, h_a\}) = -E(\{v_i, h_a\}) + H(\{v_i\}). \tag{48}$$

This then implies that

$$\frac{e^{H_\lambda^{RG}(\{h_a\})}}{\mathcal{Z}} = \text{tr}_{v_i} \frac{e^{T(\{v_i, h_a\}) - H(\{v_i\})}}{\mathcal{Z}}$$
$$= \text{tr}_{v_i} \frac{e^{-E(\{v_i, h_a\})}}{\mathcal{Z}}$$
$$= \frac{e^{-H_\lambda^{RBM}(\{h_a\})}}{\mathcal{Z}}, \tag{49}$$

which is the central claim of [14].

The above argument proves an equivalence between deep learning and RG if and only if the equations (44) provide a unique characterization of the joint probability function

$p_\lambda(\{v_i, h_a\})$. This is not the case: it is easy to construct functions $p_\lambda(\{v_i, h_a\})$ that obey (44), but are nothing like either the RBM or RG joint probability functions. As an example, define

$$\rho(\{v_i\}) = \frac{\text{tr}_{h_a} \left(e^{T(\{v_i, h_a\}) - H(\{v_i\})} \right)}{\mathcal{Z}},$$
$$\tilde{\rho}(\{h_a\}) = \frac{\text{tr}_{v_i} \left(e^{T(\{v_i, h_a\}) - H(\{v_i\})} \right)}{\mathcal{Z}}, \tag{50}$$

where

$$\mathcal{Z} = \sum_{v_i, h_a} e^{T(\{v_i, h_a\}) - H(\{v_i\})}. \tag{51}$$

We clearly have $\text{tr}_{v_i}(\rho(\{v_i\})) = 1 = \text{tr}_{h_a}(\tilde{\rho}(\{h_a\}))$ which implies that

$$A_\lambda(\{v_i, h_a\}) = \tilde{\rho}(\{h_a\})\rho(\{v_i\}), \tag{52}$$

obeys (44). It is quite clear that in $A_\lambda(\{v_i, h_a\})$ there are no correlations between the hidden and visible spins

$$\langle v_j h_b \rangle = \text{tr}_{v_i, h_a}(v_j h_b \, A_\lambda(\{v_i, h_a\}))$$
$$= \text{tr}_{v_i}(\rho(\{v_i\})v_j) \, \text{tr}_{h_a}(\tilde{\rho}(\{h_a\})h_b)$$
$$= \langle v_j \rangle \langle h_b \rangle, \tag{53}$$

so that we would reject it as a possible model of either the RG quantity

$$\mathcal{Z}^{-1} e^{T(\{v_i, h_a\}) - H(\{v_i\})}, \tag{54}$$

or of the RBM quantity

$$\mathcal{Z}^{-1} e^{-E(\{v_i, h_a\})}. \tag{55}$$

In addition to clarifying aspects of the argument of [14], the joint correlation functions between visible and hidden spins can be used to characterize the RG flow, as we now explain. The RG flow "coarse grains" in position space: a "block of spins" is replaced by an effective spin, whose magnitude is the average of the spins it replaces. Since correlations between microscopic spins fall off with distance, an RG coarse graining implies that because the hidden spin is a linear combination of nearby visible spins, the correlation function between hidden and visible spins reflects a correlation between a hidden spin and a cluster of visible spins. This produces distinctive correlation functions, some examples of which are plotted in Figure 7. We will search for this distinctive signal in the $\langle v_i h_a \rangle$ correlator, to find quantitative evidence that deep learning is indeed performing an RG coarse graining.

A. NUMERICAL RESULTS

Our numerical study aims to do two things: First, we establish whether there are RG-like patterns present within the correlator $\langle v_i h_a \rangle$, for correlators computed using the patterns generated by an RBM flow. If these patterns are indeed present, this constitutes strong evidence in favor of the connection
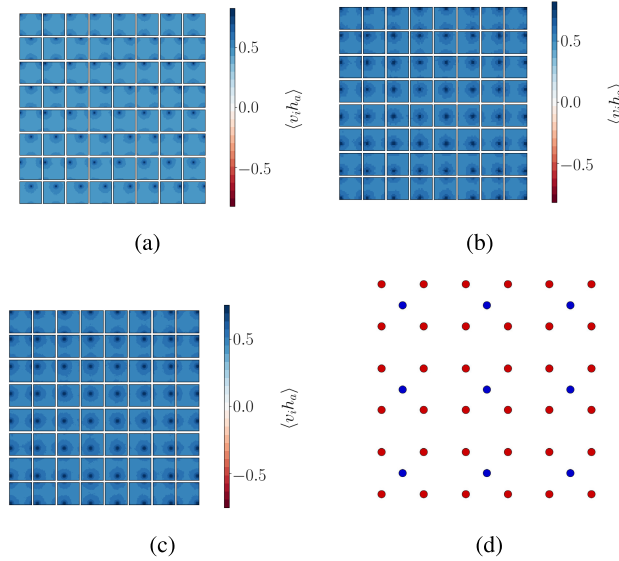
IEEE Access·

E. de Mello Koch *et al.*: Is Deep Learning a Renormalization Group Flow?

FIGURE 7. Correlation plots for Ising model visible data with lattice size 32 by 32 at T_c and RG decimated Ising data of sizes 16 by 16 (one step of RG) and 8 by 8 (two steps of RG). (a) shows visible Ising data correlated with configurations resulting after one step of RG. (b) shows correlations between configurations resulting after one step of RG and configurations resulting after two steps of RG. (c) shows correlations between Ising model visible data and configurations resulting after two steps of RG. (d) shows one step of RG. The red dots show the original visible lattice and the blue dots show the lattice obtained after one step of RG. Each blue dot is surrounded by four red dots. The value of the blue dot is determined by averaging the surrounding four red dots.

between RG and deep learning. The $\langle v_i h_a \rangle$ correlator is calculated using

$$\langle v_i h_a \rangle = \frac{1}{N_s} \sum_{A=1}^{N_s} v_i^{(A)} h_a^{(A)}, \tag{56}$$

where $A = 1, 3, \ldots, N_s$ with N_s being the number of samples, $i = 1, 2, \ldots, N_v$ labels the visible nodes within a visible vector and $a = 1, 2, 3, \ldots, N_h$ labels the nodes within a hidden vector.

For each hidden node h_a we can produce a plot which shows how this hidden node is correlated to the $i = 1, 2, 3, \ldots, N_v$ visible nodes, v_i. This gives us a total of N_h plots. Each panel within the plot for h_a shows the N_v correlation values for $\langle v_i h_a \rangle$. By arranging these panels according to the lattice sites of the visible spins we get a grid of $L_v \times L_v = N_v$ values for the correlators $\langle v_i h_a \rangle$, where a is fixed for the given plot and i runs from 1 to N_v.

By doing this we can determine if a given hidden node is correlated to a local patch of visible nodes which are neighbors on the original lattices produced from MC. This local information is not encoded inherently in the RBM so learning about the nearest neighbour interactions present in the 2D Ising model would show promise that RBMs are performing a coarse graining related to that of RG. We find that RG-like patterns do indeed emerge.

Second, according to the proposal of [14], in a deep network each layer that is stacked to produce the depth of the

network performs one step in the RG flow. With this interpretation in mind, it may be useful to compare how a network with multiple stacked RBMs learns as compared to a network with a single layer. This issue is explored below.

The training data is a set of 30000 configurations of Ising model 32 by 32 lattices, near the critical temperature $T = 2.269$. The dataset is generated using Monte Carlo simulations. An input lattice length of 32 allows a large enough final configuration even after two steps of RG, corresponding to stacking two RBMs. In each step of the RG, the number of lattice sites is reduced by a factor of 4. Thus, we flow from lattices with 1024 sites to lattices with 64 sites. We enforce periodic boundary conditions. To find signals of RG in the correlation functions the maximum distance between operators in a correlator must be large enough that the spin-spin correlation has dropped to zero. We have confirmed that our lattice is large enough, judged by this criterion.

Having described the conditions of our numerical experiment, we consider the correlators $\langle v_i h_a \rangle$ generated when the hidden neurons h_a are generated from the visible neurons v_i using RG. Our goal is to understand the patterns appearing in correlation functions, that are a signature of the RG. In Figure 7d the process of decimation used in our RG is explained. The red dots, representing the visible lattice, are averaged (coarse grained) to produce the blue dots which define the lattice after a single step of the RG. The four spin values located at the red dots surrounding each blue dot are averaged to obtain the value of the spin at the new (blue) lattice point. This process clearly reduces the number of lattice sites by a factor of four.

Using the visible data which populates a 32 by 32 lattice, we populate lattices of size 16 by 16 and 8 by 8 spins by applying the RG and then calculate the various possible $\langle v_i h_a \rangle$ correlations.

Figure 7a shows the $\langle v_i h_a \rangle$ correlation function that results from a single RG step. Each panel of the three Figures 7a, 7b and 7c, shows how a given hidden spin is correlated with the visible spins. We can clearly see a peak in correlation values around the spatial location of the hidden spin. This is the signal of RG coarse graining: small spatially localized collections of spins are replaced by their average value. We can go into a little more detail: the patches of large correlation in Figures 7b and 7c are larger in size than those of Figure 7a. This makes sense since each step of the RG implies ever larger spatial regions of the spins are being averaged to produce the coarse grained variables. The fact that the spins that are averaged are spatially localized is a direct consequence of the fact that the Ising model Hamiltonian is local in space so that spatially adjacent spins have similar behaviors. In more general big data settings it may be harder to decide if the coarse graining is RG-like or not, since it might not be clear what is meant by spatial locality.

Having established the signal characteristic of the RG flow, we will now search for this signal in the $\langle v_i h_a \rangle$ correlators computed using the configurations generated from the RBM flow. We consider configurations generated by a stacked
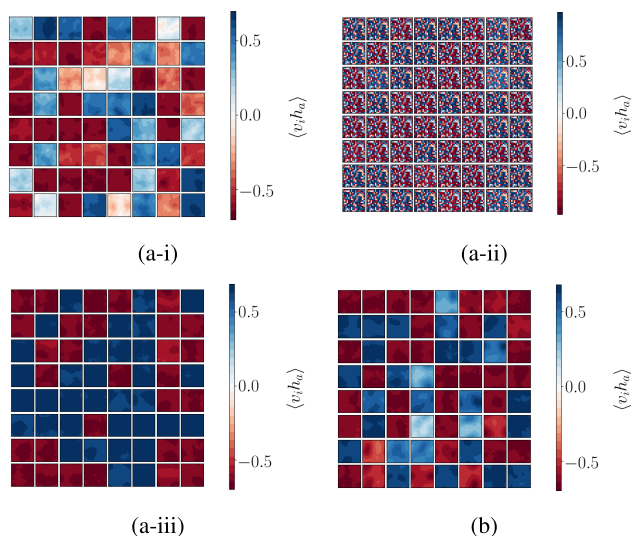
(a-i) (a-ii)

(a-iii) (b)

FIGURE 8. Plots showing the correlation values for (a) the stacked RBMs various layers and (b) the single RBM. (a-i) shows correlations between visible Ising data (1024 nodes) at T_c and outputs from the first stacked RBM (256 nodes). (a-ii) shows correlations between outputs from the first stacked RBM (256 nodes) and outputs from the second stacked RBM (64 nodes). (a-iii) shows correlations between visible Ising data (1024 nodes) at T_c and outputs from the second stacked RBM (64 nodes). (b) shows correlations between visible Ising data (1024 nodes) at T_c and outputs from the single RBM (64 nodes).

network with an RBM having 1024 visible nodes and 256 hidden nodes cascading into a second RBM having 256 visible nodes and 64 hidden nodes. We also consider configurations generated by a single RBM network with 1024 visible nodes and 64 hidden nodes. The factor of 4 relating the number of visible to hidden nodes is chosen to mimic the decimation of lattice sites in each step of the RG. The networks are trained on the same data used as input for the RG considered above. Training is through 10000 steps of contrastive divergence [28].

Figures 8a-i to 8a-iii show plots for the stacked RBM and Figure 8b for the single RBM network. Figure 8a-iii shows the correlation functions between the visible vectors input to the first network in the stack and the final hidden vectors output from the stack and is to be compared to the corresponding RG result in Figure 7c. The two patterns are very similar suggesting that the trained RBM is indeed performing something like the RG coarse graining.

To quantitatively compare the patterns we observe in the $\langle v_i h_a \rangle$ correlators produced by RG to those produced by the RBM we make use of a two point correlation function. When we perform an RG coarse graining we average local nearby nodes from the input (visible lattice) to obtain the output (hidden lattice). This local averaging is encoded in the $\langle v_i h_a \rangle$ plots by bright spots. The bright spots correspond to a specific hidden node being highly correlated to a patch of local visible spins. In each $\langle v_i h_a \rangle$ plot, the hidden node we consider is fixed and we plot its correlation with all visible nodes. If we denote each value of $\langle v_i h_a \rangle$ by x_i we calculate the two point correlator $\langle x_i x_j \rangle$ between values $\langle v_i h_a \rangle$ and $\langle v_j h_a \rangle$ summed
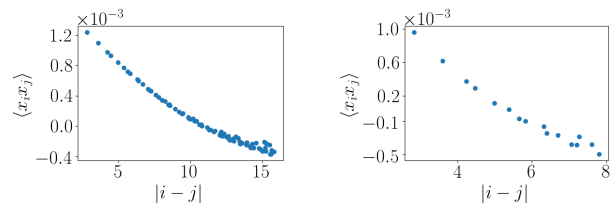
(a) Two point correlator versus distance for Figure 7a.

(b) Two point correlator versus distance for Figure 7b.
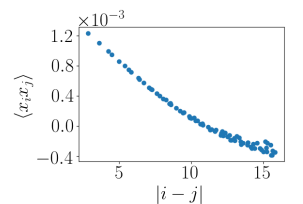
(c) Two point correlator versus distance for Figure 7c.

FIGURE 9. Plots showing $\langle x_i x_j \rangle$ for RG $\langle v_i h_a \rangle$ plots shown in Figures 7a, 7b and 7c.

over all hidden nodes

$$\langle x_i x_j \rangle = \frac{1}{N_h} \sum_{a=1}^{N_h} \langle v_i h_a \rangle \times \langle v_j h_a \rangle. \qquad (57)$$

By calculating this quantity, we learn about the size of the correlated patches in the $\langle v_i h_a \rangle$ plots. We can plot the value of $\langle x_i x_j \rangle$ versus the distance, $|i-j|$. This quantity tells us important information about the size of the correlated patches. We average the values of $\langle x_i x_j \rangle$ where the distances $|i-j|$ are equal. The patches present in $\langle v_i h_a \rangle$ will thus be detected regardless of where they appear in the plot. If we do have local patches of high correlation, $\langle x_i x_j \rangle$ will be peaked at short distances and as distance increases, $\langle x_i x_j \rangle$ will decrease in value.

For RG, the plots seen in Figure 9 show a linear fall off in the correlator as distance increases. The fall off of these correlators is in the order of magnitude of 10^{-4}. In Figure 10 the behavior of the RBM correlator is shown. Figures 10a and 10c show similar behavior to that seen for RG. There is a linear decrease in $\langle x_i x_j \rangle$ in the same order of magnitude of 10^{-4}. A difference between these plots is that the RBM correlators are offset. We do not have an explanation for this offset.

We also study $\langle x_i x_j \rangle$ where the visible lattice is of size $48 \times 48 = 2304$ and the hidden lattice is of size $24 \times 24 = 576$. The $\langle v_i h_a \rangle$ correlators are determined using a visible set of 40000 lattices at $T_c = 2.269$ and the hidden set produced by an application of RG and by applying a trained RBM (which is trained on this same visible data set). The results for $\langle x_i x_j \rangle$ are shown in Figure 11. For the RBM (Figure 11d) the fall off of the correlator again matches the behavior of RG (Figure 11b). For the RBM the fall off trend is clearer with these larger lattice sizes when compared to the RBMs of smaller lattice sizes. There is a slight increase in correlation in Figure 11d as the distance nears $L_v/2 = 24$. This suggests

IEEE *Access*

E. de Mello Koch *et al.*: Is Deep Learning a Renormalization Group Flow?
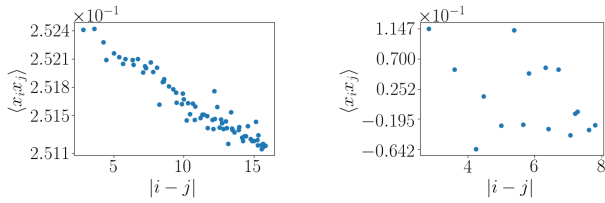
(a) Two point correlator versus distance for Figure 8a-i.

(b) Two point correlator versus distance for Figure 8a-ii.
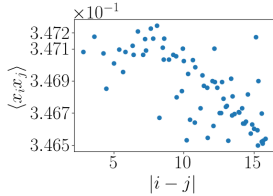
(c) Two point correlator versus distance for Figure 8a-iii.

FIGURE 10. Plots showing $\langle x_i x_j \rangle$ for RBM $\langle v_i h_a \rangle$ plots shown in Figures 8a-i, 8a-ii and 8a-iii.

(a) RG $\langle v_i h_a \rangle$.

(b) RG $\langle x_i x_j \rangle$.

(c) RBM $\langle v_i h_a \rangle$.

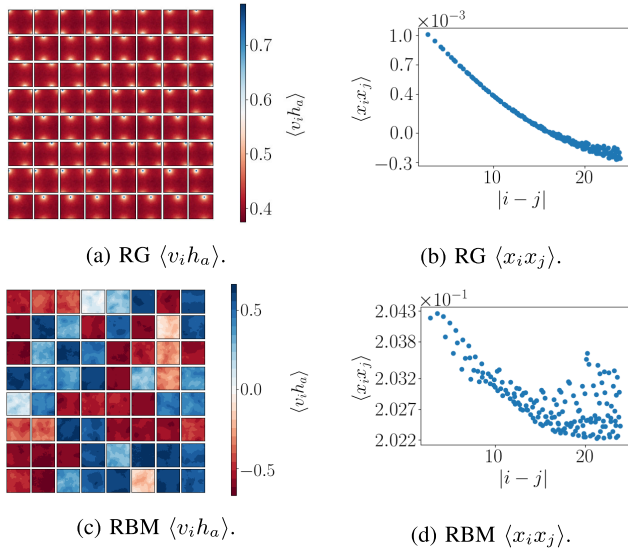(d) RBM $\langle x_i x_j \rangle$.

FIGURE 11. Plots showing the $\langle v_i h_a \rangle$ correlators and corresponding two point correlator $\langle x_i x_j \rangle$ values for one step of RG and a single RBM starting from an input lattice of size $48 \times 48 = 2304$ which is reduced to an output lattice of size $24 \times 24 = 576$.

that there are more than 1 local patches of correlation in the RBM $\langle v_i h_a \rangle$ plots. In Figure 11c we can see that some plots show some speckle with a few highly correlated spots in a single $\langle v_i h_a \rangle$ plot. We verify this observation by considering specific $\langle v_i h_a \rangle$ correlation patterns below.

To gain more understanding of the information encoded in the two point correlator we consider $\langle v_i h_a \rangle$ patterns of white noise in addition to a checkerboard shape with various sizes for the sub-blocks on the checkerboard. This allows us to explore the benefit of studying $\langle x_i x_j \rangle$ in probing patterns present in $\langle v_i h_a \rangle$. We show an example of a single hidden
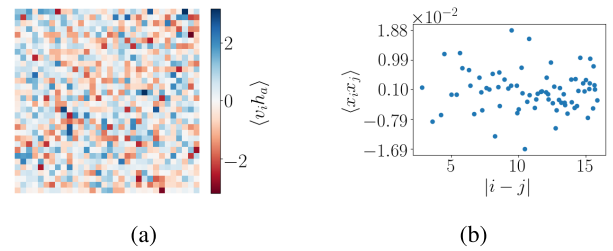

(a)

(b)

FIGURE 12. White noise: Plots showing (a) a hypothetical $\langle v_i h_a \rangle$ correlator (for a single hidden node with all visible nodes) consisting of white noise and (b) the two point correlator $\langle x_i x_j \rangle$ calculated from the values of $\langle v_i h_a \rangle$ in (a).

node's correlation with all visible nodes constructed using white noise in Figure 12a. In Figure 12b we can see $\langle x_i x_j \rangle$ calculated from the values shown in Figure 12a. We see different behavior to that observed in Figures 9 and 10. As expected, there is no clear relationship between the value of the two point correlator $\langle x_i x_j \rangle$ and the distance between values x_i and x_j.

We also study $\langle v_i h_a \rangle$ with a checkerboard pattern as shown in Figure 13. We explore various sub-block sizes within the checkerboard pattern. In Figures 13a, 13c and 13e we show the $\langle v_i h_a \rangle$ plot with a checkerboard pattern on a lattice of size 32×32 with sub-blocks of size 4 by 4, 8 by 8 and 16 by 16 respectively. The corresponding two point correlators are shown in Figures 13b, 13d and 13f. We can see from these plots that having many correlated patches in $\langle v_i h_a \rangle$ which are of size $< L_v/2$, produces a two point correlator which is peaked at a number of points. In the case of Figure 13f, where the sub-block sizes equal $L_v/2$ we see similar behavior to that seen in the RBM and RG correlator plots. The additional peaks seen in Figures 13b and 13d are due to multiple patches in the image being correlated. This behavior is not characteristic of the RG local patches as a single highly correlated patch is present in the RG $\langle v_i h_a \rangle$ plots.

There is one more interesting comparison that can be carried out and it quantitatively tests the flow. The temperature is a relevant coupling so it grows as the flow proceeds. In the block spin RG that we are considering, the length of the lattice keeps halving. Thus, after 7 steps our unit of length is $2^7 = 128 \approx 100$ times larger than it was. To get some insight into the effect of this change of units, imagine we change units from centimeters to meters. In the new units, a length of 100cm is now 1m. Anything with the units of length will roughly halve with each step of the flow. In contrast to this, the temperature of the system, which in suitable units has a dimension of inverse length, will roughly double. There will be small departures from precise doubling due to interactions, but the temperature must increase by roughly a factor of 2 as each new layer is stacked. If the RBM is performing an RG-like coarse graining, the temperature should grow in a similar way as we pass through the layers of the deep network. Figure 14a plots the temperature of coarse grained lattices, generated by applying three steps of RG to an input lattice
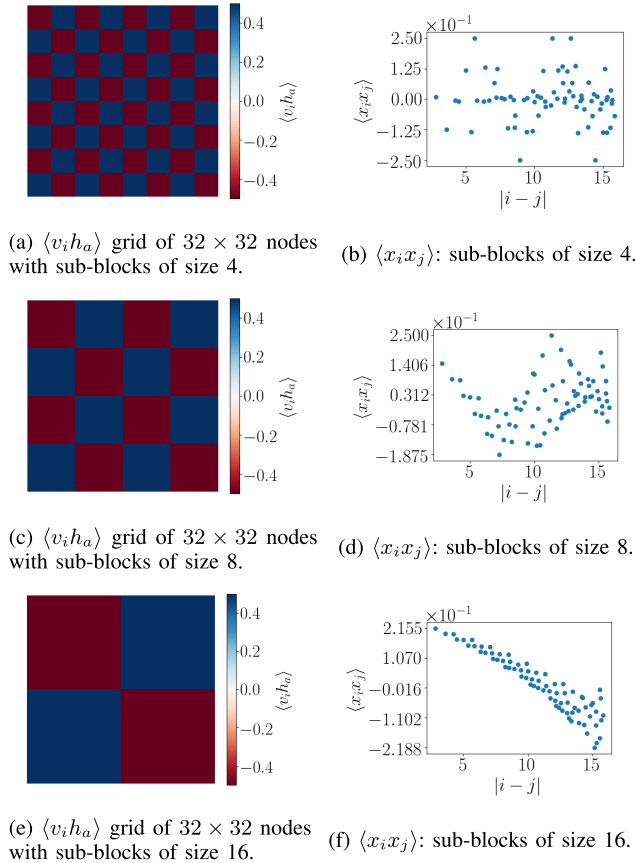
E. de Mello Koch *et al.*: Is Deep Learning a Renormalization Group Flow?

IEEE Access

(a) $\langle v_i h_a \rangle$ grid of 32×32 nodes with sub-blocks of size 4.

(b) $\langle x_i x_j \rangle$: sub-blocks of size 4.

(c) $\langle v_i h_a \rangle$ grid of 32×32 nodes with sub-blocks of size 8.

(d) $\langle x_i x_j \rangle$: sub-blocks of size 8.

(e) $\langle v_i h_a \rangle$ grid of 32×32 nodes with sub-blocks of size 16.

(f) $\langle x_i x_j \rangle$: sub-blocks of size 16.

FIGURE 13. Checkerboard: Plots showing $\langle v_i h_a \rangle$ correlators generated to depict a checkerboard with varying block sizes as well as the two point correlator $\langle x_i x_j \rangle$ corresponding to the given $\langle v_i h_a \rangle$ plots. Plot (b) corresponds to plot (a), plot (d) corresponds to plot (c) and plot (f) corresponds to plot (e).

(a)

(b-i)

(b-ii)

(b-iii)

FIGURE 14. (a) shows the average probability of the measured temperature of lattices resulting after 3 steps of RG, applied to an input lattice at T_C with 4096 sites. (b) shows the average probability plot of the measured temperature of outputs produced by a stacked RBM with 4096 input nodes, 1024 nodes in the first layer, 256 nodes in the second layer and 64 nodes in the output layer. (b-i) is given input Ising samples at $T = 2.269$, (b-ii) is given input Ising samples at $T = 2$ and (b-iii) is given input Ising samples at $T = 2.7$.

of size 64 by 64, at a temperature of $T = 2.7$. There is a clear increase in the measured temperature as the number of RG steps increase. The temperature of each layer is roughly $T = 2.3, 4.8$ and 11 for layers 1, 2 and 3 respectively, which is indeed consistent with the rough rule that the temperature doubles with each step.

Now consider a deep network made by stacking three RBMs. The first network has 4096 visible nodes and 1024 hidden nodes, the second 1024 visible nodes and 256 hidden nodes and the third 256 visible nodes and 64 hidden nodes. The network is trained on Ising data at the critical temperature, as described above. Figures 14b-i, 14bii and 14b-iii give the temperatures of the outputs of the layers of the RBM, given input lattices at temperatures of $T = 2.269, 2$ and 2.7 respectively. Temperatures of $T = 2$ and $T = 2.269$ lead to the same behavior for the temperature flow, as exhibited in Figures 14b-i and 14b-ii. The temperature jumps rapidly to a high temperature in the first step of the flow, and remains fixed when the second step is taken. This is an important difference that deserves to be understood better. It questions the identification of layers of a deep network with steps in an RG flow.
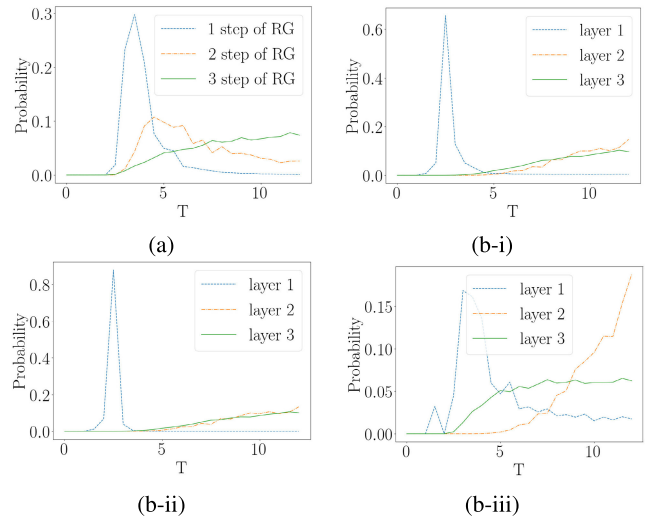
Figure 14b-iii shows different characteristics to those of 14b-i and 14b-ii. Here the temperature of the input is above T_c at 2.7. Layer 1 is not as sharply peaked near T_c as observed in Figures 14b-i and 14b-ii. In addition to this, layers 2 and 3 are not at the same temperature but rather layer 2 is at a higher temperature than layer 3. This differs to the RG flow, where temperature increases along the flow. Figure 14b-iii shows a decrease in temperature from layer 2 to layer 3 rather than an increase. These plots demonstrate that the flow defined by multiple layers in a "deep" network show important differences to the RG flow. The discrepancies we have uncovered are important and precise quantitative mismatches that may provide useful clues in understanding the relationship between unsupervised deep learning by an RBM and the RG flows.

The results above have shown that the correlator $\langle v_i h_a \rangle$ exhibits RG-like characteristics. This is evident from the comparison between the $\langle v_i h_a \rangle$ plots from RG, a stacked RBM network and a network with a single RBM. We can see RG-like patterns in the correlators produced by the two RBM networks. This is a promising result that demonstrates that a form of coarse graining is taking place when networks are stacked.

V. CONCLUSIONS AND DISCUSSION
Our main goal has been to explore the possibility that RG provides a framework within which a theoretical understanding of deep learning can be pursued. We have focused on a single model, the Ising model, which is naturally related to RBMs. Thus, at best our conclusions and discussion can only suggest interesting avenues for further study. We are not able to draw general definite conclusions about the applicability of

IEEE *Access*

E. de Mello Koch *et al.*: Is Deep Learning a Renormalization Group Flow?

RG as a framework within which a theoretical understanding of deep learning can be achieved. Our data set contains the possible states of an Ising magnet, generated using Monte Carlo simulation. This is an interesting data set, since we know that there is a well defined theory for the magnet defined on large length scales. The existence of this long distance theory guarantees that there is some emergent order for the unsupervised learning to identify. Another point worth stressing is that the RG treatment of this system is well understood and is easily implemented numerically. It is therefore an ideal setting in which both deep learning and RG can be implemented and their results can be compared. At the critical temperature, where the system is on the verge of spontaneous magnetization, there is an interesting scale invariant theory which is well understood. By working at this critical point, we have managed to probe the patterns generated by the RBM at different length scales and to compare it to the expected results from an RG treatment.

Our first set of numerical results compare the RBM flow introduced in [15] and further pursued in [16]. From a theoretical point of view the RBM flow looks rather different to RG since the RBM flow appears to drive configurations towards the critical temperature. The RG would drive configurations to ever higher temperatures due to the fact that the temperature corresponds to a relevant perturbation. Another important difference between the RBM flow and RG is that the number of spins is a constant of the RBM flow, but decreases with the RG flow. Our numerical results confirm that the RBM flow does indeed generate RG-like Ising configurations and we have reproduced the scaling dimension of the spin variable from the spatial statistics of the patterns generated by the RBM. This is a remarkable result and it extends and supports results reported and discussed in [15], [16]. The spin variable has the smallest possible scaling dimensions and consequently probes the largest possible scales in the pattern. When considering correlation functions of the next primary operators we find that the RBM data does not reproduce the correct scaling dimension, proving that the spatial statistics of the patterns generated by the RBM flow and those generated by RG start to differ as smaller scales are tested. We therefore conclude that the RBM flow and RG are distinct, but they do agree on the largest scale structure of the generated patterns. This is a hint into the mechanism behind the RBM flow and it deserves an explanation.

Our second numerical study has explored the idea that deep learning is an RG flow with each stacked layer performing a step of RG. We have explained why correlation functions between the visible and hidden neurons, $\langle v_i h_a \rangle$ are capable of diagnosing RG-like coarse graining and we have computed these correlation functions using the patterns generated by the RBM. The basic signal of RG coarse graining is a "bright spot" in the $\langle v_i h_a \rangle$ correlation function, since this indicates that spins in a localized region were averaged to produce the coarse grained spin. The numerical results do indeed show a dark background with emerging bright spots. It would be interesting if the emergent patterns again guarantee

agreement on the largest length scales, similar to what was found for the RBM flows, but we can not confidently make this assertion yet.

Our final numerical study considered the flow of the temperature, a relevant operator according to the RG. We find three distinct behaviors. Section III-B reviewed that RBM flows converge to the critical temperature. This is borne out in our results. The RG flows to ever higher temperatures, with (roughly) a doubling in temperature for each step. Again, this is precisely what we observe. Finally, for a deep network made by stacking three RBMs, the temperature appears to flow when moving between the first and second layers of a deep network, but is fixed when moving between the second and third layers. This is an important difference that deserves to be understood better. It questions the identification of layers of a deep network with steps in an RG flow.

Our results are encouraging. There are enough similarities between unsupervised learning by an RBM and the RG flow that the relationship between the two should be developed further. Regarding future studies, it maybe useful to explore models other than Ising. The Ising model has an unstable fixed point due to the presence of relevant operators. Consequently, finite flows starting near the critical point all terminate on different models. In this case its not easy to know if the RBM has flowed to the "right answer" because there are many possible right answers! The stable fixed point of the model is at infinite temperature and the configurations at this fixed point are random with correlators that have a correlation length of zero. This is hardly a promising answer to shoot for. It maybe more instructive to study models that have an attractive RG fixed point. In this case the minimum that the RBM is looking for would be unique and the connection between the two may be easier to recognize. We have in mind systems that exhibit self organized criticality [36], including models constructed to understand the spread of forest fires [34] and models for the spread of infectious diseases [35].

By using Ising model data, generated by Monte Carlo simulation, starting from a local Hamiltonian we know how a coarse graining capable of identifying emergent patterns should proceed: spatially neighboring spins should be averaged. For more general data sets, this may not be the case. It is fascinating to ask what the rules determining the correct coarse graining are and in fact, with respect to this question, deep learning has the potential to shed light on RG.

Another interesting comparison worth mentioning is the similarity between an average pooling layer within a convolutional neural network (CNN) and the averaging performed in variational RG. CNNs are known for their excellent performance in image recognition and classification tasks [42]–[44]. CNNs have a number of layers which act on groups of nearby pixels in the image. One of these layers which is similar to the coarse graining performed in variational RG is called a pooling layer. The pooling layer performs a down-sampling on the data it receives from previous layers in the network [45]–[47]. The down-sampled data is more robust to changes in position of features and gives

E. de Mello Koch *et al.*: Is Deep Learning a Renormalization Group Flow?

IEEE *Access*

the network the property of local translation invariance. One way in which the pooling operation is implemented is by averaging all values in the given patch of data it acts on to obtain a new value to replace these values. Pooling usually averages blocks of data which are of size 2×2. This results in an input block of data being reduced by a factor of 2 in length and by a factor of 4 in the number of values which is the same factor of rescaling which occurs in variational RG.

In recent years a connection between the renormalization group and tensor networks [48] has been discovered, providing a connection to the field of quantum information. The discovered connection demonstrates that the multi-scale entanglement renormalization ansatz (MERA) tensor networks carry out a coarse graining that agrees in many ways with the coarse graining performed by the renormalization group [49]. This suggests that there maybe a link between tensor networks and deep learning. For related ideas see [50], [51]. Since tensor networks have been extensively studied for calculations the connection may prove to be useful for better understanding deep learning.

Apart from the exciting possibility that the link to RG might contribute towards a theoretical understanding of deep learning, one might also ask if the connection would have any practical applications. One possibility that we are currently pursuing, is a Callan-Symanzik like equation governing the learning process. Roughly speaking, one might mimic RG by dividing the weights to be learned into relevant, marginal and irrelevant parameters, depending on gross statistical properties of the training data. If this classification is itself not too expensive, one could pursue a more efficient approach towards training, since the classification of weights would provide an understanding of which weights are important, and which can simply be set to zero. We hope to report on this possibility in the future.

APPENDIX A
RBM EXPECTATION VALUES
The expectation values quoted in equations (7), (8) and (9) are derived using (2). Data expectation values are evaluated by summing over all samples, $\hat{v}_i^{(A)}$ in the training set. On the other hand model expectation values employ sums over the entire space of visible and hidden vectors. This is such an enormous sum that its numerically intractable. Consequently, the approximations described in Section II-A are used. The complete set of expectation values needed to describe the RBM are given by

$$\langle v_i h_a \rangle_{data} = \frac{1}{N_s} \sum_{A=1}^{N_s} \hat{v}_i^{(A)} \tanh\left(\sum_k W_{ka}\hat{v}_k^{(A)} + b_a^{(h)}\right), \quad (58)$$

$$\langle v_i h_a \rangle_{model} = \sum_{\{v,h\}} \tanh\left(\sum_k W_{ka}v_k + b_a^{(h)}\right)$$
$$\cdot \tanh\left(\sum_j W_{ij}h_j + + b_i^{(v)}\right), \quad (59)$$

$$\langle v_i \rangle_{data} = \frac{1}{N_s} \sum_{A=1}^{N_s} \hat{v}_i^{(A)}, \quad (60)$$

$$\langle v_i \rangle_{model} = \sum_{\{h\}} \tanh\left(\sum_a W_{ia}h_a + b_i^{(v)}\right), \quad (61)$$

$$\langle h_a \rangle_{data} = \frac{1}{N_s} \sum_{A=1}^{N_s} \tanh\left(\sum_i W_{ia}\hat{v}_i^{(A)} + b_a^{(h)}\right), \quad (62)$$

$$\langle h_a \rangle_{model} = \sum_{\{v\}} \tanh\left(\sum_i W_{ia}v_i + b_a^{(h)}\right), \quad (63)$$

with $\hat{v}_i^{(A)}$ the Ath sample of the data set, $\hat{v}$.

APPENDIX B
TWO VERSIONS OF RG
In this section we review two versions of the RG that are needed in this article. The first of these, the variational renormalization group, was introduced by Kadanoff [52]–[54] as a method to approximately perform the renormalization group in practice.

A. VARIATIONAL RG
Consider a system of N spins $\{v_i\}$ which each take the values ± 1. The partition function describing the system is given by

$$Z = \sum_{v_i} e^{-H(\{v_i\})}. \quad (64)$$

Here the sum is over all possible configurations of the system of spins and the function $H(\{v_i\})$, called the Hamiltonian, gives the energy of the system. This would include the energy of each individual spin as well as the energy associated to the fact that the collection of spins is interacting. The Hamiltonian $H(\{v_i\})$ can be an arbitrarily complicated function of the spins

$$H(\{v_i\}) = -\sum_i K_i v_i - \sum_{i,j} K_{ij}v_i v_j$$
$$- \sum_{i,j,k} K_{ijk}v_i v_j v_k + \cdots. \quad (65)$$

The RG flows maps the original Hamiltonian to a new Hamiltonian with a different set of coupling constants. The new Hamiltonian

$$H(\{h_a\}) = -\sum_a K_a' h_a - \sum_{a,b} K_{ab}' h_a h_b$$
$$- \sum_{a,b,c} K_{abc}' h_a h_b h_c + \cdots, \quad (66)$$

gives the energy for the coarse grained spins h_a. After many RG iterations many coupling constants (the so called irrelevant terms) flow to zero. A much smaller number may remain constant (marginal terms) or even grow (relevant terms). To implement this conceptual framework a concrete RG mapping is needed. Variational RG provides a mapping which is not exact but can be implemented numerically. It does this

IEEE Access

E. de Mello Koch *et al.*: Is Deep Learning a Renormalization Group Flow?

by introducing an operator $T_\lambda(\{v_i, h_a\})$ which is a function of a set of parameters $\{\lambda\}$. The Hamiltonian after a step of RG flow is

$$e^{-H_{RG}(\{h_a\})} = \sum_{v_i} e^{T_\lambda(\{v_i, h_a\}) - H(\{v_i\})}. \qquad (67)$$

The form of $T_\lambda(\{v_i, h_a\})$ must be chosen cleverly, for each problem we consider. This is the tough step in variational RG and it is carried out using physical intuition, but essentially on a trial and error basis. Once a given $T_\lambda(\{v_i, h_a\})$ has been chosen, we minimize the following quantity by choosing the parameters $\{\lambda\}$

$$\log(\sum_{v_i} e^{-H(\{v_i\})}) - \log(\sum_{h_a} e^{-H_{RG}(\{h_a\})}). \qquad (68)$$

The minimum possible value for this quantity is zero. Notice that when

$$\sum_{h_a} e^{T_\lambda(\{v_i, h_a\})} = 1, \qquad (69)$$

(68) attains its minimum value of 0 and the RG transformation is called exact.

B. BLOCK SPIN AVERAGING
Block spin averaging is a pedagogical version of RG. To illustrate the method, consider a rectangular lattice of interacting spins. Divide the lattice into blocks of 2×2 squares. Block spin averaging describes the system in terms of *block variables*, which are variables describing the average behavior of each block. The "block spin" is literally the average of the four spins in the block. The plots shown in Figures 7a use block spin averaging. The block spins h_a are each an average of four visible spins v_i.

ACKNOWLEDGEMENT

REFERENCES
[1] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, Jul. 2015.
[2] L. Deng and D. Yu, "Deep learning: Methods and applications," *Found. Trends Signal Process.*, vol. 7, nos. 3–4, pp. 197–387, Jun. 2014.
[3] Y. Bengio, "Learning deep architectures for AI," *Found. Trends Mach. Learn.*, vol. 2, no. 1, pp. 1–127, 2009.
[4] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.
[5] N. Le Roux and Y. Bengio, "Deep belief networks are compact universal approximators," *Neural Comput.*, vol. 22, no. 8, pp. 2192–2207, Aug. 2010.
[6] N. Le Roux and Y. Bengio, "Representational power of restricted Boltzmann machines and deep belief networks," *Neural Comput.*, vol. 20, no. 6, pp. 1631–1649, Jun. 2008.
[7] Y. Bengio and Y. LeCun, "Scaling learning algorithms towards AI," *Large-Scale Kernel Mach.*, vol. 34, no. 5, pp. 1–41, 2007.
[8] A. Paul and S. Venkatasubramanian, "Why does deep learning work?-a perspective from group theory," 2014, *arXiv:1412.6621*. [Online]. Available: http://arxiv.org/abs/1412.6621

[9] Y. Bengio, A. C. Courville, and P. Vincent, "Unsupervised feature learning and deep learning: A review and new perspectives," *CoRR*, vol. abs/1206.5538, pp. 1–30, Jun. 2012.
[10] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning requires rethinking generalization," 2016, *arXiv:1611.03530*. [Online]. Available: http://arxiv.org/abs/1611.03530
[11] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 153–160.
[12] H. W. Lin, M. Tegmark, and D. Rolnick, "Why does deep and cheap learning work so well?" *J. Stat. Phys.*, vol. 168, no. 6, pp. 1223–1247, 2017.
[13] J. Kogut and K. G. Wilson, "The renormalization group and the ϵ expansion," *Phys. Rep. C*, vol. 12, no. 2, pp. 75–200, 1974.
[14] P. Mehta and D. J. Schwab, "An exact mapping between the variational renormalization group and deep learning," 2014, *arXiv:1410.3831*. [Online]. Available: http://arxiv.org/abs/1410.3831
[15] S. Iso, S. Shiba, and S. Yokoo, "Scale-invariant feature extraction of neural network and renormalization group flow," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 97, no. 5, May 2018, Art. no. 053304.
[16] S. S. Funai and D. Giataganas, "Thermodynamics and feature extraction by machine learning," 2018, *arXiv:1810.08179*. [Online]. Available: http://arxiv.org/abs/1810.08179
[17] C. Bény, "Deep learning and the renormalization group," 2013, *arXiv:1301.3124*. [Online]. Available: http://arxiv.org/abs/1301.3124
[18] M. Koch-Janusz and Z. Ringel, "Mutual information, neural networks and the renormalization group," *Nature Phys.*, vol. 14, no. 6, pp. 578–582, Jun. 2018.
[19] B. M. McCoy and T. T. Wu, *The Two-Dimensional Ising Model*. North Chelmsford, MA, USA: Courier Corporation, 2014.
[20] J. Carrasquilla and R. G. Melko, "Machine learning phases of matter," *Nature Phys.*, vol. 13, no. 5, pp. 431–434, May 2017.
[21] A. Morningstar and R. G. Melko, "Deep learning the Ising model near criticality," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 5975–5991, 2017.
[22] S. Saremi and T. J. Sejnowski, "Hierarchical model of natural images and the origin of scale invariance," *Proc. Nat. Acad. Sci. USA*, vol. 110, no. 8, pp. 3071–3076, Feb. 2013.
[23] P. Smolensky, "Information processing in dynamical systems: Foundations of harmony theory," Dept. Comput. Sci., Colorado Univ. Boulder, Boulder, CO, USA, Tech. Rep. CU-CS-321-86, 1986.
[24] Y. Freund and D. Haussler, "Unsupervised learning of distributions on binary vectors using two layer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 1992, pp. 912–919.
[25] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Comput.*, vol. 14, no. 8, pp. 1771–1800, Aug. 2002.
[26] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Nat. Acad. Sci. USA*, vol. 79, no. 8, pp. 2554–2558, 1982.
[27] G. E. Hinton, "A practical guide to training restricted Boltzmann machines," in *Neural Networks: Tricks of the Trade*. Berlin, Germany: Springer, 2012, pp. 599–619.
[28] M. A. Carreira-Perpinan and G. E. Hinton, "On contrastive divergence learning," in *Proc. Aistats*. Princeton, NJ, USA: CiteSeer, vol. 10, 2005, pp. 33–40.
[29] R. Salakhutdinov, A. Mnih, and G. Hinton, "Restricted Boltzmann machines for collaborative filtering," in *Proc. 24th Int. Conf. Mach. Learn. (ICML)*, 2007, pp. 791–798.
[30] M. Ranzato, A. Krizhevsky, and G. Hinton, "Factored 3-way restricted Boltzmann machines for modeling natural images," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 621–628.
[31] T. Tieleman, "Training restricted Boltzmann machines using approximations to the likelihood gradient," in *Proc. 25th Int. Conf. Mach. Learn. (ICML)*, 2008, pp. 1064–1071.
[32] I. Sutskever and T. Tieleman, "On the convergence properties of contrastive divergence," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 789–795.
[33] C. G. Callan, "Broken scale invariance in scalar field theory," *Phys. Rev. D, Part. Fields*, vol. 2, no. 8, p. 1541, 1970.
[34] V. Loreto, L. Pietronero, A. Vespignani, and S. Zapperi, "Renormalization group approach to the critical behavior of the forest-fire model," *Phys. Rev. Lett.*, vol. 75, no. 3, p. 465, 1995.

E. de Mello Koch *et al.*: Is Deep Learning a Renormalization Group Flow?

IEEE *Access*

[35] S. Clar, B. Drossel, and F. Schwabl, "Forest fires and other examples of self-organized criticality," *J. Phys., Condens. Matter*, vol. 8, no. 37, p. 6803, 1996.

[36] A. Diaz-Guilera, "Dynamic renormalization group approach to self-organized critical phenomena," *Europhys. Lett.*, vol. 26, no. 3, p. 177, 1994.

[37] D. Poland, S. Rychkov, and A. Vichi, "The conformal bootstrap: Theory, numerical techniques, and applications," *Rev. Mod. Phys.*, vol. 91, no. 1, Jan. 2019, Art. no. 015002.

[38] R. Ruger. (2014). SSMC—Monte Carlo Simulation Code for Classical Spin Systems Like the Ising Model. Ssmc. [Online]. Available: https://github.com/robertrueger/SSMC

[39] (2020). *Ellen de Mello Koch. Dl_and_rg*. [Online]. Available: https://github.com/ellendmk/DL_and_RG

[40] F. Chollet and J. J. Allaire. (2015). *Keras*. [Online]. Available: https://github.com/fchollet/keras

[41] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986.

[42] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[43] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4700–4708.

[44] A. Khan, A. Sohail, U. Zahoora, and A. Saeed Qureshi, "A survey of the recent architectures of deep convolutional neural networks," 2019, *arXiv:1901.06032*. [Online]. Available: http://arxiv.org/abs/1901.06032

[45] C.-Y. Lee, P. W. Gallagher, and Z. Tu, "Generalizing pooling functions in convolutional neural networks: Mixed, gated, and tree," in *Proc. Artif. Intell. Statist.*, 2016, pp. 464–472.

[46] D. Yu, H. Wang, P. Chen, and Z. Wei, "Mixed pooling for convolutional neural networks," in *Proc. Int. Conf. Rough Sets Knowl. Technol.* Cham, Switzerland: Springer, 2014, pp. 364–375.

[47] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2921–2929.

[48] G. Vidal, "Entanglement renormalization," *Phys. Rev. Lett.*, vol. 99, no. 22, Nov. 2007, Art. no. 220405.

[49] G. Evenbly and G. Vidal, "Tensor network renormalization," *Phys. Rev. Lett.*, vol. 115, no. 18, Oct. 2015, Art. no. 180405.

[50] A. Cichocki, N. Lee, I. Oseledets, A.-H. Phan, Q. Zhao, and D. P. Mandic, "Tensor networks for dimensionality reduction and large-scale optimization: Part 1 low-rank tensor decompositions," *Found. Trends Mach. Learn.*, vol. 9, nos. 4–5, pp. 249–429, 2016.

[51] A. Cichocki, A.-H. Phan, Q. Zhao, N. Lee, I. Oseledets, M. Sugiyama, and D. P. Mandic, "Tensor networks for dimensionality reduction and large-scale optimization: Part 2 applications and future perspectives," *Found. Trends Mach. Learn.*, vol. 9, no. 6, pp. 431–673, 2017.

[52] L. P. Kadanoff, *Statistical Physics: Statics, Dynamics and Renormalization*. Singapore: World Scientific, 2000.

[53] L. P. Kadanoff, A. Houghton, and M. C. Yalabik, "Variational approximations for renormalization group transformations," *J. Stat. Phys.*, vol. 14, no. 2, pp. 171–203, Feb. 1976.

[54] E. Efrati, Z. Wang, A. Kolan, and L. P. Kadanoff, "Real-space renormalization in statistical mechanics," *Rev. Mod. Phys.*, vol. 86, no. 2, p. 647, 2014.

ELLEN DE MELLO KOCH received the B.Sc. (Eng.) and M.Sc. (Eng.) degrees in electrical engineering from the University of the Witwatersrand in 2014 and 2016, respectively. She is currently pursuing the Ph.D. degree with the Department of Electrical and Information Engineering. She is currently a Lecturer with the Department of Electrical and Information Engineering. Her Ph.D. research investigates the link between deep learning and the renormalization group, as an attempt to develop a theoretical framework for deep learning. Her research interests include unsupervised learning algorithms and their application to the real world.

ROBERT DE MELLO KOCH received the B.Sc. (Eng.), B.Sc.Hon. (Phys.), M.Sc. (Phys.), and Ph.D. (Phys.) degrees from the University of the Witwatersrand in 1992, 1993, 1994, and 1998, respectively. He is a Professor with the University of the Witwatersrand, where he holds the DST/NRF Research Chair in fundamental physics and string theory and is a Distinguished Visiting Professor with South China Normal University. He is a Fellow of the Durham Institute for Advanced Studies, the Stellenbosch Institute for Advanced Studies, and the Academy of Science of South Africa. His current research interests include the gauge theory/string theory duality, the application of representation theory of discrete groups, and Lie groups to quantum field theory and deep learning.

LING CHENG (Senior Member, IEEE) received the B.Eng. degree *(cum laude)* in electronics and information from the Huazhong University of Science and Technology (HUST), in 1995, the M.Ing. degree *(cum laude)* in electrical and electronics in 2005, and the D.Ing. degree in electrical and electronics from the University of Johannesburg (UJ), in 2011. His research interests include telecommunications and artificial intelligence. In 2010, he joined the University of the Witwatersrand, where he was promoted to Associate Professor in 2015. He has served as the Vice-Chair for the IEEE South African Information Theory Chapter. He has been a visiting professor at five universities and the principal advisor for over 40 full research post-graduate students. He has published more than 80 research papers in journals and conference proceedings. He was awarded the Chancellor's medals in 2005 and 2019 and the National Research Foundation Rating in 2014. The IEEE ISPLC 2015 Best Student Paper Award was made to his Ph.D. student in Austin.

• • •